



KEMENTERIAN
PENDIDIKAN DASAR
DAN MENENGAH
2025

INFORMATIKA

Edisi Revisi

Erlangga
Erna Piantari
Khairur Rosyidi

SMP/MTs Kelas IX

Hak Cipta pada Kementerian Pendidikan Dasar dan Menengah Republik Indonesia.

Dilindungi Undang-Undang.

Penafian: Buku ini disiapkan oleh Pemerintah dalam rangka pemenuhan kebutuhan buku pendidikan yang bermutu, murah, dan merata sesuai dengan amanat dalam UU No. 3 Tahun 2017. Buku ini disusun dan ditelaah oleh berbagai pihak di bawah koordinasi Kementerian Pendidikan Dasar dan Menengah. Buku ini merupakan dokumen hidup yang senantiasa diperbaiki, diperbarui, dan dimutakhirkan sesuai dengan dinamika kebutuhan dan perubahan zaman. Masukan dari berbagai kalangan yang dialamatkan kepada penulis atau melalui alamat surel buku@kemendikdasmen.go.id diharapkan dapat meningkatkan kualitas buku ini.

Informatika untuk SMP/MTs Kelas IX (Edisi Revisi)

Penulis

Erlangga
Erna Piantari
Khairur Rosyidi

Penelaah

Budi Permana
Aris Sunawar

Penyelia/Penyelarass

Supriyatno
Maharani Prananingrum
Frandi Yuanda
Ervina

Kontributor

Armia
Sahrudin

Ilustrator

Prehatin

Editor

Misianita Hapsari
Ervina
Frandi Yuanda

Editor Visual

M Rizal Abdi

Desainer

Syndhi Renolarisa

Penerbit

Kementerian Pendidikan Dasar dan Menengah

Dikeluarkan oleh

Pusat Perbukuan
Kompleks Kemendikdasmen Jalan RS. Fatmawati, Cipete, Jakarta Selatan
<https://buku.kemendikdasmen.go.id>

Edisi Revisi, 2025

ISBN 978-623-118-482-5 (no.jil.lengkap)

ISBN 978-634-00-3299-4 (jil.3 PDF)

Isi buku ini menggunakan Noto Serif 9/12 pt, Steve Matteson, Open Font License.
xvi, 264 hlm.: 17,6 x 25 cm.



KATA PENGANTAR

Pusat Perbukuan; Badan Standar, Kurikulum, dan Asesmen Pendidikan; Kementerian Pendidikan Dasar dan Menengah memiliki tugas dan fungsi mengembangkan buku pendidikan pada satuan Pendidikan Anak Usia Dini, Pendidikan Dasar, dan Pendidikan Menengah, termasuk Pendidikan Khusus. Sebagai sumber belajar utama dalam pembelajaran, buku senantiasa menyesuaikan diri dengan arah kebijakan pendidikan dan perkembangan kurikulum dari waktu ke waktu.

Sejalan dengan dinamika tersebut, pemerintah melakukan penyesuaian terhadap buku teks utama agar tetap relevan dan mendukung proses pembelajaran yang bermutu. Buku teks utama yang terdiri atas buku siswa dan panduan guru, disiapkan sebagai sumber belajar utama bagi peserta didik dan menjadi salah satu referensi atau inspirasi bagi guru dalam merancang dan mengembangkan pembelajaran sesuai karakteristik, potensi, dan kebutuhan peserta didik. Keberadaan buku teks utama ini diharapkan, dapat membentuk generasi yang beriman dan bertakwa kepada Tuhan yang Maha Esa, dan berakhlak mulia; berpengetahuan luas; mandiri; kritis; kreatif; mampu bekerja sama; serta berdaya saing di tingkat nasional maupun global.

Kami mengucapkan terima kasih kepada semua pihak yang telah berkolaborasi dalam upaya menghadirkan buku teks utama ini. Semoga buku teks utama ini dapat menjadi tonggak perubahan yang menginspirasi, membimbing, dan mengangkat kualitas pendidikan kita ke puncak keunggulan.

Jakarta, Oktober 2025
Kepala Pusat Perbukuan,

Supriyatno, S.Pd., M.A.





PRAKATA

Puji syukur kehadiran Tuhan Yang Maha Esa senantiasa kita panjatkan atas rahmat dan karunia-Nya, sehingga buku **Informatika untuk Siswa SMP Kelas IX** ini dapat diselesaikan. Buku ini disusun untuk membantu peserta didik memahami serta menguasai konsep dan keterampilan Informatika sesuai dengan pembelajaran mendalam (*deep learning*), sekaligus menumbuhkan karakter positif sebagai pengguna teknologi yang cerdas, kreatif, dan bertanggung jawab.

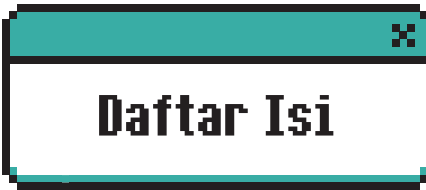

Perkembangan teknologi informasi dan komunikasi yang sangat pesat menuntut generasi muda memiliki literasi digital yang kuat. Mata pelajaran Informatika menjadi jembatan penting agar peserta didik tidak hanya berperan sebagai pengguna, tetapi juga mampu menjadi pencipta dan pengembang solusi berbasis teknologi. Oleh karena itu, materi dalam buku ini disusun secara sistematis, kontekstual, dan aplikatif, sehingga peserta didik dapat mempelajari Informatika secara menyeluruh dan menyenangkan.

Isi buku mengacu pada capaian pembelajaran mata pelajaran Informatika kelas IX, terdiri atas dua elemen utama yaitu Berpikir Komputasional (BK) dan Literasi Digital (LD). Kedua elemen tersebut dipelajari secara tematik dan diperkuat melalui berbagai aktivitas, baik individu maupun kelompok, untuk menumbuhkan profil lulusan peserta didik yang bernalar kritis, kreatif, dan mampu bergotong royong. Aktivitas juga dirancang dengan moda *plugged* (menggunakan komputer), sehingga peserta didik dapat memahami konsep sekaligus mengimplementasikan Informatika secara mendalam dan bermakna.

Materi dan aktivitas dalam buku ini telah disesuaikan dengan kebutuhan peserta didik IX dan dikemas dalam bentuk yang menarik. Akhir kata, penulis berharap buku siswa ini dapat bermanfaat dan digunakan sebagai pendamping belajar Informatika sebaik-baiknya. Saran dan kritik yang membangun sangat diharapkan untuk meningkatkan kualitas buku ini di masa mendatang.

Jakarta, September 2025

Penulis



Daftar Isi

Kata Pengantar	iii
Prakata.....	iv
Daftar Isi.....	v
Daftar Gambar	vii
Daftar Tabel.....	xi
BAB I Berpikir Komputasional dalam Analisis Data	1
A. Struktur Data <i>Tree</i>	7
B. Struktur Data <i>Graph</i>	15
C. Analisis Himpunan Data Terstruktur dengan Teknik Visualisasi.....	19
1. Menentukan Tujuan Analisis Data	20
2. Mengumpulkan Data yang Diperlukan.....	21
3. Menyiapkan dan Membersihkan Data	21
4. Mengeksplorasi Data.....	22
5. Memvisualisasikan dan Memublikasikan Hasil Analisis Data	23
BAB II Berpikir Komputasional dalam Algoritma dan Pemrograman	39
A. Mengembangkan <i>Library</i> dalam Pemrograman Visual Blok.....	42
1. Modularisasi Program	43
2. <i>Library</i>	64
3. Penggunaan <i>Library</i>	66
B. Pemrograman Visual Blok vs Pemrograman Tekstual.....	76
1. Transisi dari Pemrograman Visual Blok ke Pemrograman Tekstual	76
2. Analisis <i>Pseudocode</i>	106
BAB III Literasi Digital untuk Produktivitas	115
A. Pemrosesan Data.....	119
1. Permasalahan Dasar Pemrosesan Data.....	122
2. Dampak Penggunaan Perangkat yang Baik dan Benar.....	125
B. Penyajian Data	130
1. Jenis dan Format Data.....	130
2. Perangkat Lunak Produktivitas	150
3. Integrasi Data Konten	164

C. Perancangan Konten	170
1. Strategi Perancangan Konten Digital	171
2. Perancangan Konten Digital Menggunakan Kecerdasan Artifisial (KA)..	174
BAB IV Keamanan Digital	181
A. Jenis Informasi	185
1. Data Pribadi Umum	187
2. Data Pribadi Spesifik	187
B. Serangan Siber	194
1. Jenis Serangan Siber	194
2. Dampak Pada Teknologi Modern	202
3. Penanganan	203
C. Keamanan Jaringan Nirkabel di Ruang Publik	209
1. Jaringan Nirkabel	209
2. Dampak Pengabaian Keamanan Jaringan	210
3. Risiko Pengiriman Data di Jaringan Publik	211
D. Perlindungan Data Pribadi	215
1. Kata Sandi yang Kuat	215
2. Autentikasi Dua Faktor	217
3. Enkripsi dan Dekripsi	218
4. <i>Update Software</i>	222
E. Perilaku dan Kesadaran Penuh (<i>Mindfulness</i>) pada Dunia Digital	225
1. Perilaku Digital	226
2. <i>Mindfulness</i> Dunia Digital	229
3. Pola Hidup Sehat Mendukung Kesehatan Mental	235
Glosarium	241
Daftar Pustaka	249
Daftar Sumber Gambar	252
Indeks	254
Profil Pelaku Perbukuan	256



Daftar Gambar



Gambar 1.1	Peta Materi Bab I Berpikir Komputasional dalam Analisis Data	2
Gambar 1.2	Empat Kunci dalam Konsep Berpikir Komputasional	4
Gambar 1.3	Ilustrasi Penyimpanan Alat Makan di Dapur	5
Gambar 1.4	Ilustrasi Perbandingan Struktur Pohon dan Struktur Data <i>Tree</i> (Pohon)	7
Gambar 1.5	Struktur Data <i>Tree</i>	8
Gambar 1.6	Contoh Struktur Penyimpanan Folder di Komputer.....	8
Gambar 1.7	Contoh Struktur Data <i>Tree</i> pada Struktur Organisasi Sekolah	9
Gambar 1.8	Ilustrasi Labirin Air	10
Gambar 1.9	Ilustrasi Labirin Air Setelah Satu Jam.....	11
Gambar 1.10	Contoh Representasi Aturan Permasalahan Jadwal Bermain dalam Struktur Data <i>Tree</i>	12
Gambar 1.11	Contoh Representasi Aturan Permasalahan Jadwal Bermain Dalam Struktur Data <i>Tree</i>	13
Gambar 1.12	Contoh Jaringan Pertemanan Dalam Struktur Data <i>Graph</i>	15
Gambar 1.13	Contoh Representasi Jalur Transportasi pada Struktur Data <i>Graph</i>	16
Gambar 1.14	Contoh Representasi Rute Truk Pengangkut Sampah dalam <i>Graph</i>	17
Gambar 1.15	Diagram Batang	23
Gambar 1.16	Grup Diagram Batang	23
Gambar 1.17	Diagram Garis	24
Gambar 1.18	Diagram Lingkaran.....	24
Gambar 1.19	Diagram Pencar	25
Gambar 1.20	Laju Pertumbuhan Penduduk.....	26
Gambar 1.21	Jumlah Wisatawan Mancanegara	26
Gambar 1.22	Distribusi Peredaran Uang	26
Gambar 1.23	Hubungan Kinerja Infrastruktur dengan Ekonomi Setiap Provinsi	27
Gambar 1.24	Tabel Data Diagram Batang	27
Gambar 1.25	Pemilihan Data untuk Membuat Diagram Batang.....	27
Gambar 1.26	<i>Tampilan Menu Insert</i>	27
Gambar 1.27	Pemilihan Diagram Batang Vertikal atau Horizontal	28
Gambar 1.28	Tabel Data Diagram Batang Grup	28
Gambar 1.29	Pemilihan Data untuk Membuat Diagram Batang Grup.....	28
Gambar 1.30	<i>Tampilan Menu Insert</i>	28
Gambar 1.31	Pemilihan Diagram Batang	29
Gambar 1.32	Tabel Data Diagram Garis.....	29

Gambar 1.33 Pemilihan Data untuk Membuat Diagram Garis.....	29
Gambar 1.34 Pemilihan Diagram Garis.....	29
Gambar 1.35 Menambahkan Informasi Sumbu X dan Sumbu Y.....	30
Gambar 1.36 Tabel Data Diagram Lingkaran.....	30
Gambar 1.37 Pemilihan Data untuk Membuat Diagram Lingkaran	30
Gambar 1.38 Pemilihan Diagram Lingkaran.....	31
Gambar 1.39 Tabel Data Diagram Pencar	31
Gambar 1.40 Diagram Pencar Korelasi Positif.....	31
Gambar 1.41 Diagram Pencar Korelasi Negatif.....	32
Gambar 1.42 Diagram Pencar Tanpa Korelasif.....	32
Gambar 1.43 Pemilihan Data untuk Membuat Diagram Pencar.....	32
Gambar 1.44 Pemilihan Diagram Pencar	32
Gambar 1.45 Menambahkan Informasi pada Diagram Pencar.....	32
Gambar 2.1 Peta Materi Bab II Berpikir Komputasional dalam Algoritma dan Pemrograman.....	40
Gambar 2.2 Kode Blok Program Cek Nilai.....	41
Gambar 2.3 Mainan Robot-Robotan Menggunakan Blok Kayu.....	43
Gambar 2.4 Alur Program Perhitungan x.....	51
Gambar 2.5 Percakapan 1	53
Gambar 2.6 Percakapan 2	54
Gambar 2.7 Percakapan Anto dan Ibu.....	55
Gambar 2.8 Ilustrasi Konsep <i>Library</i> dalam Pemrograman.....	65
Gambar 2.9 <i>Flowchart</i> Proses Pembelian Produk	92
Gambar 2.10 <i>Flowchart</i> Penentuan Bilangan Nol, Positif, atau Negatif.....	93
Gambar 2.11 Program Blok Visual dan <i>Flowchart</i>	94
Gambar 2.12 Kondisi Awal dan Kondisi Akhir Program dalam Scratch	94
Gambar 2.13 <i>Flowchart</i> Menghitung Luas Lingkaran.....	95
Gambar 2.14 Ilustrasi Penelusuran <i>Flowchart</i> di Buku Kerja Siswa	97
Gambar 2.15 <i>Flowchart</i> Menghitung Luas Persegi.....	98
Gambar 2.16 <i>Flowchart</i> Menghitung Luas Permukaan Kubus.....	99
Gambar 2.17 <i>Flowchart</i> Menghitung Luas Lingkaran.....	101
Gambar 2.18 <i>Flowchart</i> Menghitung Luas Persegi.....	105
Gambar 2.19 <i>Flowchart</i> Menghitung Luas Permukaan Kubus.....	106
Gambar 2.20 <i>Flowchart</i> untuk Menghitung Membagi Pembilang dengan Penyebut....	111
Gambar 3.1 Peta Materi Bab III Literasi Digital untuk Produktivitas.....	116
Gambar 3.2 Contoh Ilustrasi Surat Undangan Kegiatan Proyek Kebersihan Sekolah ...	118
Gambar 3.3 Pemrosesan Data pada Komputer	121
Gambar 3.4 Contoh Naskah Dokumen Buku yang Menggunakan Format Data Teks.....	131

Gambar 3.5	Contoh <i>File</i> dalam Format .docx.....	131
Gambar 3.6	Contoh <i>File</i> dalam Format .xlsx.....	132
Gambar 3.7	Contoh <i>File</i> dalam Format .pptx.....	132
Gambar 3.8	Contoh <i>File</i> dalam Format .odt.....	133
Gambar 3.9	Contoh <i>File</i> dalam Format .ods.....	133
Gambar 3.10	Contoh <i>File</i> dalam Format .odp.....	134
Gambar 3.11	Contoh <i>File</i> dalam Format .rtf.....	134
Gambar 3.12	Contoh <i>File</i> dalam Format .pdf.....	135
Gambar 3.13	Contoh <i>File</i> dalam Format .csv.....	135
Gambar 3.14	Contoh <i>File</i> dalam Format .txt.....	136
Gambar 3.15	Contoh Representasi Gambar dalam Bentuk Kode Biner 1 dan 0	136
Gambar 3.16	Contoh <i>File</i> dalam Format .jpg.....	137
Gambar 3.17	Contoh <i>File</i> dalam Format .png.....	138
Gambar 3.18	Contoh <i>File</i> dalam Format .gif.....	139
Gambar 3.19	Contoh <i>File</i> dalam Format .bmp.....	140
Gambar 3.20	Contoh <i>File</i> dalam Format .tiff.....	140
Gambar 3.21	Contoh <i>File</i> dalam Format .svg.....	141
Gambar 3.22	Contoh <i>File</i> dalam Format .raw.....	142
Gambar 3.23	Contoh <i>Icon File</i> dalam Format .heif.....	143
Gambar 3.24	Contoh <i>File</i> dalam Format .psd.....	143
Gambar 3.25	Gelombang Suara dari Analog ke Digital Menjadi Kode Biner	144
Gambar 3.26	Tangkapan Layar Tampilan Antarmuka Perangkat Lunak Pengolah Kata Microsoft Word (kiri) dan Google Docs (kanan)	151
Gambar 3.27	Tangkapan Layar Tampilan Antarmuka Perangkat Lunak Pengolah Kata (spreadsheet) Microsoft Excel dan Google Sheet.....	153
Gambar 3.28	Tangkapan Layar Tampilan Antarmuka Perangkat Lunak Presentasi Microsoft PowerPoint (kiri) dan Google Slides (kanan)	154
Gambar 3.29	Tangkapan Layar Blogger (kiri) dan Wordpress (kanan)	156
Gambar 3.30	Tampilan Halaman Web CapCut	158
Gambar 3.31	Tampilan Halaman Youtube Creators.....	159
Gambar 3.32	Tangkapan Layar Contoh Grafik Rata-Rata Setiap Jenis Sampah yang Dikumpulkan oleh Siswa	168
Gambar 3.33	Contoh Grafik Pemasukan Jumlah Saldo Uang yang Ditabung di Bank Sampah oleh Siswa	168
Gambar 3.34	Presentasi Grafik di Aplikasi Presentasi	169
Gambar 4.1	Peta Materi Bab IV Keamanan Digital	183
Gambar 4.2	<i>Pop-Up</i> Iklan pada <i>Website</i> Ilegal	184
Gambar 4.3	Poster Film Missing (2023).....	186
Gambar 4.4	Ilustrasi Dokter Memberikan Map Data Medis pada Pasien	187

Gambar 4.5	Ilustrasi Buka Kunci Layar dengan Pengenalan Wajah	188
Gambar 4.6	Ilustrasi DNA.....	188
Gambar 4.7	Ilustrasi Map SKCK.....	189
Gambar 4.8	Ilustrasi Nama Anggota Keluarga di Belakang Mobil	190
Gambar 4.9	Ilustrasi Profil Media Sosial Menyertakan Informasi Privat	190
Gambar 4.10	Ilustrasi Virus	196
Gambar 4.11	Ilustrasi <i>Worm</i>	197
Gambar 4.12	Ilustrasi <i>Ransomware</i>	197
Gambar 4.13	Ilustrasi <i>Backdoor</i>	198
Gambar 4.14	Ilustrasi Kuda Troya	198
Gambar 4.15	Ilustrasi <i>Spyware</i>	199
Gambar 4.16	Ilustrasi Ikon Antivirus.....	204
Gambar 4.17	Ilustrasi Ikon <i>Firewall</i>	205
Gambar 4.18	Ilustrasi Ikon <i>Backup File</i>	205
Gambar 4.19	Tampilan Hasil Pengecekan Kekuatan Kata Sandi	216
Gambar 4.20	Ilustrasi Proses Enkripsi dan Dekripsi.....	220
Gambar 4.21	Ilustrasi <i>Caesar's Cipher</i>	221
Gambar 4.22	Ilustrasi Proses Enkripsi <i>Caesar's Cipher</i> Secara Acak.....	222
Gambar 4.23	Ilustrasi Kecemasan Pengguna Media Sosial.....	225
Gambar 4.24	Ilustrasi Membagikan Data Pribadi.....	230
Gambar 4.25	Ilustrasi Penanda Lokasi pada Unggahan di Media Sosial	231
Gambar 4.26	Ilustrasi Berpikir Ulang Sebelum Membagikan Konten	231
Gambar 4.27	Ilustrasi Menerima Permintaan Pertemanan	231
Gambar 4.28	Ilustrasi <i>Phishing</i>	232
Gambar 4.29	Ilustrasi Melaporkan Akun	233





Daftar Tabel

Tabel 1.1	Contoh Data Nilai Siswa	20
Tabel 1.2	Contoh Data Tinggi Badan Siswa	21
Tabel 1.3	Aktivitas Membuat Grafik Sederhana	34
Tabel 1.4	Data Penggunaan Energi Listrik di Rumah	35
Tabel 1.5	Konsumsi Penggunaan Energi Listrik Harian.....	35
Tabel 1.6	Refleksi Pemahaman Belajar Bab I Berpikir Komputasional dalam Analisis Data.....	38
Tabel 2.1	Langkah-Langkah Membuat Modul Program di Scratch.....	44
Tabel 2.2	Contoh <i>Input</i> dan <i>Output</i> Program Penyelesaian Persamaan Linear.....	46
Tabel 2.3	Langkah-Langkah Membuat Modul <i>Linear1</i> dalam Pemrograman Visual Scratch.....	46
Tabel 2.4	Hasil Eksperimen Modul <i>Linear1</i>	51
Tabel 2.5	Hasil Eksperimen Terkait Program $f(x, y)$	52
Tabel 2.6	Tabel Eksperimen Terhadap Parameter Persamaan 3	53
Tabel 2.7	Contoh <i>Input</i> dan <i>Output</i> Program Perhitungan Luas Lingkaran	56
Tabel 2.8	Cara Membuat Modul pada Blockly	57
Tabel 2.9	Langkah Pembuatan Modul Program Luas Lingkaran.....	58
Tabel 2.10	Langkah Pembuatan Modul <i>Tampil</i> untuk Hasil Luas Lingkaran	59
Tabel 2.11	Eksekusi Modul <i>Utama</i> dan Hasil Program	60
Tabel 2.12	Tabel Eksperimen Terhadap Rumus Luas Lingkaran.....	62
Tabel 2.13	Contoh <i>Input</i> dan <i>Output</i> Program Volume Tabung	62
Tabel 2.14	Tabel Analisis <i>Library</i> Modul Program (<i>Function</i> dan <i>Procedure</i>) Bangun Datar	68
Tabel 2.15	Contoh <i>Input</i> dan <i>Output</i> Luas Persegi	68
Tabel 2.16	Contoh <i>Input</i> dan <i>Output</i> Luas Persegi Panjang.....	68
Tabel 2.17	Contoh <i>Input</i> dan <i>Output</i> Luas Segitiga	69
Tabel 2.18	Tabel Eksperimen Terhadap Rumus Luas Persegi.....	73
Tabel 2.19	Tabel Eksperimen Terhadap Rumus Luas Persegi Panjang	73
Tabel 2.20	Tabel Eksperimen Terhadap Rumus Luas Segitiga	73
Tabel 2.21	Contoh <i>Input</i> dan <i>Output</i> Keliling Persegi.....	74
Tabel 2.22	Contoh <i>Input</i> dan <i>Output</i> Keliling Persegi Panjang.....	75
Tabel 2.23	Contoh <i>Input</i> dan <i>Output</i> Keliling Segitiga.....	75
Tabel 2.24	Perbandingan Bahasa Pemrograman Visual Blok dan Bahasa Pemrograman Tekstual	77

Tabel 2.25 Eksperimen Terhadap Rumus Luas Lingkaran Berdasarkan <i>Flowchart</i>	97
Tabel 2.26 Translasi <i>Flowchart</i> ke Kode Pemrograman Visual Blok	101
Tabel 2.27 Translasi Kode Pemrograman Visual Blok ke Pemrograman Teksual.....	103
Tabel 2.28 Translasi Kode Pemrograman Teksual ke Pemrograman Blok.....	108
Tabel 2.29 Translasi Kode Pemrograman Visual Blok ke Pemrograman Teksual.....	108
Tabel 2.30 Refleksi Pemahaman Belajar Bab II Berpikir Komputasional dalam Algoritma dan Pemrograman.....	114
Tabel 3.1 Identifikasi Gangguan Digital dan Solusi <i>Mindfulness</i>	129
Tabel 3.2 Contoh Desain Tabel Pencatatan Tabungan Sampah Siswa SMP Merdeka .	166
Tabel 3.3 Contoh Hasil Pengisian ke Tabel Pencatatan Tabungan Sampah Siswa SMP Merdeka dalam Perangkat Lunak Lembar Kerja.....	167
Tabel 3.4 Laporan Harian Masalah Komputer dan Cara Penanganannya.....	178
Tabel 3.5 Refleksi Pemahaman Belajar Bab III Literasi Digital untuk Produktivitas ..	180
Tabel 4.1 Aktivitas Analisis Kasus Kebocoran Data.....	192
Tabel 4.2 Dampak Pengabaian Keamanan Jaringan.....	210
Tabel 4.3 Asesmen Informasi Privat atau Publik.....	236
Tabel 4.4 Asesmen <i>Caesar's Cipher</i>	238
Tabel 4.5 Asesmen Persoalan <i>Mindfulness</i>	238
Tabel 4.6 Refleksi Pemahaman Belajar Bab IV Keamanan Digital	240

Petunjuk Penggunaan Buku



Buku teks Informatika ini disusun untuk membantu kamu dalam proses pembelajaran, baik di kelas maupun secara mandiri. Buku ini tidak hanya memuat konsep dasar dan lanjutan tentang informatika, tetapi juga dilengkapi aktivitas yang dirancang untuk menumbuhkan rasa ingin tahu dan kreativitas. Melalui perpaduan teori dan praktik, kamu akan didorong untuk belajar lebih aktif, kritis, dan mandiri dalam menggali dan memahami materi yang dipelajari.

Struktur buku disusun secara sistematis agar mudah dipahami. Mulai dari pertanyaan pemantik, tujuan pembelajaran, hingga refleksi, setiap bagian menjadi panduan langkah demi langkah untuk memahami materi, mengasah keterampilan, dan menilai pemahaman diri. Buku ini diharapkan dapat membantumu, tidak hanya memahami Informatika secara konseptual tetapi juga mampu mengaitkannya dengan kehidupan sehari-hari serta perkembangan teknologi di sekitarmu.

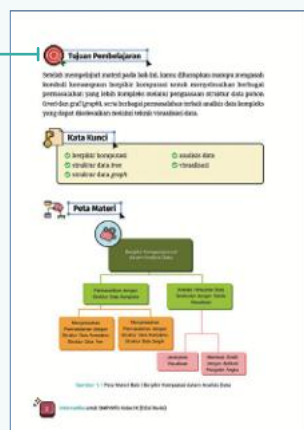


Pertanyaan Pemantik

Memuat pertanyaan sederhana atau situasi sehari-hari untuk membangkitkan rasa ingin tahumu dan menghubungkan materi dengan pengalaman nyata sebelum memulai pembelajaran.

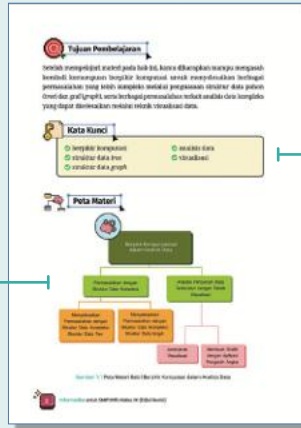
Tujuan Pembelajaran

Berisi tujuan dan kompetensi utama yang harus dicapai dalam satu bab. Setelah mempelajari bab tersebut, kamu diharapkan dapat menguasai materi dan menerapkannya baik di sekolah maupun dalam kehidupan sehari-hari.



Peta Materi

Menyajikan gambaran umum isi bab untuk menunjukkan keterkaitan antartopik yang akan dipelajari.



Kata Kunci

Berisi istilah-istilah penting yang sering muncul dalam materi.

Berpikir kritis adalah cara berfikir yang baik. Kita harus bisa membedakan mana yang benar dan mana yang salah. Kita harus bisa membedakan mana yang penting dan mana yang tidak penting. Kita harus bisa membedakan mana yang benar dan mana yang salah. Kita harus bisa membedakan mana yang penting dan mana yang tidak penting.

Siap-Siap Belajar

Komputer akan kita gunakan untuk membantu menyelesaikan berbagai permasalahan, seperti menghitung, mencatat, dan lain-lain. Kita akan menggunakan komputer untuk membantu kita dalam menyelesaikan berbagai permasalahan. Kita akan menggunakan komputer untuk membantu kita dalam menyelesaikan berbagai permasalahan. Kita akan menggunakan komputer untuk membantu kita dalam menyelesaikan berbagai permasalahan.

1. **Alasan**
Alasan adalah alasan yang paling penting yang kita gunakan untuk menjelaskan sesuatu. Alasan adalah alasan yang paling penting yang kita gunakan untuk menjelaskan sesuatu. Alasan adalah alasan yang paling penting yang kita gunakan untuk menjelaskan sesuatu.
2. **Keuntungan**
Keuntungan adalah keuntungan yang kita dapatkan dari sesuatu. Keuntungan adalah keuntungan yang kita dapatkan dari sesuatu. Keuntungan adalah keuntungan yang kita dapatkan dari sesuatu.

Siap-Siap Belajar

Bagian pengantar berisi pembahasan sederhana yang menghubungkan peristiwa sehari-hari dengan inti materi, untuk membentuk pemahaman awal sebelum mempelajari materi.

Ayo, Mengingat Kembali!

Pertanyaan yang membantu peserta didik meninjau kembali pengetahuan sebelumnya yang berkaitan dengan materi yang akan dipelajari.

Apakah kamu pernah mendengar kata 'peristiwa'?

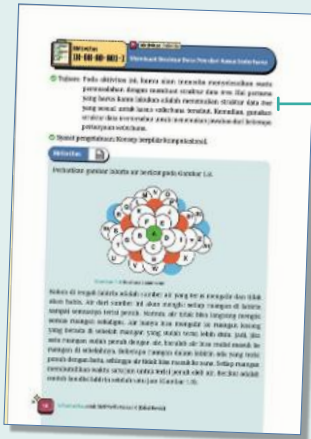
Peristiwa adalah kejadian yang terjadi di sekitar kita. Peristiwa adalah kejadian yang terjadi di sekitar kita. Peristiwa adalah kejadian yang terjadi di sekitar kita.

Ayo, Mengingat Kembali!

Sebelum mempelajari lebih lanjut, mari kita ingat kembali pengetahuan yang sudah kita miliki.

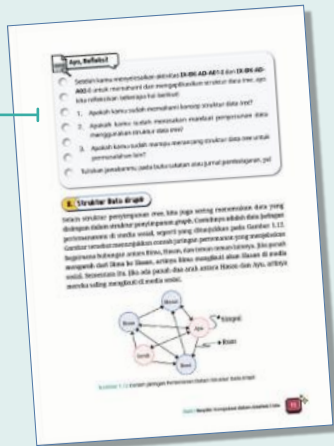
1. Bagaimana peristiwa terjadi?
2. Apa saja peristiwa yang terjadi di sekitar kita?
3. Bagaimana peristiwa terjadi di sekitar kita?
4. Apa saja peristiwa yang terjadi di sekitar kita?





Aktivitas

Berisi latihan soal atau aktivitas yang dikerjakan secara mandiri atau kelompok untuk menguji pemahaman terhadap materi yang sudah dipelajari.



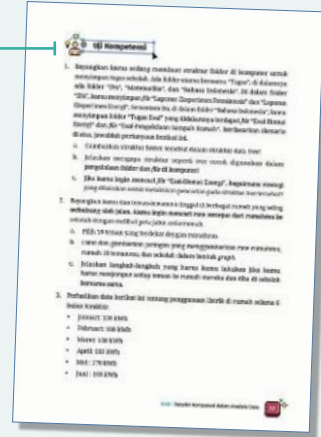
Ayo, Refleksi!

Mengajak peserta didik untuk menilai kembali apa yang telah dipelajari dan dipahami pada setiap subbab.



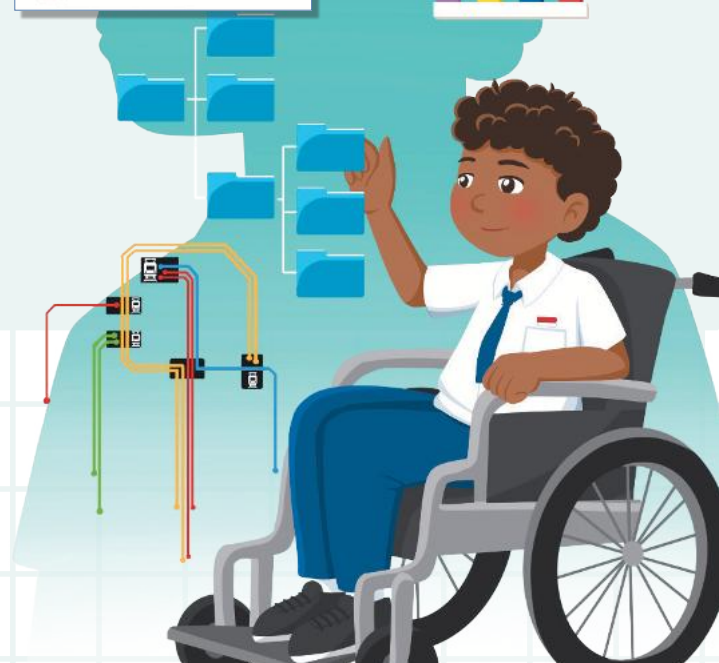
Uji Kompetensi

Bertujuan untuk mengetahui sejauh mana kamu telah mencapai tujuan pembelajaran, sekaligus mengidentifikasi kekuatan dan kelemahan dirimu, serta memberi umpan balik untuk memperbaiki proses belajar.



Refleksi

Membantumu menilai sejauh mana pemahamanmu terhadap materi. Melalui kegiatan refleksi, kamu dilatih berpikir kritis, mengenali kekurangan, serta menemukan strategi belajar yang lebih efektif.



KEMENTERIAN PENDIDIKAN DASAR DAN MENENGAH
REPUBLIK INDONESIA, 2025

Informatika untuk SMP/MTs Kelas IX (Edisi Revisi)

Penulis : Erlangga, Erna Piantari, dan Khairur Rosyid
ISBN : 978-634-00-3299-4 (jil.3 PDF)



Berpikir Komputasional dalam Analisis Data



“

Pernahkah kamu melihat bagan peta jaringan transportasi? Mengapa hubungan antartitik tersebut disajikan dalam bentuk cabang atau garis? Apakah cara penyajian tersebut membantu memahami rute dengan lebih baik?

”



Tujuan Pembelajaran

Setelah mempelajari materi pada bab ini, kamu diharapkan mampu mengasah kembali kemampuan berpikir komputasional untuk menyelesaikan berbagai permasalahan yang lebih kompleks melalui penguasaan struktur data pohon (*tree*) dan graf (*graph*), serta berbagai permasalahan terkait analisis data kompleks yang dapat diselesaikan melalui teknik visualisasi data.

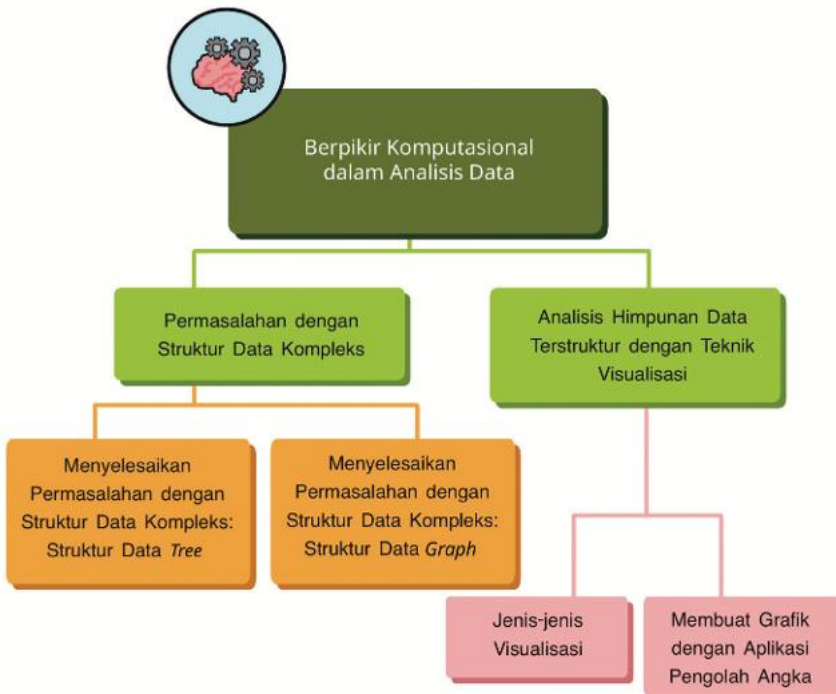


Kata Kunci

- ✓ berpikir komputasional
- ✓ struktur data *tree*
- ✓ struktur data *graph*
- ✓ analisis data
- ✓ visualisasi



Peta Materi



Gambar 1.1 Peta Materi Bab I Berpikir Komputasional dalam Analisis Data

Berpikir komputasional tentu bukan hal baru bagimu. Di kelas VII dan VIII, kamu telah mempelajari dan menyelesaikan berbagai permasalahan menggunakan konsep ini. Sebagai pengingat, pada bab ini kamu akan mengulas kembali penggunaan berpikir komputasional dengan permasalahan yang lebih kompleks dan melibatkan struktur penyimpanan data yang lebih rumit. Kamu juga akan berkenalan dengan beberapa struktur data seperti *tree* dan *graph*, serta mempelajari proses analisis data melalui teknik visualisasi.



Komputer biasa kita gunakan untuk membantu menyelesaikan berbagai permasalahan, seperti menghitung, mencari informasi, atau membuat gambar. Sebelum hal tersebut dapat dilakukan, kita perlu memahami permasalahan terlebih dahulu dan merancang solusi yang tepat agar komputer dapat menjalankan solusi tersebut. Salah satu cara yang dapat kita gunakan untuk menemukan solusi tersebut adalah dengan menggunakan konsep berpikir komputasional. Konsep ini merupakan cara berpikir logis yang dapat membantu kita dalam memahami masalah dan mengembangkan solusi. Sehingga solusi tersebut lebih mudah dipahami dan dapat dijalankan oleh komputer. Dalam konsep berpikir komputasional, kita mengenal empat *cornerstone* atau batu penyangga (Gambar 1.2) yang menjadi kunci untuk membantu kita dalam memahami dan menyelesaikan masalah, yaitu:

1. Abstraksi

Abstraksi adalah proses menyaring detail yang tidak relevan dan berfokus pada informasi penting dari suatu kondisi atau permasalahan kompleks. Sebagai contoh, saat menganalisis pemakaian listrik di sekolah, kita hanya memusatkan perhatian pada data utama seperti jumlah siswa dan total konsumsi listrik per kelas, alih-alih mempertimbangkan variabel yang tidak relevan seperti suhu ruangan atau warna cat kelas.

2. Dekomposisi

Dekomposisi adalah proses memecah masalah kompleks menjadi beberapa bagian kecil yang lebih sederhana agar lebih mudah diselesaikan. Contohnya, masalah besar seperti penentuan hubungan antara konsumsi listrik dan jumlah siswa dipecah menjadi beberapa langkah kecil, antara lain:



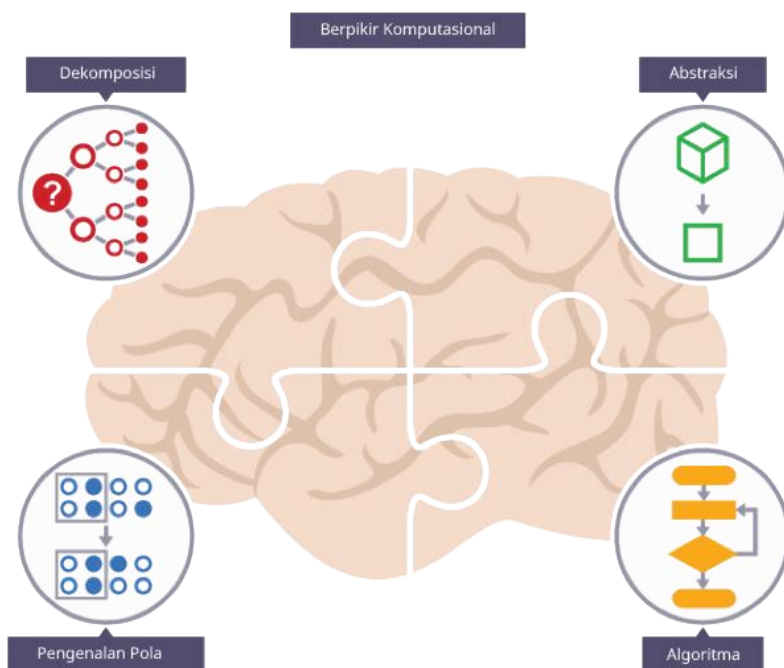
- mengumpulkan data,
- membersihkan data,
- menyusun data kedalam bentuk tabel,
- membuat grafik, dan
- menarik kesimpulan.

3. Pengenalan pola

Pengenalan pola adalah proses mencari kesamaan pola dalam berbagai permasalahan untuk menemukan solusi. Sebagai contoh, hasil analisis data grafik menunjukkan bahwa peningkatan jumlah siswa di suatu kelas berbanding lurus dengan peningkatan konsumsi listriknya.

4. Penyusunan algoritma

Penyusunan algoritma adalah proses menyusun langkah-langkah logis dan terstruktur untuk menyelesaikan suatu permasalahan. Dengan algoritma, serangkaian instruksi dapat dijalankan secara berulang untuk menghasilkan solusi yang konsisten. Contohnya adalah menyusun tahapan sistematis dalam melakukan analisis data.



Gambar 1.2 Empat Kunci dalam Konsep Berpikir Komputasional

Sumber : Diolah dari BBC Bitesize (2025)

Dalam bab ini, kita akan mencoba menggunakan konsep berpikir komputasional untuk menyelesaikan berbagai permasalahan kompleks yang melibatkan pengelolaan data dan proses analisis data. Beberapa permasalahan kompleks biasanya terjadi karena bentuk, struktur data yang kompleks, atau jumlah data yang besar. Di kelas VIII, kamu sudah berkenalan dengan struktur data *stack* atau tumpukan. Apakah kamu masih ingat, apa yang dimaksud dengan struktur data atau bentuk data? Struktur data adalah cara kita mengatur dan menyimpan data, agar lebih mudah digunakan. Sebelum kita belajar lebih lanjut, coba kamu perhatikan benda-benda yang ada di sekitarmu. Pernahkah kamu berurusan dengan masalah yang melibatkan data atau informasi? Bagaimana cara data itu disusun, disimpan, atau ditampilkan? Amati dan catat temuanmu di buku catatanmu masing-masing!



Gambar 1.3 Ilustrasi Penyimpanan Alat Makan di Dapur

Untuk membantumu memahami konsep struktur data, ayo kita bayangkan atau ingat kembali seperti apa dapur rumahmu. Coba perhatikan benda-benda atau data apa saja yang ada di sana. Misalnya, kamu bisa melihat alat makan seperti piring, gelas, atau sendok. Perhatikan bagaimana ibumu menyimpan alat-alat makan tersebut. Apakah ada cara khusus atau aturan tertentu dalam menyusunnya? Coba tuliskan apa yang kamu temukan:

1. Apa saja alat makan yang disimpan?
2. Bagaimana cara penyimpanannya?
3. Apakah ada aturan penyusunan yang menarik?
4. Berapa banyak jumlah alat makan yang disimpan?
5. Hal apa saja yang menjadi pertimbangan Ibu dalam membuat aturan penyimpanan alat makan?
6. Apa tujuan ibu menyimpan alat makan dengan aturan tertentu?

Apa yang akan terjadi jika semua benda atau data di dapur tidak disimpan sesuai aturan yang berlaku? Misalnya, kita lihat piring atau gelas yang sering digunakan disusun dengan cara berjajar menyamping di atas rak.

Dari hasil pengamatan tersebut, apakah kamu sudah menyadari bahwa aturan dalam penyimpanan data sangat penting untuk mempermudah pengelolaan data, khususnya saat mencari data tertentu? Bayangkan jika piring yang sering digunakan disimpan dengan cara ditumpuk. Masalah apa yang mungkin terjadi? Nah, cara menyimpan data dengan aturan tertentu bisa diibaratkan seperti struktur data. Semakin banyak kita mengenal cara-cara menyimpan dan mengatur data, maka semakin mudah juga kita menyelesaikan berbagai pekerjaan. Pada bab ini, kita akan belajar tentang struktur data pohon (*tree*) dan graf (*graph*). Selain itu, kita juga akan mempelajari cara menganalisis data yang sudah tersusun rapi, agar bisa dipahami dan digunakan dengan lebih mudah melalui teknik visualisasi.

*



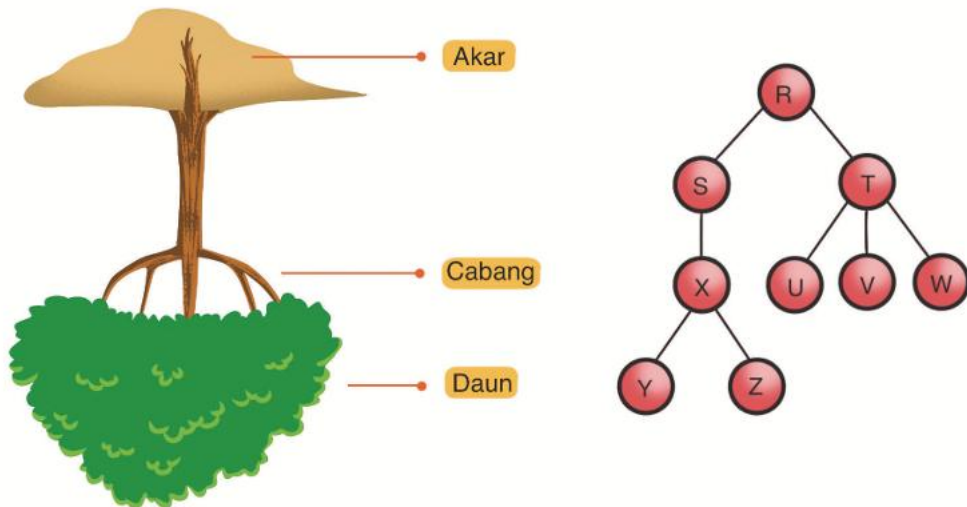
Ayo, Mengingat Kembali!

Sebelum membahas lebih lanjut mengenai pembahasan pada bab ini, jawablah pertanyaan-pertanyaan di bawah ini:

1. Mengapa data perlu diolah terlebih dahulu sebelum digunakan?
2. Masih ingatkah struktur data yang sebelumnya pernah dipelajari, misalnya struktur data tumpukan (*stack*)? Bagaimana peran struktur data dalam membantu pengolahan data?
3. Jika kita ingin menggambarkan silsilah keluarga, struktur dan cara penyusunan data seperti apa yang paling tepat digunakan?
4. Jika memiliki data nilai siswa dari berbagai mata pelajaran, bagaimana caramu mengelola dan menampilkannya selain menggunakan tabel agar data tersebut lebih mudah dipahami? Berikan penjelasan singkat.

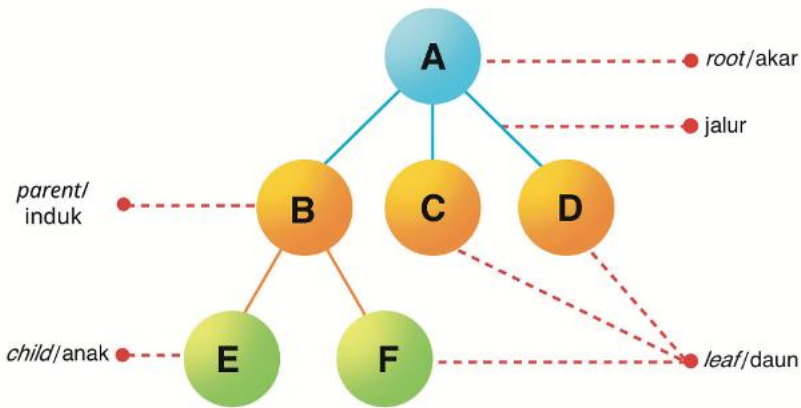
A. Struktur Data *Tree*

Struktur data *tree* sering disebut sebagai struktur data pohon. Seperti namanya, struktur data ini menyediakan bentuk penyimpanan data layaknya sebuah pohon yang memiliki akar, batang, cabang, dan daun. Untuk memudahkan kamu membayangkan bagaimana struktur data ini, perhatikan gambar pohon terbalik pada Gambar 1.4.



Gambar 1.4 Ilustrasi Perbandingan Struktur Pohon dan Struktur Data *Tree* (Pohon)

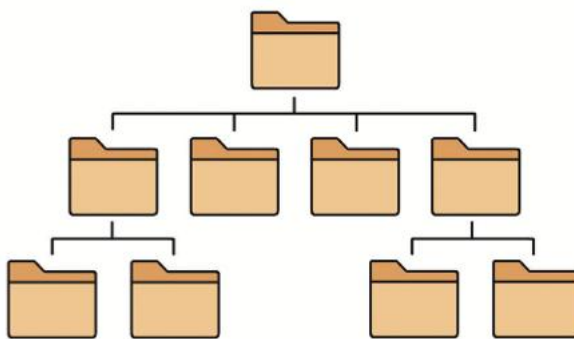
Jika kamu perhatikan Gambar 1.4, struktur pohon dalam Informatika terlihat mirip dengan pohon sungguhan, tetapi posisinya terbalik. Pada pohon biasa (sebelah kiri), akar berada di bawah dan daun ada di atas. Sementara itu, dalam struktur data *tree*, justru akar atau yang lebih dikenal sebagai *root* berada di atas, lalu cabang serta daunnya menyebar ke bawah. Agar lebih jelas, mari kita uraikan kembali struktur data *tree* pada Gambar 1.5. Secara umum, struktur data *tree* terdiri dari simpul yang saling dihubungkan dengan jalur. Simpul yang paling awal disebut sebagai akar (*root*), sedangkan simpul yang paling ujung disebut sebagai daun (*leaf*). Setiap simpul, kecuali akar, memiliki satu simpul yang berperan sebagai induk (*parent*). Contoh pada Gambar 1.5, simpul B adalah *parent* dari simpul E. Sebaliknya, simpul E adalah simpul anak (*child*) dari simpul B. Simpul yang tidak punya *child* kita sebut sebagai *leaf*. Pada Gambar 1.5, contoh simpul *leaf* adalah C, D, E, dan F.



Gambar 1.5 Struktur Data *Tree*

Struktur data *tree* dalam Informatika digunakan untuk menyimpan data yang memiliki hubungan bertingkat atau disebut juga struktur hierarki. Misalnya, dalam sebuah keluarga, ada kakek-nenek di tingkat paling atas, lalu orang tua, dan terakhir anak-anak. Nah, struktur seperti ini cocok disimpan dalam bentuk pohon karena setiap data bisa dihubungkan *parent* dan *child*.

Penyimpanan data dalam bentuk pohon, mempermudah kita untuk melakukan pencarian. Sebagai contoh dalam kehidupan sehari-hari adalah struktur data folder komputer. Satu folder utama dalam komputer dapat menyimpan banyak subfolder, dan subfolder tersebut dapat berisi folder lagi. Sistem penyimpanan seperti ini akan memudahkan kita dalam mengelola dan mencari berkas (*file*) atau folder yang disimpan.



Gambar 1.6 Contoh Struktur Penyimpanan Folder di Komputer

Selain mempermudah proses pencarian, penyimpanan data dengan struktur data *tree* juga mempermudah kita untuk memahami susunan dan hubungan

antardata. Struktur organisasi OSIS di sekolah adalah contoh penggunaan struktur data *tree* yang biasa kita temui dalam kehidupan sehari-hari (Gambar 1.7). Dalam struktur data ini, data disimpan dalam sebuah hirarki bertingkat. Oleh karena itu, kita dapat memahami bahwa kepala sekolah memiliki tingkat tertinggi di penyimpanan tersebut. Sedangkan sekretaris, sekretaris bidang, dan bendahara bertanggung jawab secara langsung kepada ketua OSIS.



Gambar 1.7 Contoh Struktur Data *Tree* pada Struktur Organisasi Sekolah

Bisa dibayangkan apa yang terjadi jika data struktur organisasi OSIS sekolah tidak disimpan dalam struktur data *tree*. Tentu saja, kita akan kesulitan menentukan bagaimana hubungan antardata yang tersimpan. Sebagai contoh, kita akan kehilangan informasi yang menjelaskan hubungan antara ketua OSIS dan kepala sekolah. Siapa yang memiliki level kedudukan paling tinggi dalam organisasi. Dari contoh tersebut, bisakah kamu menyebutkan seperti apa karakteristik data yang sebaiknya disimpan dalam struktur data *tree*?

Ayo, Mengamati!

Setelah kamu memahami contoh penyimpanan data:

1. Carilah beberapa contoh penyimpanan data yang biasa disimpan dalam struktur data *tree*.
2. Amati dari contoh yang kamu temukan, kemudian identifikasi dampak yang mungkin terjadi jika data-data tersebut tidak disusun menggunakan struktur data *tree*.



Aktivitas

IK-BK-AD-A01-I



Aktivitas Individu

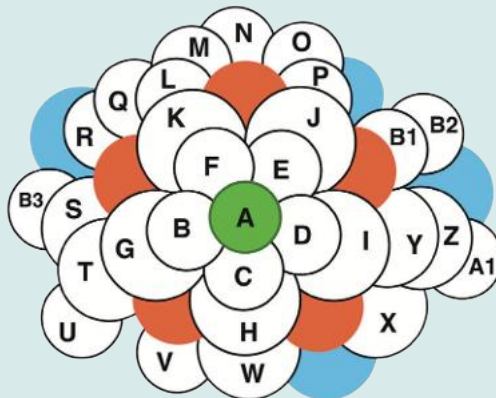
Membuat Struktur Data *Tree* dari Kasus Sederhana

- ✔ Tujuan: Pada aktivitas ini, kamu akan mencoba menyelesaikan suatu permasalahan dengan membuat struktur data *tree*. Hal pertama yang harus kamu lakukan adalah menemukan struktur data *tree* yang sesuai untuk kasus sederhana tersebut. Kemudian, gunakan struktur data *tree* tersebut untuk menemukan jawaban dari beberapa pertanyaan sederhana.
- ✔ Syarat pengetahuan: Konsep berpikir komputasional.

Aktivitas



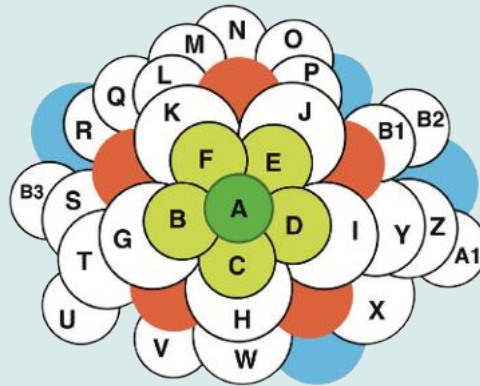
Perhatikan gambar labirin air berikut pada Gambar 1.8.



Gambar 1.8 Ilustrasi Labirin Air

Kolom di tengah labirin adalah sumber air yang terus mengalir dan tidak akan habis. Air dari sumber ini akan mengisi setiap ruangan di labirin sampai semuanya terisi penuh. Namun, air tidak bisa langsung mengisi semua ruangan sekaligus. Air hanya bisa mengalir ke ruangan kosong yang berada di sebelah ruangan yang sudah terisi lebih dulu. Jadi, jika satu ruangan sudah penuh dengan air, barulah air bisa mulai masuk ke ruangan di sebelahnya. Beberapa ruangan dalam labirin ada yang terisi penuh dengan batu, sehingga air tidak bisa masuk ke sana. Setiap ruangan membutuhkan waktu satu jam untuk terisi penuh oleh air. Berikut adalah contoh kondisi labirin setelah satu jam (Gambar 1.9).





Gambar 1.9 Ilustrasi Labirin Air Setelah Satu Jam

Coba perhatikan gambar labirin tersebut dan jawablah pertanyaan berikut.

1. Dari kasus labirin tersebut, buatlah struktur data *tree* yang menggambarkan aliran air.

Petunjuk: Pembuatan struktur data *tree* dimulai dari sumber air utama yaitu A. Simpul anak dari simpul A adalah simpul lain yang dapat terisi air dari simpul A, yaitu B,C,D,E, dan F. Selanjutnya simpul anak dari simpul C adalah H, karena simpul C akan mengalirkan air ke simpul H. Lakukan proses ini sampai selesai.

2. Berdasarkan gambar struktur *tree* yang sudah dibuat, berapa lama waktu yang dibutuhkan sehingga semua ruang kosong dapat terisi air?
3. Labirin mana yang paling terakhir terisi air?
4. Telusuri struktur data *tree* untuk menentukan jalur labirin mana yang harus dialiri air, agar labirin B3 terisi penuh?

Ayo, Berpikir Kritis!

Setelah membuat struktur data *tree* dari permasalahan labirin air, coba pikirkan dan tuliskan keuntungan penggunaan struktur data *tree* dalam penyimpanan data!



Aktivitas
IK-BK-AD-A02-I



Aktivitas Individu

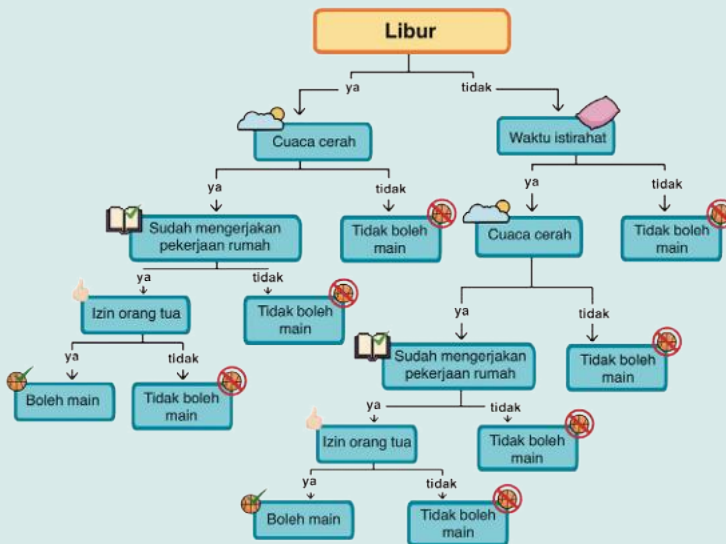
Melakukan Penelusuran Aturan pada Struktur Data Tree

- ✓ Tujuan: Pada aktivitas ini, kamu akan memahami pemanfaatan lain dalam struktur data *tree*, yaitu menyajikan aturan. Selain itu, kamu akan berlatih menggunakan struktur data *tree* untuk menentukan suatu kondisi berdasarkan aturan yang tersimpan di dalamnya.
- ✓ Syarat pengetahuan: Konsep berpikir komputasional

Aktivitas 1



Dalam informatika, penyimpanan data menggunakan struktur data *tree* juga biasa digunakan untuk merepresentasikan aturan untuk memenuhi kondisi tertentu. Dalam hal ini sekumpulan aturan adalah data yang harus dikelola. Penyimpanan aturan dengan struktur data *tree* akan mempermudah penelusuran aturan yang disimpan. Untuk lebih memahami bagaimana aturan dapat disimpan dalam struktur data *tree*, amati Gambar 1.10 dibawah ini. Gambar tersebut merepresentasikan aturan untuk menentukan kondisi “boleh main” atau “tidak boleh main”. Dengan memperhatikan struktur data *tree* yang diberikan, tentukan kondisi yang berlaku: boleh main atau tidak boleh main.



Gambar 1.10 Contoh Representasi Aturan Permasalahan Jadwal Bermain dalam Struktur Data Tree

Selanjutnya, ayo kita coba telusuri aturan yang disimpan dalam *tree* untuk menentukan kondisi yang terpenuhi dari pernyataan berikut:

Pernyataan 1 : Hari ini adalah Kamis sore dan cuacanya cerah. Roni sudah pulang sekolah dan memastikan dirinya sudah mengerjakan tugas. Apakah Roni boleh bermain atau tidak?

Pernyataan 2 : Desi sudah meminta izin kepada orang tua untuk bermain pada hari Minggu, tetapi hujan lebat. Apakah Desi boleh bermain atau tidak?

Pernyataan 3 : Jika hari ini adalah hari libur yang cerah dan orang tua sudah mengizinkan Amir untuk bermain, tetapi Amir belum mengerjakan pekerjaan rumah. Apakah Amir boleh bermain atau tidak?

Untuk menentukan kondisi boleh bermain atau tidak, coba identifikasi terlebih dahulu kondisi mana yang harus dicek dari setiap pernyataan tersebut.

Contoh:

Kondisi pada pertanyaan 1

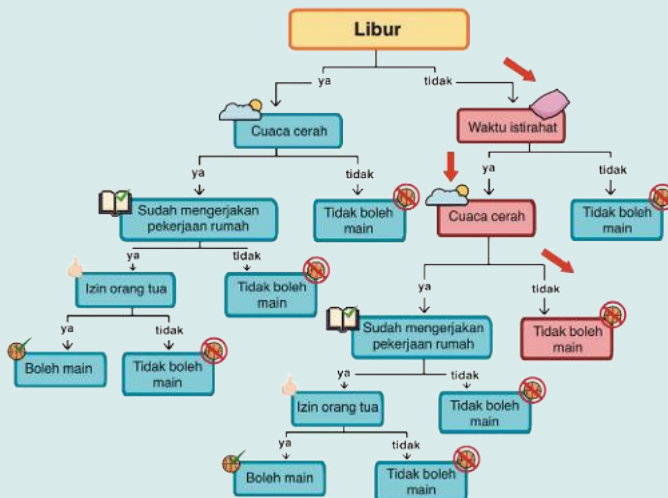
Hari libur: tidak

Mengerjakan tugas: ya

Cuaca cerah: tidak

Waktu istirahat: ya

Kemudian telusuri *tree* sesuai kondisi berikut (Gambar 1.11).



Gambar 1.11 Contoh Representasi Aturan Permasalahan Jadwal Bermain Dalam Struktur Data *Tree*

Hasil penelusuran untuk pernyataan 1, menunjukkan kondisi yang terpenuhi adalah “**tidak boleh main**”. Selanjutnya, tentukan kondisi yang terpenuhi untuk pernyataan 2 dan 3 berdasarkan contoh pernyataan 1.

Ayo, Bereksplorasi!

Setelah menelusuri struktur data *graph* untuk membuat keputusan berdasarkan aturan untuk kondisi “boleh bermain atau tidak”, coba kembangkan struktur data serupa untuk menentukan cara terbaik dalam menangani sampah sekolah agar ramah lingkungan. Contoh aturan yang dapat disusun dalam struktur data *tree* adalah sebagai berikut:

1. Apakah sampahnya organik?
 - Ya → Apakah bisa dijadikan kompos?
 - Ya → Masukkan ke komposter.
 - Tidak → Buang ke tempat sampah organik.
 - Tidak → Lanjut ke pertanyaan berikutnya.
2. Apakah sampahnya anorganik?
 - Ya → Apakah bisa didaur ulang (plastik, kertas, logam)?
 - Ya → Pisahkan untuk didaur ulang.
 - Tidak → Masukkan ke tempat sampah anorganik.
 - Tidak → lanjut ke pertanyaan berikutnya
3. Apakah sampah berbahaya (baterai, obat kadaluarsa)?
 - Ya → Simpan di wadah khusus dan bawa ke tempat penampungan limbah B3.
 - Tidak → Buang sesuai jenisnya.

Kembangkan struktur data *tree* dan modifikasi aturannya sesuai dengan aturan dan kondisi yang berlaku di sekolahmu.

Ayo, Berpikir Kritis!

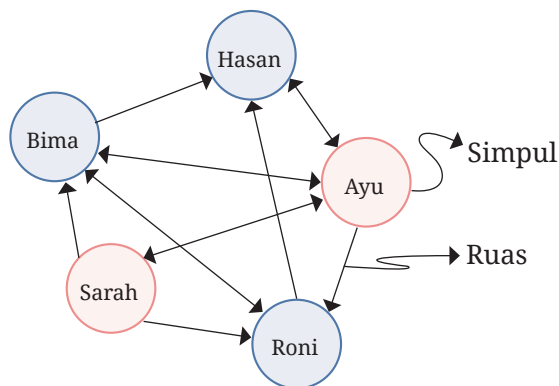
Setelah mengembangkan struktur data *tree* untuk menentukan strategi penanganan sampah sekolah yang ramah lingkungan, coba renungkan pengalamammu. Bagaimana perasaan kamu ketika membuat keputusan? Dari pengalaman tersebut, coba tuliskan manfaat penyusunan data menggunakan struktur data *tree*.

Ayo, Refleksi!

- Setelah kamu menyelesaikan aktivitas **IX-BK-AD-A01-I** dan **IX-BK-AD-A02-I** untuk memahami dan mengaplikasikan struktur data *tree*, ayo kita refleksikan beberapa hal berikut!
- 1. Apakah kamu sudah memahami konsep struktur data *tree*?
- 2. Apakah kamu sudah merasakan manfaat penyusunan data menggunakan struktur data *tree*?
- 3. Apakah kamu sudah mampu merancang struktur data *tree* untuk permasalahan lain?
- Tuliskan jawabanmu pada buku catatan atau jurnal pembelajaran, ya!

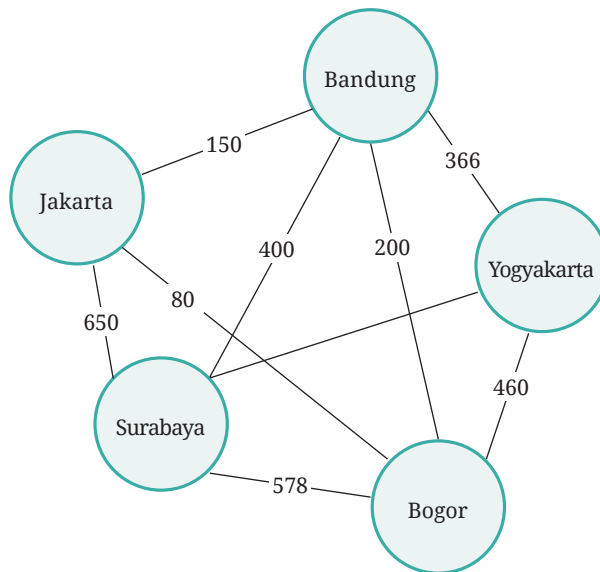
B. Struktur Data *Graph*

Selain struktur penyimpanan *tree*, kita juga sering menemukan data yang disimpan dalam struktur penyimpanan *graph*. Contohnya adalah data jaringan pertemananmu di media sosial, seperti yang ditunjukkan pada Gambar 1.12. Gambar tersebut menunjukkan contoh jaringan pertemanan yang menjelaskan bagaimana hubungan antara Bima, Hasan, dan teman-teman lainnya. Jika panah mengarah dari Bima ke Hasan, artinya Bima mengikuti akun Hasan di media sosial. Sementara itu, jika ada panah dua arah antara Hasan dan Ayu, artinya mereka saling mengikuti di media sosial.



Gambar 1.12 Contoh Jaringan Pertemanan Dalam Struktur Data *Graph*

Dalam struktur data *graph*, kita mengenal istilah **simpul** dan **ruas**. Simpul adalah sesuatu yang dihubungkan, sedangkan ruas adalah penghubung antarsimpul. Pada contoh sebelumnya, nama-nama dalam jaringan pertemanan berperan sebagai simpul, sedangkan ruas digambarkan oleh tanda panah yang menunjukkan adanya hubungan antarsimpul. Contoh data lain yang biasa disajikan dalam struktur data *graph* adalah jalur transportasi (Gambar 1.13). Berbeda dengan contoh sebelumnya, pada contoh kedua setiap jalur yang menghubungkan antarkota memiliki nilai. Misalnya, jarak transportasi dari Jakarta ke Bandung adalah 150 km.



Gambar 1.13 Contoh Representasi Jalur Transportasi pada Struktur Data *Graph*

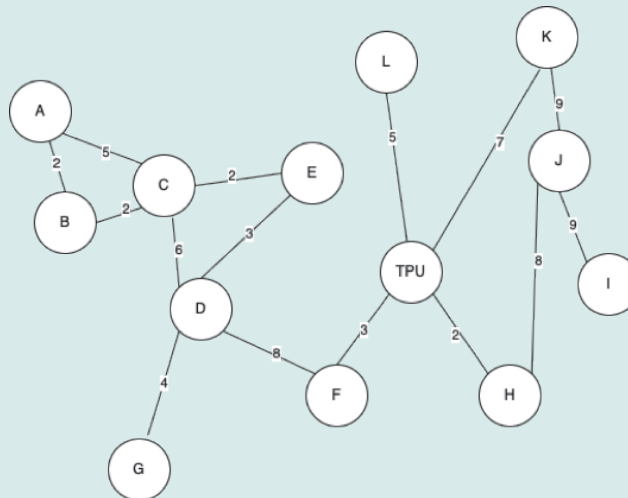
Ayo, Mengamati!

1. Dari dua contoh sebelumnya, kamu bisa melihat bahwa struktur data *graph* berbeda dengan struktur data *tree*. Menurutmu, apa perbedaan yang paling jelas di antara keduanya?
2. Pernahkah kamu menemukan masalah dalam kehidupan sehari-hari yang melibatkan data saling terhubung seperti pada struktur *graph*?



- ✔ Tujuan: Pada aktivitas kali ini, kamu diharapkan dapat menganalisis kasus keamanan jaringan nirkabel pada ruang publik.
- ✔ Syarat pengetahuan: Konsep berpikir komputasional.

Perhatikan struktur data *graph* yang disajikan pada Gambar 1.14 di bawah ini. Struktur data tersebut menggambarkan rute sebuah truk pengangkut sampah dengan kapasitas medium. Truk tersebut harus mengangkut sampah dari setiap tempat penampungan (*shelter*) A, B, C, D, E, F, G, H, I, J, K, dan L menuju Tempat Pembuangan Akhir (TPA). Jarak setiap *shelter* ditunjukkan dengan nilai yang tertulis pada setiap jalur. Karena keterbatasan bahan bakar, truk tersebut hanya mampu mengangkut sampah dari *shelter* yang berjarak tidak lebih dari 18 km dari TPA. Dengan ketentuan ini, *shelter* mana yang tidak dapat dijangkau oleh truk tersebut melalui jalur manapun?



Gambar 1.14 Contoh Representasi Rute Truk Pengangkut Sampah dalam *Graph*

Petunjuk

Lakukan penelusuran mulai dari TPA, kemudian ke *shelter* terdekat. Hitung jarak maksimal yang bisa ditelusuri dari TPA ke *shelter* terdekat, selanjutnya ke setiap *shelter* lain yang terhubung. Hentikan penelusuran jika jarak totalnya sudah melebihi 18 km dari TPA.



Ayo, Berpikir Kritis!

1. Ketika melakukan penelusuran rute dalam struktur data *graph*, apa keuntungan menyimpan data jalur transportasi dalam struktur data *graph*?
2. Jika kamu membuat *graph* untuk lingkungan sekitar rumah atau sekolahmu, simpul dan ruas apa saja yang akan kamu buat?
3. Apakah konsep *graph* bisa digunakan untuk hal lain selain menentukan rute? Coba berikan contohnya!



Aktivitas

IX-BK-AD-B02-I



Aktivitas Individu

Membuat Struktur Data *Graph* untuk Permasalahan Sederhana

- ✔ Tujuan : Pada aktivitas ini, kamu diharapkan dapat memahami konsep penyimpanan struktur data *graph* secara mendalam dan mampu membuat struktur data *graph* untuk permasalahan sederhana.
- ✔ Syarat pengetahuan: Konsep berpikir komputasional.

Pada aktivitas ini, kamu akan mencoba menyelesaikan permasalahan lain menggunakan struktur data *graph*. Bacalah cerita berikut dengan saksama.

Sebuah gedung ajaib memiliki beberapa kamar yang dapat mengubah satu binatang menjadi binatang lainnya. Berikut adalah daftar kamar dan perubahan yang dapat dilakukan.

1. Kamar A: kucing -> ikan atau anjing -> kodok
2. Kamar B: ikan -> kura-kura atau burung -> semut
3. Kamar C: semut -> anjing atau burung -> kodok
4. Kamar D: kucing -> kura-kura atau kucing -> semut
5. Kamar E: kucing -> anjing atau burung -> kelinci
6. Kamar F: burung -> anjing atau semut -> ikan
7. Kamar G: semut -> burung atau kucing -> kodok
8. Kamar H: kucing -> kodok atau kura-kura -> ikan

Jika Amir ditugaskan untuk mengubah kucing menjadi kelinci, maka ruang mana saja yang harus dilewati Amir dalam gedung ajaib tersebut? Untuk menjawab permasalahan ini, kamu bisa membuat struktur data *graph*.

Petunjuk

Ketika membuat struktur data *graph*, kita mengenal simpul dan ruas. Simpul adalah suatu kondisi dalam struktur *graph*, sedangkan ruas adalah penghubung antarsimpul. Dalam kasus ini simpul (kondisi) adalah binatang, sedangkan ruas (penghubung) adalah kamar yang dapat menghubungkan perubahan dari satu kondisi (binatang) menjadi kondisi lain (binatang lain). Untuk menjawab permasalahan Amir, buatlah *graph* untuk merepresentasikan aturan dalam gedung ajaib tersebut.



Ayo, Berpikir Kritis!

Ketika mengembangkan struktur data *graph* pada **Aktivitas IX-BK-AD-B02-I**, apa hal yang perlu diperhatikan? Bagaimana cara membedakan simpul dan jalur pada struktur data *graph*?



Ayo, Refleksi!

1. Apakah kamu sudah bisa mengidentifikasi data yang dapat disimpan dalam bentuk struktur data *graph*?
 2. Menurutmu, kapan data dapat disimpan dalam struktur data *tree* dan kapan dapat disimpan dalam struktur data *graph*?
 3. Apakah kamu dapat membuat struktur data *graph* dengan baik untuk kasus lain dalam kehidupan sehari-hari?
- Tuliskan jawabanmu pada buku catatan atau jurnal pembelajaran, ya!

C. Analisis Himpunan Data Terstruktur dengan Teknik Visualisasi

Setelah mengenal struktur data *tree* dan *graph*, selanjutnya kita akan mempelajari proses analisis data untuk data yang terstruktur. Apakah kamu masih ingat apa yang dimaksud data terstruktur di kelas VIII? Data terstruktur adalah data yang dapat disusun secara rapi dan sistematis dalam bentuk kolom dan baris, sehingga mudah untuk dibaca. Sebagai contoh, perhatikan data nilai siswa yang disajikan pada Tabel 1.1 berikut.



Tabel 1.1 Contoh Data Nilai Siswa

Nama	Informatika	IPA	Matematika
Budi	90	90	80
Amir	86	88	70
Dina	87	90	80
Suti	100	90	85

Dari Tabel 1.1 di atas, kita dapat melihat bahwa data sudah disusun dengan rapi. Namun, penyusunan data dalam bentuk tabel atau daftar saja belum tentu langsung membuat kita mudah memahaminya. Untuk menganalisis data lebih efektif, kita dapat menggunakan teknik visualisasi data, seperti grafik atau diagram.

Visualisasi data membantu kita memahami isi data dengan lebih jelas. Bahkan, kita dapat menemukan pola atau perbandingan yang mungkin tidak terlihat jika hanya melihat angka-angka saja. Tampilan yang menarik dan mudah dibaca membuat informasi dari data lebih cepat dipahami. Hal ini sangat berguna dalam pengambilan keputusan berdasarkan data. Pada aktivitas pembelajaran berikutnya, kita akan melakukan analisis data dengan teknik visualisasi, mulai dari mengumpulkan, membersihkan, dan mengolah data sampai menampilkan hasil analisisnya. Agar dapat menganalisis data dengan baik, terdapat beberapa langkah yang perlu dilakukan.

1. Menentukan Tujuan Analisis Data

Sebelum menganalisis data, kita perlu mengetahui apa yang ingin kita cari. Cara paling mudah untuk menentukan tujuan analisis data adalah dengan merumuskan pertanyaan. Pertanyaan ini akan membantu kita fokus pada informasi yang ingin kita temukan dari data. Contoh pertanyaan yang bisa digunakan untuk menganalisis data:

- Bagaimana perbandingan jumlah sampah organik dan anorganik sepanjang tahun 2024? Apakah lebih banyak sampah organik atau anorganik?
- Apakah ada hubungan antara jumlah sampah yang dihasilkan dengan jumlah penghuni di setiap rumah di Desa Mekarjaya? Apakah jumlah penghuni rumah tangga berpengaruh terhadap volume sampah yang dihasilkan?

2. Mengumpulkan Data yang Diperlukan

Setelah kita tahu tujuan dari analisis data, langkah selanjutnya adalah menentukan data yang dibutuhkan. Data ini bisa kita dapatkan dari berbagai sumber, yaitu:

- a. Sumber data primer, yaitu data yang kita kumpulkan sendiri. Misalnya dengan cara mengamati langsung (observasi), melakukan wawancara, atau membagikan kuesioner kepada teman-teman.
- b. Sumber data sekunder, yaitu data yang sudah dikumpulkan dan dibagikan oleh pihak atau lembaga lain. Contohnya data dari Badan Pusat Statistik (BPS), *website* pemerintah, atau laporan hasil penelitian. Berikut adalah beberapa *website* yang digunakan untuk mendapatkan data sekunder.
 - 1) Data BPS: <https://bukupusbuk.id/s/bhuv85>
 - 2) Data OpenData Jawa Barat: <https://bukupusbuk.id/s/2mubxf>
 - 3) Data OpenData Jakarta: <https://bukupusbuk.id/s/alo6hq>

3. Menyiapkan dan Membersihkan Data

Setelah mengumpulkan data yang akan dianalisis, langkah selanjutnya adalah memeriksa data tersebut. Kita harus memastikan bahwa data yang kita miliki sudah lengkap, tidak ada kesalahan dan tidak ada duplikasi (ganda). Jika kita menemukan data yang belum lengkap, kita harus memutuskan apakah data tersebut perlu dihapus atau dilengkapi sebelum digunakan.

Sebagai contoh, perhatikan Tabel 1.2 yang berisi data tinggi badan siswa berikut.

Tabel 1.2 Contoh Data Tinggi Badan Siswa

Nama	Tinggi Badan
Ani	160
Amir	170
Citra	162
Dodi	-
Vina	158

Dari Tabel 1.2, kita dapat melihat bahwa data tinggi badan Dodi belum terisi. Sebelum melakukan analisis data, kita harus membersihkan data ini dengan dua cara, yaitu:

- a. Melengkapi data yang kosong, jika kamu bisa menanyakan langsung kepada Dodi.
- b. Menghapus data Dodi, jika tidak bisa melengkapinya agar tidak mengganggu hasil analisis.

4. Mengeksplorasi Data

Setelah data dikumpulkan dan diperiksa, langkah selanjutnya adalah eksplorasi data. Eksplorasi data artinya kita mulai mengamati isi data, mencoba memahami pola, atau hubungan di dalam data tersebut. Kita dapat mencari tahu beberapa hal seperti:

- a. Apakah terdapat data yang muncul secara berulang?
- b. Apakah terdapat data yang nilainya sangat tinggi atau sangat rendah?
- c. Bagaimana hubungan antara satu data dengan data lainnya?

Untuk membantu proses ini, kita dapat menggunakan beberapa fungsi di aplikasi pengolah angka, seperti:

- a. *COUNT* → untuk menghitung banyaknya data tertentu.
- b. *MIN* → untuk mencari data yang paling kecil.
- c. *MAX* → untuk mencari data yang paling besar.
- d. *AVERAGE* → untuk menghitung rata-rata.

Fungsi-fungsi ini sudah pernah kamu pelajari di kelas VII dan VIII. Selain itu, dalam eksplorasi data kita juga bisa membuat grafik atau diagram. Tujuannya adalah agar data yang kita miliki menjadi lebih mudah dibaca dan dipahami. Pada bab ini, kita akan mulai mengenal beberapa jenis grafik atau diagram yang bisa kita gunakan untuk membantu proses analisis data. Dengan bantuan grafik, kita bisa melihat pola, perbandingan, atau tren dari data dengan lebih jelas. Oleh karena itu, belajar membaca dan membuat grafik sangat penting dalam analisis data.

5. Memvisualisasikan dan Memublikasikan Hasil Analisis Data

Tahap terakhir dari proses analisis data adalah menyampaikan hasilnya kepada orang lain. Hasil analisis data bisa kita tampilkan dalam bentuk grafik atau diagram, agar lebih mudah dimengerti. Selain dapat membantu kita mengeksplorasi data, grafik dan diagram juga berguna untuk menjelaskan dan memublikasikan hasil analisis data. Dengan membuat visualisasi, pembaca dapat langsung melihat informasi penting tanpa harus membaca angka satu per satu.

a. Jenis-Jenis Visualisasi

Hal pertama yang harus kita pahami untuk melakukan analisis data dengan teknik visualisasi adalah mengetahui jenis-jenis diagram yang dapat digunakan untuk memvisualisasikan data.

1) Diagram Batang



a) Diagram Batang Vertikal

b) Diagram Batang Horizontal

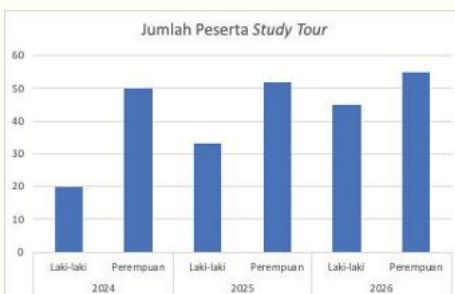
Gambar 1.15 Diagram Batang

Sumber: Erna/Kemendikdasmen (2025)

Tujuan visualisasi data: Melakukan perbandingan data angka.

Deskripsi: Diagram yang sering digunakan untuk membandingkan data angka. Terdapat dua bentuk visualisasi diagram batang, yaitu horizontal dan vertikal.

2) Grup Diagram Batang



Gambar 1.16 Grup Diagram Batang

Sumber: Erna/Kemendikdasmen (2025)

Tujuan visualisasi data: Melakukan perbandingan data angka dalam grup.

Deskripsi: Diagram yang digunakan untuk membandingkan data angka dari beberapa grup atau kelompok.

3) Diagram Garis



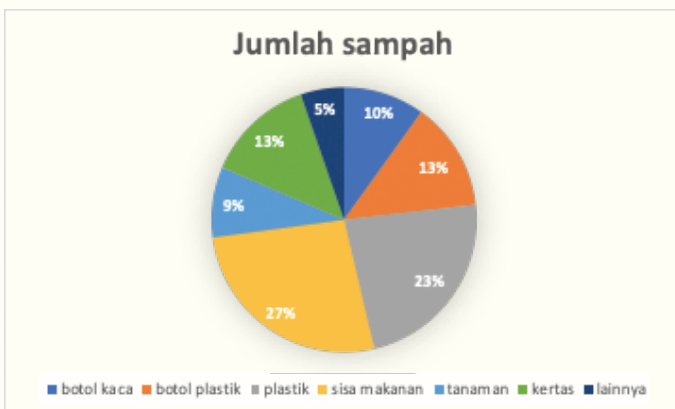
Gambar 1.17 Diagram Garis

Sumber: Erna/Kemendikdasmen (2025)

Tujuan visualisasi data: Menunjukkan tren atau perubahan data dari waktu ke waktu.

Deskripsi: Diagram yang digunakan untuk menunjukkan pola.

4) Diagram Lingkaran



Gambar 1.18 Diagram Lingkaran

Sumber: Erna/Kemendikdasmen (2025)

Tujuan visualisasi data: Menunjukkan proporsi atau komposisi data.

Deskripsi: Diagram yang digunakan untuk menunjukkan pola.

5) Diagram Pencar



Gambar 1.19 Diagram Pencar

Sumber: Erna/Kemendikdasmen (2025)

Tujuan visualisasi data: Menunjukkan hubungan dua variabel.

Deskripsi: Diagram yang menggambarkan hubungan antar dua variabel.



Aktivitas IX-BH-AD-C01-I



Aktivitas Individu

Mengidentifikasi Jenis Visualisasi dan Tujuannya

- ✓ Tujuan: Pada aktivitas ini, kamu diharapkan dapat mengidentifikasi berbagai jenis visualisasi berupa grafik dan dapat membedakan tujuan dari setiap grafik.
- ✓ Syarat pengetahuan: Data dan informasi.

Pada kegiatan ini, kamu akan mengamati berbagai jenis visualisasi data yang disajikan dalam bentuk gambar atau grafik. Kamu akan mempelajari berbagai grafik yang tersedia dan mencoba menemukan informasi yang dapat diperoleh dari masing-masing grafik. Tentukan informasi berikut dari setiap grafik yang kamu amati.

- Nama jenis grafik (contoh: diagram batang, diagram lingkaran, atau diagram garis).
- Tujuan grafik atau pesan yang ingin disampaikan melalui grafik tersebut.

Melalui kegiatan ini, kamu akan memahami berbagai jenis grafik dan mengetahui kapan harus menggunakan grafik tertentu sesuai dengan tujuan analisis datanya.

Tampilan Visualisasi



Gambar 1.20 Laju Pertumbuhan Penduduk

Sumber: Badan Pusat Statistik (BPS), 2023

Nama Grafik:

.....

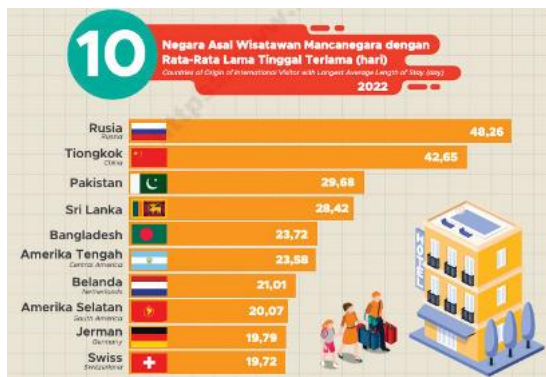
Tujuan dari Visualisasi:

(informasi apa yang ingin disampaikan dari visualisasi tersebut?)

.....

.....

Tampilan Visualisasi



Gambar 1.21 Jumlah Wisatawan Mancanegara

Sumber: Badan Pusat Statistik (BPS), 2023

Nama Grafik:

.....

Tujuan dari Visualisasi:

(informasi apa yang ingin disampaikan dari visualisasi tersebut?):

.....

.....

Tampilan Visualisasi



Gambar 1.22 Distribusi Peredaran Uang

Sumber: Badan Pusat Statistik (BPS), 2023

Nama Grafik:

.....

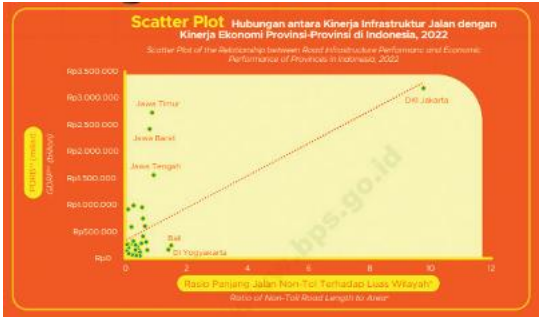
Tujuan dari Visualisasi:

(informasi apa yang ingin disampaikan dari visualisasi tersebut?):

.....

.....

Tampilan Visualisasi



Gambar 1.23 Hubungan Kinerja Infrastruktur dengan Ekonomi Setiap Provinsi

Sumber: Badan Pusat Statistik (BPS), 2023

Nama Grafik:

.....

Tujuan dari Visualisasi:

(informasi apa yang ingin disampaikan dari visualisasi tersebut?)

.....

.....

b. Membuat Grafik dengan Aplikasi Pengolah Angka

Setelah mengenal beberapa jenis diagram, ayo kita praktikkan cara membuat diagram tersebut menggunakan aplikasi pengolah angka. Pada buku ini, kita akan menggunakan aplikasi pengolah angka Microsoft Excel versi 2023. Kamu dapat menggunakan versi terbaru dari aplikasi tersebut.

1) Diagram Batang

Data yang diperlukan untuk membuat diagram batang adalah data berupa nilai atau angka yang akan dibandingkan.

Contoh tabel data untuk diagram batang:

Jenis Kelamin	Jumlah peserta
Laki-laki	20
Perempuan	30

Gambar 1.24 Tabel Data Diagram Batang

Sumber: Erna/Kemendikdasmen (2025)

Langkah Membuat Visualisasi di Microsoft Excel:

1. Pilih data yang akan divisualisasikan

	A	B	C
1			
2	Jenis Kelamin	Jumlah peserta	
3	Laki-laki	20	
4	Perempuan	30	
5			

Gambar 1.25 Pemilihan Data untuk Membuat Diagram Batang

Sumber: Erna/Kemendikdasmen (2025)

2. Pilih Insert

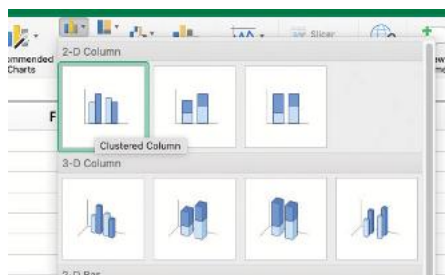


Gambar 1.26 Tampilan Menu Insert

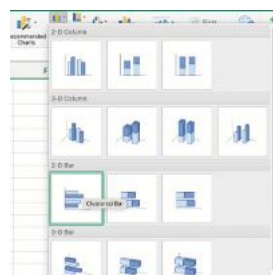
Sumber: Erna/Kemendikdasmen (2025)

3. Pilih **diagram batang**

a. Jika memilih diagram batang vertikal;



b. Jika memilih diagram batang horizontal;



Gambar 1.27 Pemilihan Diagram Batang Vertikal atau Horizontal

Sumber: Erna/Kemendikdasmen (2025)

2) Diagram Batang untuk Data Grup

Contoh tabel untuk diagram batang grup:

Pada tabel tersebut, data dikelompokkan berdasarkan tahun, setiap tahunnya memuat data untuk dua jenis kelamin, yaitu laki-laki dan perempuan. Data yang akan dibandingkan adalah jumlah laki-laki dan jumlah perempuan setiap tahun.

Tahun	Jenis Kelamin	Jumlah peserta
2020	Laki-laki	20
	Perempuan	30
2021	Laki-laki	33
	Perempuan	38
2022	Laki-laki	45
	Perempuan	48
2023	Laki-laki	40
	Perempuan	50
2024	Laki-laki	49
	Perempuan	55

Gambar 1.28 Tabel Data Diagram Batang Grup

Sumber: Erna/Kemendikdasmen (2025)

Langkah Membuat Visualisasi di Microsoft Excel:

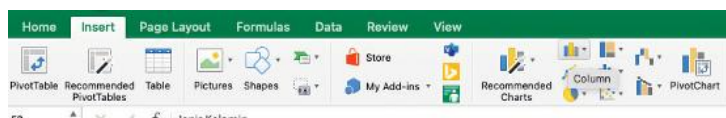
1. Pilih data yang akan divisualisasikan

A	B	C
1		
2	Tahun	Jenis Kelamin Jumlah peserta
3		Laki-laki 20
4	2020	Perempuan 30
5		Laki-laki 33
6	2021	Perempuan 38
7		Laki-laki 45
8	2022	Perempuan 48
9		Laki-laki 40
10	2023	Perempuan 50
11		Laki-laki 49
12	2024	Perempuan 55
13		
14		

Gambar 1.29 Pemilihan Data untuk Membuat Diagram Batang Grup

Sumber: Erna/Kemendikdasmen (2025)

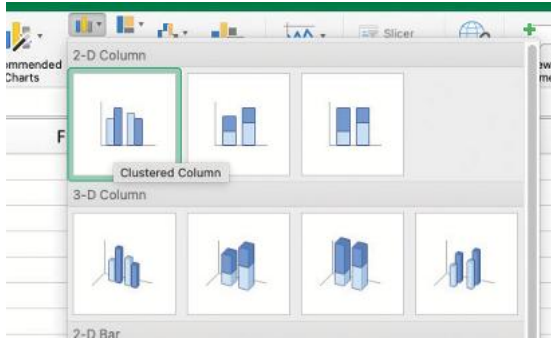
2. Pilih **Insert**



Gambar 1.30 Tampilan Menu Insert

Sumber: Erna/Kemendikdasmen (2025)

3. Pilih diagram batang



Gambar 1.31 Pemilihan Diagram Batang

Sumber: Erna/Kemendikdasmen (2025)

3) Diagram Garis

Contoh tabel untuk data diagram garis:

	2020	2021	2022	2023	2024
Laki-laki	20	33	45	40	49
Perempuan	30	38	48	50	55

Gambar 1.32 Tabel Data Diagram Garis

Sumber: Erna/Kemendikdasmen (2025)

Pada tabel tersebut, terdapat informasi tahun dan jenis kelamin. Informasi yang ingin ditampilkan adalah tren atau pola perubahan data dari tahun ke tahun pada kategori laki-laki dan perempuan. Untuk tujuan tersebut, diagram garis adalah pilihan yang tepat karena dapat digunakan untuk membandingkan tren atau pola pada kategori yang berbeda.

Langkah Membuat Visualisasi di Microsoft Excel:

1. Pilih data yang divisualisasikan.

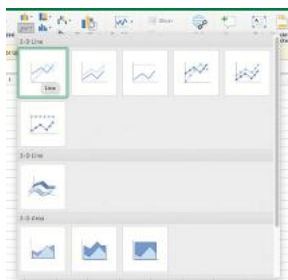


	A	B	C	D	E	F	G
1		2020	2021	2022	2023	2024	
2	Laki-laki	20	33	45	40	49	
3	Perempuan	30	38	48	50	55	
4							
5							

Gambar 1.33 Pemilihan Data untuk Membuat Diagram Garis

Sumber: Erna/Kemendikdasmen (2025)

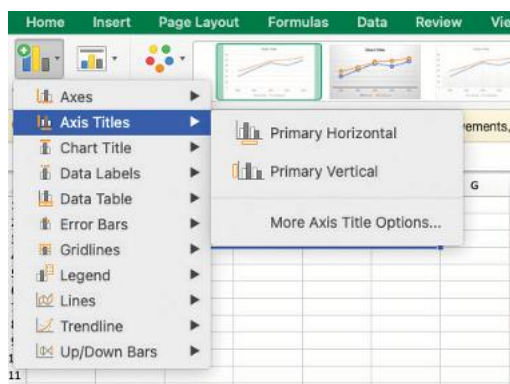
2. Pilih **Insert**
3. Pilih **diagram garis**



Gambar 1.34 Pemilihan Diagram Garis

Sumber: Erna/Kemendikdasmen (2025)

4. Tambahkan informasi judul diagram (**Add chart element -> Chart Title**)
5. Tambahkan informasi sumbu x dan sumbu y (**Add chart element-> Axis Titles**)



Gambar 1.35 Menambahkan Informasi Sumbu X dan Sumbu Y

Sumber: Erna/Kemendikdasmen (2025)

4) Diagram Lingkaran

Contoh tabel untuk data diagram lingkaran:

Tabel tersebut memuat jenis dan jumlah sampah. Diagram lingkaran digunakan untuk menunjukkan proporsi masing-masing jenis sampah terhadap total keseluruhan.

Berdasarkan contoh data yang tercatat pada tabel di samping, proporsi sampah plastik adalah,

$$20 : \text{total sampah} = 20 : 100 = 20\%$$

Langkah Membuat Visualisasi di Microsoft Excel:

1. Pilih data yang divisualisasikan

	A	B
1		Jumlah Sampah
2	bottle kaca	20
3	bottle plastik	15
4	plastik	20
5	kertas	15
6	sisa makana	25
7	lainnya	5
8		

Gambar 1.37 Pemilihan Data untuk Membuat Diagram Lingkaran

Sumber: Erna/Kemendikdasmen (2025)

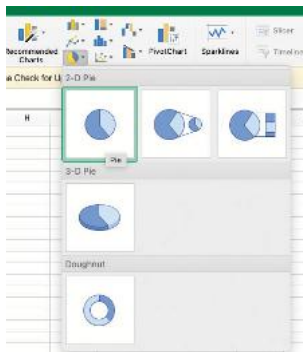
2. Pilih **Insert**

	Jumlah Sampah
bottle kaca	20
bottle plastik	15
plastik	20
kertas	15
sisa makana	25
lainnya	5

Gambar 1.36 Tabel Data Diagram Lingkaran

Sumber: Erna/Kemendikdasmen (2025)

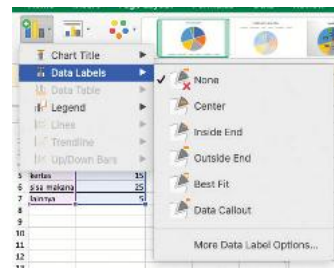
3. Pilih diagram lingkaran



Gambar 1.38 Pemilihan Diagram Lingkaran

Sumber: Erna/Kemendikdasmen (2025)

4. Tambahkan informasi judul diagram dan data label (Add chart element -> Data Labels)



5) Diagram Pencar

	berat badan	tinggi badan
rina	50	150
amir	90	170
wawan	60	160
dwi	45	150
novi	48	151
ani	55	155
edi	60	158
eka	89	170
julian	77	171
sofi	53	163

Gambar 1.39 Tabel Data Diagram Pencar

Sumber: Erna/Kemendikdasmen (2025)

Contoh tabel untuk data diagram pencar:

Pada tabel di samping, terdapat data nama siswa, berat badan, dan tinggi badan. Dengan diagram pencar, informasi yang dapat divisualisasikan adalah hubungan antara kedua data numerik yaitu data berat badan dan tinggi badan.

Hubungan yang divisualisasikan dapat menunjukkan tiga kemungkinan.

1. **Berkorelasi positif**, yaitu jika semakin besar nilai sebuah data, maka nilai data lainnya juga menunjukkan data yang semakin besar. Contoh, jika nilai berat badan semakin besar maka nilai tinggi badan menunjukkan nilai semakin besar.



Gambar 1.40 Diagram Pencar Korelasi Positif

Sumber: Erna/Kemendikdasmen (2025)

2. **Berkorelasi negatif**, jika semakin besar suatu nilai, maka nilai lainnya menunjukkan nilai sebaliknya. Contoh, jika nilai usia semakin besar maka tingkat metabolisme akan menurun.



Gambar 1.41 Diagram Pencar Korelasi Negatif

Sumber: Erna/Kemendikdasmen (2025)

3. **Tidak berhubungan**, jika ada dua variabel yang sama sekali tidak menunjukkan hubungan apapun.



Gambar 1.42 Diagram Pencar Tanpa Korelasif

Sumber: Erna/Kemendikdasmen (2025)

Langkah Membuat Visualisasi di Microsoft Excel:

1. Pilih data yang divisualisasikan
2. Pilih **Insert**
3. Pilih **diagram pencar (Scatter)**

	A	B	C
1		berat badan	tinggi badan
2	rina	50	150
3	amor	90	170
4	wawan	60	160
5	dwi	45	150
6	novi	48	151
7	ani	55	155
8	edi	60	158
9	eka	89	170
10	julian	77	171
11	sofi	53	163
12			

Gambar 1.43 Pemilihan Data untuk Membuat Diagram Pencar

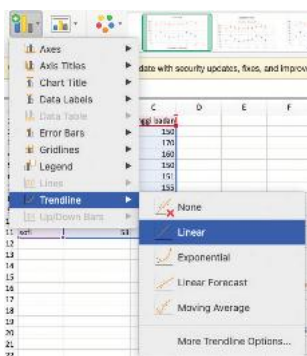
Sumber: Erna/Kemendikdasmen (2025)



Gambar 1.44 Pemilihan Diagram Pencar

Sumber: Erna/Kemendikdasmen (2025)

4. Tambahkan informasi judul diagram, *axis chart* dan garis trend (**Add chart element -> Trendline**)



Gambar 1.45 Menambahkan Informasi pada Diagram Pencar

Sumber: Erna/Kemendikdasmen (2025)



Aktivitas IX-BH-AD-C02-I



Aktivitas Individu

Membuat Visualisasi untuk Analisis Data Sederhana

✔ Tujuan: Pada aktivitas ini, kamu diharapkan mampu menentukan jenis grafik yang tepat untuk merepresentasikan data tertentu dan menggambarannya baik secara manual maupun menggunakan aplikasi pengolah angka (Microsoft Excel).

✔ Syarat pengetahuan: Aplikasi pengolah angka.

Pada kegiatan ini, kamu akan belajar membuat grafik sederhana untuk menampilkan informasi dari data. Ikuti langkah-langkah berikut ini.

Perhatikan tujuan visualisasi pada kolom 1, Tabel 1.3.

1. Pilih jenis grafik yang paling tepat dan sesuai dengan tujuan tersebut, lalu tuliskan pada kolom 2. Kamu bisa memilih salah satu jenis grafik berikut.
 - a. Grafik batang (*bar chart*),
 - b. Grafik garis (*line chart*),
 - c. Diagram lingkaran (*pie chart*), dan
 - d. Grafik pencar (*scatter plot*).
2. Buat tabel data sesuai dengan grafik yang kamu pilih. Gunakan data *dummy* (data palsu tetapi masuk akal).
3. Gambarkan grafik dari data tersebut dengan dua cara, yaitu:
 - a. Menggambarinya secara manual,
 - b. Menggunakan aplikasi pengolah angka, seperti Microsoft Excel, Google Sheets, atau aplikasi lainnya.



Tabel 1.3 Aktivitas Membuat Grafik Sederhana

Tujuan Visualiasi	Jenis Grafik yang Dapat Digunakan Dalam Visualisasi	Tabel yang Digunakan untuk Menyimpan Data (Gunakan Data <i>Dummy</i> / Palsu Sebagai Contoh)	Gambar Visualisasi yang Digambar Secara Manual (<i>Unplugged</i>)	Gambar Visualisasi yang Dihasilkan oleh Aplikasi Pengolah Angka (<i>Plugged</i>)
Membandingkan perubahan jumlah sampah organik dan anorganik yang dikumpulkan di Kota Bandung dari bulan Januari sampai Desember 2024.				
Menampilkan perbandingan jumlah siswa laki-laki dan perempuan setiap tahun dari tahun 2010 sampai 2015 yang aktif mengikuti program Bank Sampah.				
Menampilkan hubungan antara jumlah makanan yang dikonsumsi setiap hari dalam kalori dengan berat badan siswa (kg) dari 20 siswa.				
Menampilkan komposisi kandungan makanan yang dikonsumsi (karbohidrat, protein, lemak, kalsium, mineral, dan vitamin).				



Aktivitas
IX-BH-AD-C03-K

Aktivitas Kelompok

Proyek Analisis Data Sederhana dengan Menggunakan Visualisasi

- ✔ Tujuan: Pada aktivitas ini, kalian diharapkan mampu mengembangkan proyek analisis data menggunakan teknik visualisasi sederhana.
- ✔ Syarat pengetahuan: Aplikasi pengolah angka, rumus aritmatika, dan statistik sederhana.
- ✔ Tema proyek analisis data: Hemat Energi.

Kerjakan proyek analisis data ini secara berkelompok. Hal pertama yang dilakukan adalah mewawancarai teman sekelas untuk melengkapi Tabel 1.4 berikut.

Tabel 1.4 Data Penggunaan Energi Listrik di Rumah

Nama	Rata-Rata Lama TV Dinyalakan Dalam Satu Hari (Jam)	Jumlah Telepon Genggam di Rumah	Jumlah Lampu yang Rutin Menyala Setiap Hari	Menggunakan Lampu LED (Ya/Tidak)	Mencabut Charger Ketika Tidak Digunakan (Ya/Tidak/ Kadang-kadang)

Setelah itu, catatlah kebiasaan penggunaan listrik di rumahmu selama satu minggu pada Tabel 1.5.

Tabel 1.5 Konsumsi Penggunaan Energi Listrik Harian

Alat Elektronik	Senin	Selasa	Rabu	Kamis	Jumat	Sabtu	Minggu
Lama menyalakan TV (jam)							
Berapa kali mengisi ulang daya telepon genggam							
Lama menyalakan lampu di kamar (jam)							



Dari data yang telah terkumpul, buatlah visualisasi untuk menjawab permasalahan berikut.

1. Bagaimana hubungan jumlah telepon genggam di rumah dengan durasi televisi menyala?
2. Tunjukkan komposisi bagaimana kebiasaan siswa dalam menggunakan *charger* di rumah. Berapa komposisi siswa yang biasa mencabut *charger*, tidak mencabut *charger*, dan kadang-kadang mencabut *charger* ketika tidak sedang digunakan?
3. Bandingkan berapa jumlah siswa yang menggunakan lampu LED dan bukan LED!
4. Bandingkan pola penggunaan alat elektronik di rumahmu (TV, telepon genggam, dan lampu) setiap hari selama satu minggu!

Selanjutnya presentasikan hasil analisis data tersebut di depan kelasmu!

Ayo, Refleksi!

1. Apa yang kamu pelajari tentang cara menyajikan data dalam bentuk visual seperti grafik atau diagram?
2. Mengapa informasi pada visualisasi data lebih mudah dipahami daripada hanya melihat tabel?
3. Menurutmu, dalam situasi apa saja kemampuan analisis dan visualisasi data dapat diterapkan dalam kehidupan sehari-hari?
-



Uji Kompetensi

1. Bayangkan kamu sedang membuat struktur folder di komputer untuk menyimpan tugas sekolah. Ada folder utama bernama “Tugas”, di dalamnya ada folder “IPA”, “Matematika”, dan “Bahasa Indonesia”. Di dalam folder “IPA”, kamu menyimpan *file* “Laporan Eksperimen Fotosintesis” dan “Laporan Eksperimen Energi”. Sementara itu, di dalam folder “Bahasa Indonesia”, kamu menyimpan folder “Tugas Esai” yang didalamnya terdapat *file* “Esai-Hemat Energi” dan *file* “Esai-Pengelolaan Sampah Rumah”. Berdasarkan skenario di atas, jawablah pertanyaan berikut ini.
 - a. Gambarkan struktur folder tersebut dalam struktur data *tree*!
 - b. Jelaskan mengapa struktur seperti *tree* cocok digunakan dalam pengelolaan folder dan *file* di komputer!
 - c. Jika kamu ingin mencari *file* “Esai-Hemat Energi”, bagaimana strategi yang dilakukan untuk melakukan pencarian pada struktur *tree* tersebut?
2. Bayangkan kamu dan teman-temanmu tinggal di berbagai rumah yang saling terhubung oleh jalan. Kamu ingin mencari rute tercepat dari rumahmu ke sekolah dengan melihat peta jalur antarrumah.
 - a. Pilih 10 teman yang terdekat dengan rumahmu.
 - b. Catat dan gambarkan jaringan yang menggambarkan rute rumahmu, rumah 10 temanmu, dan sekolah dalam bentuk *graph*.
 - c. Jelaskan langkah-langkah yang harus kamu lakukan jika kamu harus menjemput setiap teman ke rumah mereka dan tiba di sekolah bersama-sama.
3. Perhatikan data berikut ini tentang penggunaan listrik di rumah selama 6 bulan terakhir.

Januari: 120 kWh

Februari: 100 kWh

Maret: 130 kWh

April: 110 kWh

Mei : 170 kWh

Juni : 160 kWh

Kemudian, jawablah pertanyaan berikut ini.

- a. Jenis grafik apa yang cocok untuk menampilkan data tersebut?
- b. Buatlah grafik tersebut secara sederhana!
- c. Informasi apa yang bisa diperoleh dari grafik yang kamu buat?



Refleksi

Selamat! Kamu sudah berhasil menyelesaikan bab ini. Ayo, kita periksa apa saja yang sudah kamu pelajari pada bab ini? Salinlah tabel berikut di buku tugasmu. Lalu beri tanda centang (✓) sesuai dengan pengalaman yang kamu rasakan.

Tabel 1.6 Refleksi Pemahaman Belajar Bab I Berpikir Komputasional dalam Analisis Data

Pengalaman Saya	Sudah Bisa	Masih Perlu Belajar
Memahami konsep struktur data <i>tree</i>		
Mampu membuat struktur data <i>tree</i> untuk masalah sehari-hari		
Mampu menyelesaikan permasalahan yang melibatkan struktur data <i>tree</i>		
Memahami konsep struktur data <i>graph</i>		
Mampu membuat struktur data <i>graph</i> untuk masalah sehari-hari		
Mampu menyelesaikan permasalahan yang melibatkan struktur data <i>graph</i>		
Memahami proses analisis data sederhana		
Mampu membedakan fungsi dari setiap jenis grafik		
Mampu membuat grafik untuk menampilkan data sederhana secara manual		
Mampu membuat berbagai grafik dengan aplikasi tertentu		



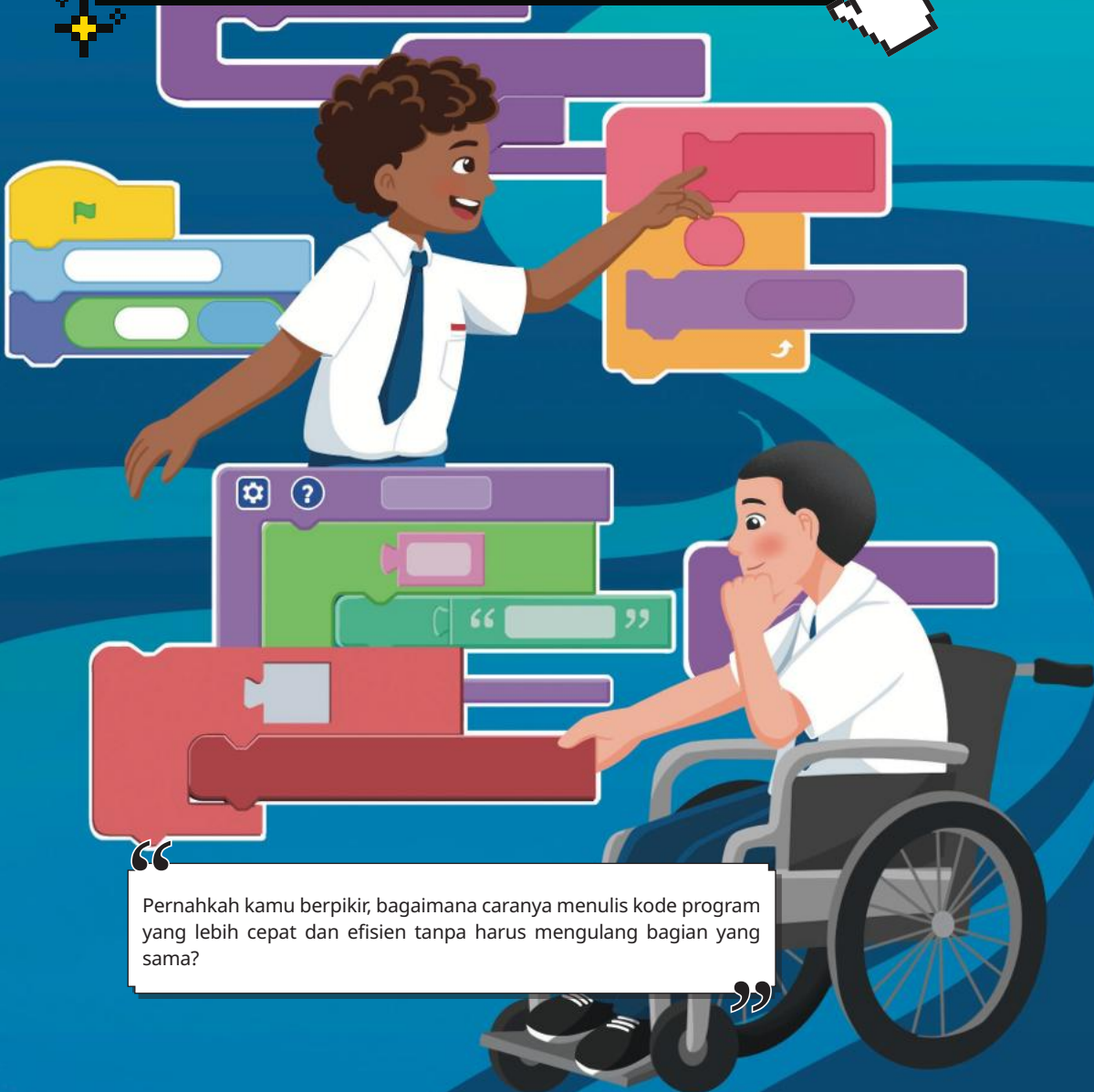
Pengayaan

Kunjungi tautan <https://bukupusbuk.id/s/if9p1> atau pindai kode QR di samping!





Berpikir Komputasional dalam Algoritma dan Pemrograman



“

Pernahkah kamu berpikir, bagaimana caranya menulis kode program yang lebih cepat dan efisien tanpa harus mengulang bagian yang sama?

”



Tujuan Pembelajaran

Setelah mempelajari bab ini, Kamu diharapkan mampu mengembangkan artefak komputasional dengan menganalisis, merancang, mengimplementasi, dan menguji program menggunakan *library* yang disediakan oleh bahasa pemrograman visual. Selain itu, kamu juga akan mengenal salah satu bahasa pemrograman tekstual melalui bahasa pemrograman visual yaitu Scratch dan Blockly. Kamu juga diharapkan dapat mengevaluasi efektivitas *pseudocode* atau *flowchart* yang telah dibuat sebagai solusi dari suatu permasalahan, serta merefleksikan bagaimana *pseudocode* atau *flowchart* dapat membantu dalam memahami dan merancang solusi secara sistematis dan efisien.

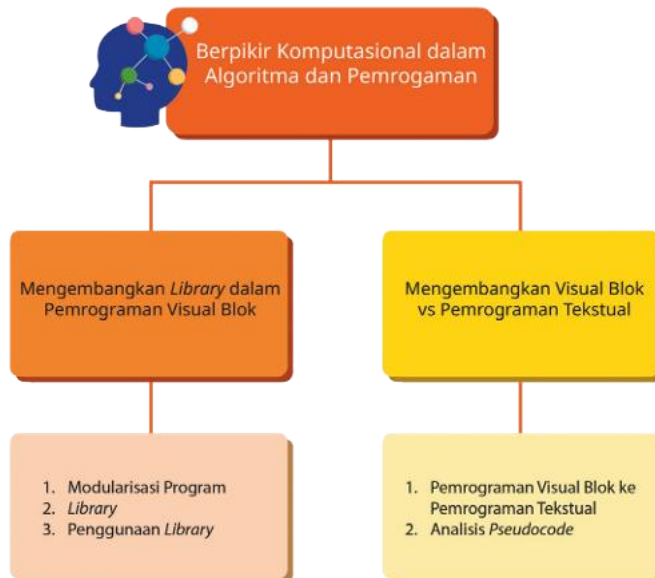


Kata Kunci

- ✔ modularisasi
- ✔ *library*
- ✔ pemrograman visual dan tekstual
- ✔ *pseudocode*
- ✔ *flowchart*



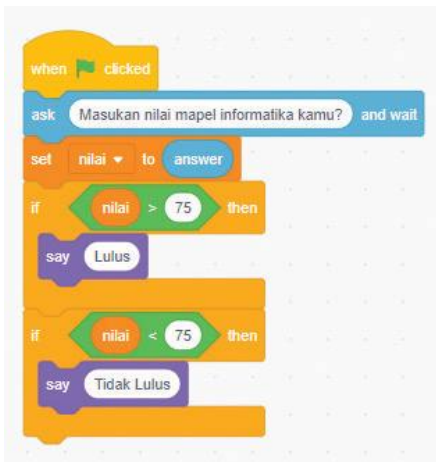
Peta Materi



Gambar 2.1 Peta Materi Bab II Berpikir Komputasional dalam Algoritma dan Pemrograman

Siap-Siap Belajar

Apakah kamu masih ingat pernah belajar tentang fungsi (*function*) di kelas VII? Sekarang, kamu akan mempelajari lebih lanjut tentang penggunaan *library* dalam pemrograman visual. *Library* merupakan kumpulan kode yang dapat digunakan kembali untuk efisiensi. Dengan memahami penggunaan *library*, kamu akan mampu menganalisis, merancang, mengimplementasi, dan menguji sebuah artefak komputasional menggunakan fitur yang tersedia dalam bahasa pemrograman visual berbasis blok. Kamu juga akan mempelajari transisi dari pemrograman visual seperti Scratch dan Blockly ke pemrograman tekstual menggunakan bahasa pemrograman C dan Python. Selain itu, kamu akan belajar menganalisis *flowchart* dan *pseudocode*, melakukan *tracing* ke pemrograman tekstual, serta mengoptimalkan algoritma sederhana. Berbagai contoh penerapan *flowchart* dan *pseudocode* juga akan dipelajari untuk menyelesaikan masalah sederhana dalam kehidupan sehari-hari. Sebelum mempelajari lebih lanjut tentang *library*, yuk amati terlebih dahulu penggalan kode blok program “Cek Nilai” pada Gambar 2.2 berikut ini.



Gambar 2.2 Kode Blok Program Cek Nilai

Sumber: Erlangga/Kemendikdasmen (2025)

1. Apakah kode blok program pada Gambar 2.2 sudah tepat? Jelaskan alasanmu!
2. Coba tawarkan satu alternatif perbaikan agar kode blok program tersebut lebih efisien dan tidak mengulang bagian kode yang sama.



Ayo, Mengingat Kembali!

1. Kamu diminta membuat program menggunakan pemrograman visual berbasis blok untuk menghitung total nilai dari lima siswa, lalu mencetak rata-ratanya.
2. Jelaskan langkah-langkah berpikir komputasional yang kamu gunakan untuk menyusun algoritmanya!
3. Program visual berbasis blok untuk menghitung rata-rata ini akan digunakan kembali pada program lainnya dengan jumlah siswa yang lebih banyak dari soal sebelumnya. Oleh karena itu, sebelum membuat kode program dalam bentuk visual atau tekstual, penting untuk mempelajari terlebih dahulu konsep *library*, modularisasi program, serta analisis *flowchart* dan *pseudocode* agar program yang disusun menjadi lebih efisien, terstruktur, dan mudah digunakan kembali.

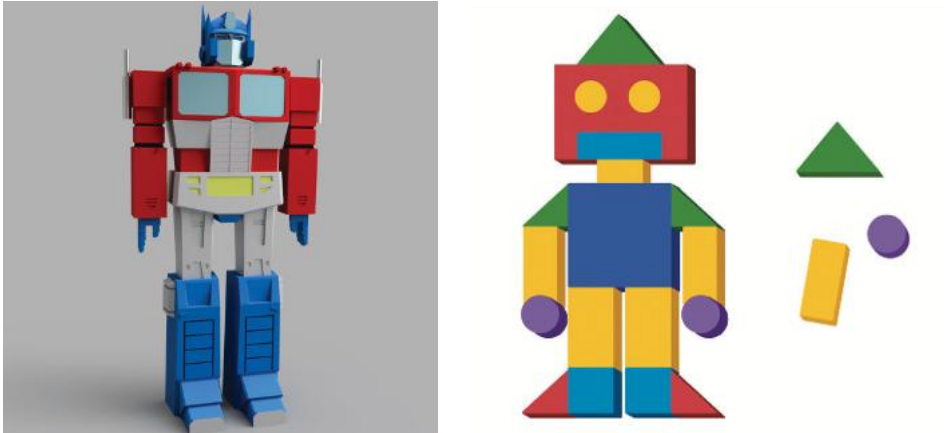
A. Mengembangkan *Library* dalam Pemrograman Visual Blok

Untuk memahami *library* dalam pemrograman visual, bayangkan kamu sedang mengirim pesan yang sama kepada banyak orang. Jika hanya satu orang, tentu kamu cukup menuliskan pesan tersebut satu kali pada secarik kertas. Namun, bagaimana jika orangnya lebih dari satu, misalkan seratus orang. Bagaimana caranya agar pesan yang sama bisa digunakan tanpa harus menulis ulang untuk setiap orang? Kamu dapat menuliskan pesan tersebut satu kali, lalu menggandakannya sebanyak 100 lembar untuk dibagikan. Hal yang sama berlaku dalam menulis kode program. Kamu dapat menggunakan kembali kode yang sama tanpa harus menuliskannya berulang kali. Dalam istilah penulisan kode program, hal ini disebut *reusable code*. Melalui pembelajaran ini, kamu akan mempelajari bagaimana menulis kode program secara lebih singkat tanpa mengulang kembali kode yang sama.

1. Modularisasi Program

a. Pengenalan Modularisasi Program

Pernahkan kamu membuat robot-robotan dari sekumpulan blok kayu seperti ilustrasi pada gambar di bawah ini?



Gambar 2.3 Mainan Robot-Robotan Menggunakan Blok Kayu

Pada Gambar 2.3, kamu dapat melihat secara keseluruhan, bahwa tubuh robot tersusun dari beberapa blok kayu yang berbeda-beda. Namun, ada juga blok yang sama dan dapat digunakan kembali. Seperti blok kayu untuk tangan kanan dan kiri memiliki bentuk yang mirip, sehingga blok kayu tangan kanan dapat digunakan kembali untuk membentuk tangan kiri. Begitu juga blok kayu bagian kaki. Blok kayu pada kaki kanan dapat digunakan kembali untuk kaki kiri. Blok kayu yang dapat digunakan berulang kali disebut modul, yaitu komponen yang dapat digunakan kembali lebih dari satu kali. Program komputer umumnya terdiri atas beberapa modul yang dapat digunakan kembali. Menurut Kamus Besar Bahasa Indonesia (KBBI), modul adalah komponen dari suatu sistem yang berdiri sendiri, tetapi menunjang program dari sistem tersebut. Seperti pada penggunaan blok kayu dalam penyusunan robot-robotan, modul program dapat digunakan berulang kali meskipun hanya ditulis sekali. Penggunaan modul secara berulang dapat diibaratkan seperti mencantumkan referensi saat menulis tugas esai sekolah. Cukup ditulis sekali, tetapi dapat dirujuk berkali-kali dalam teks. Agar dapat memahami modul program, ayo kerjakan aktivitas berikut ini. Ingat, perhatikan penamaan *file* aktivitas latihanmu pada program Scratch, agar mudah digunakan dan dipelajari kembali.



Aktivitas IX-BK-AP-A01-I




Aktivitas Individu

Algoritma Menghitung Persamaan Linear

- ✔ Tujuan: Pada aktivitas ini, kamu akan mengimplementasikan perhitungan persamaan linear melalui pembuatan modul-modul program.
- ✔ Syarat pengetahuan: Persamaan linear.

Kamu akan mencoba membuat program terdiri atas dua buah modul. Ikuti langkah-langkah berikut untuk membuat modul pada Scratch. Kamu juga dapat menggunakan *tools* pemrograman visual blok lainnya.

Tabel 2.1 Langkah-Langkah Membuat Modul Program di Scratch

Block	Keterangan
	<ol style="list-style-type: none">Berikut ini langkah untuk membuat sebuah modul program di Scratch:<ol style="list-style-type: none">Pilihlah tab menu Code dan pilih My BlocksSelanjutnya pilih Make a Block
	<ol style="list-style-type: none">Setelah memilih Make a Block, akan muncul pilihan untuk menambahkan <i>input</i>, seperti Add an input number or text, Add an input boolean, dan Add a label. Setiap blok dapat diberi nama sesuai kebutuhan, yang akan ditampilkan pada area berwarna hijau.

Setelah memahami bagaimana membuat dan menggunakan blok atau modul dalam Scratch, sekarang saatnya kamu menerapkannya untuk menyelesaikan masalah nyata. Salah satu contohnya adalah menyelesaikan soal matematika seperti persamaan linear sederhana. Dengan bantuan pemrograman visual, kamu bisa membuat program untuk menyelesaikan persamaan secara sistematis dan efisien. Perhatikan persamaan linear berikut ini.

$$4x - 5 = 15$$

Persamaan 1

Ikuti langkah-langkah berikut untuk menyelesaikan persamaan menggunakan pemrograman visual berbasis blok.

1. Identifikasi variabel, konstanta, dan nilai yang tidak diketahui dalam persamaan.
2. Kembangkan rumus matematika untuk menyelesaikan nilai yang tidak diketahui (yaitu mengisolasi x).
3. Buat dan inialisasi variabel dalam program visual.
4. Lakukan perhitungan yang diperlukan dan tetapkan hasilnya ke nilai yang tidak diketahui (variabel x).
5. Cetak hasil nilai variabel x menggunakan format yang sesuai.

Sebelumnya kamu perlu memahami terlebih dahulu mengenai variabel, konstanta, dan nilai. Variabel dapat disebut sebagai wadah atau tempat untuk menyimpan nilai. Nilai tersebut dapat berubah sesuai kebutuhan. Sedangkan konstanta adalah nilai yang bersifat tetap artinya tidak dapat berubah. Nilai merupakan isi yang tersimpan di dalam variabel atau konstanta. Setelah memahaminya, mari kita uraikan setiap langkah secara mendetail.

Langkah pertama yaitu mengidentifikasi variabel, konstanta, dan nilai yang tidak diketahui dalam persamaan. Dalam contoh ini, kita memiliki komponen berikut.

$$4 \text{ (Konstanta 1)} * x \text{ (Nilai yang belum diketahui)} - 5 \text{ (Konstanta 2)} = 15 \text{ (Konstanta 3)}$$

Langkah kedua yaitu mengembangkan rumus matematika untuk menyelesaikan nilai yang tidak diketahui (mengisolasi x).

$$x = \frac{15 + 5}{4}$$

Persamaan 2



Langkah ketiga adalah mengembangkan program untuk menghitung nilai dari x .

Input:	4 buah variabel konstanta 1, 2, 3, dan x
Output:	Tampilan nilai x pada layar



Tabel 2.2 merupakan contoh *input* dan *output* dari program ini.



Tabel 2.2 Contoh *Input* dan *Output* Program Penyelesaian Persamaan Linear

No	Contoh Input Variabel Konstanta 1 (K1)	Contoh Input Variabel Konstanta 2 (K2)	Contoh Input Variabel Konstanta 3 (K3)	Output Variabel x
1	4	5	15	5
2	2	3	17	10

Ikuti langkah berikut ini untuk membuat modulnya.

Tabel 2.3 Langkah-Langkah Membuat Modul `Linear1` dalam Pemrograman Visual Scratch

Block	Keterangan
<p>Modul <code>Linear1</code></p> 	<p>1. Buatlah modul <code>linear1</code> dengan parameter <code>k1</code>, <code>k2</code>, dan <code>k3</code>. Klik tombol Ok.</p>
<p>Parameter inputan</p> 	<p>2. Maka modul <code>linear1</code> dengan parameter <code>k1</code>, <code>k2</code>, dan <code>k3</code> akan muncul. Seperti yang kamu ketahui, persamaan 1 akan bergantung pada variabel <code>k1</code>, <code>k2</code>, <code>k3</code>, dan x. Oleh karena itu, kamu harus membuat modul tersebut, agar dapat menerima nilai-nilai dari keempat variabel tersebut.</p>

Block	Keterangan
<p>Parameter inputan</p> 	<p>3. Perhatikan bahwa k_1, k_2, dan k_3 yang ditandai warna kuning disebut sebagai parameter. Dengan demikian, modul <code>linear1</code> adalah modul yang memiliki tiga buah parameter.</p>
<p>Modul persamaan linear mencari nilai x</p> 	<p>4. Salinlah blok kode berikut ke layar kerja proyek Scratch kamu. Pada blok kode bagian A, akan melakukan operasi perhitungan untuk mencari nilai x dengan operasi perhitungan $(k_3 + k_2)$ dibagi dengan k_1. Nilai k_1, k_2, dan k_3 diperoleh dari parameter yang telah didefinisikan dalam modul <code>linear1</code>, dapat diperhatikan dari garis yang ditandai dengan warna oranye. Hasil perhitungan tersebut kemudian disimpan dalam variabel <code>nilai x</code>.</p> <p>Sedangkan pada blok kode bagian B, akan menampilkan pesan <code>nilai x</code> adalah dengan menampilkan hasil perhitungan dari variabel <code>nilai x</code>. Modul program akan memberi tahu nilai dari variabel x kepada modul lain yang memanggil modul <code>linear1</code> ini.</p>

Sampai disini kamu telah membuat sebuah modul dengan nama `linear1` yang bertujuan untuk menghitung `linear1` berdasarkan persamaan 2. Setelah membuat satu blok modul `linear1`, ceritakan dengan kata-katamu sendiri mengenai definisi dari modul program! Tuliskan pada jurnalmu! Mari buat sebuah modul baru yang dapat berinteraksi dengan modul `linear1`. Ikuti langkah-langkah berikut ini.

Membuat Blok Modul Utama



1

Buatlah modul utama pada **Make a Block**. Karena kamu sedang membuat modul utama, yaitu modul yang akan memanggil modul `linear1`, gantilah nama modul yang kamu buat dengan nama `utama`. Klik tombol **Ok**.

Modul Utama



2

Salinlah blok kode berikut pada layar kerja proyek Scratch kamu. Blok tersebut menunjukkan bahwa kamu sedang memanggil modul `linear1` dari dalam modul `utama`.

Bagian yang ditandai dengan kotak hijau menunjukkan bahwa kamu sedang menentukan nilai untuk tiga buah parameter modul `linear1` yaitu `k1`, `k2`, dan `k3`.

Bagian yang ditandai dengan kotak biru menunjukkan pemanggilan modul `linear1`.

Dengan demikian, modul `F1` yang bertujuan untuk menghitung nilai $f(x)$ berdasarkan persamaan 1 yang semula adalah: $4x - 5 = 15$ Kemudian diubah menjadi: $x = \frac{15 + 5}{4}$

Maka nilai dari x adalah 5.

Menjalankan Modul Utama



3

Untuk menjalankan program ini, kamu perlu melakukan pemanggilan terhadap modul “utama”. Pilihlah nama modul “utama” tersebut dari daftar modul yang ada pada menu **My Blocks**. Letakkan modul “utama” pada layar kerja.

4

Gambar berikut menunjukkan tampilan yang seharusnya ada pada layar kerja proyek Scratch kamu.

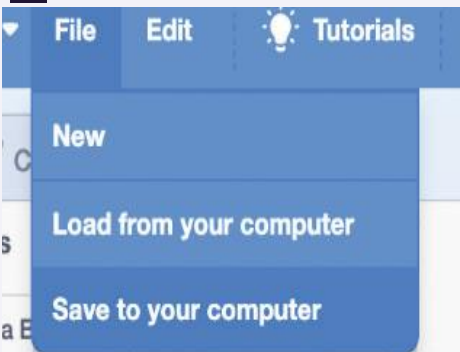
Hasil Program



5

Perhatikan bahwa setelah kamu meletakkan nama modul “utama” pada layar kerjamu, Scratch akan menampilkan hasil nilai dari x , yaitu 5.

Simpan File



6

Untuk menyimpan *file* proyek Scratch, pilih menu **File** >> **Save to your computer**.

7

Beri nama *file* dengan format: **IX-BK-AP-A01-I-<No. Absen>-<Nama>**. Berikut contoh nama *file* jika nama siswa adalah Aldi dengan nomor absen 7 : **IX-BK-AP-A01-I-7-Aldi.sb3**.

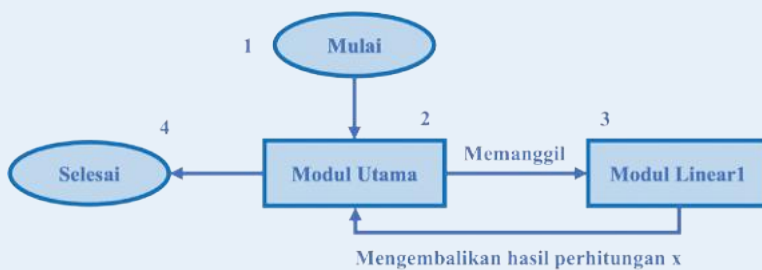
Catatan: Penamaan *file* bisa saja berubah sesuai dengan instruksi guru masing-masing.

Selamat! Kamu sudah berhasil membuat program pertama yang terdiri atas dua buah modul!

Ayo, Berpikir Kritis!

1. Pada program Scratch yang telah kamu buat, terdapat dua buah modul bernama `linear1` dan `utama`.
2. Ketika kamu sudah selesai membuat modul `linear1`, tetapi belum membuat modul `utama`, maka program belum menampilkan apapun. Hal serupa juga terjadi jika kamu telah membuat modul `utama`, tetapi belum memanggilnya.

3. Pada program tersebut, modul utama dapat dikatakan memanggil modul `linear1`.
4. Modul `linear1` mengembalikan sebuah nilai hasil perhitungan ke modul utama. Nilai tersebut disimpan pada variabel `x`.
5. Ketika kamu menempatkan modul utama pada layar kerja Scratch, itu berarti kamu sedang memanggil modul utama tersebut. Saat sebuah modul dipanggil, modul tersebut akan menjalankan perintah-perintah yang terdapat di dalamnya. Secara sederhana, alur prosesnya dapat kamu lihat pada Gambar 2.4.



Gambar 2.4 Alur Program Perhitungan `x`

Pada dasarnya, setiap program memiliki sebuah modul yang menjadi titik awal jalannya program tersebut. Pada aktivitas ini, titik awal jalannya program dimulai dari modul utama. Pada beberapa bahasa pemrograman lain, modul yang menjadi titik awal jalannya program biasanya disebut modul `main` (dalam bahasa Inggris, yang artinya utama). Cobalah untuk mengubah nilai parameter-parameter pada modul utama, lalu amati perubahan nilai keluarannya. Catat hasil eksperimen kamu dalam Tabel 2.4 berikut ini.

Tabel 2.4 Hasil Eksperimen Modul `Linear1`

No	Parameter			Hasil Perhitungan Manual	Hasil dari Program	Sesuai (Ya/Tidak)
	K1	K2	K3			
1	2	4	16			
2	3	4	23			
3	5	5	25			

Kesimpulan apa yang kamu dapatkan berdasarkan hasil eksperimen pada Tabel 2.4? Tuliskan pada jurnalmu ya!



Aktivitas

IX-BK-AP-A02-I



Aktivitas Individu

Mengembangkan Modul Algoritma Menghitung Persamaan Linear

- ✔ Tujuan: Mampu mengimplementasikan perhitungan persamaan linear melalui praktik pembuatan modul program.
- ✔ Syarat pengetahuan: Persamaan linear.

Pada aktivitas ini, kamu tidak diberi panduan berupa blok program. Silakan berlatih membuat blok program berdasarkan contoh yang sudah diberikan pada aktivitas sebelumnya. Salah satu contoh bentuk persamaan linear adalah sebagai berikut.

$$f(x,y) = ax + by$$

Persamaan 3 - Contoh Persamaan Linear

Buatlah sebuah program untuk menghitung nilai $f(x, y)$ seperti yang ditunjukkan pada **persamaan 3**.

Input : 4 buah variabel, yaitu $a, b, x,$ dan y

Output : Tampilan pada layar berupa nilai dari $f(x, y)$

Tabel 2.5 adalah contoh *input* dan *output* dari program ini.

Tabel 2.5 Hasil Eksperimen Terkait Program $f(x, y)$

No	Contoh Input Variabel a	Contoh Input Variabel b	Contoh Input Variabel x	Contoh Input Variabel y	Contoh Output
1	3	2	3	2	13
2	10	30	2	3	110

Analisis:

Untuk membuat program yang dapat menghitung persamaan linear tersebut, diperlukan dua buah modul berikut ini.

1. Buatlah modul bernama `persamaanLinear` yang bertujuan untuk menghitung $f(x, y)$ dari persamaan 2 tersebut.
2. Buatlah sebuah modul bernama `utama` yang memanggil modul `persamaanLinear`. Parameter yang diterima oleh modul `persamaanLinear` didapatkan dari modul `utama`. Modul ini akan menampilkan hasil perhitungan $f(x, y)$ ke layar.



Jalankan program tersebut dan isilah tabel berikut ini.

Tabel 2.6 Tabel Eksperimen Terhadap Parameter Persamaan 3

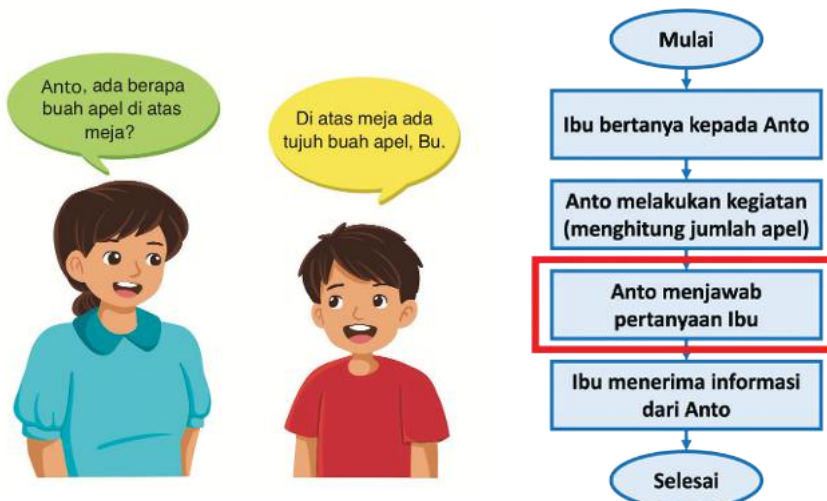
No	Parameter				Hasil Perhitungan Manual	Hasil dari Program	Sesuai (Ya/Tidak)
	a	b	x	y			
1	20	10	3	2			
2	30	15	4	3			
3	40	20	5	4			

Simpan *file* program dengan format nama *file*: “**IX-BK-AP-A02-I-<No. Absen>-<>Nama>.sb3**”.

b. Modularisasi Program *Function* dan *Procedure*

Sebelumnya kamu sudah mempelajari pengantar tentang modularisasi program. Dalam buku Informatika kelas VII, Bab Berpikir Komputasional kamu juga telah dikenalkan pada konsep fungsi. Pada pembahasan kali ini, kamu akan mempelajari lebih lanjut tentang modularisasi program yang berfokus pada *function* (fungsi) dan *procedure* (prosedur). Agar kamu dapat memahami konsep ini dengan lebih baik, perhatikan ilustrasi percakapan antara Ibu dan Anto berikut.

Percakapan 1



Gambar 2.5 Percakapan 1

Keterangan: pada percakapan ini, ibu bertanya dan menunggu jawaban dari Anto. Anto memberi jawaban kepada ibunya.

Percakapan 2



Gambar 2.6 Percakapan 2

Keterangan: pada percakapan ini, ibu memberi perintah kepada Anto. Anto melakukan perintah tersebut dan melaksanakannya secara mandiri.

Anto tidak wajib melaporkan hasil pekerjaannya kepada ibu karena ibu tidak memintanya. Namun, ibu tetap dapat mengetahui status hasil pelaksanaan pekerjaan Anto setelah Anto selesai mengerjakan tugasnya.

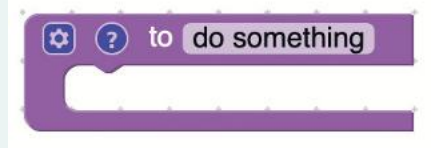
Percakapan antara Ibu dan Anto dapat digunakan sebagai analogi untuk mempelajari *function* dan *procedure* pada program. Istilah “*function*” dan “*procedure*” di sini tidak diterjemahkan ke dalam bahasa Indonesia. Dua istilah ini dipakai untuk menjelaskan makna konsep “*function*” dan “*procedure*” dalam pemrograman. Setiap bahasa pemrograman dapat mengimplementasikan *function* dan *procedure* dengan cara dan kata kunci yang berbeda. Seperti yang telah kamu pelajari sebelumnya, saat membuat modul program pada Scratch dikenal dengan My Block. Akan tetapi, kita tidak dapat mempelajari penggunaan *function* dan *procedure* secara menyeluruh pada Scratch. Oleh karena itu, pada pembahasan kali ini kamu akan menggunakan Blockly sebagai alat bantu untuk memahami *function* dan *procedure*. Kamu tentu sudah terbiasa menggunakan Blockly bukan? Pada buku Informatika kelas VIII telah dijelaskan dasar pemrograman visual dengan Blockly. Dalam Blockly, kamu dapat memilih dua jenis blok untuk *function* dan *procedure*.

Block Program untuk *Function*



Pada blok program ini, terdapat bagian *return statement* (ditandai dengan kotak merah). Modul program yang memiliki *return statement*, akan mengembalikan jawaban atas aksi yang telah dilakukan pada blok tersebut. Hal ini dapat dianalogikan dengan Anto yang menjawab pertanyaan ibunya setelah menghitung apel di atas meja. Dalam pemrograman, modul program yang memiliki *return statement* dapat disebut **function**.

Block Program untuk *Procedure*



Pada blok program ini, tidak terdapat *return statement*. Modul hanya menjalankan perintah-perintah di dalamnya, tetapi tidak mengembalikan nilai apa pun. Seperti Anto yang melaksanakan perintah ibunya untuk mencuci apel dan meletakkannya kembali di atas meja, tanpa melaporkan hasilnya. Dalam pemrograman, modul program yang tidak memiliki *return statement* dapat disebut **procedure**.

Ayo, Berpikir Kritis!

Perhatikan percakapan lain antara Anto dan Ibu pada Gambar 2.7 berikut.



Gambar 2.7 Percakapan Anto dan Ibu

Coba kamu analisis, apakah Gambar 2.7 merupakan ilustrasi dari *function* atau *procedure*? Jelaskan jawabanmu! Semoga kesamaan nama menu **function** pada Blockly tidak membuat kamu bingung dengan istilah *function* dan *procedure* yang baru saja dipelajari. Agar semakin memahami *function* dan *procedure*, kerjakanlah aktivitas berikut ini.



Aktivitas

IX-BK-AP-A03-I



Aktivitas Individu

Algoritma Menghitung Luas

- ✔ Tujuan: Mampu membuat program menghitung luas lingkaran.
- ✔ Syarat pengetahuan: Konsep lingkaran dan luas lingkaran.



Ayo, Bereksplorasi!

Dari pelajaran matematika, kamu telah mengetahui bahwa rumus umum untuk menghitung luas lingkaran adalah:

$$\text{Luas lingkaran} = \pi \times r^2$$

dimana:

$$\pi = 3,14$$

$$r = \text{jari-jari lingkaran}$$

Input : 1 variabel, yaitu variabel r dan konstanta π

Output : Tampilan pada layar berupa hasil perhitungan nilai lingkaran

Tabel 2.7 merupakan contoh *input* dan *output* dari program ini.

Tabel 2.7 Contoh *Input* dan *Output* Program Perhitungan Luas Lingkaran

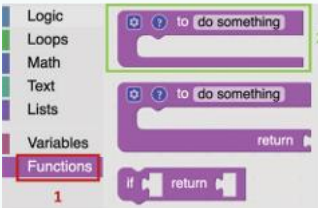
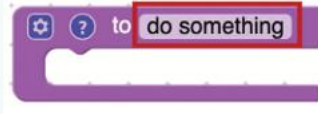
No	Contoh <i>Input</i> Variabel r	Contoh <i>Input</i> Variabel π	Contoh <i>Output</i>
1	10	3,14	314
2	5	3,14	78,5

Untuk membuat program yang dapat menampilkan luas lingkaran diperlukan tiga buah modul, yaitu:

1. *Procedure* utama. Modul ini adalah modul utama dari program. Pada modul ini, akan dilakukan pengulangan untuk pemanggilan modul-modul lainnya.
2. *Function* luasLingkaran. Modul ini berisi perhitungan luas lingkaran. Perhatikan bahwa modul ini akan mengirimkan hasil perhitungan fungsi ke modul utama agar dapat dimanfaatkan oleh modul utama.
3. *Procedure* tampil. Modul ini bertujuan untuk menampilkan hasil perhitungan luas lingkaran dari *function* luasLingkaran. Perhatikan bahwa prosedur ini tidak mengembalikan nilai ke modul utama, melainkan hanya menampilkan hasil di layar. Setelah prosedur selesai dijalankan, kontrol program akan kembali ke modul utama dan melanjutkan perintah berikutnya.

Sekarang kamu akan mencoba membuat program dengan dua buah modul menggunakan Blockly. Ikuti langkah-langkah berikut untuk membuat modul pada Blockly. Kamu dapat menggunakan *tools* ini melalui tautan <https://bukupusbuk.id/s/wfpasx>

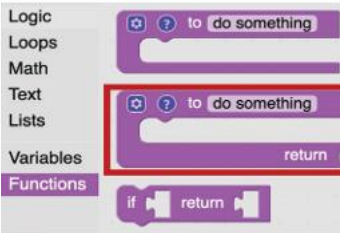

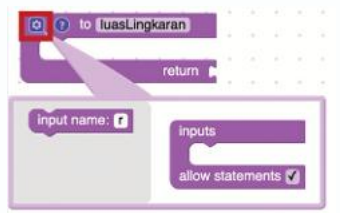
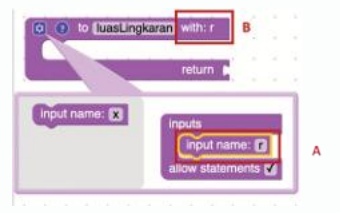
Tabel 2.8 Cara Membuat Modul pada Blockly

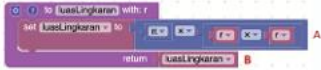
Block	Keterangan
	<p>1. Untuk membuat sebuah modul program, pilih menu Functions pada Blockly (lihat kotak merah nomor 1 pada gambar). Lalu pilih blok [to do something] (lihat kotak hijau nomor 2).</p>
	<p>2. Letakkan blok tersebut pada lembar kerja Blockly. Setiap modul pada program memiliki nama yang ditampilkan pada bagian yang ditandai dengan kotak merah. Jika kamu tidak mengganti nama modul, maka nama modul akan tetap tertulis [do something].</p>

Sampai tahap ini, kamu sudah memiliki sebuah modul kosong bernama [do something]. Nama “do something”, yang artinya ialah “lakukan sesuatu”, merepresentasikan bahwa setiap modul pada program memiliki fungsi atau

tugas tertentu. Selanjutnya kamu akan membuat program untuk menghitung luas lingkaran. Ikuti langkah-langkah berikut ini.

Tabel 2.9 Langkah Pembuatan Modul Program Luas Lingkaran

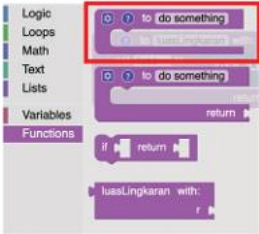

Nama	Block	Keterangan
Pilihan blok modul program		<p>1. Pada menu Functions pilih blok kedua dari atas untuk membuat modul program <i>function</i>.</p>
Modul LuasLingkaran		<p>2. Buatlah modul dengan nama luasLingkaran.</p>
Parameter <i>input</i>		<p>3. Seperti yang kamu ketahui, luas lingkaran akan bergantung pada variabel r. Dengan demikian, kamu harus membuat modul yang dapat menerima nilai-nilai variabel tersebut. Klik bagian yang ditandai dengan kotak merah.</p>
		<p>4. Buatlah satu variabel yaitu r, yang akan menjadi <i>input</i> untuk modul yang dibuat. Perhatikan bahwa <i>input</i> yang kamu buat pada bagian A akan tertera pada bagian B. Bagian B ini disebut dengan parameter. Dengan demikian, modul luasLingkaran adalah modul yang memiliki empat buah parameter.</p>


Nama	Block	Keterangan
Modul luasLingkaran		<p>5. Salinlah blok berikut pada layar kerjamu. Pada bagian A, kamu akan menghitung nilai luas lingkaran dan menyimpannya ke dalam variabel luasLingkaran.</p> <p>Pada bagian B, modul program akan mengirimkan nilai dari variabel luasLingkaran kepada modul lain yang memanggil modul luasLingkaran ini.</p>

Sampai saat ini, kamu sudah membuat sebuah modul dengan nama luasLingkaran yang bertujuan untuk menghitung nilai luas lingkaran. Modul luasLingkaran disebut sebagai modul *function* karena terdapat *return statement* yang berfungsi untuk mengembalikan nilai lingkaran. Setelah membuat satu blok modul luasLingkaran, ceritakan dengan kata-katamu sendiri apa yang dimaksud dengan modul *function*! Tuliskan pada buku jurnalmu, ya!

Ayo, buat modul lain yang dapat berinteraksi dengan modul luasLingkaran, untuk menampilkan pesan teks hasil luas lingkaran. Ikuti langkah-langkah berikut ini.




Tabel 2.10 Langkah Pembuatan Modul `tampil` untuk Hasil Luas Lingkaran



Nama	Block	Keterangan
Pilihan blok modul program		<p>1. Pada menu Function, pilihlah blok program yang paling atas. Perhatikan bahwa pada menu Function saat ini, sudah terdapat sebuah blok modul bernama luasLingkaran.</p>
Modul tampil		<p>2. Karena kamu sedang membuat modul <code>tampil</code>, yaitu modul yang akan menampilkan hasil dari modul luasLingkaran, gantilah nama modul tersebut menjadi "tampil".</p>

Nama	Block	Keterangan
		<p>3. Salinlah blok berikut pada layar kerjamu. Blok tersebut menunjukkan bahwa kamu sedang melakukan pemanggilan modul <code>luasLingkaran</code> dari modul <code>tampil</code>. Dengan demikian, modul <code>luasLingkaran</code> berfungsi untuk menghitung nilai luas lingkaran dan hasilnya adalah 78,5.</p>

Selamat! kamu sudah berhasil membuat dua modul dengan nama `luasLingkaran` dan modul `tampil`. Selanjutnya, kamu akan menampilkan kedua modul tersebut di dalam modul `utama`. Ikuti langkah-langkah berikut ini.

Tabel 2.11 Eksekusi Modul `utama` dan Hasil Program

Nama	Block	Keterangan
Pilihan blok modul program		<p>1. Pada menu Function, pilihlah blok program yang paling atas. Perhatikan bahwa pada menu Function saat ini, sudah terdapat dua blok modul bernama <code>luasLingkaran</code> dan <code>tampil</code>.</p>
Modul utama		<p>2. Karena kalian sedang membuat modul <code>utama</code>, yaitu modul yang akan menampilkan hasil dari modul <code>tampil</code>, kemudian ganti namanya menjadi <code>utama</code>.</p>
		<p>3. Salinlah blok berikut pada layar kerja kamu. Arti dari blok tersebut adalah kamu sedang melakukan pemanggilan modul <code>tampil</code> dari modul <code>utama</code>.</p>

Nama	Block	Keterangan
Menjalankan Modul utama		<p>4. Untuk menjalankan program ini, kamu perlu melakukan pemanggilan terhadap modul utama. Pilihlah nama modul utama tersebut dari daftar modul yang ada pada menu Functions. Letakkan modul utama tersebut pada layar kerja kamu.</p> <p>5. Gambar berikut menunjukkan apa yang seharusnya terdapat pada layar kerja kamu.</p>
Hasil dari program		<p>6. Perhatikan bahwa setelah kamu meletakkan nama modul utama pada layar kerja, Blockly akan menampilkan hasil luas lingkaran, yaitu 78,5.</p>

Selamat! Kamu sudah berhasil membuat program pertama yang terdiri atas tiga modul.

Ayo, Berpikir Kritis!

Cobalah untuk mengubah nilai parameter-parameter pada modul tampil dan isilah Tabel 2.12 berikut.

Tabel 2.12 Tabel Eksperimen Terhadap Rumus Luas Lingkaran

No	Parameter	Hasil Perhitungan Manual	Hasil dari Program	Sesuai (Ya/Tidak)
	r			
1	6			
2	7			
3	8			

Apa kesimpulan yang dapat kamu ambil berdasarkan Tabel 2.12? Tuliskan dalam jurnal kamu.



Aktivitas
IX-BK-AP-A04-I



Aktivitas Individu

Mengembangkan Modul untuk Menghitung Luas dan Volume

- ✔ Tujuan: Mampu membuat program menghitung volume tabung.
- ✔ Syarat pengetahuan: Konsep lingkaran, luas, dan volume lingkaran.

Pada aktivitas ini, kamu tidak diberi panduan berupa blok program. Silakan berlatih membuat blok program berdasarkan contoh yang sudah diberikan pada aktivitas sebelumnya. Dari pelajaran matematika, kamu telah mengetahui bahwa rumus umum untuk menghitung volume bangun ruang ialah menghitung hasil perkalian antara luas alas dan tinggi bangun ruang tersebut. Buatlah program untuk menghitung volume sebuah tabung.

Input : Dua buah variabel yaitu r dan t .

Output : Tampilan pada layar berupa hasil perhitungan nilai volume.

Contoh *input* dan *output* dari program ini dapat dilihat pada Tabel 2.13.

Tabel 2.13 Contoh *Input* dan *Output* Program Volume Tabung

No	Contoh <i>Input</i> Variabel r	Contoh <i>Input</i> Variabel t	Contoh <i>Output</i>
1	10	5	1570
2	5	10	785



Analisis:

Program tersebut terdiri atas tiga buah modul, yaitu:

1. Modul `luasLingkaran`. Modul ini bertujuan untuk menghitung luas lingkaran yang menjadi alas dari tabung. Modul ini memiliki sebuah parameter, yaitu `r` yang merepresentasikan jari-jari lingkaran, dapat digunakan kembali dari aktivitas **IX-BK-AP-A03-I Algoritma Menghitung Luas**.
2. Modul `hitungVolume`. Modul ini bertujuan untuk menghitung volume tabung. Parameter dari modul ini adalah `r` yang merepresentasikan jari-jari lingkaran dan `t` yang merepresentasikan tinggi tabung. Modul ini memanggil modul `hitungLuasLingkaran`.
3. Modul `utama`. Modul ini bertujuan untuk memanggil modul `hitungVolume` dan menampilkan hasil perhitungannya. Simpan *file* yang kamu buat dengan format nama *file* “**IX-BK-AP-A04-I-<No. Absen>-<>Nama>.cwc**”.

Ayo, Refleksi!

1. Setelah memahami bahwa program komputer dapat terdiri dari beberapa modul subprogram, coba tuliskan kesulitan apa saja yang kamu temui selama proses ini.
 2. Bagaimana perasaanmu setelah menyadari bahwa sekarang kamu mampu membuat program terdiri dari beberapa modul?
 3. Apa saja kesimpulan yang dapat kamu ambil dari pembelajaran hari ini?
 4. Ceritakan dengan kalimatmu sendiri, apa persamaan dan perbedaan antara cara kerja *function* (misalnya modul `luasLingkaran`) dan *procedure* (misalnya modul `tampil`).
 5. Menurut kamu, dalam kondisi seperti apa sebuah modul perlu mengembalikan nilai (*return value*) dan kapan sebuah modul tidak perlu mengembalikan nilai?
 6. Mengapa sebuah modul pada umumnya memerlukan parameter?
 7. Apa yang terjadi jika sebuah modul *function* atau modul *procedure* tidak berisi atau memanggil *function* atau *procedure* lain?
- Tuliskan jawabanmu pada jurnal pembelajaran, ya.

2. Library

Selamat! kamu sudah mempelajari konsep modularisasi program. Ayo ingat kembali konsep tersebut sebelum melanjutkan materi berikutnya. Sekarang, kita lanjutkan dengan mempelajari konsep *library*. Ketika mempelajari konsep modularisasi program, kamu sudah memahami bahwa modul program dapat dibuat terpisah dan dapat digunakan berulang kali, tanpa perlu menuliskan ulang kode yang sama. Kamu hanya perlu memanggil modul program tersebut dan menggunakannya kembali. Lalu apa kaitannya dengan *library*? Arti *library* dalam bahasa Indonesia adalah “pustaka”.

Pernahkah kamu membuat tugas laporan merangkum suatu materi? Biasanya, kamu akan mencari pustaka rujukan terlebih dahulu untuk memahami dan merangkumnya. Dalam konteks tersebut, pustaka dapat diartikan sebagai kumpulan pengetahuan yang dapat digunakan kembali karena bermanfaat. Dalam pemrograman, *library* atau pustaka adalah kumpulan kode program yang telah dibuat sebelumnya dan dapat digunakan kembali untuk menyelesaikan suatu tugas agar lebih optimal. Kumpulan kode ini memang sebelumnya dirancang khusus untuk tugas tertentu. Sehingga kamu dapat menggunakannya kembali tanpa harus menulis ulang seluruh instruksi dari awal.

Sederhananya *library* merupakan kumpulan modul program yang sebelumnya sudah dikembangkan dan siap digunakan kembali. Perhatikan gambar ilustrasi dibawah ini agar kamu semakin memahami konsep *library* dalam bahasa pemrograman.



Menulis modul kode program sama seperti ketika kamu membuat modul pekerjaan terpisah yang suatu saat dapat digunakan kembali.



Setelah itu kamu menyimpan modul-modul program yang sudah dikerjakan ke dalam *library* atau pustaka program. Kegiatan ini dapat dianalogikan seperti menyimpan modul buku yang telah kamu buat di dalam rak buku.



Modul program yang telah kamu kerjakan dapat digunakan kembali, sesuai dengan kebutuhan saat mengerjakan proyek pengembangan program. Hal ini dapat dianalogikan saat kamu mengerjakan tugas, kamu dapat melihat referensi dari modul lain atau modul yang pernah dibuat sebelumnya untuk membantu menyelesaikan tugasmu.

Gambar 2.8 Ilustrasi Konsep *Library* dalam Pemrograman

Library dalam pemrograman memiliki beberapa fungsi utama, antara lain:

- a. **Reusability (Keterpakaian ulang).** *Library* memungkinkan penggunaan kembali kode yang telah ada, sehingga kamu tidak perlu menulis ulang kode yang sama berulang kali.
- b. **Abstraction (abstraksi).** *Library* menyembunyikan kompleksitas logika kode program, sehingga kamu dapat fokus pada alur program tanpa harus memahami detail implementasi di balik layar.
- c. **Efficiency (efisiensi).** Dengan menggunakan *library*, produktivitas dan efisiensi meningkat karena kamu dapat memanfaatkan kode yang telah dioptimalkan dan teruji.

Berikut ini contoh penggunaan *library* dalam pemrograman visual blok.

a. Penggunaan *Library* Standar dalam Pemrograman Visual Blok.

Dalam pemrograman visual berbasis blok seperti Scratch, *library* mencakup berbagai blok yang berfungsi untuk kontrol, pergerakan, suara, penampilan,



dan lainnya. Misalnya, ada blok untuk menggerakkan *sprite*, mengulangi tindakan, atau memutar suara.

Contoh:

- 1) Blok gerakan seperti ***move 10 steps*** atau ***turn 15 degrees*** adalah bagian dari *library* gerakan.
- 2) Blok kontrol seperti ***repeat 10*** atau ***if-then*** berasal dari *library* kontrol.

Sementara itu, dalam *platform* pemrograman visual lain seperti Blockly, *library* mencakup berbagai kategori blok seperti logika, *loop*, matematika, teks, dan variabel. Blok-blok tersebut dapat dikombinasikan untuk membuat program.

b. Penggunaan *Library* Eksternal dalam Pemrograman Visual Blok.

Selain menggunakan *library* standar, kamu juga dapat memanfaatkan *library* eksternal yang dapat dikembangkan sendiri dan digunakan kembali dalam program. Kamu telah mempelajarinya dalam aktivitas tentang modularisasi program. Pada pembahasan selanjutnya, kamu akan mempelajari penggunaan *library* dalam pemrograman visual blok baik menggunakan Scratch maupun Blockly.

3. Penggunaan *Library*

Dalam bahasa pemrograman visual blok, penggunaan *library* merujuk pada kumpulan blok yang telah dirancang sebelumnya dan dapat digunakan untuk membangun program dengan cara menyeret dan menyusunnya pada antarmuka visual. *Library* ini berfungsi untuk menyediakan blok-blok yang merepresentasikan berbagai fungsi, operasi logika, manipulasi data, dan interaksi dengan perangkat keras atau perangkat lunak lain. Untuk menambah pemahamanmu tentang *library* dan penggunaannya dalam bahasa pemrograman visual blok, aktivitas-aktivitas dalam pembahasan ini akan memandu kamu mempelajarinya lebih dalam. Apakah kamu sudah siap?



Aktivitas IX-BK-AP-A05-I



Aktivitas Individu

Penggunaan *Library* dalam Bahasa Pemrograman Visual Blok

- ✔ Tujuan: Mampu mendefinisikan modul program yang disimpan dalam *library*, menambahkan modul program baru, dan menggunakan modul program dalam sebuah *library*.
- ✔ Syarat pengetahuan: Konsep luas suatu bangun datar.

Pada aktivitas ini, kamu diminta membuat program untuk menghitung luas berbagai bangun datar. Program ini disebut sebagai *library* luas bangun datar karena berisi kumpulan modul program untuk menghitung luas berbagai bentuk bangun datar. Kamu diminta untuk menghitung luas persegi, persegi panjang, segitiga, dan lingkaran. Modul program luas lingkaran sudah pernah kamu kerjakan pada aktivitas **IX-BK-AP-A03-I Menghitung Luas**. Kamu hanya perlu menambahkan modul program luas bangun datar lainnya. Berikut ini adalah beberapa kebutuhan modul program (*function* dan *procedure*) yang dapat kamu definisikan.

1. *Procedure* utama. Modul ini adalah modul utama dari program. Pada modul ini, akan dilakukan pengulangan untuk pemanggilan modul-modul lainnya.
2. *Function* luasPersegi, luasPersegiPanjang, luasSegitiga, dan luasLingkaran. Modul-modul ini berisi perhitungan luas persegi, persegi panjang, segitiga, dan lingkaran. Setiap *function* akan mengembalikan hasil perhitungan ke modul utama agar dapat digunakan lebih lanjut.
3. *Procedure* tampil. Modul ini bertugas untuk menampilkan hasil perhitungan luas dari semua *function*: luasPersegi, luasPersegiPanjang, luasSegitiga, dan luasLingkaran. Perhatikan bahwa prosedur ini tidak mengembalikan nilai ke modul utama, melainkan hanya menampilkan hasil di layar. Setelah menampilkan hasil, kontrol program akan kembali ke modul utama dan melanjutkan perintah berikutnya.

Sebelum mengembangkan modul program untuk disimpan dalam *library*, kamu perlu menganalisis dan merancang nama serta deskripsi setiap modul ke dalam Tabel 2.14 berikut ini.



Tabel 2.14 Tabel Analisis *Library* Modul Program (*Function* dan *Procedure*) Bangun Datar

No	Nama Modul Program (<i>Function</i> dan <i>Procedure</i>)	Deskripsi Modul Program (Formulasi Matematis Jika Ada)
1	<i>Function</i> luasPersegi	
2	<i>Function</i> luasPersegiPanjang	
3	<i>Function</i> luasSegitiga	
4	<i>Function</i> luasLingkaran	
5	<i>Procedure</i> tampil	
6	<i>Procedure</i> utama	

Setelah melakukan analisis, langkah selanjutnya adalah mengimplementasikan hasil rancangan ke dalam kode program dengan ketentuan *input* dan *output* sebagai berikut.

Input : 6 buah variabel yaitu variabel *s*, *p*, *l*, *a*, *t*, dan *r*.

Output : Tampilan pada layar berupa hasil perhitungan nilai luas persegi, persegi panjang, segitiga, dan lingkaran.

Tabel 2.15 adalah contoh *input* dan *output* dari program ini untuk luas persegi.

Tabel 2.15 Contoh *Input* dan *Output* Luas Persegi

No	Contoh <i>Input</i> Variabel <i>s</i>	Contoh <i>Output</i>
1	10	100
2	5	25

Tabel 2.16 adalah contoh *input* dan *output* dari program ini untuk luas persegi panjang.

Tabel 2.16 Contoh *Input* dan *Output* Luas Persegi Panjang

No	Contoh <i>Input</i> Variabel <i>p</i>	Contoh <i>Input</i> Variabel <i>l</i>	Contoh <i>Output</i>
1	10	2	20
2	5	3	15

Tabel 2.17 adalah contoh *input* dan *output* dari program ini untuk luas segitiga.

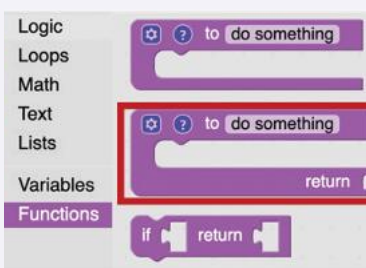
Tabel 2.17 Contoh *Input* dan *Output* Luas Segitiga

No	Contoh <i>Input</i> Variabel <i>a</i>	Contoh <i>Input</i> Variabel <i>t</i>	Contoh <i>Output</i>
1	10	2	10
2	5	3	7,5

Ayo, Berpikir Kritis!

Kamu dapat menggunakan *tools* ini melalui tautan <https://bukupusbuk.id/s/wfpasx>. Sebelumnya, kamu sudah mengerjakan aktivitas **IX-BK-AP-A03-I Menghitung Luas**. Silakan buka kembali aktivitas tersebut untuk menambahkan beberapa modul program bangun datar lainnya.

Pilihan Blok Modul Program



1

Pada menu **Function** pilih blok kedua dari atas untuk membuat modul program function.

Menambahkan Modul *luasPersegi*, *luasPersegiPanjang*, dan *luasSegitiga*



2

Buatlah modul lainnya setelah modul *luasLingkaran* dengan nama modul *luasLingkaran*, *luasPersegi*, *luasPersegiPanjang*, dan *luasSegitiga*.

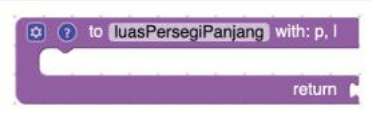
Parameter *Input*



```
to luasPersegi with: s
  return
```

3

Seperti yang kalian ketahui, luas persegi akan bergantung pada variabel *s*. Maka buatlah blok *function* seperti di samping dengan parameter variabel *s*.



```
to luasPersegiPanjang with: p, l
  return
```

4

Luas persegi panjang akan terdiri atas variabel *p* dan *l*, maka buatlah blok *function* seperti di samping dengan parameter variabel *p* dan *l*.

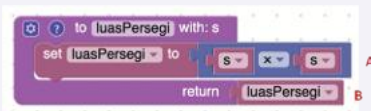


```
to luasSegitiga with: a, t
  return
```

5

Luas segitiga akan terdiri atas variabel *a* dan *t*, maka buatlah blok *function* seperti di samping dengan parameter variabel *a* dan *t*.

Modul luasPersegi



```
to luasPersegi with: s
  set luasPersegi to s * s
  return luasPersegi
```


6

Salinlah blok berikut pada layar kerjamu.

Bagian A, menghitung nilai luas persegi dan menyimpan hasilnya pada variabel *luasPersegi*.

Bagian B akan mengirimkan nilai luas persegi kepada modul lain yang memanggil modul *luasPersegi*.

Modul luasPersegiPanjang



```
to luasPersegiPanjang with: p, l
  set luasPersegiPanjang to p * l
  return luasPersegiPanjang
```

7

Salinlah blok berikut pada layar kerjamu.

Bagian A menghitung nilai luas persegi panjang dan menyimpan hasilnya pada variabel *luasPersegiPanjang*.

Bagian B akan mengirimkan nilai luas persegi panjang kepada modul lain yang memanggil modul *luasPersegiPanjang*.

Modul luasSegitiga



```
to luasSegitiga with: a, t
  set luasSegitiga to 0.5 * a * t
  return luasSegitiga
```

8

Salinlah blok berikut pada layar kerjamu.

Bagian A menghitung nilai luas segitiga dan menyimpan hasilnya pada variabel luasSegitiga.

Bagian B akan mengirimkan nilai luas segitiga kepada modul lain yang memanggil modul luasSegitiga.

Sampai saat ini, kamu telah menambahkan sebuah modul dengan nama luasPersegi, luasPersegiPanjang, dan luasSegitiga yang bertujuan untuk menghitung nilai luas persegi, persegi panjang, dan segitiga. Modul luasPersegi, luasPersegiPanjang, dan luasSegitiga disebut sebagai modul *function* dikarenakan terdapat *return statement* untuk mengembalikan nilai luas masing-masing.

Selanjutnya, mari modifikasi dan tambahkan modul tampil yang berinteraksi dengan modul lain yaitu luasPersegi, luasPersegiPanjang, dan luasSegitiga untuk menampilkan pesan teks yang berisi hasil perhitungan luas masing-masing bangun datar. Ikuti langkah-langkah berikut ini.

Modifikasi Modul Tampil



```
to tampil
  set luasPersegi to luasPersegi with: s 10
  set luasPersegiPanjang to luasPersegiPanjang with: p 5 l 3
  set luasSegitiga to luasSegitiga with: a 10 t 2
  set luasLingkaran to luasLingkaran with: r 5
  print create text with "Luas persegi adalah" luasPersegi
  print create text with "Luas persegi panjang adalah" luasPersegiPanjang
  print create text with "Luas segitiga adalah" luasSegitiga
  print create text with "Luas lingkaran adalah" luasLingkaran
```

Salinlah blok berikut pada layar kerjamu. Arti dari blok tersebut adalah kamu sedang melakukan pemanggilan modul luasPersegi, luasPersegiPanjang, luasSegitiga, dan luasLingkaran dari modul tampil.

Kamu telah menambahkan luasPersegi, luasPersegiPanjang, luasSegitiga, dan luasLingkaran ke dalam modul tampil. Selanjutnya kamu akan menjalankan modul tersebut di dalam modul utama. Ikuti langkah-langkah berikut ini.

Menjalankan Modul Utama

```
to luasPersegi with: s
  set [luasPersegi] to (s * s)
  return luasPersegi
end

to luasPersegiPanjang with: p, l
  set [luasPersegiPanjang] to (p * l)
  return luasPersegiPanjang
end

to luasSegitiga with: a, t
  set [luasSegitiga] to (0.5 * a * t)
  return luasSegitiga
end

to luasLingkaran with: r
  set [luasLingkaran] to (pi * r * r)
  return luasLingkaran
end

to tampil
  set [luasPersegi] to luasPersegi with: s 10
  set [luasPersegiPanjang] to luasPersegiPanjang with: p 5 l 3
  set [luasSegitiga] to luasSegitiga with: a 10 t 2
  set [luasLingkaran] to luasLingkaran with: r 5
  print create text with "Luas persegi adalah"
  print create text with "Luas persegi panjang adalah"
  print create text with "Luas segitiga adalah"
  print create text with "Luas lingkaran adalah"
end

utama
  tampil
```

1

Gambar berikut menunjukkan tampilan yang seharusnya ada pada layar kerjamu.

Hasil dari Program

```
Luas persegi adalah 100
Luas persegi panjang adalah 15
Luas segitiga adalah 10.0
Luas lingkaran adalah 78.53981633974483
```

Salinlah blok berikut pada layar kerjamu. Arti dari blok tersebut adalah kamu sedang melakukan pemanggilan modul `luasPersegi`, `luasPersegiPanjang`, `luasSegitiga`, dan `luasLingkaran` dari modul `tampil`.



Ayo, Berpikir Kritis!

Cobalah untuk mengubah nilai parameter-parameter pada modul tampil dan isilah tabel berikut.

Tabel 2.18 Tabel Eksperimen Terhadap Rumus Luas Persegi

No	Input Variabel s	Hasil Perhitungan Manual	Hasil Perhitungan Program	Sesuai (Ya/Tidak)
1.	3			
2.	7			

Tabel 2.19 Tabel Eksperimen Terhadap Rumus Luas Persegi Panjang

No	Input Variabel p	Input Variabel l	Hasil Perhitungan Manual	Hasil Perhitungan Program	Sesuai (Ya/Tidak)
1.	3	7			
2.	5	4			

Tabel 2.20 Tabel Eksperimen Terhadap Rumus Luas Segitiga

No	Input Variabel a	Input Variabel t	Hasil Perhitungan Manual	Hasil Perhitungan Program	Sesuai (Ya/Tidak)
1.	5	6			
2.	7	3			

Kesimpulan apa yang dapat kamu tangkap berdasarkan tabel 2.18, tabel 2.19 dan tabel 2.20? Ayo tuliskan pada jurnalmu!



Aktivitas

IX-BK-AP-A06-I



Aktivitas Individu

Penggunaan *Library*, Menambah, dan Menggunakan *Library* Keliling Bangun Datar

- ✔ Tujuan: Mampu mendefinisikan modul program yang disimpan dalam *library*, menambahkan modul program baru, dan menggunakan modul program dalam sebuah *library*.
- ✔ Syarat pengetahuan: Konsep keliling suatu bangun datar.

Pada aktivitas ini, kamu tidak diberi panduan berupa blok program. Silakan berlatih membuat blok program berdasarkan contoh yang sudah diberikan pada aktivitas sebelumnya. Kamu diminta menambahkan modul program untuk menghitung keliling berbagai bangun datar. Program ini akan menjadi bagian dari *library* luas dan keliling bangun datar yang berisi kumpulan modul untuk menghitung luas dan keliling bangun datar. Kamu diminta untuk menghitung keliling persegi, persegi panjang, segitiga, dan lingkaran. Modul program luas bangun datar sudah pernah kamu kembangkan di aktivitas **IX-BK-AP-A05-I Penggunaan *Library* dalam Bahasa Pemrograman Visual Blok**. Sekarang, kamu hanya perlu menambahkan modul program kelilingnya dengan ketentuan *input* dan *output* seperti berikut.

Input : 4 buah variabel, yaitu variabel *sPersegi*, *p*, *l*, dan *sSegitiga*.

Output : Tampilan pada layar berupa hasil perhitungan nilai keliling persegi, persegi panjang, segitiga, dan lingkaran.

Tabel 2.21 adalah contoh *input* dan *output* dari program ini untuk keliling persegi.

Tabel 2.21 Contoh *Input* dan *Output* Keliling Persegi

No	Contoh <i>Input</i> Variabel <i>sPersegi</i>	Contoh <i>Output</i>
1	10	40
2	5	20

Tabel 2.22 adalah contoh *input* dan *output* dari program ini untuk keliling persegi panjang.

Tabel 2.22 Contoh *Input* dan *Output* Keliling Persegi Panjang

No	Contoh <i>Input</i> Variabel p	Contoh <i>Input</i> Variabel 1	Contoh <i>Output</i>
1	10	2	24
2	5	3	16

Tabel 2.23 adalah contoh *input* dan *output* dari program ini untuk keliling segitiga.

Tabel 2.23 Contoh *Input* dan *Output* Keliling Segitiga

No	Contoh <i>Input</i> Variabel sSegitiga	Contoh <i>Output</i>
1	10	30
2	5	15

Analisis:

Berikut ini beberapa kebutuhan modul program (*function* dan *procedure*) yang dapat didefinisikan.

1. *Procedure* utama. Modul ini adalah modul utama dari program. Pada modul ini, akan dilakukan pengulangan untuk pemanggilan modul-modul lainnya.
2. *Function* kelilingPersegi, kelilingPersegiPanjang, kelilingSegitiga, dan kelilingLingkaran. Modul ini adalah modul yang berisikan perhitungan keliling persegi, persegi panjang, segitiga, dan lingkaran. Perhatikan bahwa modul ini akan mengirimkan hasil perhitungan fungsi ke modul utama agar dapat dimanfaatkan oleh modul utama.
3. *Procedure* tampil. Modul ini bertujuan untuk menampilkan hasil perhitungan keliling bangun datar dari *function* kelilingPersegi, kelilingPersegiPanjang, kelilingSegitiga, dan kelilingLingkaran. Perhatikan bahwa prosedur ini tidak mengembalikan nilai ke modul utama, melainkan hanya menampilkan hasil di layar. Setelah prosedur selesai dijalankan, kontrol program akan kembali ke modul utama, lalu menjalankan perintah berikutnya.
4. Simpan *file* yang kalian buat dengan format nama *file* “IX-BK-AP-A06-I-<No. Absen>-<>Nama>.cwc”.

Ayo, Refleksi!

1. Apakah kamu sudah memahami penggunaan *library* dalam program? Coba tuliskan kesulitan apa saja yang kamu temui.
2. Sekarang kamu sudah berhasil membuat dan menggunakan *library* yang terdiri dari beberapa modul. Ceritakan perasaanmu tentang peningkatan *skill* pemrogramanmu!
3. Kesimpulan apa saja yang kamu dapatkan dari pertemuan ini? Tuliskan jawabanmu pada jurnal pembelajaran, ya!

B. Pemrograman Visual Blok vs Pemrograman Tekstual

1. Transisi dari Pemrograman Visual Blok ke Pemrograman Tekstual

a. Perbandingan Pemrograman Visual Blok ke Pemrograman Tekstual

Kamu telah menerapkan algoritma dan pemrograman pada elemen berpikir komputasional dengan menggunakan bahasa pemrograman visual blok. Bahasa pemrograman visual blok merupakan pendekatan dalam pemrograman yang menggunakan antarmuka visual untuk menyusun kode program dalam bentuk blok-blok yang bisa dihubungkan secara grafis, tanpa harus menulis kode secara manual. Pendekatan ini sangat populer bagi kamu yang ingin belajar tentang algoritma dan pemrograman, tanpa harus mempelajari dan memahami sintaks kode pemrograman tekstual. Bahasa pemrograman visual blok memfasilitasi kamu belajar dengan memberikan representasi visual dari konsep seperti variabel, perulangan, kondisi, modularisasi program, sehingga kalian dapat fokus pada logika pemrograman tanpa harus khawatir mengenai kesalahan sintaks.

Sedangkan, bahasa pemrograman tekstual merupakan jenis bahasa pemrograman di mana kamu harus menulis kode dalam bentuk teks yang terdiri dari rangkaian instruksi menggunakan sintaksis khusus yang ditentukan oleh bahasa tersebut. Bahasa pemrograman tekstual memungkinkan penulisan kode secara langsung, memberikan fleksibilitas yang lebih besar dalam kontrol, dan kemampuan program dibandingkan dengan bahasa pemrograman visual blok. Bahasa pemrograman tekstual

menawarkan lebih banyak kontrol atas detail implementasi program dan fleksibilitas yang lebih besar dibandingkan dengan bahasa visual blok. Namun, hal ini juga memerlukan pemahaman yang lebih mendalam tentang sintaksis dan struktur bahasa pemrograman tertentu. Bahasa pemrograman visual blok dan bahasa pemrograman tekstual adalah dua pendekatan yang berbeda dalam menulis dan memahami kode. Keduanya memiliki kelebihan dan kekurangan tergantung pada konteks penggunaan, tujuan, dan tingkat pengalaman pengguna. Berikut adalah tabel perbandingan antara keduanya.

Tabel 2.24 Perbandingan Bahasa Pemrograman Visual Blok dan Bahasa Pemrograman Tekstual

1) Antarmuka dan Cara Penggunaan

Bahasa Pemrograman Visual Blok

- Menggunakan antarmuka grafis di mana pengguna menyusun program dengan menyeret dan menghubungkan blok-blok kode yang mewakili instruksi logika.
- Sangat intuitif, cocok bagi pemula yang baru belajar konsep dasar pemrograman.
- Tidak memerlukan pengetahuan tentang sintaksis bahasa pemrograman, sehingga mengurangi kemungkinan kesalahan sintaks.

Bahasa Pemrograman Tekstual

- Pengguna menulis kode secara manual menggunakan editor teks atau lingkungan pengembangan terintegrasi (IDE).
- Memerlukan pemahaman tentang sintaksis dan struktur bahasa pemrograman tertentu.
- Memberikan fleksibilitas dan kontrol penuh atas kode yang ditulis, tetapi juga lebih rentan terhadap kesalahan sintaks.

2) Penggunaan

Bahasa Pemrograman Visual Blok

- Sering digunakan dalam pendidikan dasar, terutama untuk memperkenalkan konsep pemrograman.

- Contoh penggunaan termasuk pengembangan aplikasi sederhana, *game* edukasi, dan prototipe cepat.

Bahasa Pemrograman Tekstual

- Bahasa pemrograman tekstual digunakan secara luas dalam pengembangan perangkat lunak profesional, mulai dari aplikasi web, sistem operasi, hingga pengembangan *game* dan kecerdasan buatan.
- Bahasa ini lebih disukai dalam industri karena fleksibilitasnya dan kemampuannya untuk menangani proyek-proyek besar dan kompleks.

3) Proses Pembelajaran

Bahasa Pemrograman Visual Blok

- Memudahkan kamu untuk memulai belajar pemrograman tanpa harus menguasai sintaksis yang rumit.
- Visualisasi yang intuitif membantu kamu memahami bagaimana logika program bekerja.
- Bisa menjadi pembatas ketika kamu ingin beralih ke pemrograman tekstual, karena perbedaan yang signifikan dalam cara berpikir dan bekerja.

Bahasa Pemrograman Tekstual

- Memberikan fondasi yang kuat untuk berkarir di bidang pemrograman.
- Memungkinkan kamu memahami lebih dalam cara kerja perangkat lunak, termasuk manajemen memori, *debugging*, dan optimasi kinerja.
- Menyediakan lebih banyak sumber daya dan komunitas yang besar untuk belajar dan berkembang.

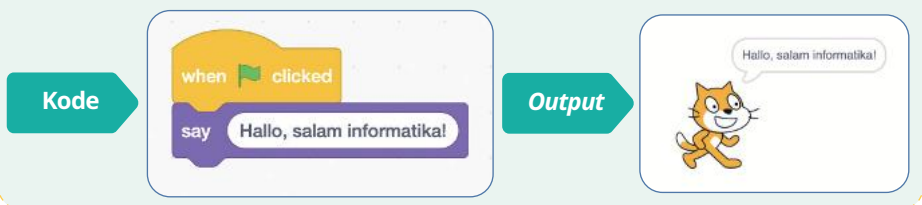
Pemilihan antara bahasa pemrograman visual blok dan tekstual tergantung pada kebutuhan spesifik, tingkat pengalaman, dan tujuan. Selanjutnya, kamu akan mempelajari bagaimana transisi dari pemrograman visual blok ke pemrograman tekstual. Pembahasan lebih mendalam mengenai hal ini akan kamu pelajari di jenjang SMA.

b. Konversi Pemrograman Visual Blok Scratch ke Pemrograman Tekstual

Konversi dari pemrograman visual blok seperti Scratch ke pemrograman tekstual melibatkan penerjemahan blok-blok visual yang intuitif ke dalam kode teks yang eksplisit. Meskipun Scratch adalah alat yang sangat berguna untuk belajar dasar-dasar pemrograman, bahasa pemrograman tekstual memberikan lebih banyak fleksibilitas dan kontrol. Berikut adalah cara mengonversi konsep-konsep dasar dari Scratch ke bentuk pemrograman tekstual.

Konsep *Input* dan *Output*

✦ Pemrograman Visual Blok Scratch



✦ Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>
int main(void) {
    printf("Halo, salam
informatika!\n");}
```

Output

Halo, salam informatika!

✦ Pemrograman Tekstual Bahasa Python


Kode

```
print ('Halo, salam informatika!')
```

Output

Halo, salam informatika!

✦ Deskripsi Program

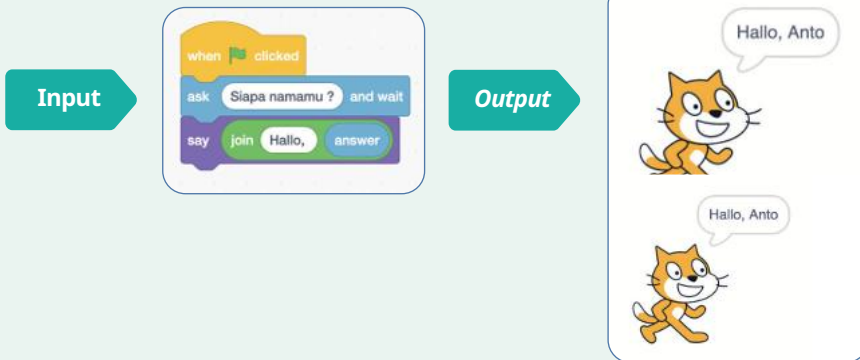
Program tersebut menampilkan pesan teks **"Halo, salam informatika"**. Perhatikan perbedaannya. Dalam pemrograman visual blok, perintah disusun oleh blok  .

Sementara itu, dalam pemrograman tekstual:

- bahasa C menggunakan sintaks: `printf("Halo, salam informatika!\n"),`
- bahasa Python menggunakan perintah: `print ('Halo, salam informatika!')`.

Konsep *Input* dan *Output*

Pemrograman Visual Blok Scratch



Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>
int main(void)
{
    char nama[50];
    printf("Siapa namamu ? ");
    scanf("%s", &nama);
    printf("Halo,%s\n", nama);
}
```

Output

```
Siapa namamu ? Anto
Halo, Anto
```

Pemrograman Tekstual Bahasa Python


Kode

```
nama = input("Siapa namamu ? ")
printf("Halo, {nama}")
```

Output

```
Siapa namamu ? Anto
Halo, Anto
```


Deskripsi Program

Program tersebut menerima *input* dari pengguna berupa pertanyaan **"Siapa namamu?"**, lalu mengeluarkan pesan teks berupa **"Halo, {Nama pengguna yang diinputkan}"**. Ayo perhatikan perbedaannya! Dalam pemrograman visual blok, perintah disusun oleh blok  untuk menampilkan pertanyaan **"Siapa namamu?"**.

Sedangkan dalam pemrograman tekstual:

- bahasa C menggunakan sintaks: `printf("Siapa namamu?")` dan `scanf("%s", &nama)`.

- Dalam bahasa Python menggunakan perintah:
`nama=input("Siapa namamu?")`.

Untuk menampilkan pesan teks **"Hallo, {Nama pengguna yang diinputkan}"** pada pemrograman visual blok menggunakan .

Sedangkan dalam pemrograman tekstual:

- bahasa C menggunakan sintaks: `printf("Hallo,%s\n", nama)`,
- bahasa Python menggunakan perintah:
`printf("Hallo, {nama}") {nama}"`.

Konsep Variabel

✦ Pemrograman Visual Blok Scratch

Kode



✦ Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

int main(void)
{
    int penjumlahan = 3+2;
    int pengurangan = 5-2;
    int perkalian = 4*3;
    int pembagian = 6/2;
}
```

✦ Pemrograman Tekstual Bahasa Python

Kode

```
penjumlahan = 3 + 2
pengurangan = 5 - 2
perkalian = 4 * 3
pembagian = 6 / 2
```

✦ Deskripsi Program

Program tersebut merupakan konsep variabel untuk melakukan operasi aritmatika seperti penjumlahan, pengurangan, perkalian, dan pembagian.

Konsep Percabangan

Pemrograman Visual Blok Scratch

Kode



Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

int main(void)
{
  if (x < y)
  {
    printf("x kurang dari y\n");
  }
  else if (x > y)
  {
    printf("x lebih dari y\n");
  }
  else
  {
    printf("x sama dengan y\n");
  }
}
```

Pemrograman Tekstual Bahasa Python

Kode

```
if x < y:
    print("x kurang dari y")
elif x > y:
    print("x lebih dari y")
else:
    print("x sama dengan y")
```

Deskripsi Program

Program tersebut merupakan konsep percabangan untuk membandingkan nilai x dan y . Apakah nilai tersebut kurang dari, lebih besar, atau sama dengan.

Konsep Perulangan

Pemrograman Visual Blok Scratch

Kode



Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

int main(void)
{
    for (int i = 0; i < 3; i++)
    {
        printf("Meong\n");
    }
}
```

```
#include <stdio.h>

int main(void)
{
    int i = 0;
    while (i < 3)
    {
        printf("Meong\n");
        i++;
    }
}
```

Pemrograman Tekstual Bahasa Python

Kode

```
for i in range(3):
    print("Meong")
```

```
i = 0
while i < 3:
    print("Meong")
    i += 1
```

Deskripsi Program

Program tersebut merupakan konsep perulangan, dimana dalam pemrograman visual blok menggunakan blok pengulangan **repeat** untuk mengulang pesan "Meong" sebanyak tiga kali. Dalam pemrograman tekstual baik menggunakan bahasa C atau bahasa Python, menggunakan konsep perulangan **for** dan **while**.

Konsep Modularisasi Program

Pemrograman Visual Blok Scratch

Kode

```
define meong n kali
repeat n
  play sound Meow until done

when clicked
  meong 3 kali
```

Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

void meong(int n)
{
    for (int i = 0; i < n; i++)
    {
        printf("Meong\n");
    }
}

int main(void)
{
    meong(3);
}
```

Pemrograman Tekstual Bahasa Python

Kode

```
def meong(n):
    for i in range(n):
        print("Meong")

if __name__ == "__main__":
    meong(3)
```

Deskripsi Program

Program tersebut merupakan konsep modularisasi program. Terdapat *procedure* *meong* yang menerima parameter *n* dan akan menampilkan pesan "meong" sebanyak *n* kali. Prosedur ini kemudian dipanggil dalam program utama dengan memasukan nilai parameter sebanyak 3, yang artinya pesan "meong" akan ditampilkan sebanyak tiga kali.

Konversi dari Scratch ke bahasa pemrograman tekstual seperti bahasa C dan Python melibatkan penerjemahan blok-blok visual menjadi kode yang lebih eksplisit. Meskipun sintaksis teks lebih kompleks, namun dapat memberikan lebih banyak kontrol dan fleksibilitas dalam pengembangan program. Pemahaman dasar tentang logika pemrograman yang diperoleh dari Scratch sangat berguna dalam transisi ini.

c. Konversi Pemrograman Visual Blok Blockly ke Pemrograman Tekstual

Konversi dari pemrograman visual blok seperti Blockly ke pemrograman tekstual serupa dengan konversi dari Scratch. Blockly juga memungkinkan pengguna untuk menyusun logika program dengan menghubungkan blok-blok visual, tetapi dapat menghasilkan kode tekstual langsung dalam beberapa bahasa pemrograman seperti bahasa C, Python, atau lainnya. Berikut adalah konversi konsep dasar dari Blockly ke bahasa pemrograman tekstual.

Konsep *Input* dan *Output*



Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

int main(void)
{
    printf("Halo,
salam informatika!\n");
}
```

Output

```
Hallo, salam informatika!
```

Pemrograman Tekstual Bahasa Python

Kode

```
print
('Hallo, salam informatika!')
```

Output

```
Hallo, salam informatika!
```

Deskripsi Program

Program tersebut menampilkan pesan teks **“Hallo, salam informatika”**. Coba perhatikan perbedaannya! Dalam pemrograman visual blok, perintah disusun oleh blok



Sedangkan dalam pemrograman tekstual:

- Bahasa C menggunakan sintaks: `printf(“Hallo, salam informatika!\n”)`,
- Bahasa Python menggunakan perintah: `print(‘Hallo, salam informatika!’)`.

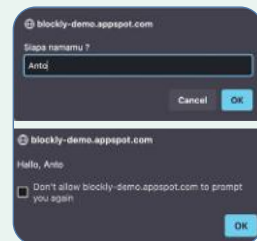
Konsep *Input* dan *Output*

Pemrograman Visual Blok Scratch

Kode



Output



Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

int main(void)
{
    char nama[50];
    printf(“Siapa namamu ? “);
    scanf(“%s”, &nama);
    printf(“Hallo,%s\n”, nama);
}
```

Output

```
Siapa namamu ? Anto
Hallo, Anto
```

Pemrograman Tekstual Bahasa Python

Kode

```
nama = input(“Siapa namamu ? “)
printf(“Hallo, {nama}”)
```

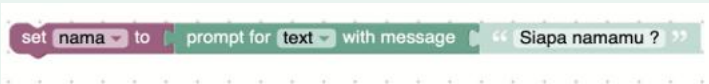
Output

```
Siapa namamu ? Anto
Hallo, Anto
```

Deskripsi Program

Program tersebut menerima *input* dari pengguna berupa pertanyaan **“Siapa namamu?”**, lalu mengeluarkan pesan teks berupa **“Hallo, {Nama pengguna yang diinputkan}”**. Perhatikan perbedaannya!

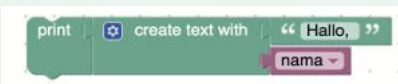
Dalam pemrograman visual blok, perintah disusun oleh blok untuk menampilkan pertanyaan **“Siapa namamu?”**.



Sedangkan dalam pemrograman tekstual:

- Bahasa C menggunakan sintaks: `printf(“Siapa namamu?”)` dan `scanf(“%s”, &nama)`,
- Bahasa Python menggunakan perintah: `nama=input(“Siapa namamu?”)`.

Untuk pesan teks berupa **“Hallo, {Nama pengguna yang diinputkan}”** pada pemrograman visual blok menggunakan



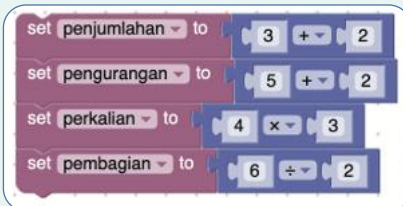
Sedangkan dalam pemrograman tekstual:

- Bahasa C menggunakan sintaks: `printf(“Hallo,%s\n”, nama)`,
- Bahasa Python menggunakan perintah: `nama=printf(“Hallo, {nama}”)`.

Konsep Variabel

Pemrograman Visual Blok Scratch

Kode



Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

int main(void)
{
int penjumlahan = 3+2;
int pengurangan = 5-2;
int perkalian = 4*3;
int pembagian = 6/2;
}
```

Pemrograman Tekstual Bahasa Python

Kode

```
penjumlahan = 3 + 2
pengurangan = 5 - 2
perkalian = 4 * 3
pembagian = 6 / 2
```

Deskripsi Program

Program tersebut merupakan konsep variabel untuk melakukan operasi aritmatika seperti penjumlahan, pengurangan, perkalian, dan pembagian.

Konsep Percabangan

Pemrograman Visual Blok Scratch

Kode



Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

int main(void)
{
    if (x < y)
    {
        printf("x kurang dari y\n");
    }
    else if (x > y)
    {
        printf("x lebih dari y\n");
    }
    else
    {
        printf("x sama dengan y\n");
    }
}
```

Pemrograman Tekstual Bahasa Python

Kode

```
if x < y:
    print("x kurang dari y")
elif x > y:
    print("x lebih dari y")
else:
    print("x sama dengan y")
```

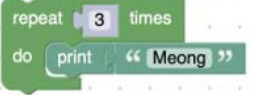
Deskripsi Program

Program tersebut merupakan konsep percabangan untuk membandingkan nilai x dan y. Apakah nilai tersebut kurang dari, lebih besar, atau sama dengan.

Konsep Perulangan

Pemrograman Visual Blok Scratch

Kode



```
repeat 3 times
do print "Meong"
```

Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

int main(void)
{
    for (int i = 0; i < 3; i++)
    {
        printf("Meong\n");
    }
}
```

Pemrograman Tekstual Bahasa Python

Kode

```
for i in range(3):
    print("Meong")
```

Pemrograman Visual Blok Scratch

Kode



```
count with i from 1 to 3 by 1
do print "Meong"
```

Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

int main(void)
{
    int i = 0;
    while (i < 3)
    {
        printf("Meong\n");
        i++;
    }
}
```

Pemrograman Tekstual Bahasa Python

Kode

```
i = 0
while i < 3:
    print("Meong")
    i += 1
```

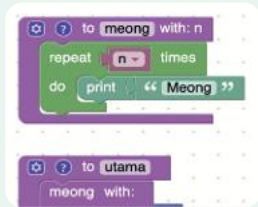
Deskripsi Program

Program tersebut merupakan konsep perulangan. Dalam pemrograman visual blok menggunakan blok pengulangan **repeat** untuk mengulang pesan "Meong" sebanyak tiga kali. Sedangkan dalam pemrograman tekstual baik bahasa C maupun bahasa Python menggunakan konsep perulangan **for** dan juga **while**.

Konsep Modularisasi Program

Pemrograman Visual Blok Scratch

Kode



Pemrograman Tekstual Bahasa C

Kode

```
#include <stdio.h>

void meong(int n)
{
    for (int i = 0; i < n; i++)
    {
        printf("Meong\n");
    }
}

int main(void)
{
    meong(3);
}
```

Pemrograman Tekstual Bahasa Python

Kode

```
def meong(n):
    for i in range(n):
        print("Meong")

if __name__ == "__main__":
    meong(3)
```

Deskripsi Program

Program tersebut merupakan contoh konsep modularisasi program, dimana terdapat *procedure* bernama *meong* yang menerima parameter sebanyak *n*. Prosedur ini akan mengulang perintah untuk menampilkan pesan "Meong" sebanyak *n* kali. Dalam program utama, prosedur *meong* dipanggil dengan memasukkan nilai parameter 3, yang berarti pesan "Meong" akan ditampilkan sebanyak tiga kali.

Konversi dari Blockly ke bahasa pemrograman tekstual seperti Bahasa C dan Python adalah proses menerjemahkan blok-blok visual ke dalam kode teks yang eksplisit. Blockly mempermudah proses ini karena menyediakan fitur untuk melihat langsung representasi kode dari blok yang disusun. Namun, pemahaman tentang bagaimana blok-blok tersebut diterjemahkan ke dalam kode teks sangat penting untuk pengembangan lebih lanjut dan penguasaan pemrograman di lingkungan yang lebih kompleks.

d. *Flowchart*

Diagram alir (*flowchart*) adalah representasi visual dari urutan langkah atau aliran proses pada algoritma. *Flowchart* menggunakan simbol-simbol grafis untuk menunjukkan aksi (proses), aliran aksi, keputusan, dan pengulangan. *Flowchart* juga digunakan dalam dunia bisnis, teknologi, dan manajemen untuk menggambarkan langkah-langkah yang terlibat dalam suatu proses atau sistem. *Flowchart* membantu memahami alur kerja dan dapat digunakan untuk analisis, dokumentasi, perbaikan proses, atau pelatihan. Berikut adalah simbol-simbol dalam *flowchart* beserta deskripsinya.



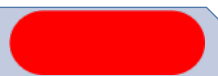
Garis Alir (*flowline*)



Anak panah berarah yang menunjukkan arah aliran langkah



Terminator



Titik terminasi *flowchart*, yang berupa titik awal dan titik akhir



Langkah



Suatu langkah yang akan dilaksanakan oleh eksekutor



Keputusan



Titik percabangan yang membuat aliran langkah akan bercabang dengan kondisi tertentu



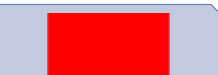
Masukan (*input*)/keluaran (*output*)



Melambatkan titik saat langkah akan menerima *input* atau menghasilkan *output*



Subprogram



Melambatkan suatu kegiatan atau langkah lain yang telah didefinisikan sebelumnya



Penghubung antartitik



Digunakan untuk menghubungkan suatu titik pada *flowchart* ke titik lain pada halaman yang sama

Penghubung antarhalaman



- Digunakan untuk menghubungkan suatu titik pada *flowchart* ke titik lain pada halaman yang berbeda
- Digunakan apabila diagram lain cukup kompleks sehingga tidak dapat digambar dalam satu halaman

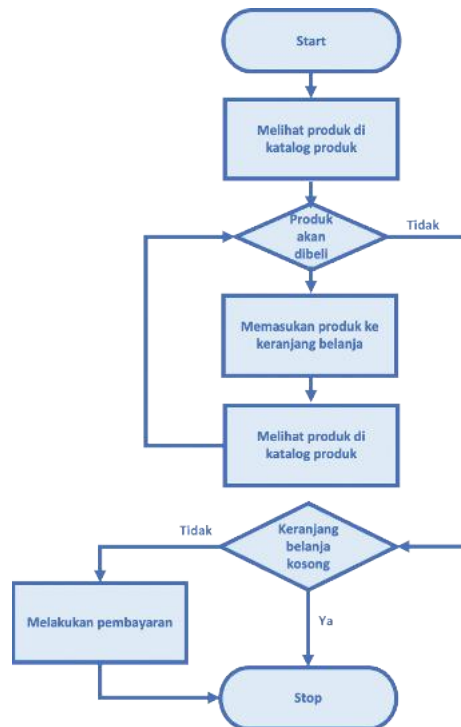
Untuk memahami bagaimana *flowchart* digunakan untuk menggambarkan suatu algoritma, perhatikan contoh beberapa *flowchart* berikut ini.

1) **Flowchart Proses Pembelian Produk yang Dilakukan oleh Pembeli**

Flowchart pada Gambar 2.9, menggambarkan aliran proses dengan tingkat abstraksi tinggi sehingga tidak dapat langsung diterjemahkan ke dalam aksi yang dapat dieksekusi bahasa pemrograman. Berbeda halnya dengan contoh berikut, yang menggambarkan proses pembelian produk seperti yang mungkin pernah dilakukan oleh orang tuamu saat berbelanja *online*, atau yang pernah kamu lakukan di warung, minimarket, atau supermarket. Berikut adalah contoh *flowchart* untuk proses pembelian produk.

Diagram alir dimulai dari simbol **Start** atau **mulai**

Pembeli melihat produk yang tersedia di katalog produk.



Gambar 2.9 *Flowchart* Proses Pembelian Produk

Pembeli akan mendapatkan pertanyaan, “**Apakah produk yang dilihat akan dibeli?**”

Jika **Tidak**, maka pengguna akan menuju pengecekan keranjang belanja dan jika keranjang belanja kosong maka diagram alir **Stop** atau **selesai**.

Jika **Ya**, maka pembeli akan memasukkan produk tersebut ke dalam keranjang belanja. Setelah itu, pembeli melihat produk lain di katalog produk dan akan ditanyakan kembali: “**Apakah produk akan dibeli?**”

Jika **Ya**, maka produk akan ditambahkan pada keranjang belanja lagi, sampai tidak ada lagi produk yang akan dibeli.

Setelah itu, sistem akan memeriksa keranjang belanja. Jika keranjang belanja tidak kosong, pembeli dapat melanjutkan ke proses pembayaran.

Jika pembayaran telah selesai dilaksanakan, maka *flowchart* **Stop** atau **selesai**.

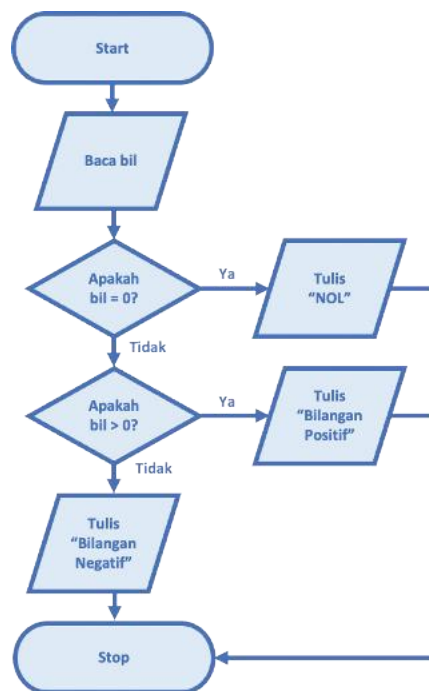
2) *Flowchart* Penentuan Bilangan Nol, Positif, atau Negatif

Contoh *flowchart* pada Gambar 2.10 adalah *flowchart* untuk sebuah program yang dapat menentukan apakah suatu bilangan yang diinputkan adalah nol, positif, atau negatif. Diagram alir ini siap ditranslasi ke program dalam bahasa pemrograman.

Diagram alir dimulai dari simbol **Start** atau **mulai**.

Program akan meminta pengguna untuk memasukkan sebuah bilangan yang akan disimpan pada variabel **bil**.

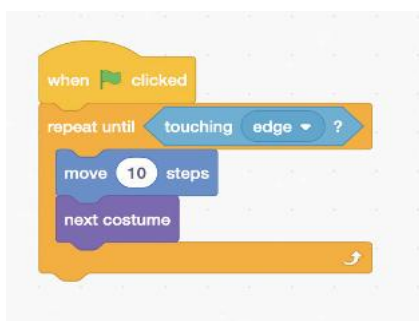
Bilangan **bil** akan diperiksa apakah bernilai 0. Jika **Ya** maka program akan menuliskan **NOL**. Jika **Tidak**, maka bilangan **bil** akan diperiksa kembali apakah lebih besar dari 0. Jika **Ya**, maka program akan menuliskan **Bilangan Positif**. Jika **Tidak**, program akan menuliskan **Bilangan Negatif**.



Gambar 2.10 *Flowchart* Penentuan Bilangan Nol, Positif, atau Negatif

3) **Flowchart dan Program dalam Bahasa Pemrograman Visual Scratch**

Bahasa pemrograman yang dipelajari di fase D adalah bahasa pemrograman visual berbasis blok. Bahasa ini mudah dipahami karena berbentuk visual. Selain itu, pendekatan visual juga digunakan melalui *flowchart*, yaitu representasi algoritma dalam bentuk simbol-simbol yang mempermudah pemahaman. Gambar 2.11 menampilkan contoh program yang dibuat menggunakan Scratch. Saat dijalankan, program ini membuat *sprite* kucing bergerak dari posisi awal hingga posisi akhir sebagaimana terlihat pada Gambar 2.12.

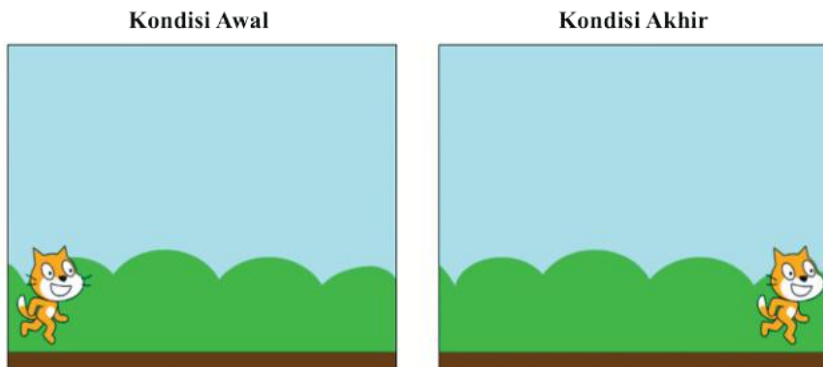


(a) Kode dalam Program Scratch

(b) Algoritma dalam *Flowchart*

Gambar 2.11 Program Blok Visual dan *Flowchart*

Untuk hasil *output* dari kode program blok Scratch di atas, dapat dilihat pada Gambar 2.12 yang menjelaskan kondisi awal dan kondisi akhir program.



Gambar 2.12 Kondisi Awal dan Kondisi Akhir Program dalam Scratch



Aktivitas IX-BH-AP-B01-I



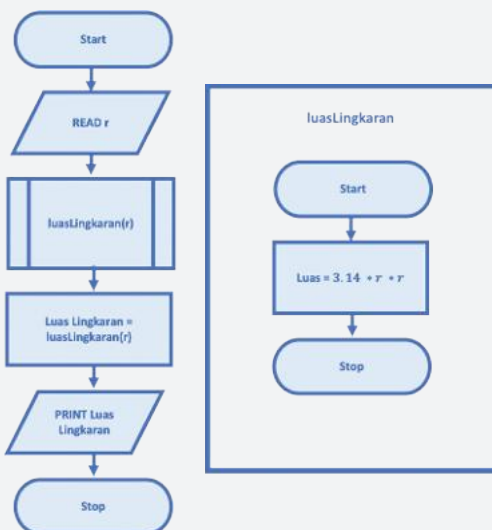
Aktivitas Individu

Penelusuran *Flowchart* LuasLingkaran

- ✔ Tujuan: Mampu melakukan penelusuran *flowchart* sederhana untuk menghitung luas lingkaran dan menganalisis hasil keluaran *flowchart*.
- ✔ Syarat pengetahuan: Konsep lingkaran dan luas lingkaran.

Pada aktivitas ini, kamu diminta untuk menelusuri *flowchart* sederhana yang digunakan untuk menghitung luas lingkaran seperti di bawah ini.

Saat menyusun solusi untuk menyelesaikan suatu permasalahan, kamu seringkali membutuhkan solusi dari permasalahan lain yang lebih sederhana. Misalnya, saat harus menghitung luas sebuah lingkaran. Pada prosesnya, kamu



Gambar 2.13 Flowchart Menghitung Luas Lingkaran

perlu menghitung luas lingkaran tersebut. Hal ini dapat digambarkan melalui *flowchart* menggunakan simbol subprogram atau modul program. Pada *flowchart* ini, terlihat bahwa proses pemrograman akan memanggil subprogram atau modul program **menghitung luas lingkaran** yang telah kamu buat sebelumnya. Subprogram tersebut menggambarkan penerapan konsep abstraksi dan dekomposisi yang telah kamu pelajari pada berpikir komputasional.

Flowchart utama pada gambar 2.13 dapat ditulis sebagai berikut.

INPUT: r

PROSES: memanggil subprogram atau modul `luasLingkaran`

OUTPUT: `LuasLingkaran`

Diagram alir dari subprogram dapat ditulis sebagai:

INPUT: r

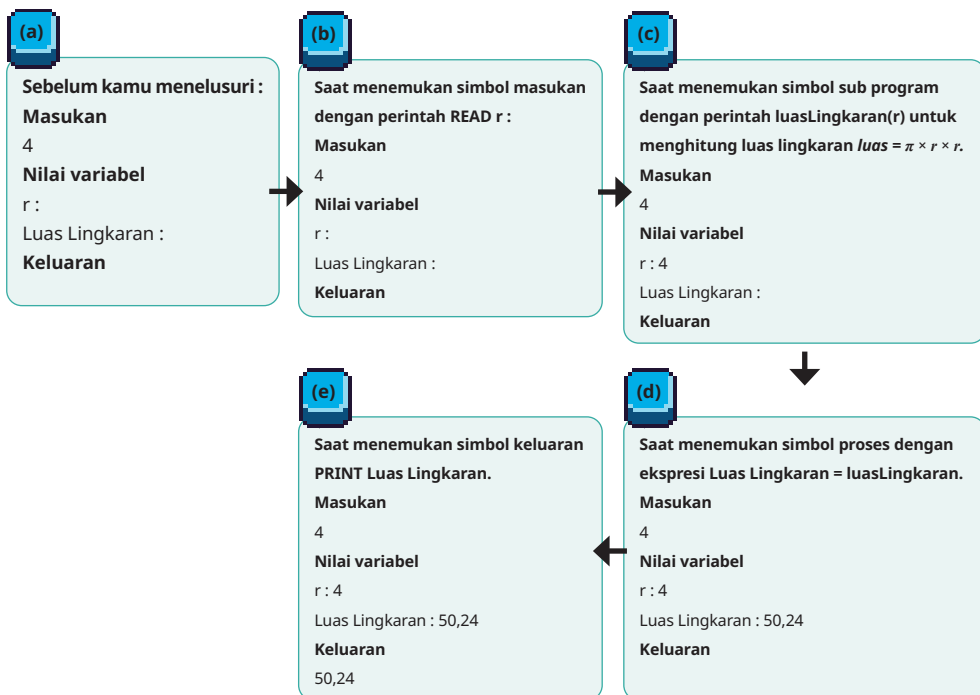
PROSES: $Luas = 3.14 * r * r$

OUTPUT: `Luas`



Untuk menelusuri *flowchart* tersebut, siapkan lembar kerja seperti yang ditunjukkan pada Gambar 2.14. Sebagai contoh, kamu dapat mencari luas lingkaran dengan jari-jari (r) = 4 menggunakan *flowchart* di atas. Berikut adalah langkah-langkah yang harus kamu lakukan dalam penelusuran.

1. Siapkan lembar kerja penelusuran (Gambar 2.14 (a)), dengan menuliskan data yang akan diolah dan nama variabel yang digunakan pada *flowchart*. Kosongkan bagian keluaran. Setelah itu, kamu mulai telusuri *flowchart* dari bagian awal.
2. Ketika menemukan simbol untuk membaca suatu data, kamu dapat mencoret masukan yang dibaca, kemudian meletakkannya ke variabel yang tepat. Misal, kamu berada pada simbol masukan **READ** r . Letakkan bilangan pertama di bagian masukan (yaitu 4) ke variabel r (Gambar 2.14 (b)).
3. Ketika kamu tiba pada simbol proses yang terhubung dengan subprogram atau modul luasLingkaran(r), artinya kamu akan menjalankan perintah untuk menghitung luas lingkaran menggunakan rumus $luas = \pi \times r \times r$. Nilai r diisi dengan 4, sehingga hasil perhitungannya adalah $luas = \pi \times 4 \times 4 = 50,24$ (Gambar 2.14 (c)).
4. Proses menugaskan variabel LuasLingkaran diisi oleh hasil perhitungan subprogram atau modul program luasLingkaran (r) dengan hasil 50,24, maka nilai luas lingkaran = 50,24. Lembar kerja (Gambar 2.14 (d)).
5. Kemudian, kamu menemukan simbol keluaran **PRINT** LuasLingkaran. Pada tahap ini, kamu dapat menuliskan isi dari variabel LuasLingkaran ke dalam bagian keluaran di lembar kerjamu (Gambar 2.14 (e)).
6. Terakhir, kamu akan menemukan simbol terminator Stop yang menandakan penelusuran telah selesai. *Flowchart* tersebut menghasilkan keluaran berupa luas lingkaran = 50,24. Selamat! Kamu berhasil menelusuri *flowchart* dengan benar.



Gambar 2.14 Ilustrasi Penelusuran *Flowchart* di Buku Kerja Siswa

Ayo, Berpikir Kritis!

Cobalah untuk mengubah nilai parameter variabel r pada *flowchart* di atas. Lakukan penelusuran seperti lembar kerja sebelumnya dan isilah tabel berikut sesuai hasil penelusuranmu.

Tabel 2.25 Eksperimen Terhadap Rumus Luas Lingkaran Berdasarkan *Flowchart*

No	Parameter	Hasil Perhitungan Manual dari <i>Flowchart</i>
	r	
1	9	
2	10	
3	11	

Kesimpulan apa yang dapat kamu tangkap berdasarkan tabel 2.25? Tuliskan pada jurnalmu, ya.



- ✔ Tujuan: Mampu melakukan penelusuran *flowchart* sederhana untuk menghitung luas persegi dan luas permukaan kubus serta menganalisis hasil keluaran *flowchart*.
- ✔ Syarat pengetahuan: Konsep persegi, luas persegi, dan luas permukaan kubus.

Flowchart 1: Menghitung Luas Persegi

Flowchart pada gambar 2.15 dapat dibaca sebagai berikut.

1. *Flowchart* dibaca mulai dari simbol **Start** lalu mengikuti arah panah.
2. Untuk menghitung luas persegi, kamu memerlukan sebuah data, yaitu panjang sisi. Panjang sisi ini dibaca pada *flowchart* dengan menggunakan kata kunci **READ** dan disimpan di dalam sebuah variabel bernama **sisi**.
3. Setelah itu, kamu melakukan suatu proses ekspresi matematika untuk menghitung luas persegi menggunakan rumus yang telah kamu ketahui, yaitu **luas Persegi = sisi x sisi**. Hasil perhitungan tersebut disimpan pada sebuah variabel bernama **LuasPersegi**.
4. Walaupun telah memperoleh jawaban yang dicari, komputer perlu diinstruksikan secara spesifik untuk mengeluarkan jawaban tersebut. Pada diagram ini, kata kunci **PRINT** digunakan untuk mencetak nilai dari variabel **luas** yang telah diperoleh pada tahap sebelumnya.
5. Setelah **PRINT**, algoritma berakhir karena simbol berikutnya adalah simbol **END (Stop)**.



Gambar 2.15 *Flowchart* Menghitung Luas Persegi

Secara ringkas, kamu pun dapat menulis *flowchart* tersebut dalam tiga bagian, yaitu masukan (*input*), proses, dan luaran (*output*) menjadi:

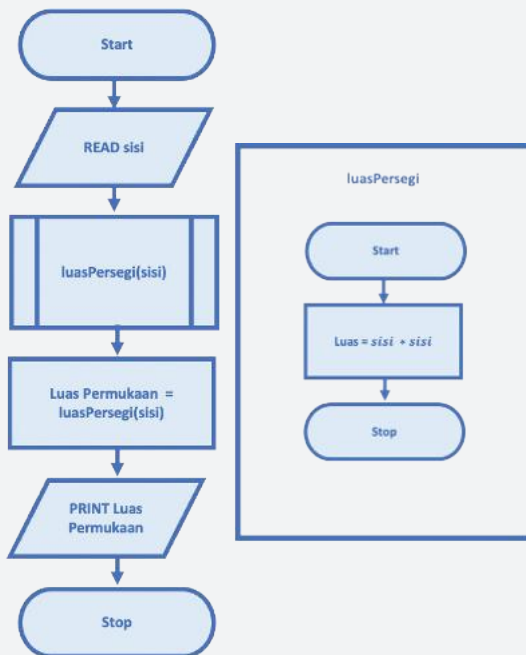
INPUT: sisi

PROSES: LuasPersegi=sisi*sisi

OUTPUT: LuasPersegi

Flowchart 2: Menghitung Luas Permukaan Kubus

Saat menyusun solusi untuk menyelesaikan suatu permasalahan, kamu seringkali membutuhkan solusi dari permasalahan lain yang lebih sederhana. Misalnya, untuk menghitung luas permukaan kubus, kamu perlu menghitung luas salah satu sisi persegi yang membentuk kubus tersebut. Hal ini dapat digambarkan pada *flowchart* menggunakan simbol subprogram atau modul program. Pada *flowchart* ini, terlihat bahwa proses akan memanggil subprogram atau modul program menghitung luas persegi yang telah kamu buat sebelumnya. Subprogram tersebut menggambarkan penerapan konsep abstraksi dan dekomposisi yang telah kamu pelajari pada berpikir komputasional.



Gambar 2.16 Flowchart Menghitung Luas Permukaan Kubus

Flowchart pada gambar 2.16 dapat ditulis sebagai berikut.

INPUT: sisi

PROSES: luas*6

OUTPUT: LuasPermukaan

Diagram alir dari subprogram dapat ditulis sebagai:

INPUT: sisi

PROSES: Luas=sisi*sisi

OUTPUT: LuasPersegi

Ayo, Berpikir Kritis!

Sebelum mencoba membuat *flowchart* sendiri, kamu perlu memiliki kemampuan untuk membaca *flowchart* yang tersedia. Oleh karena itu, pada latihan ini kamu akan mencoba menelusuri *flowchart* yang telah disediakan berdasarkan kasus yang diberikan. Walaupun kamu mungkin telah mengetahui apa hasilnya, lakukan penelusuran secara terstruktur mulai dari awal hingga akhir *flowchart*.

1. Hitunglah luas persegi dengan panjang sisi 3 cm menggunakan *flowchart* pertama.
2. Hitunglah luas permukaan kubus yang memiliki panjang sisi 20 cm menggunakan *flowchart* kedua.
3. Sajikan langkah pekerjaanmu dalam lembar kerja, seperti yang telah dicontohkan dalam kegiatan aktivitas.



Aktivitas

IX-BK-AP-B03-I



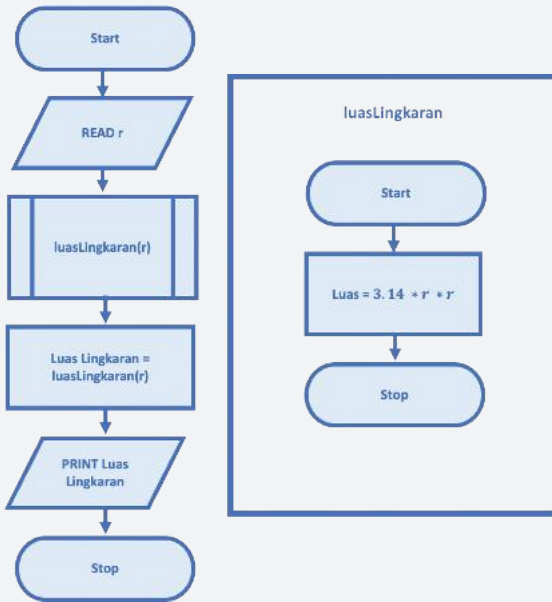
Aktivitas Individu

Translasi *Flowchart* ke Kode Program Visual Blok dan Kode Program Tekstual Luas Lingkaran

- ✔ Tujuan: Mampu mentranslasi *flowchart* sederhana untuk menghitung luas lingkaran ke kode program visual blok dan kode program tekstual.
- ✔ Syarat pengetahuan: Konsep lingkaran dan luas lingkaran.

Pada aktivitas ini, kamu diminta untuk mentranslasi program dari *flowchart* sederhana untuk menghitung luas lingkaran seperti di bawah ini.

Saat menyusun solusi untuk menyelesaikan suatu permasalahan, kamu seringkali membutuhkan solusi dari permasalahan lain yang lebih sederhana. Misalnya, saat harus menghitung luas sebuah lingkaran. Pada prosesnya, kamu perlu menghitung luas lingkaran tersebut. Hal ini dapat digambarkan melalui *flowchart* menggunakan simbol subprogram atau modul program. Pada *flowchart* ini, terlihat bahwa proses akan memanggil subprogram atau modul program **menghitung luas lingkaran** yang telah kamu buat sebelumnya. Subprogram tersebut menggambarkan penerapan konsep abstraksi dan dekomposisi yang telah kamu pelajari pada berpikir komputasional.



Gambar 2.17 Flowchart Menghitung Luas Lingkaran

Flowchart pada gambar 2.17 dapat ditulis sebagai berikut.

INPUT: r

PROSES: memanggil subprogram atau modul luasLingkaran

OUTPUT: luasLingkaran

Diagram alir dari subprogram dapat ditulis sebagai:

INPUT: r

PROSES: $Luas = 3.14 * r * r$

OUTPUT: Luas

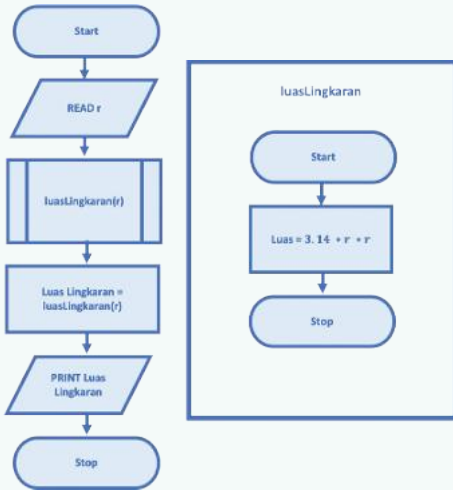
Untuk mentranslasi flowchart tersebut, siapkan lembar kerja seperti yang ditunjukkan pada flowchart berikut ini.

Tabel 2.26 Translasi Flowchart ke Kode Pemrograman Visual Blok

Flowchart	Kode Pemrograman Visual Blok
<p>1. Langkah pertama yaitu mentranslasi flowchart subprogram atau modul program menghitung luas lingkaran.</p> <p style="text-align: center;">luasLingkaran</p>	<p>1. Kode program visual blok berdasarkan flowchart di samping adalah:</p> <p>Kode program visual blok Scratch</p> <p>Kode program Blockly</p>

Flowchart

2. Langkah kedua yaitu mentranslasi *flowchart* program utama menghitung luas lingkaran.



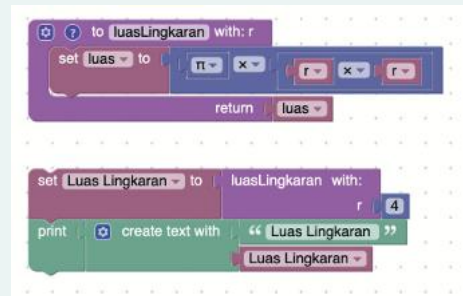
Kode Pemrograman Visual Blok

2. Kode program visual blok berdasarkan *flowchart* di samping adalah:

Kode program visual blok Scratch




Kode program Blockly



Sedangkan, translasi dari kode pemrograman visual blok ke kode pemrograman tekstual dapat dilihat pada konversi kode di bawah ini. Kamu dapat menggunakan <https://bukupusbuk.id/s/3ulcga> pada saat mengkode ke program tekstual.

Tabel 2.27 Translasi Kode Pemrograman Visual Blok ke Pemrograman Tekstual

Kode Pemrograman Visual Blok	Kode Pemrograman Tekstual
<p>1. Kode program visual blok berdasarkan <i>flowchart</i> adalah:</p> <p>Kode program visual blok Scratch</p>  <p>Kode program Blockly</p> 	<p>1. Kode program tekstual berdasarkan kode program visual blok di samping adalah:</p> <p>Kode program tekstual bahasa C</p> <pre>double luasLingkaran(double r) { luas = M_PI * (r * r); return luas; }</pre> <p>Kode program tekstual bahasa Python</p> <pre>def luasLingkaran(r): global Luas_Lingkaran, luas luas = math.pi * (r * r) return luas</pre>
<p>2. Kode program visual blok berdasarkan <i>flowchart</i> adalah:</p> <p>Kode program visual blok Scratch</p>  	<p>2. Maka kode program tekstual berdasarkan kode program visual blok di samping adalah:</p> <p>Kode program tekstual bahasa C</p> <pre>int main() { Luas_Lingkaran = luasLingkaran(4); printf("Luas Lingkaran %.2f\n", Luas_Lingkaran); return 0; }</pre> <p>Kode program tekstual bahasa Python</p> <pre>Luas_Lingkaran = luasLingkaran(4) print('Luas Lingkaran ' + str(Luas_Lingkaran))</pre>

Kode Pemrograman Visual Blok

Kode Pemrograman Tekstual

Kode program Blockly

```
function to luasLingkaran with: r
  set luas to pi * r * r
  return luas
end

set Luas Lingkaran to luasLingkaran with: r and 4
print create text with "Luas Lingkaran" and Luas Lingkaran
```

Ayo, Berpikir Kritis!

Kesimpulan apa yang dapat kamu tangkap berdasarkan Tabel 2.27? Tuliskan pada jurnalmu, ya.

Aktivitas

IX-BK-AP-B04-I

Aktivitas Individu

Translasi *Flowchart* ke Kode Program Visual Blok dan Kode Program Tekstual Menghitung Luas Persegi dan Luas Permukaan Kubus

- ✔ Tujuan: Mampu melakukan translasi dari *flowchart* sederhana untuk menghitung luas persegi dan luas permukaan kubus ke kode program visual blok dan kode program tekstual.
- ✔ Syarat pengetahuan: Konsep persegi, luas persegi, dan luas permukaan kubus.

Flowchart 1: Menghitung Luas Persegi

Flowchart pada gambar 2.18 dapat dibaca sebagai berikut.

1. *Flowchart* dibaca mulai dari simbol **Start** lalu mengikuti arah panah.
2. Untuk menghitung luas persegi, kamu memerlukan sebuah data, yaitu panjang sisi. Panjang sisi ini dibaca pada *flowchart* dengan menggunakan kata kunci **READ** dan disimpan di dalam sebuah variabel bernama *sisi*.

3. Setelah itu, kamu melakukan suatu proses ekspresi matematika untuk menghitung luas persegi menggunakan rumus yang telah kamu ketahui, yaitu $luas = sisi \times sisi$. Hasil perhitungan tersebut disimpan pada sebuah variabel bernama `LuasPersegi`.
4. Walaupun telah memperoleh jawaban yang dicari, komputer perlu diinstruksikan secara spesifik untuk mengeluarkan jawaban tersebut. Pada diagram ini, kata kunci **PRINT** digunakan untuk mencetak nilai dari variabel `LuasPersegi` yang telah diperoleh pada tahap sebelumnya.
5. Setelah **PRINT**, algoritma berakhir karena simbol setelahnya adalah simbol **END**.



Gambar 2.18 Flowchart Menghitung Luas Persegi

Secara ringkas, kamu pun dapat menulis *flowchart* tersebut dalam tiga bagian, yaitu masukan (*input*), proses, dan luaran (*output*) menjadi:

INPUT: `sisi`

PROSES: `luas = sisi * sisi`

OUTPUT: `LuasPersegi`

Flowchart 2: Menghitung Luas Permukaan Kubus

Saat kamu menyusun solusi untuk menyelesaikan suatu masalah, seringkali kamu membutuhkan solusi dari permasalahan lain yang lebih sederhana. Misalnya, untuk menghitung luas permukaan sebuah kubus, kamu perlu menghitung luas persegi yang membentuk kubus tersebut. Proses ini dapat digambarkan pada *flowchart* menggunakan simbol subprogram atau modul program. Pada *flowchart* ini, terlihat bahwa proses akan memanggil subprogram atau modul program menghitung luas persegi yang telah kamu buat sebelumnya. Subprogram tersebut menggambarkan konsep abstraksi dan dekomposisi yang telah kamu pelajari pada berpikir komputasional.

Flowchart pada gambar 2.19 dapat ditulis sebagai berikut.

INPUT: sisi

PROSES: luas * 6

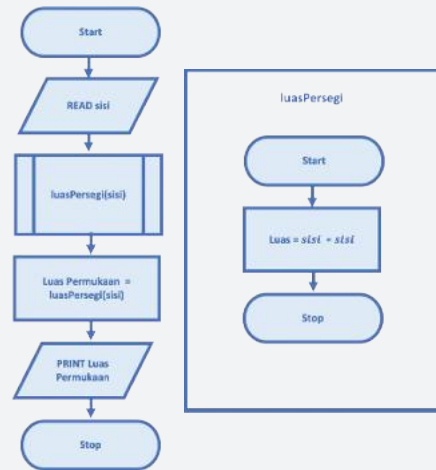
OUTPUT: LuasPermukaan

Diagram alir dari subprogram dapat ditulis sebagai:

INPUT: sisi

PROSES: Luas = sisi * sisi

OUTPUT: Luas



Gambar 2.19 Flowchart Menghitung Luas Permukaan Kubus

Ayo, Berpikir Kritis!

Setelah kamu berlatih menelusuri *flowchart* pada aktivitas **IX-BK-AP-B03-I**, ini saatnya kamu mentranslasikan *flowchart* tersebut ke dalam kode pemrograman visual blok dan kode pemrograman tekstual seperti yang dicontohkan dalam aktivitas **IX-BK-AP-B04-I Translasi *flowchart* ke Kode Program Visual Blok dan Kode Program Tekstual**. Dengan ketentuan sebagai berikut.

1. Hitunglah luas persegi dengan panjang sisi 3 cm menggunakan *flowchart* pertama.
2. Hitunglah luas permukaan kubus yang memiliki panjang sisi 20 cm menggunakan *flowchart* kedua.
3. Sajikan langkah pekerjaanmu dalam lembar kerja, seperti yang telah dicontohkan dalam kegiatan aktivitas.

2. Analisis Pseudocode

Pseudocode adalah versi algoritma atau kode program yang disederhanakan agar lebih mudah dibaca dan dipahami oleh manusia. Pada fase D, pendekatan pemrograman yang digunakan adalah prosedural tekstual. *Pseudocode* berfungsi sebagai kerangka awal solusi, yang nantinya dikembangkan menjadi kode program yang lebih rinci dan dapat dijalankan oleh komputer. Instruksi dalam *pseudocode* bersifat umum dan belum siap dieksekusi oleh komputer. Contohnya, seorang pemrogram dapat menuliskan sebagai berikut.

```

FOR sekumpulan nama dalam daftar
IF kata yang dibaca adalah ADI THEN
    RETURN "Data ditemukan"
ENDIF
ENDFOR

```

Pada teks *pseudocode* di atas, terdapat beberapa kata yang biasa digunakan dalam bahasa pemrograman (*reserved word*) dan dituliskan dalam huruf kapital. Selain itu, terdapat juga kata-kata yang ditulis dengan bahasa natural manusia. *Pseudocode* merupakan bentuk perantara bahasa natural dan bahasa pemrograman. *Pseudocode* umumnya tidak menggambarkan struktur penyimpanan data (misal *array* atau *list*) maupun nama variabel.



Aktivitas IX-BK-AP-B05-I

Aktivitas Individu

Translasi *Pseudocode* ke Kode Program Visual Blok dan Kode Program Tekstual Luas Lingkaran

- ✓ Tujuan: Mampu mentranslasi *pseudocode* sederhana untuk menghitung luas lingkaran ke kode program visual blok dan kode program tekstual.
- ✓ Syarat pengetahuan: Konsep lingkaran dan luas lingkaran.



Pada aktivitas ini, kamu diminta untuk mentranslasi program dari *pseudocode* sederhana untuk menghitung luas lingkaran seperti di bawah ini.

Deskripsi Naratif	Pseudocode
<ol style="list-style-type: none"> 1. Baca nilai jari-jari (r) lingkaran 2. Hitung luas lingkaran dengan mengalikan konstanta $\pi=3,14$ dengan jari-jari (r) yang dikuadratkan 3. Cetak luas lingkaran 	<p>Algoritma Menghitung Luas Lingkaran</p> <p>INPUT Nilai jari-jari (r) lingkaran</p> <p>PROSES BACA r luasLingkaran $\leftarrow 3,14 * r * r$ CETAK luasLingkaran</p> <p>OUTPUT luasLingkaran tercetak.</p>

Untuk mentranslasi *pseudocode* tersebut, siapkan lembar kerja seperti yang ditunjukkan pada konversi *pseudocode* berikut ini.





Tabel 2.28 Translasi Kode Pemrograman Teksual ke Pemrograman Blok

Pseudocode	Kode Pemrograman Visual Blok
<p>Algoritma Menghitung Luas Lingkaran</p> <p>INPUT Nilai jari-jari (r) lingkaran.</p> <p>PROSES BACA r $luasLingkaran \leftarrow 3,14 * r * r$ CETAK $luasLingkaran$</p> <p>OUTPUT Luas lingkaran tercetak</p>	<p>Kode program visual blok berdasarkan <i>pseudocode</i> di samping adalah:</p> <p>Kode program visual blok Scratch</p>  <p>Kode program Blockly</p> 

Sedangkan, dari kode pemrograman visual blok ke kode pemrograman teksual dapat dilihat pada konversi kode Tabel 2.29. Kamu dapat menggunakan <https://bukupusbuk.id/s/3ulcga> pada saat mengkode ke program teksual.

Tabel 2.29 Translasi Kode Pemrograman Visual Blok ke Pemrograman Teksual

Kode Pemrograman Visual Blok	Kode Pemrograman Visual Teksual
<p>Kode program visual blok berdasarkan <i>pseudocode</i> di samping adalah:</p> <p>Kode program visual blok Scratch</p>  <p>Kode program Blockly</p> 	<p>Kode program teksual berdasarkan kode program visual blok di samping adalah:</p> <p>Kode program teksual bahasa C</p> <pre>luas = M_PI * (r * r);</pre> <p>Kode program teksual bahasa Python</p> <pre>luas = math.pi * (r * r)</pre>



Ayo, Berpikir Kritis!

Kesimpulan apa yang dapat kamu tangkap berdasarkan tabel di atas? Tuliskan pada jurnalmu, ya.



Aktivitas IX-BH-AP-B06-I

Aktivitas Individu

Translasi *Pseudocode* ke Kode Program Visual Blok dan Kode Program Tekstual Luas Persegi dan Luas Permukaan Kubus

- ✓ Tujuan: Mampu melakukan translasi dari *pseudocode* sederhana ke kode program visual blok dan kode program tekstual untuk menghitung luas persegi dan luas permukaan kubus.
- ✓ Syarat pengetahuan: Konsep persegi, luas persegi, dan luas permukaan kubus.

Pseudocode 1: Menghitung Luas Persegi

Deskripsi Naratif	Pseudocode
<ol style="list-style-type: none">1. Baca nilai sisi persegi2. Hitung luas dengan mengkuadratkan nilai sisi nya3. Cetak luas	<p>Algoritma Menghitung Luas Persegi</p> <p>INPUT Nilai panjang sisi persegi</p> <p>PROSES BACA sisi luas ← sisi * sisi CETAK luas</p> <p>OUTPUT Luas persegi tercetak</p>

Pseudocode 2: Menghitung Luas Permukaan Kubus

Deskripsi Naratif	Pseudocode
<ol style="list-style-type: none">1. Baca nilai sisi kubus2. Hitung luas (persegi) dari sisi kubus. Kalikan luas kubus dengan angka 6 (banyaknya jumlah persegi pada kubus) untuk mendapatkan luas permukaan3. Cetak luas permukaan	<p>Algoritma Menghitung Luas Permukaan Kubus</p> <p>INPUT Nilai panjang sisi kubus</p> <p>PROSES BACA sisi luas_permukaan ← luas(sisi) * 6 CETAK luas_permukaan</p> <p>OUTPUT Luas permukaan kubus tercetak</p>



Ayo, Berpikir Kritis!

Setelah kamu berlatih menelusuri *pseudocode* pada aktivitas **IX-BK-AP-B05-I Translasi *Pseudocode* ke Kode Program Visual Blok dan Kode Program Tekstual**, selanjutnya kamu mentranslasikan *pseudocode* tersebut ke dalam kode pemrograman visual blok dan kode pemrograman tekstual dengan ketentuan berikut.

1. Hitunglah luas persegi dengan panjang sisi 3 cm menggunakan *flowchart* pertama.
2. Hitunglah luas permukaan kubus yang memiliki panjang sisi 20 cm menggunakan *flowchart* kedua.
3. Sajikan langkah pekerjaanmu dalam lembar kerja seperti yang telah dicontohkan dalam kegiatan aktivitas.

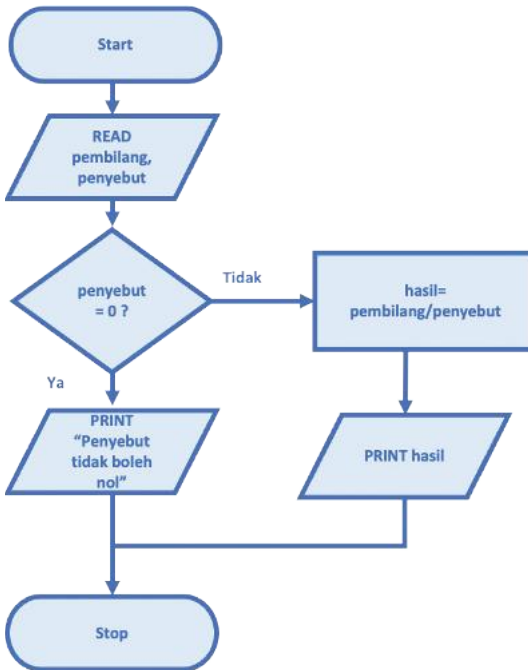
Ayo, Refleksi!

- Jawablah pertanyaan berikut dalam lembar refleksi pada buku kerjamu. Kemudian, catat kegiatanmu dalam jurnal belajar.
- 1. Apakah kamu merasa bahwa membuat algoritma dapat memudahkanmu dalam menyelesaikan masalah?
- 2. Mana yang paling mudah untuk kamu gunakan: *flowchart* atau *pseudocode*?
- 3. Mana yang paling mudah untuk dikerjakan: pemrograman visual blok atau pemrograman tekstual?
- 4. Apa yang akan kamu lakukan agar bisa membaca dan menulis algoritma dengan baik?
- 5. Apa pelajaran paling berkesan yang kamu dapatkan dari pertemuan ini?

Uji Kompetensi

1. Menelusuri flowchart

Perhatikan *flowchart* membagi pembilang dan penyebut di bawah ini.



Gambar 2.20 *Flowchart* untuk Menghitung Membagi Pembilang dengan Penyebut

Flowchart ini merupakan proses untuk membagi **pembilang** dengan **penyebut**. Akan tetapi, sebelum operasi pembagian dilakukan, diagram akan mengecek terlebih dahulu nilai dari **penyebut**. Apabila **penyebut** bernilai **0**, operasi pembagian tidak dilakukan dan pesan yang sesuai akan ditampilkan. Jika tidak, operasi dapat dilakukan dengan aman dan hasil pembagian dapat ditampilkan.

Kamu di minta untuk menelusuri *flowchart* pada lembar kerja sesuai dengan *flowchart* yang ditunjukkan pada Gambar 2.20.

2. Menelusuri pseudocode

Perhatikan deskripsi naratif membagi **pembilang** dan **penyebut** di bawah ini.

Deskripsi Naratif	<i>Pseudocode</i>
1. Baca nilai pembilang dan penyebut .	
2. Jika penyebut bernilai 0 , cetak tulisan "Penyebut tidak boleh nol"	

Deskripsi Naratif	Pseudocode
3. Jika penyebut tidak bernilai nol, lakukan pembagian pembilang dengan penyebut dan simpan hasilnya 4. Cetak hasil pembagian	

Tuliskan *pseudocode* kamu disebelah deskripsi naratif tentang membagi **pembilang** dan **penyebut**. Buatkan kode program visual blok berdasarkan *pseudocode* yang telah kamu rancang.

3. Translasi *pseudocode* menjadi kode program visual blok (membuat modul program atau subprogram)

Perhatikan deskripsi naratif dan *pseudocode* menghitung mundur dari N ke 1 di bawah ini:

Deskripsi Naratif	Pseudocode
1. Baca nilai N 2. Selama $N > 0$ Ulangi 3. Cetak tulisan N 4. Kurangi nilai N dengan 1	INPUT Nilai N PROSES BACA N WHILE $N > 0$ CETAK N $N \leftarrow N - 1$ OUTPUT Angka hasil hitung mundur dari nilai N sampai 1 tercetak

Translasikan *pseudocode* di atas ke dalam bahasa program visual blok menggunakan Scratch atau Blockly.

4. Translasi dari kode program visual blok menjadi kode program tekstual

Translasikan kode program visual blok menjadi kode program tekstual dengan ketentuan *input*, proses, dan *output* berikut.



Perhatikan gambar *sprite* dan layar Scratch di samping. Kamu diminta membuat program sederhana untuk menentukan nilai huruf siswa, dengan ketentuan:

INPUT

nama, pts (nilai tengah semester), pas (nilai akhir semester), na (nilai akhir), dan nh (nilai huruf).

PROSES

BACA nama, pts, pas
 $na \leftarrow (0,4 * pts) + (0,6 * pas)$

IF $na < 100$ AND $na > 79$

nh \leftarrow "A"

ELSE IF $na < 80$ AND $na > 69$

nh \leftarrow "B"

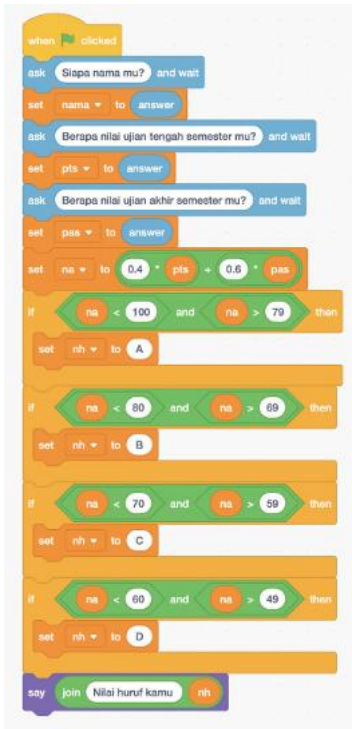
ELSE IF $na < 70$ AND $na > 59$

nh \leftarrow "C"

ELSE IF $na < 60$ AND $na > 49$

nh \leftarrow "D"

CETAK nh



OUTPUT

Nilai huruf tercetak

Berdasarkan alur dari kode program visual blok tersebut, translasikan kode program visualnya ke dalam bahasa program tekstual menggunakan bahasa C atau Python.



Refleksi

Selamat! Kamu sudah berhasil menyelesaikan bab ini. Ayo kita periksa, apa saja yang sudah kamu pelajari pada bab ini? Salinlah tabel berikut di buku tugasmu. Lalu beri tanda centang sesuai dengan pengalaman yang kamu rasakan.

Tabel 2.30 Refleksi Pemahaman Belajar Bab II Berpikir Komputasional dalam Algoritma dan Pemrograman

Pengalaman Saya	Sudah Bisa	Masih Perlu Belajar
Memahami modul program atau subprogram serta mengimplementasikannya dalam kode program visual blok		
Mampu mengembangkan dan menggunakan <i>library</i> serta mengimplementasikannya dalam kode program visual blok		
Memahami perbedaan antara kode program visual blok dan kode program tekstual, serta dapat mentranslasikan kode program visual blok dan kode program tekstual		
Memahami penelusuran <i>flowchart</i> serta dapat mentranslasikannya dalam kode program visual blok dan kode program tekstual		
Memahami <i>pseudocode</i> serta translasi dalam kode program visual blok dan kode program tekstual		



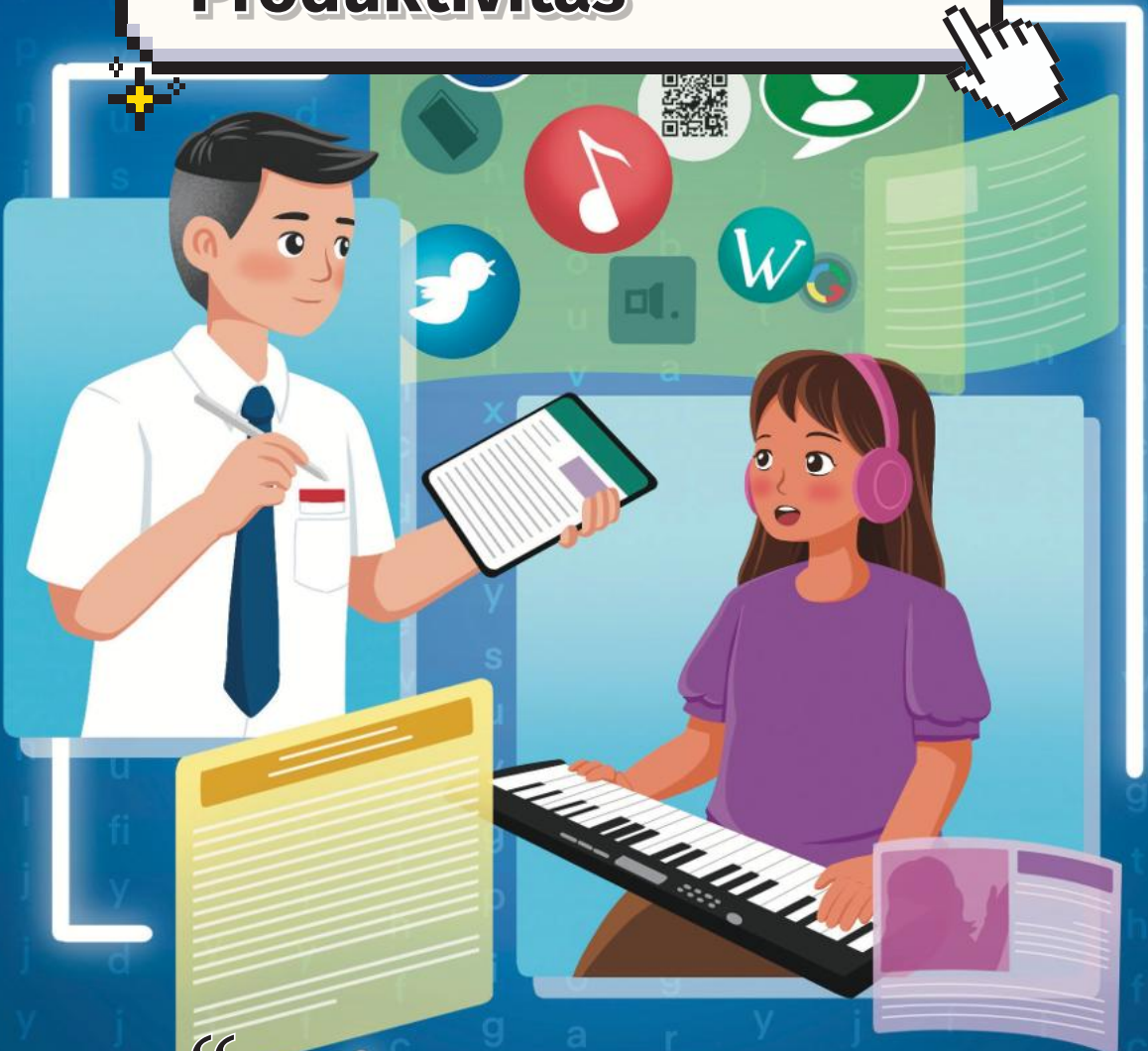
Pengayaan

Kunjungi tautan <https://bukupusbuk.id/s/if9p2> atau pindai kode QR di samping!





Literasi Digital untuk Produktivitas



“
Apakah Kamu sudah mengetahui alat produktivitas apa saja yang dapat digunakan untuk menyelesaikan tugas-tugas sekolah?
”



Tujuan Pembelajaran

Setelah mempelajari bab ini, kamu diharapkan dapat memahami pemanfaatan alat/perkakas teknologi digital untuk membuat laporan, presentasi, dan melakukan analisis serta interpretasi data. Selain itu, kamu diharapkan dapat mengintegrasikan konten antara perangkat lunak yang berisi teks, data, dan gambar agar menjadi informasi yang utuh dan mudah dipahami. Kamu juga diharapkan dapat merancang konten dengan bantuan Kecerdasan Artifisial (KA) atau *Artificial Intelligence* (AI) dalam mendiseminasikan/menyebarkan konten.

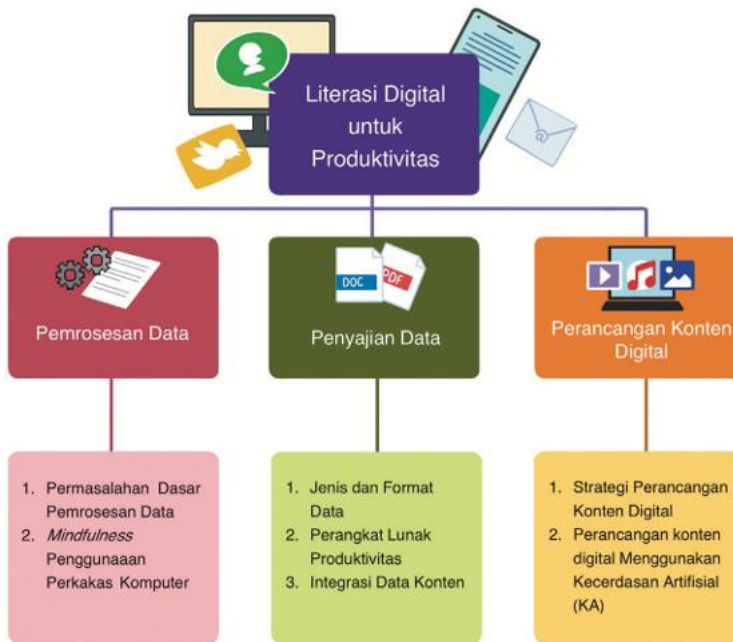


Kata Kunci

- ✓ integrasi dan pemrosesan data
- ✓ jenis data
- ✓ perancangan konten
- ✓ perangkat lunak produktivitas



Peta Materi



Gambar 3.1 Peta Materi Bab III Literasi Digital untuk Produktivitas



Siap-Siap Belajar

Saat mencari informasi di internet, kamu membutuhkan mesin pencari untuk menelusuri berbagai sumber dari situs web yang relevan. Mesin pencari terbagi menjadi dua jenis, yaitu mesin pencari tradisional dan mesin pencari berbasis *Artificial Intelligence* (AI) atau Kecerdasan Artifisial. Kamu akan mempelajari keduanya termasuk penggunaan *prompt* dalam mesin pencari berbasis AI. Selain itu, hampir semua pekerjaan untuk mendukung produktivitas melibatkan perangkat lunak digital.

Kemajuan teknologi informasi saat ini tidak dapat terlepas dari berbagai jenis dan format data berupa teks, gambar, audio, dan video. Gabungan dari data tersebut dapat diolah untuk menghasilkan sebuah informasi sesuai kebutuhan kita. Untuk mengolah berbagai jenis dan format data tersebut, kamu membutuhkan perangkat lunak perkantoran yang dapat mendukung produktivitas untuk menghasilkan informasi yang bernilai.

Jika kamu ingin mengolah teks, gunakan perangkat lunak pengolah kata (*word processor*). Untuk membuat, mengelola, dan menganalisis data dengan formulasi matematis dan statistik mulai dari yang sederhana sampai kompleks, gunakan perangkat lunak pengolah lembar kerja (*spreadsheet*). Kamu juga dapat menggunakan perangkat lunak presentasi untuk menyusun presentasi sederhana dari data yang berbentuk teks, gambar, audio, atau video. Kamu juga akan mempelajari blog dan vlog, sebagai media untuk membagikan tulisan, gambar, dan video yang dapat kamu sebar di internet. Terakhir, kamu akan belajar mengintegrasikan data, merancang konten, dan mendiseminasikannya agar menghasilkan informasi yang bermakna. Sebelum mempelajari literasi digital untuk mendukung produktivitas kerja, amati terlebih dahulu Gambar 3.2 berikut.



SMP Negeri Cerdas Bangsa

Jl. Pendidikan No. 45, Jakarta

Telp. (021) 12345678 – Email: smpcb@example.sch.id

Nomor : 017/SMP-CB/VI/2025
Lampiran : -
Hal : Undangan Kegiatan Proyek Kebersihan

Kepada
Yth. Siswa Kelas IX
SMP Negeri Cerdas Bangsa
Di Tempat

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Dengan hormat,

Dalam rangka menumbuhkan kesadaran dan kepedulian terhadap kebersihan lingkungan sekolah, kami mengundang seluruh siswa kelas IX untuk berpartisipasi dalam kegiatan:

Proyek Sekolah : Jaga Kebersihan, Jaga Masa Depan!

Kegiatan ini bertujuan membangun budaya hidup bersih, kerja sama, dan tanggung jawab siswa terhadap lingkungan sekolah.

Adapun kegiatan ini akan dilaksanakan pada:

Hari/Tanggal : Jumat, 4 Juli 2025
Waktu : Pukul 07.30 – 11.00 WIB
Tempat : Area halaman dan kelas SMP Negeri Cerdas Bangsa
Peserta : Seluruh siswa kelas IX (wajib hadir)
Kegiatan : a) Kerja bakti bersama
b) Pembuatan poster kebersihan
c) Kampanye bersih lewat media sosial
d) Kuis interaktif dan pembagian hadiah

Diharapkan seluruh siswa membawa perlengkapan kebersihan masing-masing seperti sapu, kain lap, dan kantong sampah. Demikian undangan ini kami sampaikan. Atas perhatian dan kerja samanya, kami ucapkan terima kasih. Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Hormat kami,
Kepala SMP Negeri Cerdas Bangsa

Ibu Dra. Rahmawati. M.Pd

Gambar 3.2 Contoh Ilustrasi Surat Undangan Kegiatan Proyek Kebersihan Sekolah

Sumber: Erlangga/Kemendikdasmen (2025)

Sebutkan dua contoh perkakas digital yang dapat kamu gunakan untuk meningkatkan produktivitas dalam proyek sekolah sesuai dengan surat undangan pada Gambar 3.2. Jelaskan secara singkat fungsi masing-masing perkakas digital dalam meningkatkan produktivitas!



Ayo, Mengingat Kembali

1. Kamu diminta membuat rencana kerja digital untuk membuat video edukatif bersama teman-teman. Buatlah daftar perkakas digital yang akan digunakan dan jelaskan peran masing-masing perkakas digital tersebut dalam meningkatkan produktivitas tim.
2. Di era digital, banyak peserta didik yang lebih sering menggunakan teknologi untuk hiburan, daripada untuk belajar atau bekerja. Menurut pendapatmu, apa tantangan terbesar dalam membangun literasi digital yang produktif dan bagaimana cara mengatasinya?

Ayo kita pelajari lebih lanjut tentang literasi digital untuk mendukung produktivitas kerja dalam bab ini.

A. Pemrosesan Data

Komputer adalah alat elektronik yang digunakan manusia untuk membantu pekerjaan sehari-hari. Berbagai hal dapat dilakukan oleh komputer, seperti menghitung, menulis, menggambar, bahkan menjalankan permainan (*game*). Semua itu dapat terjadi berkat suatu proses yang disebut **pemrosesan data**. Pemrosesan data adalah serangkaian langkah yang dilakukan oleh komputer untuk mengubah data mentah menjadi informasi yang berguna. Data mentah dapat berupa angka, huruf, suara, gambar, atau video yang belum memiliki makna khusus. Setelah diproses, data tersebut akan menjadi informasi yang dapat membantu manusia untuk mengambil keputusan atau menyelesaikan pekerjaan tertentu.

Dalam konteks ini, kamu harus memahami tiga proses utama yang dilakukan komputer dalam memproses data. Ketiga proses tersebut adalah *input*, proses, dan *output*. Misalnya, ketika kamu membuat jus buah: buah-buahan yang kamu siapkan di atas talenan adalah **input/data**, penghancuran buah di dalam blender adalah tahap **proses**, dan jus yang dihasilkan adalah **output**. Sama halnya seperti manusia, komputer juga bekerja melalui tahapan tertentu. Pemrosesan data pada komputer biasanya terdiri dari tiga tahapan utama, yaitu *input*, proses, dan *output*. Mari kita bahas satu per satu.



1. **Input (Masukan)**

Input adalah data yang diberikan kepada komputer. Data ini bisa berupa angka, huruf, gambar, suara, atau video. Misalnya, saat kamu mengetik menggunakan *keyboard*, kamu sedang memberikan *input* berupa teks ke komputer. Begitu juga saat kamu mengklik *mouse*, itu juga termasuk *input* dalam bentuk perintah atau navigasi. Semua perangkat yang memberikan data ke komputer disebut perangkat *input*, seperti *keyboard*, *mouse*, *scanner*, mikrofon, dan kamera.

2. **Proses (Pengolahan)**

Setelah menerima *input*, komputer akan memproses data di bagian penting yang disebut *Central Processing Unit* (CPU) atau dalam bahasa Indonesia disebut Unit Pemrosesan Pusat. CPU bisa disebut sebagai “otaknya komputer”. Di sinilah semua perhitungan, pengaturan, dan pengambilan keputusan dilakukan. CPU memiliki dua bagian penting yaitu *Arithmetic Logic Unit* (ALU) dan *Control Unit* (CU). ALU bertugas melakukan perhitungan matematika dan logika, seperti penjumlahan, pengurangan, perbandingan, dan sebagainya. Sementara itu CU mengatur alur data seperti lalu lintas data di dalam komputer, agar semuanya berjalan teratur sesuai instruksi. Namun, sebelum data diproses oleh CPU, data biasanya akan disimpan sementara di dalam *Random Access Memory* (RAM). RAM berfungsi sebagai tempat penyimpanan sementara yang membantu CPU bekerja lebih cepat. Data dan program yang sedang dipakai akan disimpan di RAM agar bisa diakses dengan cepat. Setelah proses selesai, barulah hasilnya dikeluarkan melalui tahap berikutnya. Pada tahap ini, komputer akan mengubah data mentah menjadi informasi yang lebih berguna. Contohnya, jika kamu mengetik “5 + 3” lalu tekan enter, maka CPU akan menghitungnya dan menghasilkan angka 8.

3. **Output (Keluaran)**

Output adalah hasil akhir dari pemrosesan data. *Output* ini bisa muncul di layar monitor, keluar dari *speaker* dalam bentuk suara atau tercetak di atas kertas melalui printer. Misalnya, jika kamu mengetik dan menyimpan tugasmu, maka hasil ketikan tersebut akan terlihat di layar monitor (*output*) dan bisa dicetak sebagai dokumen fisik. *Output* bisa berupa teks, gambar, grafik, suara, dan bentuk lain yang bisa kita lihat, dengar, atau gunakan. Selain ketiga tahap utama tadi, komputer juga bisa menyimpan

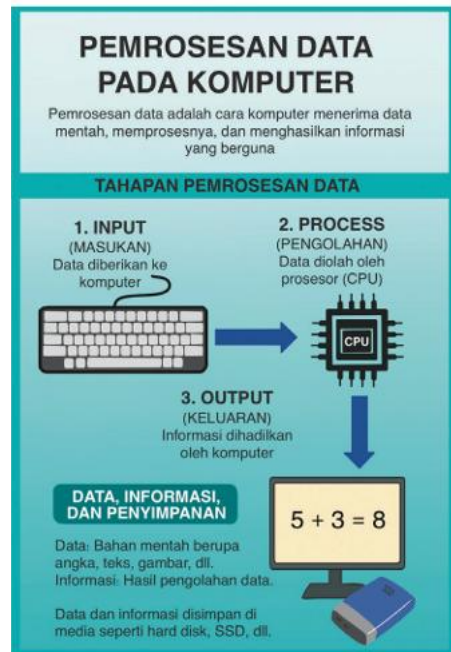
hasil pemrosesan secara permanen di perangkat penyimpanan, seperti hard disk, flashdisk, atau SSD. Penyimpanan ini penting agar data tidak hilang saat komputer dimatikan. Data yang sudah disimpan dapat dibuka kembali kapan saja saat dibutuhkan.

Ilustrasi dari rangkaian pemrosesan data pada komputer dapat dilihat pada Gambar 3.3.

Pemrosesan data pada komputer berlangsung sangat cepat, bahkan dalam hitungan sepersekian detik. Kecepatan ini memungkinkan komputer menyelesaikan ribuan hingga jutaan tugas dalam waktu singkat. Misalnya saat kamu membuka video YouTube, komputer harus memproses data gambar, suara, dan koneksi jaringan internet secara bersamaan agar video dapat diputar dengan lancar. Semua ini terjadi melalui proses *input*, pemrosesan, dan *output* yang sangat cepat dan akurat.

Dalam kehidupan sehari-hari, pemrosesan data tidak hanya terjadi pada komputer pribadi, tetapi juga pada perangkat teknologi lain, seperti *smartphone*, mesin ATM, mesin kasir, robot, bahkan mobil pintar. Semua perangkat tersebut bekerja dengan prinsip pemrosesan data yang sama. Misalnya saat kamu memindai wajahmu untuk membuka HP, sistem akan memproses data wajah dan mencocokkannya dengan data yang tersimpan. Jika cocok, HP akan terbuka. Ini merupakan contoh nyata bagaimana data diproses untuk menghasilkan *output* yang sesuai.

Pemahaman tentang pemrosesan data sangat penting di era digital, karena hampir semua kegiatan manusia sekarang berhubungan dengan komputer dan teknologi. Jika kamu memahami cara kerja komputer, kamu akan lebih mudah mempelajari teknologi lainnya, seperti pemrograman, membuat aplikasi, atau bahkan menciptakan robot. Pengetahuan ini dapat membuka peluang besar di masa depan, terutama di bidang teknologi informasi.



Gambar 3.3 Pemrosesan Data pada Komputer

1. Permasalahan Dasar Pemrosesan Data

Komputer adalah alat yang dapat membantu manusia untuk menyelesaikan berbagai pekerjaan, seperti menetik tugas, membuat presentasi, bermain *game*, hingga menjelajahi internet. Namun dalam penggunaannya, komputer juga bisa mengalami berbagai masalah. Bagi kamu yang menggunakan komputer baik di rumah maupun di sekolah, penting untuk mengenali permasalahan dasar yang sering terjadi pada komputer serta solusi praktis yang dapat dilakukan. Beberapa permasalahan dasar yang berkaitan dengan pemrosesan data pada komputer, akan dijelaskan sebagai berikut.

a. Komputer Menjadi Lambat

Salah satu masalah yang paling sering terjadi adalah komputer menjadi lambat atau lemot. Hal ini biasanya terjadi karena terlalu banyak aplikasi yang dibuka dalam waktu bersamaan, memori (RAM) penuh, atau ruang penyimpanan *hard disk* hampir habis. Misalnya, ketika mengerjakan tugas di aplikasi pengolah dokumen bersamaan membuka layanan video *streaming*, *game*, dan *browser* internet sekaligus, performa komputer seringkali menjadi lambat. Solusinya, tutuplah aplikasi yang sedang tidak digunakan. Selain itu, rutinlah membersihkan *file* yang tidak penting, seperti *file* sementara (*temporary files*) dan *file* yang ada di *recycle bin*. Gunakan juga aplikasi pembersih seperti CCleaner untuk membantu membersihkan sistem secara otomatis.

b. Blue Screen of Death (BSOD)

Masalah kedua yang sering dialami adalah layar komputer tiba-tiba menjadi biru, atau yang biasa disebut *Blue Screen of Death* (BSOD). Kondisi ini sering membuat panik karena tampilan layar mendadak berubah dan komputer tidak bisa digunakan. BSOD biasanya terjadi karena kesalahan sistem, kerusakan pada *driver* atau gangguan pada perangkat keras (*hardware*). Jika hal ini terjadi, jangan panik. Langkah pertama yang dapat dilakukan adalah *restart* komputer terlebih dahulu. Jika BSOD sering muncul, mintalah bantuan guru TIK atau teknisi komputer untuk memeriksa RAM dan *driver* perangkat, karena mungkin ada komponen yang rusak atau perlu diperbarui.

c. Tidak Ada Jaringan Internet

Masalah berikutnya adalah komputer tidak bisa terhubung ke internet. Masalah ini sangat mengganggu, apalagi saat ingin mencari bahan tugas

di internet atau mengirim tugas melalui *email*. Misalnya, saat kamu ingin mengakses situs pencarian Google untuk mencari informasi, tetapi halaman tidak bisa dibuka. Hal ini bisa disebabkan oleh koneksi Wi-Fi yang tidak terhubung, jaringan terganggu, atau pengaturan jaringan bermasalah. Solusinya, periksa apakah Wi-Fi sudah aktif dan terhubung. Jika belum berhasil, matikan lalu hidupkan kembali koneksi Wi-Fi. Jika tetap tidak bisa, *restart* modem atau tanyakan kepada orang dewasa di rumah apakah jaringan sedang mengalami gangguan atau tidak.

d. Komputer Tidak Menyala

Masalah umum lainnya adalah komputer tidak menyala atau tiba-tiba mati dengan sendirinya. Misalnya, saat menyalakan komputer, layar tetap hitam dan tidak ada reaksi. Kondisi ini bisa disebabkan oleh kabel daya yang tidak terpasang dengan baik, baterai laptop habis, atau kerusakan pada catu daya (*power supply*). Langkah pertama, periksa apakah kabel daya sudah terpasang dengan benar ke sumber listrik dan komputer. Jika menggunakan laptop, coba sambungkan pengisi daya (*charger*) dan tunggu beberapa saat. Jika komputer atau laptop tetap tidak menyala, mintalah bantuan orang yang lebih paham atau teknisi yang lebih berpengalaman.

e. Serangan Virus atau *Malware*

Masalah kelima adalah serangan virus atau *malware*. Virus dapat menyebabkan komputer menjadi lambat, data hilang, atau program tidak berjalan normal. Contohnya, ketika membuka *file* tugas, tiba-tiba *file* tersebut rusak atau tidak bisa dibuka. Untuk mencegahnya, selalu gunakan antivirus dan lakukan pemindaian secara berkala. Jangan sembarangan mengunduh *file* dari internet atau membuka *email* yang mencurigakan. Biasakan hanya mengunduh dari situs resmi dan terpercaya.

f. Perangkat *Peripheral* Tidak Berfungsi

Masalah lainnya adalah perangkat *peripheral* seperti *keyboard* atau *mouse* tidak berfungsi. Contoh kasusnya ketika ingin mengetik, tetapi huruf tidak muncul di layar. Hal ini bisa terjadi karena kabel longgar, baterai habis (pada perangkat nirkabel) atau kerusakan perangkat. Coba lepas dan pasang kembali kabel *keyboard* atau *mouse*. Jika menggunakan perangkat nirkabel, pastikan baterainya masih mencukupi. Jika perangkat tetap tidak berfungsi,

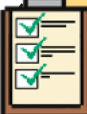


gunakan perangkat lain untuk memeriksa apakah kerusakannya berasal dari perangkat tersebut atau dari port USB di komputer. Masalah lain yang sering ditemui adalah layar monitor tidak menampilkan gambar. Saat komputer dinyalakan, suara kipas terdengar tetapi layar tetap gelap. Hal ini bisa disebabkan oleh kabel monitor tidak terhubung dengan baik, monitor dalam keadaan mati, atau masalah pada kartu grafis. Pastikan kabel monitor sudah terpasang dengan benar dan kuat. Kamu juga dapat menekan tombol daya pada monitor untuk memastikan monitor dalam keadaan hidup.


f. Permasalahan Pada Perangkat Lunak

Program atau aplikasi yang *error* juga sering terjadi. Contohnya, saat membuka aplikasi pengolah kata Microsoft Word, tiba-tiba aplikasi tertutup sendiri. Hal ini bisa disebabkan karena aplikasi yang rusak atau tidak kompatibel dengan sistem. Coba instal ulang (*reinstall*) aplikasi tersebut atau gunakan versi terbaru yang sesuai dengan perangkatmu. Pastikan untuk menyimpan pekerjaan secara berkala. Hal ini dilakukan agar dokumen yang sedang kamu kerjakan tidak hilang jika aplikasi tertutup secara tiba-tiba.

Dari berbagai contoh di atas, kamu tidak perlu panik saat menghadapi masalah komputer. Cobalah cari tahu penyebabnya dan lakukan langkah-langkah dasar untuk mengatasinya. Bila tidak berhasil, mintalah bantuan guru, teknisi, atau orang dewasa di sekitarmu. Dengan sering menggunakan komputer dan menghadapi masalah kecil seperti ini, kamu akan semakin terampil dan terbiasa merawat perangkat teknologi secara mandiri.



Aktivitas
IX-LD-A01-I

Aktivitas Individu
Permasalahan Dasar pada Komputer dan Solusi Praktisnya

- ✔ Tujuan: Memahami dasar permasalahan pada komputer dan solusi penanggulangannya.
- ✔ Syarat pengetahuan: Pemahaman dasar penggunaan komputer.

Aktivitas 1



Rina sedang mengerjakan tugas presentasi menggunakan Microsoft PowerPoint bersamaan membuka *browser* dan mendengarkan musik. Tiba-tiba komputernya menjadi sangat lambat dan macet.

Pertanyaan

1. Apa yang menyebabkan komputer Rina menjadi lambat dan macet?
2. Apa langkah yang sebaiknya dilakukan oleh Rina untuk mengatasi masalah tersebut?

Aktivitas 2



Saat Aldi mencoba membuka *file* tugasnya, *file* tersebut tidak bisa dibuka dan muncul pesan *error*. Setelah diperiksa, ternyata komputernya terinfeksi virus yang berasal dari USB temannya.

Pertanyaan

1. Apa yang harus dilakukan agar kejadian pada Aldi tidak terulang?
2. Langkah apa yang harus segera dilakukan apabila komputer telah terinfeksi virus?

2. Dampak Penggunaan Perangkat yang Baik dan Benar

Penggunaan perangkat teknologi komputer dapat memberikan dampak positif yang signifikan di berbagai bidang. Penggunaan teknologi komputer sudah hampir merambah ke semua kalangan, namun tantangan seperti kecanduan digital dan dampak negatif terhadap kesehatan mental juga semakin meningkat. *Mindfulness* menawarkan pendekatan yang bermanfaat untuk mengelola dan memanfaatkan teknologi komputer dengan cara yang lebih sehat dan produktif di berbagai bidang. Berikut adalah penerapan *mindfulness* dalam beberapa bidang.

a. Pendidikan

Pernahkah kamu menggunakan aplikasi pertemuan daring seperti Google Meet, Zoom, atau Microsoft Teams 365? Atau pernahkah kamu menggunakan Google Form untuk mengumpulkan tugas? Bahkan mungkin kamu pernah menggunakan kecerdasan artifisial seperti ChatGPT dari Open AI, Deepseek, Meta AI pada aplikasi WhatsApp, Copilot dari Microsoft, Qwen-AI dari perusahaan lokal pasar Alibaba, untuk mencari inspirasi atau keperluan tertentu? Ini merupakan salah satu contoh penggunaan teknologi pada bidang pendidikan yang diharapkan dapat mengoptimalkan efektifitas pembelajaran.



Penerapan teknologi pada lingkungan pendidikan yang semakin luas juga akan sejalan dengan gangguan digital seperti kecanduan *gadget*. Kamu mungkin sering terganggu oleh notifikasi dari *smartphone* atau aplikasi media sosial saat belajar menggunakan *gadget*, yang bisa mengurangi konsentrasi dan efektivitas saat belajar. Terkadang, kamu juga mungkin merasa kesulitan untuk menyelesaikan tugas atau membaca materi pelajaran, karena terlalu sering memeriksa notifikasi media sosial. Tidak hanya ketergantungan pada *gadget*, penggunaan AI secara berlebihan juga sangat berbahaya apabila tidak digunakan secara bijak. Penggunaan AI memang sangat memudahkan pekerjaan kita, namun kemampuan berpikir manusia juga tetap harus diasah dan dikembangkan.

Ketergantungan berlebihan pada teknologi dan kecerdasan buatan bisa berdampak buruk bagi kita. Jika terlalu sering menggunakan AI untuk mengerjakan tugas, kamu bisa menjadi malas berpikir dan kurang kreatif saat menyelesaikan masalah. Apalagi jika kamu terbiasa hanya *copy-paste* jawaban dari AI. Kamu tidak akan benar-benar memahami materi apa yang sedang dipelajari. Teknologi sebenarnya dapat membantu, tetapi jangan sampai membuatmu malas berpikir dan belajar, karena hal tersebut dapat menyebabkan hilangnya kemampuan dasar yang penting untuk masa depan.

Salah satu contoh penggunaan teknologi secara bijak untuk menjaga fokus belajar yaitu dengan membatasi *screen time* (waktu penggunaan *gadget*) serta menjaga keseimbangan antara aktivitas fisik dan interaksi sosial. Dengan demikian, kita dapat mengelola diri menjadi lebih baik. Hal ini tidak hanya tentang cara menggunakan teknologi, tetapi juga tentang cara memanfaatkan teknologi secara bijak.

b. Lingkungan

Dengan kemajuan teknologi yang pesat, kamu perlu memahami cara menggunakan teknologi secara efisien dan ramah lingkungan. Saat kamu menggunakan perangkat elektronik seperti *smartphone*, TV, laptop, dan gawai lainnya secara intensif, tanpa memperhatikan pengaturan daya atau opsi hemat energi, dapat berkontribusi pada konsumsi energi. Hal ini akan berpengaruh pada lingkungan sekitar secara tidak langsung.

Selain penggunaan daya, kebiasaan konsumtif terhadap gawai tanpa memahami dampaknya dapat meningkatkan jumlah limbah elektronik

(*e-waste*). Kamu perlu menyadari bahwa perangkat seperti ponsel, laptop, dan baterai mengandung bahan beracun seperti timbal dan merkuri. Bahan-bahan tersebut bisa mencemari tanah dan air jika dibuang sembarangan. Selain mencemari lingkungan, *e-waste* juga membahayakan kesehatan manusia, misalnya menyebabkan gangguan pernapasan. Kamu bisa mengurangi *e-waste* dengan memperbaiki gawai yang rusak atau mendaur ulang gawai tersebut. Kamu juga bisa menyumbangkan perangkat yang masih layak, alih-alih membeli baru. Secara tidak langsung, kamu dapat menghemat sumber daya alam, mengurangi polusi, dan melindungi generasi mendatang. Langkah kecil seperti ini dapat berdampak besar bagi kelestarian lingkungan.

Salah satu contoh penggunaan teknologi yang bijak di bidang lingkungan adalah mengaktifkan mode hemat energi ketika gawai sedang tidak digunakan dalam jangka waktu tertentu. Kamu juga dapat membiasakan diri untuk mematikan perangkat elektronik sebelum pulang sekolah, mematikan lampu, dan koneksi jaringan internet sebelum tidur.

c. Kesehatan

Teknologi memang membantu aktivitas kita, namun apakah kamu tahu bahwa penggunaan teknologi yang tidak terkontrol dapat berpengaruh pada kesehatan fisik dan mental? Peserta didik yang menghabiskan waktu berjam-jam melihat layar gawai atau memiliki *screen time* yang tinggi, baik untuk mengerjakan tugas sekolah, bermain *game*, menjelajahi media sosial, lokapasar, atau sekedar *browsing*, dapat mengalami gangguan fisik seperti ketegangan otot, sakit punggung, atau kelelahan mata. Bahkan hal ini dapat memicu gangguan mental seperti kesulitan tidur, *overthinking*, atau kecanduan gawai yang dapat mempengaruhi konsentrasi keesokan harinya.

Dalam bidang kesehatan, kamu dapat menjadwalkan rutinitas sebelum tidur yang mencakup teknik relaksasi, seperti meditasi malam atau membaca buku fisik. Di sekolah kamu bisa mencoba latihan pernafasan sebelum memulai pelajaran, agar lebih rileks dan siap menerima pembelajaran dengan tenang. Gunakan aplikasi pembatas *screen time* untuk mencegah penggunaan gawai secara berlebihan.



Penggunaan gawai berlebihan dapat menyebabkan kecanduan yang berdampak negatif pada kesehatan mental dan fisik. Misalnya, seorang peserta didik yang menghabiskan waktu berjam-jam untuk bermain *game online*, menjelajahi media sosial, atau menonton film baik saat jam istirahat, sepulang sekolah, maupun menjelang tidur, dapat mengabaikan waktu istirahat yang diperlukan tubuh. Kebiasaan ini berpotensi mengganggu kesehatan fisik dan mental.

Dampak positif penggunaan teknologi dapat membantu kita mengelola masalah kesehatan, seperti mengatasi gangguan postur tubuh, stres, dan kecemasan. Pembiasaan *mindfulness* juga dapat mendukung penerapan gaya hidup sehat yang berpengaruh dalam jangka panjang.

d. Ekonomi

Pernahkan kamu melihat teman yang boros? Mereka bisa menghabiskan uang sakunya untuk membeli barang-barang yang tidak dibutuhkan, hanya karena terpengaruh iklan di media sosial, lokapasar, atau promosi dari *influencer* terkenal. Disinilah *mindfulness* berperan, yaitu membantu kita dalam mengelola dampak penggunaan teknologi serta mengintegrasikannya untuk mengelola keuangan secara lebih bijaksana dan bertanggung jawab.

Tidak hanya dalam hal berbelanja, kamu mungkin mengandalkan aplikasi untuk mencatat pengeluaran uang saku tanpa memahami cara membuat anggaran sederhana. Akibatnya, pengelolaan keuangan pribadi pun belum bisa berjalan optimal. Penggunaan media sosial yang tidak dirancang khusus untuk jual-beli barang juga berisiko menimbulkan penipuan *online*, jika tidak berhati-hati saat bertransaksi digital atau belum memahami prinsip keamanan akun.

Contoh dampak positif teknologi pada bidang ekonomi adalah kemampuan mengambil keputusan belanja secara tepat. Misalnya, dengan meluangkan waktu untuk mempertimbangkan tingkat kebutuhan sebelum membeli produk di lokapasar. Dengan menerapkan prinsip ini, kamu dapat belajar mengelola pengeluaran dengan lebih bijaksana khususnya saat melakukan transaksi digital, sekaligus mengembangkan keterampilan perencanaan keuangan dan meningkatkan kesadaran akan pentingnya keamanan transaksi digital.



Aktivitas IX-LD-A02-K

Aktivitas Individu

Gangguan Digital dan Solusi Praktisnya

- ✔ Tujuan: Pada aktivitas ini, kalian akan mengeksplorasi dampak penggunaan teknologi dalam kehidupan sehari-hari.
- ✔ Syarat pengetahuan: Pemahaman dasar teknologi digital.

Berikut adalah langkah-langkah pengerjaannya.

1. Guru membagi peserta didik ke dalam beberapa kelompok.
2. Setiap kelompok diminta untuk menuliskan tantangan yang dihadapi terkait gangguan digital beserta dampak negatifnya, solusi yang bisa dilakukan, dan dampak positif yang diharapkan. Tuliskan jawabannya pada tabel 3.1 berikut.
3. Tambahkan lima jenis gangguan digital lain yang pernah kalian temui pada teman-teman kalian.

Tabel 3.1 Identifikasi Gangguan Digital dan Solusi *Mindfulness*

No	Gangguan Digital		<i>Mindfulness</i>	
	Jenis Gangguan	Dampak Negatif yang Diakibatkan	Penerapan atau Solusi	Dampak Positif yang Diharapkan
1	Screen time berlebih			
2	Penggunaan transaksi elektronik yang tidak ter-pantau orang tua			
4	Tidak mematikan listrik perangkat elektronik yang sudah selesai digunakan			
5	Penggunaan akun <i>fake</i> dalam media sosial			



B. Penyajian Data

1. Jenis dan Format Data

Dalam dunia digital saat ini, komputer sudah menjadi bagian dari rutinitas sehari-hari. Hampir semua pekerjaan, terutama di bidang perkantoran, melibatkan pembuatan dokumen. Bahkan buku yang sedang kamu baca ini merupakan hasil dari proses pembuatan dokumen dengan bantuan komputer. Sebelum menggunakan perangkat lunak perkantoran untuk mendukung produktivitas, kamu perlu mengenal jenis dan format data terlebih dahulu. Pengetahuan ini penting, agar kamu dapat memilih perangkat lunak yang tepat untuk mengolah data, sehingga dapat menghasilkan dokumen yang bermakna. Jenis dan format data yang akan dipelajari meliputi teks, gambar, audio, dan video. Format-format ini memungkinkan aplikasi perkantoran untuk menangani berbagai jenis konten, mulai dari dokumen teks sederhana hingga presentasi yang kompleks.

a. Teks

Kamu tentu sering membaca buku, baik dalam format cetak maupun digital, seperti halnya buku mata pelajaran Informatika yang sedang kamu pelajari ini. Dalam proses pembuatan dokumen buku, baik cetak maupun digital, digunakan jenis dan format data berupa teks. Teks menurut Kamus Besar Bahasa Indonesia (KBBI) adalah naskah yang berupa kata-kata asli dari pengarang. Sedangkan dalam konsep Informatika, jenis dan format data teks adalah cara tertentu untuk menyimpan data dalam bentuk teks yang dapat dibaca oleh manusia atau mesin. Format ini digunakan untuk menyimpan dan memanipulasi data dalam berbagai aplikasi, mulai dari penyimpanan sederhana hingga pertukaran data antarsistem. Format data teks dalam aplikasi perkantoran mencakup berbagai jenis *file* yang memungkinkan pengguna untuk membuat, mengedit, dan menyimpan dokumen, *spreadsheet*, presentasi, serta data lainnya. Gambar 3.4 merupakan contoh dokumen buku yang menggunakan format data teks.



Ayo, Kita Renungkan

Apakah kalian pernah melihat informasi yang disajikan dengan rapi dan sistematis? Tentu akan mempermudah pembaca dalam memahami konteks informasi yang akan disampaikan. Sebagai contoh, banyak tipe grafik yang terdapat pada aplikasi pengolah lembar kerja. Menampilkan data dalam bentuk grafik memiliki tujuan agar pembaca dapat langsung mencerna informasi yang diberikan secara cepat dan mudah. Data yang disajikan dalam bentuk tabel membutuhkan waktu lebih untuk mencerna informasi yang disajikan. Informasi dalam bentuk grafik merupakan upaya untuk membantu pembaca dapat memahami dengan cepat sebuah informasi dalam bentuk visual gambar. Menurut kalian, apakah informasi yang tersusun sistematis dapat meminimalisir kesalahan dalam menafsirkan informasi yang disampaikan ke pembaca?

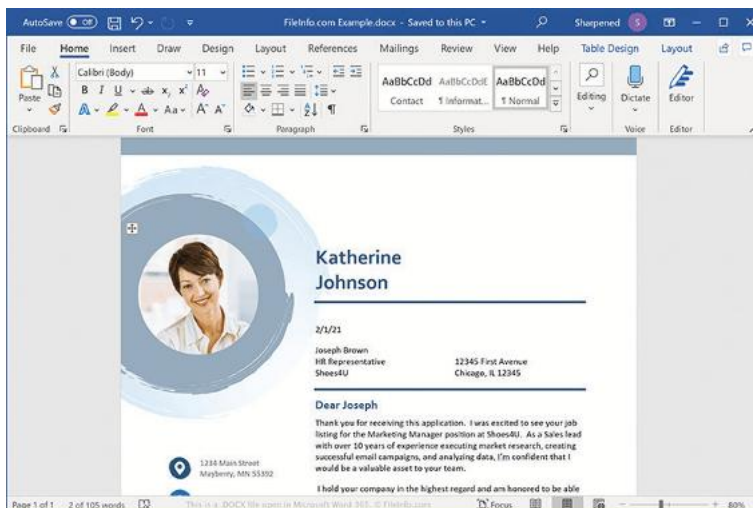
Gambar 3.4 Contoh Naskah Dokumen Buku yang Menggunakan Format Data Teks

Sumber: Hanson Prihantoro Putro, dkk. / Buku Informatika Kelas IX (2022)

Berikut adalah beberapa format teks umum yang digunakan dalam aplikasi perkantoran.

1) Docx (*Microsoft Word Document*)

- Merupakan format *file* untuk dokumen teks yang dikembangkan oleh Microsoft Word.
- Menyimpan teks dengan pemformatan kompleks, seperti jenis dan ukuran font, paragraf, tabel, dan gambar.
- Gambar berikut ini merupakan contoh *file* dalam format *.docx*.



Gambar 3.5 Contoh File dalam Format *.docx*

Sumber: FileInfo.com (2025)

2) .xlsx (Microsoft Excel Spreadsheet)

- Format *file* untuk *spreadsheet* yang dikembangkan oleh Microsoft Excel.
- Menyimpan data dalam bentuk tabel dengan kolom dan baris, serta mendukung berbagai fungsi matematika, statistik, dan pembuatan grafik.
- Gambar berikut ini merupakan contoh *file* dalam format .xlsx.



Gambar 3.6 Contoh *File* dalam Format .xlsx

Sumber: Fileinfo.com (2025)

3) .pptx (Microsoft PowerPoint Presentation)

- Format *file* untuk presentasi yang dikembangkan oleh Microsoft PowerPoint.
- Menyimpan *slide* presentasi yang berisi teks, gambar, grafik, dan animasi.
- Gambar berikut ini merupakan contoh *file* dalam format .pptx.

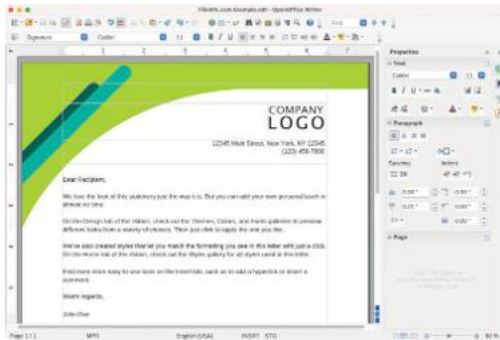


Gambar 3.7 Contoh *File* dalam Format .pptx

Sumber: Fileinfo.com (2025)

4) .odt (OpenDocument Text Document)

- a) Format *file* terbuka untuk dokumen teks dan digunakan oleh aplikasi seperti LibreOffice Writer.
- b) Menyimpan teks dengan pemformatan dan elemen dokumen sebagaimana yang terdapat dalam format .docx.
- c) Gambar berikut ini merupakan contoh *file* dalam format .odt.

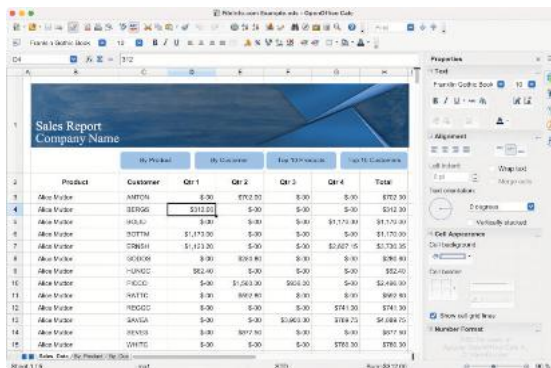


Gambar 3.8 Contoh File dalam Format .odt

Sumber: Fileinfo.com (2025)

5) .ods (OpenDocument Spreadsheet)

- a) Format *file* terbuka untuk *spreadsheet*, digunakan oleh aplikasi seperti LibreOffice Calc.
- b) Menyimpan data dalam bentuk tabel dengan dukungan berbagai fungsi matematika dan pembuatan grafik.
- c) Gambar berikut ini merupakan contoh *file* dalam format .ods.

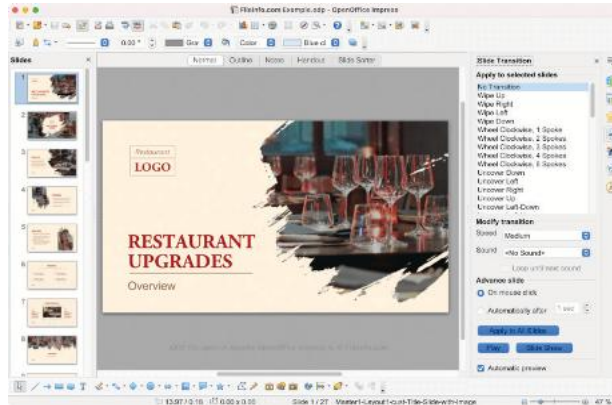


Gambar 3.9 Contoh File dalam Format .ods

Sumber: Fileinfo.com (2025)

6) .odp (OpenDocument Presentation)

- a) Format *file* terbuka untuk presentasi, digunakan oleh aplikasi seperti LibreOffice Impress.
- b) Menyimpan *slide* presentasi dengan teks, gambar, dan elemen grafis.
- c) Gambar berikut ini merupakan contoh *file* dalam format .odp.

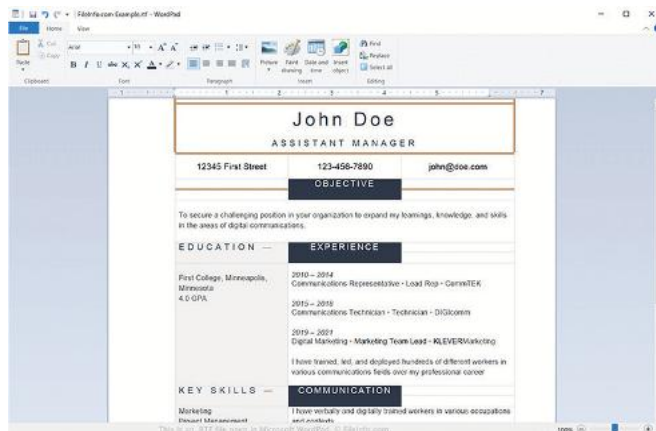


Gambar 3.10 Contoh *File* dalam Format .odp

Sumber: Fileinfo.com (2025)

7) .rtf (Rich Text Format)

- a) Format *file* teks yang mendukung pemformatan sederhana seperti font, warna, dan memberikan *style* tampilan.
- b) Dapat dibuka dan diedit oleh banyak aplikasi pengolah kata.
- c) Gambar berikut ini merupakan contoh *file* dalam format .rtf.



Gambar 3.11 Contoh *File* dalam Format .rtf

Sumber: Fileinfo.com (2025)

8) .pdf (*Portable Document Format*)

- Format *file* yang digunakan untuk mendistribusikan dokumen teks dan gambar dalam format yang konsisten di berbagai *platform*.
- Format ini memastikan dokumen tetap sama ketika dibuka pada perangkat lunak apapun.
- Gambar berikut ini merupakan contoh *file* dalam format .pdf.

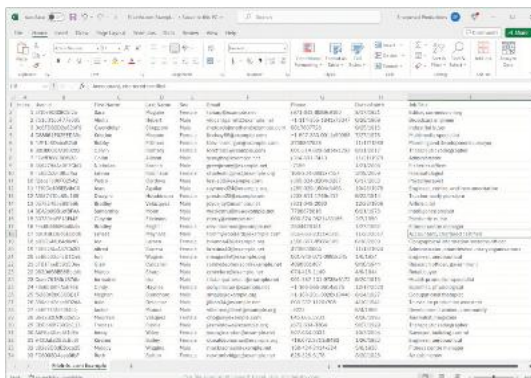


Gambar 3.12 Contoh *File* dalam Format .pdf

Sumber: Fileinfo.com (2025)

9) .csv (*Comma-Separated Values*)

- Format teks sederhana yang digunakan untuk menyimpan data tabular dan sering digunakan dalam *spreadsheet*.
- Dapat diimpor atau diekspor ke aplikasi seperti Microsoft Excel dan Google Sheets.
- Gambar berikut ini merupakan contoh *file* dalam format .csv.

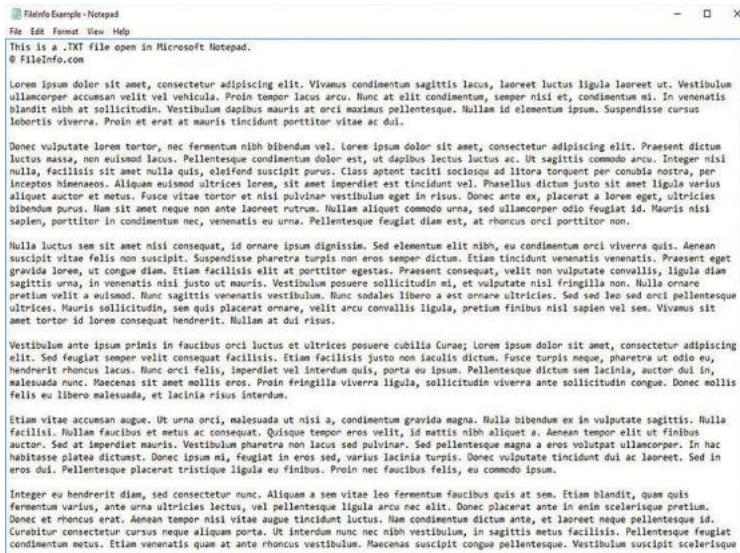


Gambar 3.13 Contoh *File* dalam Format .csv

Sumber: Fileinfo.com (2025)

10) .txt (Plain Text)

- Format *file* teks sederhana tanpa pemformatan.
- Digunakan untuk menyimpan catatan atau data yang tidak memerlukan pemformatan khusus.
- Gambar berikut ini merupakan contoh *file* dalam format .txt.

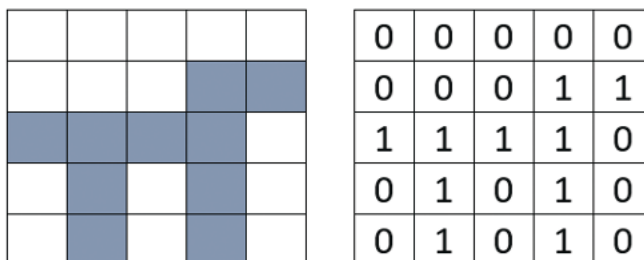


Gambar 3.14 Contoh *File* dalam Format .txt

Sumber: Fileinfo.com (2025)

b. Gambar

Saat membaca buku baik dalam bentuk cetak maupun digital, kamu tentu sering melihat gambar dalam buku tersebut. Jenis dan format data dalam bentuk gambar merupakan representasi visual yang disimpan dalam bentuk kode biner, yaitu kombinasi angka 1 dan 0. Contoh representasi gambar dalam bentuk biner dapat dilihat pada Gambar 3.15.



Gambar 3.15 Contoh Representasi Gambar dalam Bentuk Kode Biner 1 dan 0

Data gambar umumnya direpresentasikan dalam berbagai jenis dan format, tergantung pada tujuan penggunaan dan jenis kompresi yang digunakan. Berikut adalah beberapa jenis dan format data gambar yang umum digunakan:

- **Bitmap (Raster).** Data gambar yang disimpan sebagai kumpulan piksel (titik-titik warna) dalam bentuk matriks. Contoh: foto digital. Format gambar bitmap yang umum digunakan antara lain PNG, JPEG, dan BMP.
- **Vektor.** Data gambar yang disimpan sebagai serangkaian instruksi matematis untuk menggambar bentuk, garis, dan warna. Gambar vektor tidak bergantung pada resolusi, sehingga tetap tajam saat diperbesar atau diperkecil. Contoh format gambar vektor diantaranya SVG, EPS, dan AI.

Selain jenis data tersebut, berikut format data gambar yang sering digunakan dalam pengolahan dokumen.

1) **JPEG (Joint Photographic Experts Group)**

File JPG adalah gambar yang disimpan dalam format JPEG. *File* JPG biasanya berisi foto, gambar, atau grafik digital. Jika ingin membuka *file* ini di perangkatmu, cukup klik dua kali atau ketuk dua kali *file* JPG untuk membukanya di aplikasi penampil gambar *default*. Gambar berikut ini merupakan contoh *file* dalam format .jpg.



Gambar 3.16 Contoh *File* dalam Format .jpg

Sumber: Fileinfo.com (2025)

Berikut ini penjelasan lainnya terkait *file* dalam format .jpg.

- a) **Kompresi:** *lossy* (kehilangan data saat kompresi)
- b) **Penggunaan:** umum untuk foto dan gambar dengan gradasi warna yang halus
- c) **Ekstensi file:** .jpg, .jpeg

2) PNG (*Portable Network Graphics*)

File PNG adalah jenis *file* gambar yang disimpan dalam format *Portable Network Graphic* (PNG), yang umumnya digunakan untuk menyimpan grafik web, foto digital, dan gambar dengan latar belakang transparan. PNG merupakan format grafik raster yang mirip dengan .JPG tetapi menggunakan teknik kompresi *lossless* dan mendukung transparansi. Gambar berikut ini merupakan contoh *file* dalam format .png.



Gambar 3.17 Contoh *File* dalam Format .png

Sumber: Fileinfo.com (2025)

Berikut ini penjelasan lainnya terkait *file* dalam format .png.

- a) **Kompresi:** *lossless* (tanpa kehilangan data)
- b) **Penggunaan:** gambar dengan latar belakang transparan atau grafis dengan warna yang jelas
- c) **Ekstensi file:** .png

3) GIF (*Graphics Interchange Format*)

Berkas GIF adalah gambar yang disimpan dalam *Graphical Interchange Format* (GIF). Berkas ini dapat berisi hingga 256 warna yang diindeks. Palet warnanya dapat berupa serangkaian warna yang telah ditetapkan sebelumnya atau disesuaikan dengan warna dalam gambar. Berkas GIF disimpan dalam format *lossless*, yang berarti kompresi GIF tidak menurunkan kejernihan gambar. Gambar berikut ini merupakan contoh *file* dalam format .gif.



Gambar 3.18 Contoh *File* dalam Format .gif

Sumber: [Fileinfo.com](https://fileinfo.com) (2025)

Berikut ini penjelasan lainnya terkait *file* dalam format .gif.

- a) **Kompresi:** *lossless*
- b) **Penggunaan:** animasi pendek atau gambar dengan sedikit warna
- c) **Ekstensi *file*:** .gif

4) BMP (*Bitmap*)

File BMP adalah gambar yang disimpan dalam format gambar raster Bitmap (BMP) yang dikembangkan oleh Microsoft. *File* ini berisi data gambar tak terkompresi yang mendukung gambar monokrom dan berwarna pada kedalaman bit warna yang bervariasi, serta menyertakan metadata gambar. Pengguna umumnya menyimpan foto digital sebagai *file* BMP. Gambar berikut ini merupakan contoh *file* dalam format .bmp.



Gambar 3.19 Contoh *File* dalam Format .bmp

Sumber: Fileinfo.com (2025)

Berikut ini penjelasan lainnya terkait *file* dalam format .bmp.

- a) **Kompresi:** biasanya tanpa kompresi
- b) **Penggunaan:** gambar sederhana atau dalam aplikasi yang membutuhkan format asli
- c) **Ekstensi *file*:** .bmp

5) TIFF (*Tagged Image File Format*)

Berkas TIFF adalah wadah grafis yang menyimpan gambar raster dalam *Tagged Image File Format* (TIFF). Berkas ini berisi grafis berkualitas tinggi yang mendukung kedalaman warna dari 1 hingga 24 bit dan mendukung kompresi *lossy* dan *lossless*. Berkas TIFF juga mendukung beberapa lapisan dan halaman dalam satu berkas. Gambar berikut ini merupakan contoh *file* dalam format .tiff.



Gambar 3.20 Contoh *File* dalam Format .tiff

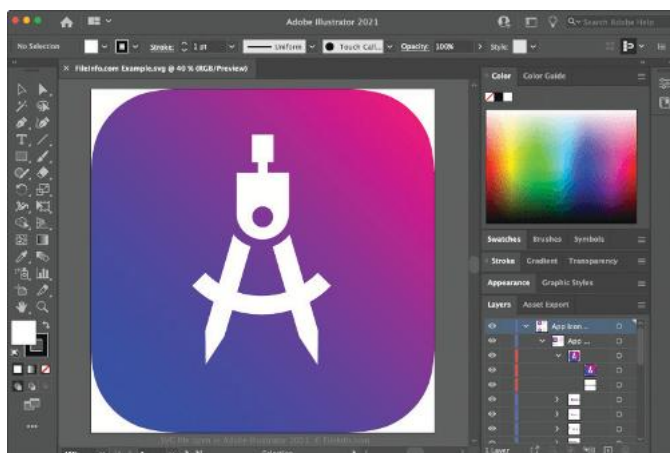
Sumber: Fileinfo.com (2025)

Berikut ini penjelasan lainnya terkait *file* dalam format .tiff.

- a) **Kompresi:** bisa *lossless* atau *lossy*
- b) **Penggunaan:** gambar berkualitas tinggi untuk keperluan cetak
- c) **Ekstensi *file* :** .tiff, .tif

6) **SVG (*Scalable Vector Graphics*)**

Berkas SVG adalah gambar yang disimpan dalam format *Scalable Vector Graphic* (SVG). Grafik vektor yang dikandungnya dapat diskalakan ke ukuran apa pun tanpa kehilangan kualitas. Karena alasan ini, ilustrator digital sering menyimpan logo, bagan, dan grafik lain dalam format .svg agar tetap tajam saat ditampilkan dalam berbagai ukuran. Gambar berikut ini merupakan contoh *file* dalam format .svg.



Gambar 3.21 Contoh *File* dalam Format .svg

Sumber: [Fileinfo.com](https://fileinfo.com) (2025)

Berikut ini penjelasan lainnya terkait *file* dalam format .svg.

- a) **Kompresi:** tidak perlu (berbasis vektor)
- b) **Penggunaan:** grafis vektor seperti logo dan ikon
- c) **Ekstensi *file*:** .svg

7) RAW

RAW adalah format data mentah yang dihasilkan oleh kamera digital, seperti kamera Panasonic, Leica, atau Casio. Berkas ini berisi data gambar mentah yang diambil oleh CCD kamera digital. Fotografer profesional sering menggunakan format .raw, karena memungkinkan penyuntingan gambar dengan tingkat akurasi yang lebih tinggi. Gambar berikut ini merupakan contoh *file* dalam format .raw.



Gambar 3.22 Contoh *File* dalam Format .raw

Sumber: Fileinfo.com (2025)

Berikut ini penjelasan lainnya terkait *file* dalam format .raw.

- a) **Kompresi:** tanpa kompresi atau sedikit kompresi
- b) **Penggunaan:** data mentah dari sensor kamera digital, untuk pengolahan lebih lanjut
- c) **Ekstensi *file*:** tergantung pada produsen kamera (.cr2, .nef, .arw)

Dalam konteks pengolahan dokumen, ada juga beberapa format gambar khusus dirancang untuk menawarkan efisiensi kompresi yang lebih baik serta fitur integrasi dengan aplikasi perkantoran. Berikut adalah format-format yang sering digunakan.

1) HEIF (*High Efficiency Image File Format*)

File HEIF adalah gambar yang disimpan dalam *High Efficiency Image Format* (HEIF). *File* ini dapat berisi gambar diam, koleksi gambar, rentetan gambar, dan rangkaian gambar. *File* HEIF juga menyimpan metadata yang menjelaskan setiap gambar. Gambar berikut ini merupakan contoh *icon file* dalam format .heif.



Gambar 3.23 Contoh *Icon File* dalam Format .heif

Sumber: [Fileinfo.com](https://fileinfo.com) (2025)

Berikut ini penjelasan lainnya terkait *file* dalam format .heif.

- a) **Kompresi:** *lossy*, lebih efisien dari JPEG
- b) **Penggunaan:** foto pada perangkat modern, seperti iPhone
- c) **Ekstensi *file*:** .heif, .heic

2) **PSD (Photoshop Document)**

File PSD merupakan format gambar yang dibuat menggunakan Adobe Photoshop, program penyuntingan gambar profesional untuk menyempurnakan foto digital dan membuat grafik web. Format ini adalah format asli yang digunakan untuk menyimpan *file* di Photoshop. *File* PSD dapat berisi lapisan gambar, lapisan penyesuaian, lapisan masker, anotasi, informasi *file*, kata kunci, dan elemen khusus Photoshop lainnya. *File* jenis ini umumnya dibuat dan dibagikan di antara para profesional grafis. Gambar berikut ini merupakan contoh *file* dalam format .psd.



Gambar 3.24 Contoh *File* dalam Format .psd

Sumber: [Fileinfo.com](https://fileinfo.com) (2025)

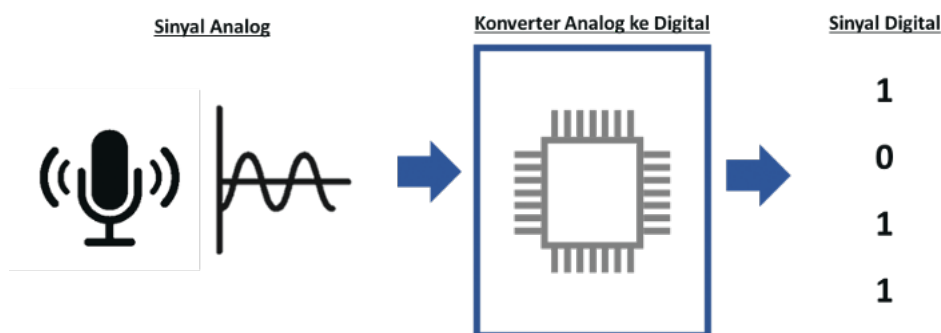
Berikut ini penjelasan lainnya terkait *file* dalam format .psd.

- a) **Kompresi:** bisa *lossless*
- b) **Penggunaan:** *file* proyek dari Adobe Photoshop yang menyimpan lapisan, teks, dan efek
- c) **Ekstensi *file*:** .psd

Setiap format gambar memiliki kelebihan dan kekurangan, tergantung pada kebutuhan spesifik seperti kualitas gambar, ukuran *file*, dukungan transparansi, atau kemampuan untuk menyimpan data tambahan seperti lapisan dan metadata.

c. Audio

Kamu tentu sering memutar lagu di laptop atau komputer menggunakan perangkat lunak pemutar musik. Pernahkah kamu berpikir bahwa suara lagu yang kamu dengarkan sebenarnya merupakan hasil olahan kode biner oleh komputer? Suara adalah fenomena fisik berupa bentuk gelombang yang merambat melalui udara. Mikروفon mengubah bentuk gelombang suara ini menjadi sinyal listrik analog. Selanjutnya, sinyal analog tersebut diambil sampelnya untuk menghasilkan pengkodean digital. Gambar 3.25 di bawah ini menunjukkan proses perubahan gelombang suara analog menjadi bentuk digital dan dikonversikan dalam kode biner agar dapat dibaca oleh komputer.



Gambar 3.25 Gelombang Suara dari Analog ke Digital Menjadi Kode Biner

Data audio adalah representasi digital dari suara yang bisa didengar. Data ini digunakan dalam berbagai aplikasi, seperti pemutar musik, video, panggilan telepon, dan asisten suara. Berikut adalah penjelasan lebih lanjut tentang data audio.

1) Cara Kerja Data Audio

Berikut ini adalah proses dari cara kerja audio.

- a) **Sampling.** Proses menangkap sinyal suara analog pada interval waktu tertentu. Setiap sampel ini kemudian diubah menjadi data digital.
- b) **Bit Depth.** Jumlah bit yang digunakan untuk merepresentasikan setiap sampel. Semakin tinggi *bit depth*, semakin akurat representasi sinyalnya (misalnya 16-bit, 24-bit).
- c) **Sample Rate.** Jumlah sampel yang diambil per detik, biasanya diukur dalam Hertz (Hz) (misalnya 44.1 kHz untuk CD audio).
- d) **Kompresi.** Proses mengurangi ukuran *file* audio, yang bisa dilakukan secara *lossy* (mengorbankan sebagian kualitas) atau *lossless* (tanpa kehilangan kualitas).

2) Format Data Audio

Format audio adalah cara data audio disimpan dalam bentuk *file* digital. Secara umum, format audio dibagi menjadi tiga kategori.

a) Tak Terkompresi

- (1) *Waveform Audio File Format* (WAV). Format audio yang umum digunakan pada Windows, sering digunakan untuk penyimpanan audio dengan kualitas tinggi.
- (2) *Audio Interchange File Format* (AIFF). Setara dengan WAV, tetapi lebih sering digunakan di sistem operasi macOS.

b) Terkompresi *Lossless*

- (1) *Free Lossless Audio Codec* (FLAC) Mengurangi ukuran *file* tanpa mengorbankan kualitas. Sering digunakan untuk distribusi musik berkualitas tinggi.
- (2) *Apple Lossless Audio Codec* (ALAC). Versi Apple dari FLAC, kompatibel dengan perangkat dan aplikasi Apple.

c) Terkompresi *Lossy*

- (1) *MPEG Audio Layer III* (MP3). Format yang paling populer, karena ukuran *file* kecil dan kompatibilitas luas.



- (2) *Advanced Audio Coding* (AAC). Format yang digunakan oleh Apple dan dianggap lebih efisien dari MP3 dengan kualitas serupa.
- (3) OGG Vorbis. Format *open source* yang sering digunakan dalam aplikasi *streaming* dan *game*.

3) Atribut Penting dari Data Audio

Untuk memahami kualitas dan karakteristik suara digital, berikut ini merupakan atribut penting dari data audio.

- a) **Sample rate.** Misalnya, 44.1 kHz berarti 44,100 sampel per detik.
- b) **Bit rate.** Jumlah data yang dihasilkan per detik, diukur dalam kilobit per detik (kbps). *Bit rate* tinggi berarti kualitas suara lebih baik, tetapi ukuran *file* lebih besar.
- c) **Channels.** Audio digital dapat memiliki jumlah saluran yang berbeda, tergantung pada cara perekaman dan pemutaran. Format audio meliputi mono (satu saluran), stereo (dua saluran) atau format lain seperti *surround sound* memiliki lebih dari dua saluran.

4) Penggunaan Data Audio

Berikut ini merupakan penggunaan data audio dalam kehidupan sehari-hari.

- a) **Musik dan Media.** Data audio digunakan untuk menyimpan dan memutar musik, film, dan *podcast*.
- b) **Komunikasi.** Data audio dikompresi untuk digunakan dalam panggilan telepon, VoIP, dan aplikasi komunikasi lainnya.
- c) **Pengolahan Audio.** Dalam produksi musik dan film, data audio diedit dan diproses menggunakan perangkat lunak audio.

Data audio memiliki peran penting dalam berbagai aspek kehidupan digital, mulai dari hiburan hingga komunikasi. Format dan kualitas data audio biasanya disesuaikan berdasarkan kebutuhan spesifik penggunaannya agar lebih optimal dan efisien.

d. Video

Pada saat kamu menonton video melalui *file* lokal laptop, komputer, atau *streaming* video di internet, pernahkah kamu pernah berpikir tentang jenis dan format data apa yang memungkinkan video tersebut bisa diputar menggunakan aplikasi tertentu? Data tersebut merupakan jenis dan format data video. Data video adalah representasi digital dari gambar bergerak yang biasanya dilengkapi dengan audio. Video terdiri dari rangkaian gambar diam (*frame*) yang ditampilkan secara cepat dan berurutan sehingga menciptakan ilusi gerakan. Berikut adalah penjelasan lebih lanjut mengenai data video.

1) Cara Kerja Data Video

Berikut ini beberapa komponen penting dalam proses kerja data video.

- a) **Frame.** Video terdiri dari rangkaian gambar diam yang disebut *frame*. Setiap *frame* adalah *snapshot* dari video pada titik waktu tertentu.
- b) **Frame Rate.** Jumlah *frame* yang ditampilkan per detik, biasanya diukur dalam *frames per second* (fps). Misalnya, 24 fps, 30 fps, atau 60 fps.
- c) **Resolusi.** Jumlah piksel dalam setiap *frame*, biasanya dinyatakan sebagai lebar×tinggi (misalnya, 1920x1080 untuk *Full HD*).
- d) **Aspect Ratio.** Perbandingan antara lebar dan tinggi *frame* video, seperti 16:9 atau 4:3.
- e) **Bit Rate.** Jumlah data yang diproses per detik dalam video, diukur dalam kilobit per detik (kbps) atau megabit per detik (Mbps). *Bit rate* yang lebih tinggi biasanya menghasilkan kualitas video yang lebih baik, tetapi dengan ukuran *file* lebih besar.

2) Format Data Video

Data video dapat disimpan dalam berbagai format *file* dan setiap format memiliki karakteristik khusus yang disesuaikan dengan kebutuhan penggunaan, kualitas, serta ukuran *file*.

- a) **Container Format.** Format kontainer berfungsi untuk menyimpan data video, audio, *subtitle*, dan metadata. Format ini mencakup berbagai elemen multimedia dalam satu *file*. Contoh: MP4, AVI, MKV, dan MOV.



- b) **Codec Video.** Codec (*compressor-decompressor*) adalah teknologi yang digunakan untuk mengodekan (*compress*) dan mendekodekan (*decompress*) data video agar dapat disimpan dan diputar dengan lebih efisien. Codec menentukan cara data video dikompresi. Contoh: H.264, H.265 (HEVC), VP9, dan AV1.

3) Format Video yang Umum

Berikut ini format video yang umum digunakan.

a) MP4 (MPEG-4 Part 14)

- Kompresi: mendukung kompresi *lossy* dan *lossless*
- Penggunaan: format yang sangat umum digunakan untuk distribusi video digital, karena mendukung banyak codec dan kompatibilitas luas
- Ekstensi *file*: .mp4

b) AVI (Audio Video Interleave)

- Kompresi: bisa tanpa kompresi atau menggunakan berbagai codec
- Penggunaan: format lama yang masih digunakan, tetapi memiliki keterbatasan dalam kompresi dan ukuran *file*
- Ekstensi *file*: .avi

c) MKV (Matroska Video)

- Kompresi: mendukung berbagai codec, baik *lossy* maupun *lossless*
- Penggunaan: sering digunakan untuk menyimpan film berkualitas tinggi, mendukung berbagai jenis data seperti video, audio, dan *subtitle* dalam satu *file*
- Ekstensi *file*: .mkv.

d) MOV (QuickTime Movie)

- Kompresi: bisa tanpa kompresi atau menggunakan codec seperti H.264
- Penggunaan: format asli untuk *QuickTime* di macOS, sering digunakan dalam produksi video.
- Ekstensi *file*: .mov.

e) **FLV (*Flash Video*)**

- Kompresi: biasanya menggunakan codec H.263 atau H.264
- Penggunaan: dulu populer untuk *streaming* video di web, tetapi sekarang jarang digunakan karena turunnya penggunaan Adobe Flash
- Ekstensi *file*: .flv

f) **WMV (*Windows Media Video*)**

- Kompresi: biasanya *lossy*, menggunakan codec Windows Media Video
- Penggunaan: format video yang dioptimalkan untuk aplikasi berbasis Windows
- Ekstensi *file*: .wmv

g) **WebM**

- Kompresi: menggunakan codec VP8 atau VP9
- Penggunaan: format *open source* yang dioptimalkan untuk web, sering digunakan oleh YouTube dan situs *streaming* lainnya
- Ekstensi *file*: .webm

4) **Atribut Penting dari Data Video**

Berikut ini atribut penting dari data video.

- a) **Frame Rate.** Jumlah *frame* per detik (fps) memengaruhi kelancaran gerakan dalam video.
- b) **Resolusi.** Ukuran *frame* video, seperti 720p, 1080p, atau 4K.
- c) **Bit Rate.** Jumlah data per detik yang digunakan oleh video, yang memengaruhi kualitas visual dan ukuran *file*.
- d) **Codec.** Algoritma yang digunakan untuk mengompresi dan mendekompresi video, yang memengaruhi kualitas dan kompatibilitas.



5) Penggunaan Data Video

Data video digunakan dalam berbagai aspek kehidupan digital, antara lain:

- a) **Streaming.** Data video dikompresi untuk efisiensi *bandwidth* dan dikirim melalui internet untuk ditonton secara *online*.
- b) **Penyimpanan.** Video disimpan dalam berbagai format *file* untuk diputar kembali menggunakan perangkat keras seperti DVD, *Blu-ray*, atau *hard drive*.
- c) **Produksi.** Dalam produksi film dan video, data video diedit dan diproses untuk menciptakan konten berkualitas tinggi.

Data video sangat penting dalam berbagai aplikasi modern, termasuk hiburan, pendidikan, komunikasi, dan media sosial. Pemilihan format video disesuaikan dengan kebutuhan spesifik seperti kualitas gambar, ukuran *file*, serta *platform* distribusinya.

2. Perangkat Lunak Produktivitas

Saat ini, orang sudah terbiasa menggunakan komputer. Hampir semua pekerjaan produktivitas berkaitan dengan pembuatan dokumen yang dilakukan dengan bantuan komputer. Pada jenjang kelas VII dan VIII, kamu telah mempelajari dasar-dasar penggunaan perangkat lunak perkantoran. Coba ingat kembali materi tersebut. Salah satu alat pendukung yang harus ada di komputer kantor adalah perangkat lunak perkantoran untuk mendukung produktivitas. Tiga jenis perangkat lunak perkantoran yang paling umum digunakan adalah pengolah kata (*word*), pengolah lembar kerja (*spreadsheet*), dan presentasi. Di era digital seperti sekarang, kamu diharapkan mampu menguasai aplikasi tersebut dengan baik. Selain itu, kamu juga akan mempelajari cara membuat blog serta mengenal video blog atau vlog. Kegiatan ini mencakup pengolahan informasi dan analisis data untuk menghasilkan dokumen atau konten yang dapat disajikan melalui vlog.

a. Perangkat Lunak Pengolah Kata (*Word*)

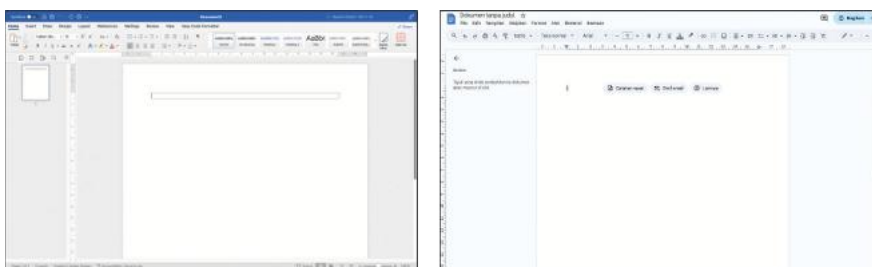
Pernahkan kamu diminta mengerjakan tugas membuat laporan yang berisi teks dan gambar? Tentu kamu harus memilih perangkat lunak yang tepat agar pekerjaan menjadi lebih mudah dan hasilnya maksimal. Salah satu

perangkat lunak yang bisa kamu gunakan adalah perangkat lunak pengolah kata (*word*). Perangkat lunak pengolah kata adalah program komputer yang digunakan untuk membuat, mengedit, dan memformat dokumen teks. Fitur umum yang dimiliki perangkat tersebut meliputi kemampuan untuk mengubah jenis dan ukuran font, menambahkan gambar, tabel, grafik, serta menyimpan dokumen dalam berbagai format *file* seperti .docx, .pdf, dan sebagainya.

Berikut adalah beberapa contoh perangkat lunak pengolah kata yang populer.

- 1) **Microsoft Word.** Salah satu perangkat lunak pengolah kata paling terkenal dan banyak digunakan, bagian dari paket Microsoft Office.
- 2) **Google Docs.** Aplikasi berbasis web yang memungkinkan pengguna untuk membuat dan mengedit dokumen secara *online* serta berkolaborasi dengan orang lain secara *real-time*.
- 3) **LibreOffice Writer.** Bagian dari suite LibreOffice, perangkat lunak *open source* yang menyediakan banyak fitur pengolah kata serupa dengan Microsoft Word.
- 4) **Apple Pages.** Aplikasi pengolah kata yang dikembangkan oleh Apple, dirancang khusus untuk pengguna macOS dan iOS.
- 5) **WPS Office Writer.** Perangkat lunak pengolah kata yang ringan dan kompatibel dengan berbagai format dokumen populer.

Gambar di bawah ini merupakan contoh tampilan antarmuka perangkat lunak pengolah kata Microsoft Word dan Google Docs.



Gambar 3.26 Tangkapan Layar Tampilan Antarmuka Perangkat Lunak Pengolah Kata Microsoft Word (kiri) dan Google Docs (kanan)

Sumber: Erlangga/Kemendikdasmen (2025)

Perangkat lunak ini digunakan secara luas di berbagai bidang, seperti pendidikan, bisnis, dan pemerintahan. Fungsinya antara lain untuk menulis laporan, surat, artikel, dan berbagai jenis dokumen lainnya.

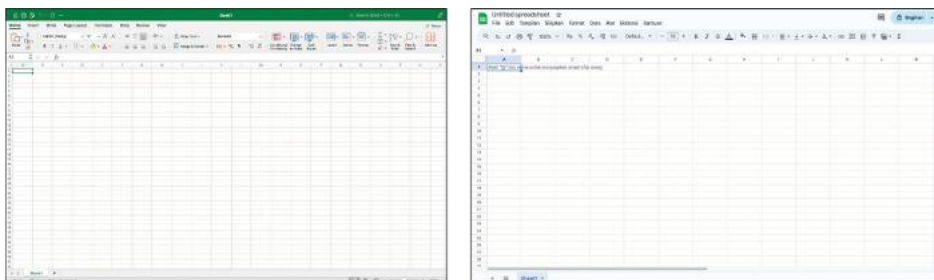
b. Perangkat Lunak Pengolah Lembar Kerja (*Spreadsheet*)

Kamu dapat membuat laporan keuangan yang sangat rinci menggunakan perangkat lunak pengolah kata (*word*). Namun, jika kamu melihat struktur data laporan keuangan, kamu akan menemukan berbagai bentuk perhitungan dan angka yang tersusun dalam bentuk tabel. Perangkat lunak pengolah kata kurang tepat digunakan untuk data yang membutuhkan perhitungan kompleks. Untuk keperluan tersebut, kamu perlu menggunakan perangkat lunak pengolah lembar kerja (*spreadsheet*). Pengolah lembar kerja adalah program komputer yang dirancang untuk mengatur, menganalisis, dan memanipulasi data dalam bentuk tabel. Perangkat lunak ini umumnya digunakan untuk membuat laporan keuangan, menganalisis data, membuat grafik, serta melakukan perhitungan matematis kompleks secara otomatis. Beberapa contoh perangkat lunak pengolah lembar kerja yang populer adalah sebagai berikut.

- 1) **Microsoft Excel.** Salah satu perangkat lunak pengolah lembar kerja paling terkenal dan banyak digunakan. Sebagai bagian dari paket Microsoft Office, Excel memiliki berbagai fitur canggih seperti fungsi, formula, pembuatan grafik, PivotTable serta kemampuan untuk mengotomatisasi tugas menggunakan makro.
- 2) **Google Sheets.** Aplikasi berbasis web yang memungkinkan pengguna untuk membuat, mengedit, dan berbagi lembar kerja secara *online*. Google Sheets juga mendukung kolaborasi *real time*, sehingga beberapa pengguna dapat bekerja secara bersamaan pada lembar kerja yang sama.
- 3) **LibreOffice Calc.** Bagian dari *suite* LibreOffice. Calc adalah perangkat lunak *open source* yang menyediakan banyak fitur serupa dengan Microsoft Excel, termasuk dukungan untuk berbagai fungsi dan formula, serta kemampuan untuk bekerja dengan berbagai format *file*.
- 4) **Apple Numbers.** Aplikasi pengolah lembar kerja yang dikembangkan oleh Apple, dirancang untuk pengguna macOS dan iOS. Apple Numbers dikenal dengan antarmuka yang bersih dan dukungan untuk grafik dan visualisasi data yang menarik.

- 5) **WPS Office Spreadsheets.** Perangkat lunak pengolah lembar kerja yang merupakan bagian dari WPS Office. Kompatibel dengan berbagai format dokumen populer dan menyediakan fitur-fitur yang serupa dengan Excel.

Gambar di bawah ini merupakan contoh tampilan antarmuka perangkat lunak pengolah lembar kerja (*spreadsheet*) Microsoft Excel dan Google Sheet.



Gambar 3.27 Tangkapan Layar Tampilan Antarmuka Perangkat Lunak Pengolah Kata (*spreadsheet*) Microsoft Excel dan Google Sheet

Sumber: Erlangga/Kemendikdasmen (2025)

Perangkat lunak pengolah lembar kerja digunakan secara luas di berbagai bidang, seperti akuntansi, manajemen proyek, dan penelitian ilmiah, untuk membantu pengguna dalam mengelola dan menganalisis data secara efisien.

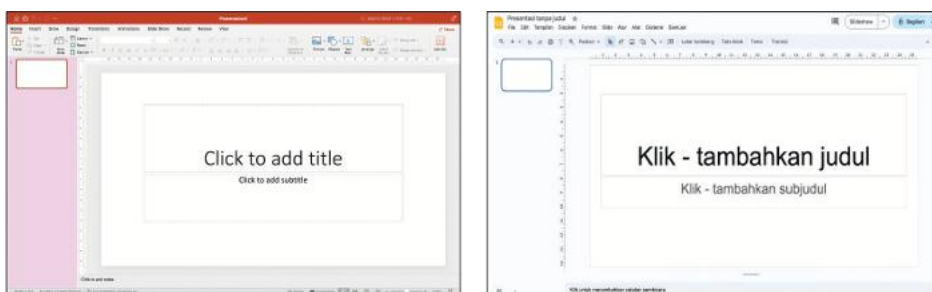
c. Perangkat Lunak Presentasi

Pernahkah kamu diminta untuk membuat laporan lalu mempresentasikannya di kelas? Biasanya, laporan tersebut ditulis menggunakan perangkat lunak pengolah kata (*word*), lalu disajikan dengan perangkat lunak presentasi. Perangkat lunak presentasi adalah jenis program komputer yang digunakan untuk membuat, mengedit, dan menampilkan *slide* presentasi yang dapat berisi teks, gambar, grafik, video, dan elemen multimedia lainnya. Perangkat lunak ini sangat berguna untuk menyampaikan informasi secara visual dan mendukung komunikasi di berbagai situasi seperti rapat, seminar, pembelajaran di kelas, dan lain-lain. Berikut adalah beberapa contoh perangkat lunak presentasi yang populer.

- 1) **Microsoft PowerPoint.** Salah satu perangkat lunak presentasi paling terkenal dan banyak digunakan, bagian dari paket Microsoft Office. PowerPoint memungkinkan pengguna untuk membuat presentasi dengan berbagai efek transisi, animasi, dan integrasi multimedia.

- 2) **Google Slides.** Aplikasi berbasis web yang memungkinkan pengguna untuk membuat, mengedit, dan berbagi presentasi secara *online*. Google Slides mendukung kolaborasi *real time*, sehingga beberapa pengguna dapat bekerja secara bersamaan pada dokumen presentasi yang sama.
- 3) **Apple Keynote.** Aplikasi presentasi yang dikembangkan oleh Apple untuk pengguna macOS dan iOS. Keynote dikenal memiliki antarmuka yang elegan serta kemampuan untuk membuat presentasi profesional dengan grafik, transisi, dan animasi yang menarik.
- 4) **LibreOffice Impress.** Bagian dari *suite* LibreOffice. Impress adalah perangkat lunak *open source* yang menyediakan fitur-fitur dasar dan lanjutan untuk membuat presentasi. Impress mendukung berbagai format *file* dan kompatibel dengan Microsoft PowerPoint.
- 5) **Prezi.** Perangkat lunak presentasi berbasis *cloud* dan menawarkan pendekatan yang berbeda dari *slide* tradisional. Prezi menggunakan kanvas besar tempat pengguna bisa membuat presentasi non-linear yang memungkinkan audiens untuk menavigasi konten dengan cara yang lebih dinamis dan interaktif.
- 6) **WPS Office Presentation.** Merupakan bagian dari WPS Office. Perangkat lunak ini menawarkan fitur serupa dengan Microsoft PowerPoint, termasuk dukungan untuk animasi, transisi, dan berbagai format *file*.

Gambar di bawah ini merupakan contoh tampilan antarmuka perangkat lunak presentasi Microsoft PowerPoint dan Google Slides.



Gambar 3.28 Tangkapan Layar Tampilan Antarmuka Perangkat Lunak Presentasi Microsoft PowerPoint (kiri) dan Google Slides (kanan)

Sumber: Erlangga/Kemendikdasmen (2025)

Perangkat lunak presentasi digunakan oleh profesional, pendidik, pelajar, dan banyak pihak lainnya untuk menyampaikan ide, laporan, proyek, atau informasi dalam format yang menarik dan mudah dipahami.

d. Blog

Kamu dapat membuat dokumen dengan berbagai jenis konten, seperti teks, grafik, tabel, bahkan gambar. Saat ingin membuat sebuah **blog**, kemampuan menyusun dokumen dengan beragam elemen ini sangat dibutuhkan. Pada awalnya, blog adalah sebuah tempat di internet yang berisi catatan harian seseorang (log) yang disusun dalam bentuk lini masa atau *timeline*. Blog dipublikasikan di internet, sehingga orang lain dapat dengan mudah membacanya tanpa harus meminjam buku asli.

Blog adalah jenis situs web atau bagian dari situs web yang berisi konten atau artikel yang diperbarui secara berkala oleh satu atau lebih penulis. Konten blog biasanya disajikan dalam urutan kronologis terbalik, dengan postingan terbaru muncul di bagian atas halaman. Blog dapat mencakup berbagai topik, mulai dari hobi pribadi, opini, berita, hingga topik khusus seperti teknologi, perjalanan, kuliner, *fashion*, dan masih banyak lagi.

Sedangkan menurut KBBI, blog adalah catatan harian atau jurnal pribadi yang tersedia secara *online* bagi siapa saja. Blog adalah alat yang dapat digunakan oleh siapa pun yang ingin berekspresi di dunia maya. Blog biasanya berisi cerita dari kehidupan penulisnya, tetapi dapat juga berisi berbagai macam konten sejak awal. Blog memberikan kesempatan kepada seseorang untuk berbagi cerita tulisan kepada siapa pun di internet. Dua penyedia layanan blog yang paling populer di Indonesia adalah Blogger dan Wordpress. Kedua layanan ini menawarkan layanan pembuatan blog secara gratis. Selain itu, kedua penyedia blog ini menawarkan layanan *hosting* blog profesional yang memungkinkan setiap pemilik blog memiliki domain pribadi mereka sendiri. Gambar di bawah ini menunjukkan layanan blog populer di Indonesia yaitu Blogger dan Wordpress.





Gambar 3.29 Tangkapan Layar Blogger (kiri) dan Wordpress (kanan)

Sumber: Erlangga/Kemendikdasmen (2025)

1) Ciri-Ciri Umum Blog

Blog memiliki beberapa ciri khas yang membedakannya dari jenis situs web lainnya. Berikut adalah ciri-ciri umum sebuah blog.

- a) **Postingan (entri).** Artikel atau entri individu yang dipublikasikan di blog. Setiap postingan biasanya memiliki judul, isi, tanggal publikasi, serta kategori atau tag yang relevan.
- b) **Komentar.** Fitur yang memungkinkan pembaca untuk memberikan tanggapan di bawah postingan blog, sehingga terjalin interaksi antara penulis dan pembaca.
- c) **Arsip.** Fitur yang memungkinkan pembaca untuk melihat postingan sebelumnya berdasarkan tanggal atau kategori.
- d) **Kategori dan Tag.** Setiap postingan blog biasanya diklasifikasikan ke dalam kategori tertentu dan diberi tag (penanda) untuk memudahkan pembaca menemukan konten dengan tema serupa.
- e) **RSS Feed.** Layanan yang memungkinkan pembaca mendapatkan notifikasi otomatis saat ada postingan baru pada blog.

2) Jenis-Jenis Blog

Blog dapat diklasifikasikan ke dalam beberapa jenis berdasarkan tujuan dan cara penyajiannya. Berikut ini beberapa jenis blog yang umum ditemukan.

- a) **Blog Pribadi.** Biasanya digunakan oleh individu untuk menulis tentang kehidupan sehari-hari, pemikiran, dan pengalaman pribadi.

- b) **Blog Profesional.** Digunakan oleh profesional untuk berbagi pengetahuan dan keahlian di bidang tertentu, seperti teknologi, bisnis, atau desain.
- c) **Blog Perusahaan.** Dikelola oleh perusahaan untuk berbagi berita, pembaruan, dan konten yang relevan dengan produk atau layanan mereka. Blog ini juga dapat digunakan untuk tujuan pemasaran konten.
- d) **Blog Niche.** Berfokus pada topik atau industri tertentu, seperti blog tentang fotografi, kesehatan, atau pendidikan.
- e) **Blog Mikro.** Platform seperti Tumblr atau Twitter yang memungkinkan pengguna untuk memposting konten pendek atau multimedia.
- f) **Blog Video (Vlog).** Alih-alih menulis, penulis vlog menggunakan video untuk menyampaikan pesan mereka. YouTube adalah salah satu platform yang paling populer untuk vlog.

3) Manfaat Blog

Blog memiliki banyak manfaat baik untuk keperluan pribadi, pendidikan, maupun profesional. Berikut ini adalah beberapa manfaat utama dari blog.

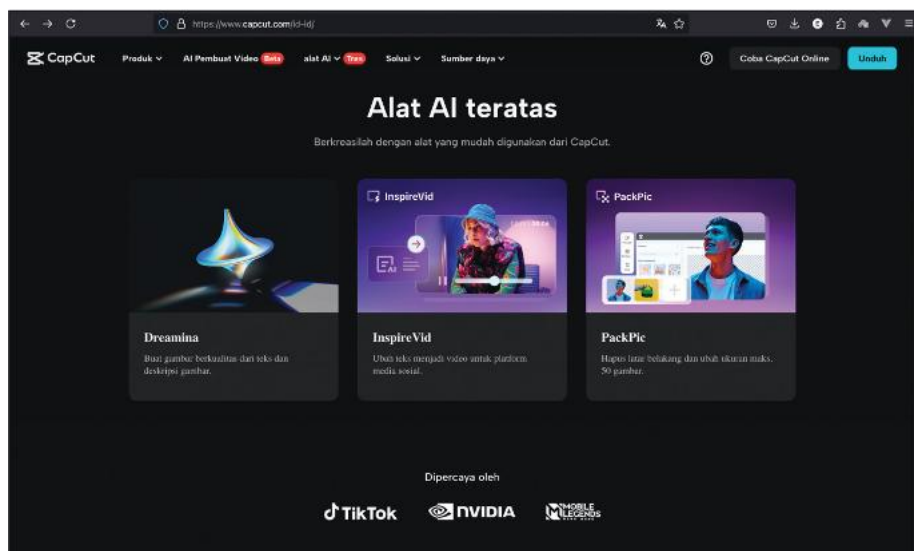
- a) **Ekspresi Diri.** Blog memberikan ruang untuk berbagi ide, opini, cerita, dan pengalaman pribadi.
- b) **Membangun Jaringan.** Melalui blog, seseorang dapat terhubung dengan pembaca, blogger lain, atau profesional yang memiliki minat dan bidang keahlian serupa.
- c) **Pemasaran dan Bisnis.** Blog digunakan oleh perusahaan untuk menarik pelanggan, meningkatkan visibilitas *online*, dan membangun merek.
- d) **Pembelajaran dan Pendidikan.** Blog dapat menjadi sumber informasi dan pendidikan, baik bagi penulis maupun pembaca.
- e) **Monetisasi.** Banyak blogger yang menghasilkan uang melalui iklan, *sponsorship*, penjualan produk, atau layanan yang ditawarkan melalui blog mereka.



Blog telah menjadi bagian penting dari budaya digital modern. Melalui blog, orang dari berbagai latar belakang dapat berbagi pengetahuan, pendapat, dan cerita mereka dengan audiens global.

e. Vlog

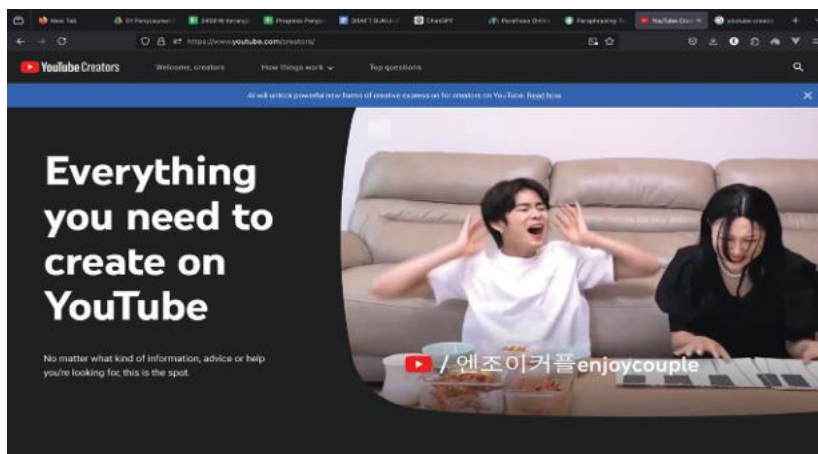
Vlog atau video blog adalah bentuk konten digital yang menggunakan video sebagai media utama untuk menyampaikan informasi, cerita, atau opini. Vlog pada dasarnya adalah blog yang disajikan dalam format video. Vlog biasanya dipublikasikan di *platform* berbagi video seperti YouTube, Vimeo, atau media sosial lainnya. Istilah vlog berasal dari kata “video blogging”, yang berarti membuat blog dalam bentuk video. Popularitas vlog meningkat seiring berkembangnya *smartphone* yang dilengkapi kamera depan, yang memudahkan siapa pun untuk merekam dirinya sendiri dan berbagi konten secara langsung. Untuk membuat vlog, kamu membutuhkan perangkat yang dapat merekam video, seperti *smartphone*. Agar hasil video yang kamu rekam dan unggah ke dalam vlog menjadi lebih maksimal dan menarik, video tersebut perlu diedit terlebih dahulu menggunakan perangkat lunak pengedit video. Salah satu perangkat lunak yang gratis (*open source*) dan mudah digunakan oleh pelajar adalah CapCut. Kamu dapat mengunduh CapCut melalui tautan berikut <https://bukupusbuk.id/s/nrtnc4>. Gambar di bawah ini menunjukkan tampilan halaman utama dari situs resmi CapCut.



Gambar 3.30 Tampilan Halaman Web CapCut

Sumber: Capcut.com (2025)

Sama halnya seperti blog, video blog juga dapat dipublikasikan kepada siapa saja. Salah satu *platform* paling populer untuk membagikan vlog adalah YouTube. YouTube awalnya adalah sebuah perusahaan kecil dengan layanan *streaming* video, tetapi akhirnya diakuisisi oleh Google. Selain berfungsi sebagai layanan untuk memublikasikan vlog, YouTube juga memiliki YouTube Creator Academy, yaitu sebuah *platform* pembelajaran *online* yang menyediakan tutorial dan panduan gratis bagi para kreator pemula (Gambar 3.31).



Gambar 3.31 Tampilan Halaman Youtube Creators

Sumber: Youtube Creators (2025)

1) Ciri-Ciri Umum Vlog

Berikut ini adalah ciri-ciri umum dari vlog yang sering dijumpai di berbagai *platform* digital.

- a) **Format video.** Vlog disajikan dalam bentuk video, dengan durasi yang bervariasi. Umumnya berdurasi antara beberapa menit sekitar 20-30 menit, tergantung pada jenis kontennya.
- b) **Konten pribadi atau profesional.** Vlog bisa berisi cerita sehari-hari, perjalanan, tutorial, ulasan produk, atau bahkan diskusi mendalam tentang topik tertentu.
- c) **Penyajian interaktif.** Vlog sering kali bersifat interaktif, ditandai dengan vlogger (orang yang membuat vlog) yang berbicara langsung ke kamera, seolah-olah berbicara langsung kepada penonton.

- d) **Platform publikasi.** Vlog biasanya dipublikasikan di *platform* video seperti YouTube, Instagram, TikTok, atau *platform* media sosial lainnya.
- e) **Komentar dan interaksi.** Sebagian besar *platform* yang mendukung vlog juga memungkinkan penonton untuk memberikan komentar, menyukai, atau berbagi video, sehingga mendorong interaksi antara vlogger dan penonton.
- f) **Regular posting.** Vlog sering kali diposting secara berkala, misalnya setiap hari, mingguan atau bulanan, tergantung pada gaya dan strategi vlogger.

2) Jenis-Jenis Vlog

Berdasarkan konten dan tujuannya, vlog dapat diklasifikasikan ke dalam beberapa jenis, yaitu:

- a) **Vlog pribadi (*daily vlog*).** Menampilkan kehidupan sehari-hari dari vlogger, sering kali mencakup kegiatan rutin, momen-momen penting, atau pengalaman sehari-hari.
- b) **Vlog perjalanan (*travel vlog*).** Berfokus pada dokumentasi perjalanan vlogger ke berbagai tempat, termasuk ulasan tentang destinasi, tips perjalanan, dan pengalaman budaya.
- c) **Vlog tutorial dan pendidikan.** Memberikan tutorial, penjelasan, atau pendidikan tentang topik tertentu, seperti tutorial makeup, tips teknologi, pelajaran memasak, atau panduan belajar.
- d) **Vlog ulasan produk (*review vlog*).** Mengulas produk atau layanan, seperti *gadget*, kosmetik, buku, film, memberikan pendapat, serta rekomendasi kepada penonton.
- e) **Vlog *fashion* dan gaya hidup.** Menampilkan tren *fashion*, tips gaya hidup, rutinitas kecantikan, dan konten lain yang berhubungan dengan gaya hidup.
- f) **Vlog hobi dan keterampilan.** Mencakup video tentang hobi tertentu, seperti memasak, berkebun, olahraga, atau bermain *game*, di mana vlogger berbagi minat mereka dengan penonton.

- g) **Vlog musik.** Berfokus pada musik, di mana vlogger mungkin menyajikan cover lagu, komposisi asli, atau diskusi tentang musik.

3) Manfaat Vlog

Vlog memiliki berbagai manfaat yang dapat dirasakan baik oleh pembuat konten maupun audiensnya, antara lain:

- a) **Membangun hubungan dengan audiens.** Vlog memungkinkan vlogger untuk menjalin hubungan yang lebih personal dengan audiens mereka, berkat interaksi langsung melalui video.
- b) **Menyampaikan informasi secara visual.** Vlog memanfaatkan kekuatan video untuk menyampaikan informasi dengan cara yang lebih menarik dan mudah dipahami.
- c) **Monetisasi.** Vlog bisa menjadi sumber pendapatan melalui iklan, *sponsorship*, penjualan *merchandise*, atau donasi dari penonton.
- d) **Ekspresi kreatif.** Vlog memberikan ruang agar dapat berekspresi secara kreatif, memungkinkan vlogger untuk berbagi cerita, bakat, dan ide dengan cara yang unik.
- e) **Pemasaran dan branding.** Banyak individu dan perusahaan menggunakan vlog sebagai bagian dari strategi pemasaran digital mereka, untuk membangun merek dan menjangkau audiens yang lebih luas.

Vlog telah menjadi salah satu format konten digital yang paling populer di era internet. Memungkinkan siapa saja yang memiliki perangkat dan ide, untuk berbagi cerita dan wawasan mereka dengan dunia.



Aktivitas IX-LD-B01-I



Aktivitas Individu

Membuat Surat Undangan Acara Gelar Karya 8 Dimensi Profil Lulusan

- ✔ Tujuan: Pembuatan surat undangan dengan perangkat lunak produktivitas.
- ✔ Syarat pengetahuan: Pemahaman dasar perangkat lunak produktivitas.

Kamu diminta untuk membuat surat undangan acara gelar karya 8 Dimensi Profil Lulusan, yang akan dipublikasikan melalui blog sekolah. Surat undangan ini ditujukan kepada orang tua atau wali peserta didik. Mintalah data susunan panitia kepada gurumu, seolah-olah kamu sedang merancang kegiatan bersama panitia dari pihak sekolah dan perwakilan orang tua. Kamu juga diminta untuk membuat surat undangan dengan susunan kepanitiaan, susunan acara kegiatan, dan konten poster promosi kegiatan. Tentukan juga perangkat lunak yang digunakan untuk membuat konten tersebut.

Proses Analisis Masalah

Lakukan analisis masalah dengan cara mengidentifikasi produk akhir yang diminta dan jenis konten yang perlu disiapkan.

Produk akhir yang diminta:

.....

Target undangan:

.....

Tujuan atau kegunaan dokumen:

.....

Jenis konten:

.....

Mencari Alternatif Solusi

Ada banyak cara untuk membuat dokumen sesuai kebutuhan produk. Kamu dapat menuliskan solusi yang relevan berdasarkan konteks permasalahan.

.....



Pemilihan Solusi dan Perangkat Lunak Pendukung

Hasil Surat Undangan

Tangkapan layar surat undangan:

Tangkapan layar poster promosi surat undangan:

Tangkapan layar surat undangan di blog:



Membuat surat undangan dapat dilakukan di komputer mana pun. Setiap anggota kelas membuat desain surat undangan dan poster promosi secara mandiri, kemudian mengirimkannya kepada salah satu anggota kelas yang bertugas sebagai *integrator* (penggabung) yang bertugas menyatukan seluruh pekerjaan. Dalam membuat surat undangan ini, apakah ada perangkat lunak yang mendukung kolaborasi langsung?

Contohnya, dalam pembuatan naskah buku, biasanya dilakukan oleh banyak penulis yang bekerja secara individu maupun kolaboratif. Setiap penulis bertanggung jawab atas bagian tertentu. Apabila penulisan naskah buku telah selesai, hasilnya digabungkan dengan penulis lainnya agar tercipta suatu naskah buku yang utuh. Menurut kamu, apakah bekerja secara kolaboratif dapat meningkatkan efisiensi pekerjaan?

3. Integrasi Data Konten

Pada pembahasan sebelumnya, kamu telah mempelajari berbagai jenis dan format data, serta mengenal perangkat lunak perkantoran. Tujuannya agar kamu dapat mengintegrasikan data dalam jenis dan format apapun dan mengolahnya dengan menggunakan perangkat lunak perkantoran, untuk menghasilkan informasi yang berguna. Pernahkah kamu memotong bungkus keripik kentang dengan gunting? Mengapa kamu lebih memilih gunting daripada pisau? Padahal, keduanya sama-sama berfungsi untuk memotong. Jawabannya terletak pada kesesuaian alat dengan benda yang dipotong dan kemudahan penggunaannya. Hal yang sama berlaku dalam memilih perangkat lunak untuk menyelesaikan suatu pekerjaan.

Sebelum membuat dokumen, kamu perlu menganalisis terlebih dahulu jenis data yang akan dimasukkan. Setelah itu kamu dapat menentukan perangkat lunak yang tepat berdasarkan jenis data tersebut. Kebutuhan pembuatan dokumen tidak sama untuk semua orang. Oleh karena itu, kita mungkin akan menggunakan perangkat lunak A dalam situasi tertentu dan memilih menggunakan perangkat lunak B dalam situasi lain. Tidak masalah perangkat lunak apa yang digunakan, asalkan sesuai dengan kebutuhan dan memberikan manfaat yang maksimal. Pada bagian ini, kamu akan memahami proses integrasi konten antarperangkat lunak yang berisi teks, data, dan gambar.

Ayo, Berpikir Kritis!

1. Bayangkan kamu adalah ketua OSIS. Pengurus OSIS diminta untuk membuat laporan tabungan bank sampah di sekolah dalam bentuk tabel dan grafik. Laporan ini bertujuan untuk memvisualisasikan data tabungan secara lebih jelas dan mudah dipahami. Apabila memungkinkan, cetaklah hasil dari laporan dan grafik tersebut.
2. Presentasikan laporan tersebut kepada guru pembina OSIS yang ada di sekolahmu. Tentukan aplikasi apa yang akan kamu gunakan dan susunlah presentasi laporan tabungan tersebut dalam bentuk tabel dan grafik.



Aktivitas
IH-LD-B02-I



Aktivitas Individu

Menyusun Laporan Kegiatan dengan Data Terintegrasi

- ✓ Tujuan: Membuat laporan kegiatan dengan data-data yang terintegrasi.
- ✓ Syarat pengetahuan: Pemahaman dasar perangkat lunak produktivitas pembuatan laporan.

Siswa SMP Merdeka sudah terbiasa memilah sampah di rumah masing-masing. Untuk mendanai kegiatan tutup tahun SMP Merdeka, para siswa bermaksud menabung sampah setiap hari Sabtu ke-4 selama 3 bulan berturut-turut. Mereka membawa sampah dari rumah masing-masing. Sampah yang bisa dibawa untuk ditabung adalah koran bekas dan botol plastik saja. Bendahara kelas mencatat data sampah yang disetorkan setiap siswa. Setiap jenis sampah dicatat beratnya. Sampah yang terkumpul akan diambil oleh pengepul dan ditukar dengan sejumlah uang dengan ketentuan sebagai berikut.

- Koran bekas Rp 3.000/kg
- Botol plastik Rp 2.000/kg

Bantulah bendahara kelas untuk mencatat dan mengolah data sampah yang disetorkan setiap anak di kelas, sehingga pada akhir bulan ke-4 menghasilkan informasi sebagai berikut.

- Rata-rata setiap jenis sampah yang dikumpulkan seluruh siswa
- Jumlah saldo uang yang ditabung setiap siswa

Buatlah laporan catatan dan pengelolaan data sampah tersebut.

Apa yang kamu perlukan?

Alat dan bahan konten

.....

Apa yang harus kamu lakukan?

1. Proses Analisis Masalah

Lakukan analisis masalah dengan melihat produk akhir yang diminta dan jenis kontennya.



Produk akhir yang diminta:

.....

Tujuan atau kegunaan dokumen:

.....

Jenis konten:

.....

2. Mencari Alternatif Solusi

Ada banyak cara untuk membuat dokumen sesuai kebutuhan produk. Kamu dapat menuliskan solusi yang relevan. Sebagai contoh solusi, sebenarnya kamu bisa memilih:

- Menggunakan fitur tabel di Microsoft Excel untuk menampilkan semua laporan hasil pengolahan sampah bulanan.
-
-

3. Mencari Alternatif Solusi

Kamu dapat memberikan penilaian untuk setiap solusi saat ini berdasarkan hasil analisis masalah dan pilihan solusi. Untuk mengubah angka menjadi grafik, kamu harus menggunakan aplikasi pengolah lembar kerja atau *spreadsheet*. Salah satu contohnya adalah memindahkan data dari buku uang kas ke tabel di perangkat lunak pengolah lembar kerja yang kamu gunakan. Buatlah tujuh kolom sederhana yang terdiri dari nomor, hari atau tanggal, pemasukan, pengeluaran, jenis pengeluaran, dan saldo. Tabel 3.2 menunjukkan contoh desain tabel yang akan dibuat.

Tabel 3.2 Contoh Desain Tabel Pencatatan Tabungan Sampah Siswa SMP Merdeka

No	Hari / Tanggal	Nama Siswa	Kelas	Jenis Sampah	Berat (Kg)	Saldo (Rp)

Kamu mungkin membutuhkan kolom tambahan sebagai informasi tambahan. Kolom apa yang mungkin kamu tambahkan untuk kasus di atas? Apa alasannya?

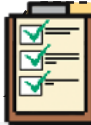
Ayo, Kita Diskusikan!

Langkah selanjutnya adalah menghasilkan data tabel tersebut sesuai dengan data yang ada di tabel catatan sampah kamu. Jangan lupa sesuaikan rumus untuk saldo pada Tabel 3.3.

Tabel 3.3 Contoh Hasil Pengisian ke Tabel Pencatatan Tabungan Sampah Siswa SMP Merdeka dalam Perangkat Lunak Lembar Kerja

No.	Hari / Tanggal	Nama Siswa	Kelas	Jenis Sampah	Berat (Kg)	Saldo (Rp)
1	7 September 2024	Anto	IX	Koran Bekas	5	15.000
2	7 September 2024	Budi	IX	Botol Plastik	3	6.000
3	7 September 2024	Cintia	IX	Koran Bekas	4	12.000
4	7 September 2024	Diva	IX	Koran Bekas	10	30.000
5	7 September 2024	Eko	IX	Botol Plastik	5	10.000
6	14 September 2024	Anto	IX	Botol Plastik	5	10.000
7	14 September 2024	Budi	IX	Koran Bekas	3	9.000
8	14 September 2024	Cintia	IX	Koran Bekas	3	3.000
9	14 September 2024	Diva	IX	Botol Plastik	10	20.000
10	14 September 2024	Eko	IX	Koran Bekas	5	15.000

Bagaimana cara kamu menghitung saldo pada baris pertama? Apakah rumus tersebut dapat digunakan pada baris kedua dan seterusnya untuk menghitung saldo?



Aktivitas IX-LD-B03-I

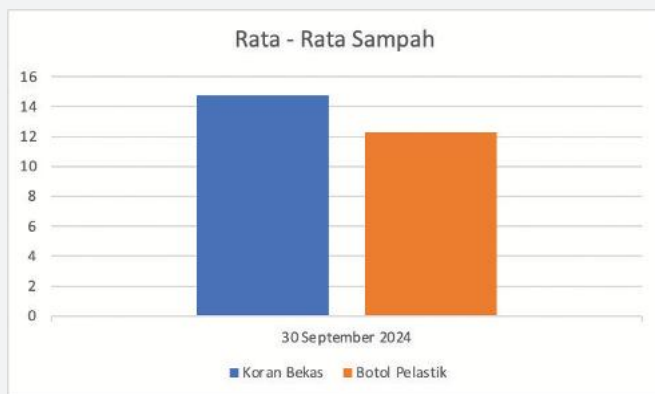


Aktivitas Individu

Menyusun Presentasi Kegiatan dengan Data Terintegrasi

- ✔ Tujuan: Penyusunan presentasi kegiatan dengan data-data terintegrasi.
- ✔ Syarat pengetahuan: Pemahaman dasar perangkat lunak produktivitas pembuatan presentasi.

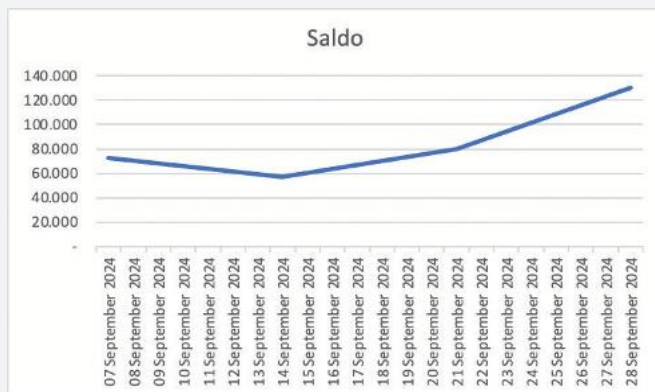
Dari aktivitas IX-LD-B02-I Menyusun Laporan Kegiatan dengan Data Terintegrasi, buatlah juga presentasi sederhana tentang visualisasi rata-rata jenis sampah terbanyak yang dikumpulkan oleh siswa dan jumlah saldo uang yang ditabung. Setelah tabel tabungan bank sampah selesai, buatlah grafik seperti Gambar 3.32.



Gambar 3.32 Tangkapan Layar Contoh Grafik Rata-Rata Setiap Jenis Sampah yang Dikumpulkan oleh Siswa

Sumber: Erlangga/Kemendikdasmen (2025)

Serta buatlah grafik yang menunjukkan jumlah saldo tabungan setiap siswa.

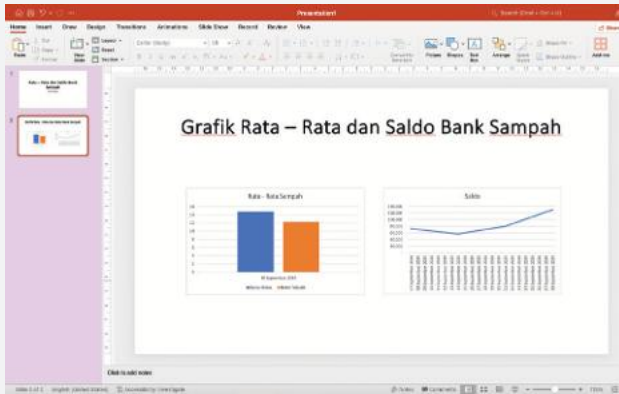


Gambar 3.33 Contoh Grafik Pemasukan Jumlah Saldo Uang yang Ditabung di Bank Sampah oleh Siswa

Sumber: Erlangga/Kemendikdasmen (2025)



Tuliskan langkah yang perlu kamu lakukan untuk memindahkan grafik tersebut ke dalam sebuah *file* presentasi seperti Gambar 3.34.



Gambar 3.34 Presentasi Grafik di Aplikasi Presentasi

Sumber: Erlangga/Kemendikdasmen (2025)

Agar informasi grafik dapat dibaca dengan mudah, kamu dapat menambahkan judul yang sesuai. Selamat! Kamu sudah berhasil membuat presentasi tentang informasi tabungan bank sampah sekolah menggunakan dua aplikasi yaitu pengolah presentasi dan lembar kerja.

Ayo, Kita Renungkan!

Apakah kamu pernah melihat informasi yang disajikan secara terorganisir? Informasi yang disusun dengan baik akan membantu pembaca memahami konteks informasi yang disampaikan. Sebagai contoh, ada banyak jenis grafik dalam aplikasi pengolah lembar kerja. Menampilkan data dalam bentuk grafik memungkinkan pembaca memahaminya dengan cepat dan mudah, sedangkan menampilkannya dalam bentuk tabel membutuhkan waktu lebih lama untuk dipahami. Coba renungkan, apakah informasi akan lebih mudah dipahami jika disusun dengan baik? Carilah tipe grafik yang dapat dibuat menggunakan aplikasi pengolah lembar kerja seperti Microsoft Excel untuk menampilkan tabungan bank sampah sekolah. Menurut kamu, grafik apa yang sesuai untuk data tersebut dan apakah itu bermanfaat? Tuliskan jawaban kamu pada lembar kerja di bawah ini.

Jenis Grafik	Data Penyusun	Hasil Grafik	Fungsi/Kegunaan

C. Perancangan Konten

Konten digital adalah semua informasi atau pesan yang kita temukan di internet dalam bentuk teks, gambar, video, suara, atau gabungan dari semuanya. Contoh konten digital antara lain video YouTube, postingan Instagram, artikel blog, meme lucu, *podcast*, hingga animasi yang sering digunakan dalam kegiatan belajar atau hiburan. Saat ini, hampir semua orang bisa membuat dan menyebarkan konten digital hanya dengan HP atau laptop dan koneksi internet. Kita bisa berbagi cerita, ilmu, pengalaman, atau hiburan kepada banyak orang melalui media sosial, *website*, atau aplikasi pesan.

Konten digital punya banyak manfaat dalam berbagai aspek kehidupan. Dalam dunia pendidikan, konten digital dapat membantu proses belajar melalui video pembelajaran, kuis *online*, atau infografis. Dalam kehidupan sehari-hari, konten digital berfungsi sebagai sumber informasi penting, seperti berita, tips kesehatan, tutorial memasak hingga cara menjaga lingkungan. Di sisi lain, konten digital juga berperan sebagai sarana hiburan, seperti *game online*, musik atau film. Bahkan, banyak orang kini bisa menghasilkan uang melalui pembuatan konten digital, misalnya menjadi YouTuber, *content creator* di TikTok atau *influencer* di media sosial.

Meskipun demikian, tidak semua konten digital bersifat positif. Ada juga konten yang menyesatkan, seperti hoaks (berita palsu), ujaran kebencian, kekerasan, atau konten yang mengandung hal-hal negatif. Kita harus berhati-hati dalam menerima dan menyebarkan konten semacam ini. Oleh karena itu, sangat penting untuk memahami literasi digital.

Literasi digital adalah kemampuan untuk memilih, memahami, dan menggunakan konten digital secara bijaksana. Selain itu, kita perlu mengembangkan kemampuan membuat konten digital yang positif, seperti konten yang menginspirasi, mengedukasi, dan membuat orang lain merasa senang atau semangat.

Saat ini, teknologi seperti kecerdasan artifisial (KA) dapat membantu kita dalam membuat konten. Contohnya, kita dapat menggunakan KA untuk menulis teks, membuat gambar digital, atau mengedit video lebih cepat. Namun, kita harus tetap bertanggung jawab dan tidak menyalahgunakan teknologi ini. Kita tidak boleh menggunakan KA untuk membuat konten yang bohong atau merugikan orang lain.

Di masa depan, konten digital akan menjadi semakin canggih. Kita mungkin akan belajar menggunakan *virtual reality* (VR), melihat dunia digital yang terasa nyata, atau membuat animasi interaktif sendiri. Tetapi apapun bentuknya, konten digital tetap harus digunakan untuk hal yang baik, bermanfaat, dan membangun. Sebagai pelajar kelas IX, kamu bisa mulai berlatih membuat konten digital yang sederhana tetapi bermakna. Kamu bisa membuat video edukasi tentang pelajaran, menulis cerita pendek di blog, membuat poster kampanye kebersihan sekolah, atau membuat *podcast* tentang topik yang kamu sukai. Dengan begitu, kamu telah berpartisipasi untuk menciptakan dunia digital yang lebih positif dan menyenangkan untuk semua orang.

1. Strategi Perancangan Konten Digital

Di era digital saat ini, hampir semua orang menggunakan internet dan media sosial. Kita sering melihat video di TikTok, YouTube, Instagram, membaca artikel, melihat meme lucu, atau belajar dari konten edukatif yang dibagikan secara *online*. Semua hal tersebut merupakan konten digital, yaitu segala informasi yang dibuat dan disebarikan secara *online* dalam bentuk teks, gambar, suara, video, atau gabungan dari semuanya. Tetapi pernahkah kamu berpikir, bagaimana cara membuat konten digital yang menarik, kreatif, dan bermanfaat? Jawabannya ada pada strategi perancangan dan pengembangan ide konten.

Membuat konten digital bukan sekadar asal membuat video atau menulis status. Dibutuhkan perencanaan yang baik, ide yang jelas, serta kreativitas agar konten tidak membosankan dan bisa memberi dampak positif. Ada beberapa langkah penting yang perlu diperhatikan dalam merancang konten digital yang efektif dan bermakna. Di antaranya adalah:

a. Menentukan Tujuan

Langkah pertama dalam membuat konten digital adalah menentukan tujuan. Apa yang ingin kamu capai melalui konten tersebut? Misalnya, kamu ingin mengedukasi teman-teman tentang bahaya sampah plastik atau menghibur orang lain dengan video lucu tetapi tetap sopan. Dengan tujuan yang jelas, kamu bisa lebih fokus dalam membuat konten.

b. Menentukan Audiens

Langkah berikutnya adalah menentukan siapa yang akan menonton atau membaca kontenmu. Dalam dunia digital, ini disebut sebagai “target audiens”.



Jika kamu membuat konten untuk teman-teman seumuran, maka gaya bahasa dan visual yang kamu gunakan harus sesuai dengan selera anak muda. Namun, jika kontenmu ditujukan untuk orang dewasa atau guru, maka pesan yang disampaikan perlu menggunakan bahasa yang lebih formal atau sopan.

c. Pengembangan Ide

Setelah kamu tahu tujuannya dan siapa penontonnya, sekarang waktunya masuk ke bagian pengembangan ide. Ide merupakan inti dari sebuah konten. Ide yang baik harus unik, menarik, dan relevan dengan kehidupan sehari-hari. Misalnya, kamu dapat membuat video pendek tentang “Tips Belajar Efektif saat Ujian” atau membuat poster digital tentang “Cara Mengurangi Sampah Plastik di Sekolah”. Untuk mendapatkan ide, kamu bisa mencari inspirasi dari masalah yang ada di sekitarmu, hal-hal yang disukai, atau tren yang sedang populer. Jangan takut untuk berpikir berbeda dan mencoba hal baru. Kreativitas bisa muncul dari keberanian untuk mencoba.

d. Membuat Rencana Konten

Setelah mempunyai ide, langkah berikutnya adalah menyusun rencana konten. Rencana ini berisi apa saja yang akan kamu buat, jadwal pembuatan, serta perkakas yang dibutuhkan. Misalnya, jika kamu ingin membuat video, kamu perlu menyiapkan naskah, kamera (bisa menggunakan HP), suara narasi, dan aplikasi pengeditan video seperti CapCut atau Kinemaster. Jika kamu ingin membuat gambar atau poster, kamu bisa menggunakan Canva, PicsArt, atau perangkat lunak desain lain yang mudah digunakan. Dengan memiliki rencana yang jelas, proses pembuatan konten akan menjadi lebih terarah, berjalan lancar, dan tidak asal-asalan.

e. Produksi Konten Digital

Selanjutnya adalah bagian yang paling seru, yaitu proses produksi. Di tahap ini, kamu mulai membuat konten sesuai dengan rencana. Kamu bisa mulai merekam video, menulis naskah, membuat ilustrasi atau gambar, serta menyusun *slide*. Perlu diingat, meskipun kamu memakai aplikasi atau alat bantu, ide dan kreatifitasmu tetap menjadi bagian terpenting dalam proses ini. Jangan hanya meniru konten orang lain. Tambahkan sentuhan khas kamu sendiri. Misalnya, gaya bicara kamu yang lucu, cara menjelaskan yang

mudah dimengerti, atau gambar-gambar buatan kamu sendiri. Hal-hal unik seperti ini akan membuat kontenmu berbeda dari yang lain.

f. Pengecekan Ulang Konten Digital Buatanmu

Setelah selesai membuat konten, jangan lupa lakukan pengecekan ulang. Cek apakah kontenmu sudah sesuai tujuan, tidak mengandung hal negatif (seperti ujaran kebencian, hoaks, atau kekerasan), dan tidak menyinggung orang lain. Kamu juga bisa minta pendapat teman atau guru untuk memberikan masukan. Setelah itu, kamu bisa mulai memublikasikan kontenmu, baik di media sosial sekolah, akun pribadi, atau *platform* digital lainnya.

g. Strategi Penyebaran Konten Digital

Agar kontenmu dilihat banyak orang, kamu bisa membuat strategi penyebaran. Misalnya, kamu bisa mengunggah konten (*upload*) di waktu yang tepat, seperti sore hari atau akhir pekan, ketika banyak orang sedang aktif menggunakan media sosial. Kamu juga bisa menambahkan *caption* menarik, *hashtag* yang sesuai, atau mengajak temanmu untuk membagikan kontenmu agar jangkuan audiens semakin luas.

h. Evaluasi Konten Digital Buatanmu

Terakhir, setelah kontenmu tersebar, kamu perlu mengevaluasi hasilnya. Coba lihat, apakah banyak orang yang menonton, memberi komentar atau membagikan kontenmu? Apakah mereka mengerti pesan yang kamu sampaikan? Dari sini, kamu bisa belajar apa yang harus ditingkatkan di konten berikutnya. Evaluasi ini sangat penting agar kamu bisa terus berkembang sebagai kreator konten.

Kesimpulannya, membuat konten digital tidak hanya soal “keren” atau “viral”, tetapi juga soal bagaimana menyampaikan ide yang baik dengan cara yang menarik dan bertanggung jawab. Kamu bisa mulai dari hal sederhana seperti membuat video edukasi singkat, poster kampanye digital atau artikel pendek tentang topik yang kamu suka. Asalkan kamu punya ide, semangat dan rencana yang jelas, kamu bisa membuat konten digital yang kreatif dan bermanfaat. Siapa tahu, konten buatanmu bisa menginspirasi banyak orang dan membuat perubahan positif di sekitarmu.



2. Perancangan Konten Digital Menggunakan Kecerdasan Artifisial (KA)

Saat ini, membuat konten digital kreatif terasa lebih mudah karena kita dapat memanfaatkan teknologi Kecerdasan Artifisial (KA). KA adalah teknologi pintar yang bisa membantu kita menulis, menggambar, membuat video, bahkan membuat suara otomatis. Dengan bantuan KA, siapapun bisa menjadi kreator konten meskipun belum ahli dalam desain atau editing. Ayo kita simak langkah-langkah mudah membuat konten digital kreatif dengan bantuan KA.

- a. Langkah pertama, tentukan dulu ide kontenmu. Misalnya, kamu ingin membuat video edukasi tentang menjaga lingkungan atau poster motivasi belajar. Pikirkan juga siapa target audiensnya (teman, guru, atau masyarakat umum). Lalu tentukan pesan apa yang ingin kamu sampaikan.
- b. Langkah kedua, buat naskah atau teks kontennya. Kamu bisa menggunakan perkakas KA untuk membantumu menulis. Misalnya, kamu bisa ketik: *“Tolong buat teks pendek tentang pentingnya buang sampah pada tempatnya”*. KA akan memberikan saran berupa teks, yang bisa kamu edit dan sesuaikan dengan gaya bahasamu sendiri.
- c. Langkah ketiga, buat desain visualnya. Kamu bisa menggunakan perkakas KA seperti Canva AI (<https://bukupusbuk.id/s/eagmdy>) atau Microsoft Designer (<https://designer.microsoft.com/>). Kamu bisa ketik: *“Desain poster tentang menjaga kebersihan sekolah”*, lalu KA akan menampilkan beberapa pilihan desain. Selanjutnya kamu dapat memilih, mengedit warna, menambahkan teks, dan menyisipkan gambar sesuai kebutuhan.
- d. Langkah keempat, jika kamu ingin membuat gambar atau ilustrasi unik yang tidak tersedia di internet, kamu bisa menggunakan perkakas KA seperti DALL-E (<https://bukupusbuk.id/s/xp3mxk>). Misalnya, ketik deskripsi gambar yang kamu inginkan: *“Anak SMP memakai seragam sekolah, sedang membersihkan taman sekolah”*. Maka KA akan menghasilkan ilustrasi sesuai permintaanmu. Gambar ini bisa kamu pakai untuk melengkapi poster atau video yang kamu buat.
- e. Langkah kelima, jika kamu ingin membuat video dengan suara, gunakan perkakas seperti Synthesia (<https://bukupusbuk.id/s/oh19c1>) atau CapCut AI (<https://bukupusbuk.id/s/2lukon>). Pada Synthesia, kamu bisa memasukkan teks, lalu kakas KA akan mengubahnya menjadi video lengkap dengan suara narasi dan presenter digital. Sementara di CapCut, kamu bisa menambahkan

teks, musik, efek, dan bahkan *subtitle* otomatis. Sangat cocok untuk membuat video edukasi berdurasi singkat!

- f. Setelah konten selesai dibuat, periksa kembali kontenmu sebelum diunggah. Pastikan tidak ada kesalahan ketik, isi kontennya sopan, tidak menyinggung pihak manapun dan bermanfaat. Jika sudah siap, kamu bisa mengunggahnya ke berbagai *platform* seperti YouTube, Instagram, TikTok atau *platform* lainnya. Jangan lupa tambahkan *caption* menarik dan *hashtag* yang relevan agar kontenmu mudah ditemukan orang lain dan dilihat banyak orang.

Meskipun perkakas KA sangat membantu, kamu tetap harus menggunakan ide dan kreativitasmu sendiri. Jadikan perkakas KA sebagai alat bantu, bukan pihak yang mengerjakan semua hal. Buatlah konten yang unik, jujur, dan positif. Dengan latihan dan semangat, kamu bisa jadi kreator konten digital yang keren sejak SMP.

 **Aktivitas**
IX-LD-C01-I

 **Aktivitas Individu**

Membuat Konten Digital Kreatif Menggunakan Kecerdasan Artifisial (KA)

- ✔ Tujuan: 1. Menjelaskan langkah-langkah pembuatan konten digital menggunakan KA,
2. Mengembangkan ide konten kreatif yang bermanfaat, dan
3. Membuat konten sederhana menggunakan bantuan perkakas KA.
- ✔ Syarat pengetahuan: Pemahaman dasar pembuatan konten digital kreatif menggunakan Kecerdasan Artifisial (KA).

Bahan dan Alat

- *Handphone* / laptop dengan koneksi internet
- Aplikasi Canva / CapCut / ChatGPT / Bing Image Creator
- Kertas kerja / perangkat lunak pengolah kata

Langkah-Langkah Aktivitas

1. **Menuliskan ide konten. Tulislah topik yang ingin kamu buat.**

.....

Apa tujuan dari konten yang kamu buat?



2. **Membuat naskah atau teks dengan bantuan KA. Gunakan ChatGPT untuk menghasilkan teks pendek sesuai topikmu. Salinlah hasilnya di bawah ini!**

.....

Kreasikan ulang dengan gayamu:.....

3. **Desain gambar atau poster dengan aplikasi Canva/DALL•E. Deskripsikan isi gambarmu.**

.....

Link atau file desain:

4. **Membuat video singkat dengan CapCut/Synthesia (opsional). Judul video:**

.....

Durasi video (menit):

Isi singkat video:

5. **Refleksi diri.**

Apa yang kamu pelajari dari kegiatan ini?

Apa tantangan yang kamu hadapi?

Bagaimana perasaanmu setelah berhasil membuat konten digital?

Catatan:

- Kegiatan ini bisa dilakukan secara individu atau kelompok.
- Gunakan konten yang sopan, positif, dan tidak melanggar hak cipta.



Uji Kompetensi

1. Mencocokkan

Cocokkanlah jenis format data berikut dengan format *file* yang sesuai.

Format <i>file</i> dokumen teks	AVI
Format <i>file</i> dokumen <i>spreadsheet</i>	.MP4
Format <i>file</i> dokumen presentasi	.MP3
Format <i>file</i> gambar foto	.GIF
Format <i>file</i> gambar animasi	.PNG
Format <i>file</i> audio digunakan di <i>windows</i>	.JPG
Format <i>file</i> audio lagu populer	.PPTX
Format <i>file</i> video untuk film	.XLSX
Format <i>file</i> video untuk <i>codec</i>	.DOCX

2. Menganalisis Bentuk Presentasi yang Sesuai Dengan Sebuah Kasus

Analisislah sebuah kasus dari **aktivitas IX-LD-B02-I Menyusun Laporan Kegiatan dengan Data Terintegrasi**. Siswa SMP Merdeka sudah terbiasa memilah sampah di rumah masing-masing. Untuk mendanai kegiatan tutup tahun SMP Merdeka, para siswa bermaksud menabung sampah setiap hari Sabtu keempat selama 3 bulan berturut-turut. Para siswa membawa sampah dari rumah masing-masing. Sampah yang dibawa untuk ditabung adalah koran bekas dan botol plastik saja. Bendahara kelas mencatat data sampah yang disetorkan setiap siswa. Setiap jenis sampah dicatat beratnya. Sampah yang terkumpul akan diambil oleh pengepul dan ditukar dengan uang sejumlah tertentu dengan ketentuan sebagai berikut.

- Koran bekas Rp 3.000/kg.
- Botol plastik Rp 2.000/kg.

Dengan tambahan persyaratan, siswa akan mendapatkan predikat berdasarkan jumlah tabungan yang berhasil dikumpulkan, dengan ketentuan sebagai berikut.

- Tabungan ≥ 500.000 predikat “Gold”.
- $200.000 \leq$ tabungan < 500.000 predikat “Silver”.
- Tabungan < 200.000 predikat “Bronze”.

Bantulah bendahara kelas untuk mencatat dan mengolah data sampah yang disetorkan oleh setiap siswa di kelas, sehingga pada akhir bulan keempat dapat menghasilkan informasi sebagai berikut.

- Predikat jumlah tabungan setiap siswa,
- Jumlah tabungan yang paling banyak,
- Jumlah tabungan yang paling sedikit, dan
- Rata-rata tabungan siswa.

Selanjutnya, analisislah bentuk presentasi dan jenis grafik yang sesuai untuk menyajikan informasi pada kasus tersebut. Lakukan tugas ini secara berkelompok.

3. Tugas Proyek Mandiri: Mengamati Permasalahan Pada Komputer

Selama satu minggu ke depan, amati dan catatlah permasalahan yang terjadi pada komputer rumah atau sekolah. Tuliskan hasil pengamatanmu pada tabel berikut (Tabel 3.4).

Tabel 3.4 Laporan Harian Masalah Komputer dan Cara Penanganannya

Hari	Masalah yang Terjadi	Penyebab (Perkiraan)	Solusi yang Dilakukan
Senin			
Selasa			
...			

4. Tugas Proyek: Kreasi Konten Digital Edukatif dengan Bantuan Kecerdasan Artifisial (KA)

Judul Tugas	: Membuat konten edukasi positif menggunakan teknologi KA.
Deskripsi Tugas	: Kamu diminta untuk membuat sebuah konten digital (poster, video pendek, atau artikel singkat) yang bertujuan untuk mengedukasi teman-teman seusiamu tentang topik yang bermanfaat, dengan bantuan perkakas KA seperti ChatGPT, Canva, DALL.E, atau CapCut.
Langkah-Langkah Pengerjaan	<ol style="list-style-type: none">1. Pilih salah satu tema berikut.<ol style="list-style-type: none">a. Pentingnya menjaga lingkungan sekolahb. Tips belajar menyenangkan saat ujianc. Bahaya penggunaan <i>gadget</i> berlebihand. Cara menjadi pelajar digital yang bijake. Anti perundungan di sekolah2. Gunakan ChatGPT untuk membuat teks singkat atau naskah sesuai tema pilihanmu.3. Gunakan Canva atau DALL•E untuk membuat desain visual (poster, ilustrasi, atau gambar pendukung).4. (Opsional) Buat video pendek (≤ 1 menit) menggunakan CapCut atau Synthesia dengan narasi otomatis atau suara kamu sendiri.5. Kumpulkan konten yang sudah dibuat dalam bentuk digital (<i>file</i> gambar/video atau <i>link</i> Google Drive).



Refleksi

Selamat! Kamu sudah berhasil menyelesaikan bab ini. Apa saja yang sudah kamu pelajari pada bab ini? Salin tabel berikut di buku tugasmu. Lalu beri tanda centang (✓) sesuai dengan pengalaman yang kamu rasakan.

Tabel 3.5 Refleksi Pemahaman Belajar Bab III Literasi Digital untuk Produktivitas

Pengalaman Saya	Sudah Bisa	Masih Perlu Belajar
Memahami jenis dan format data teks, gambar, audio, dan video		
Mampu memahami dan menggunakan perangkat lunak perkantoran: perangkat lunak pengolah kata (<i>word</i>), lembar kerja (<i>spreadsheet</i>), presentasi, blog, dan video		
Mampu memahami integrasi data konten		
Mampu memahami permasalahan pemrosesan data pada komputer		
Mampu memahami dampak penggunaan perkakas komputer yang baik dan benar		
Mampu memahami perancangan konten digital dengan menggunakan KA berbasis <i>prompt</i>		



Pengayaan

Kunjungi tautan <https://bukupusbuk.id/s/if9p3> atau pindai kode QR di samping!





Keamanan Digital



e-money



“

Sekarang informasi bisa bergerak sangat cepat. Kita bisa dengan mudah berbagi atau menerima data dari mana saja. Akan tetapi, bagaimana caranya supaya data kita tetap aman saat beraktivitas di internet?

”

Sandi:



Tujuan Pembelajaran

Setelah mempelajari bab ini, kamu diharapkan mampu mengenali perbedaan antara informasi pribadi (privat) dan publik serta memahami pentingnya menjaga kerahasiaan informasi tersebut. Selain itu, kamu juga diharapkan dapat menjelaskan jenis-jenis ancaman digital, termasuk *malware*, dan memahami cara melindungi perangkat digitalmu. Lebih lanjut, kamu akan mampu menjelaskan dampak serta risiko penggunaan ruang publik digital, membuat kata sandi yang aman dan kuat, dan menjaganya, serta menunjukkan praktik kesadaran penuh (*mindfulness*) untuk memelihara kesehatan mental.



Kata Kunci

- ✓ informasi
- ✓ jaringan publik
- ✓ *malware*
- ✓ perlindungan data
- ✓ *mindfulness*

Hai, selamat datang di dunia digital! Tempat kamu dapat mengolah data penting dengan cepat, menyimpan data, mengakses berbagai hal yang kamu butuhkan, serta berkolaborasi dengan siapa saja, kapan saja, dan di mana saja melalui jaringan internet.

Ayo kita bayangkan tentang dunia digital atau biasa disebut dengan istilah dunia maya. Dunia digital ini mirip dengan miniatur kehidupan keseharian kita. Bayangkan di sana ada sebuah kota besar dengan gedung-gedung tinggi, jalanan, lorong, dan berbagai moda transportasi yang lalu lintasnya sangat padat. Hal ini pula yang terjadi di dunia maya. Pada saat proses pengiriman data dari satu komputer ke komputer lainnya, data tersebut harus melintasi berbagai protokol pengiriman data, seperti kendaraan yang menempuh rute di jalanan kota. Setiap perintah klik pada *mouse*, *keyboard*, *LCD touchscreen* atau menekan tombol *joystick* saat bermain *game* kesukaanmu, semua itu adalah data yang mondar-mandir di “jalanan” dunia digital. Hebatnya lagi jaringan ini harus bisa memastikan setiap data yang dikirimkan sampai ke tujuannya dengan aman.

Pengiriman data telah menjangkau hampir semua aspek kehidupan kita. Data ini bisa diolah dan saling terhubung satu sama lain. Misalnya, data kita

sebagai pelajar akan terhubung dengan data keluarga, rapor *online*, sistem penerimaan murid baru, dan lainnya. Namun, tidak ada jaminan bahwa data yang kita miliki benar-benar aman. Begitu juga dengan kebenaran informasi yang kita akses. Banyaknya data di dunia digital membuat keamanan digital menjadi hal yang sangat penting untuk dipelajari di era teknologi ini. Pada bab ini, kita akan mempelajari bagaimana cara menjadi masyarakat digital yang lebih mawas diri, cerdas, kritis, dan bijak dalam mengakses serta membagikan informasi di dunia digital.

Peta Materi



Gambar 4.1 Peta Materi Bab IV Keamanan Digital

Siap-Siap Belajar

Dunia digital tentu tidak asing bagi kita. Saat ini, hampir semua hal bisa kita digitalisasi, mulai dari bidang pendidikan, transportasi, jual beli barang dan jasa, pertanian, hiburan, hingga berbagai aktivitas lainnya. Semua itu dapat diakses dengan mudah melalui aplikasi maupun situs internet. Terkadang kita menemukan iklan seperti pada Gambar 4.2 berikut. Iklan ini umumnya berupa informasi penawaran yang terkadang menggiurkan. Apa yang biasanya kamu lakukan jika menemukan hal tersebut? Apakah kamu akan membuka atau menutup *pop-up* informasi ini? Sebutkan alasan mengapa kamu melakukan hal tersebut.



Gambar 4.2 Pop-Up Iklan pada Website Ilegal

Dunia digital adalah dunia yang penuh dengan petualangan sekaligus tantangan, yang perlu kita hadapi sebagai bagian dari kemajuan peradaban manusia. Di era ini, hampir semua aspek kehidupan, mulai dari merencanakan, melaksanakan, sampai mengevaluasi aktivitas sehari-hari sangat erat kaitannya dengan teknologi. Salah satu perkembangan teknologi yang paling sering kita gunakan adalah internet. Dalam penggunaannya, kita perlu memahami betapa pentingnya menjaga keamanan digital. Setiap klik dan ketikan yang kita lakukan merupakan data yang dikirimkan melalui dunia maya dan setiap data tersebut harus dikirimkan dengan aman.

*



Ayo, Mengingat Kembali!

1. Hari ini, kamu mungkin telah menerima berbagai informasi dari internet, media sosial atau percakapan digital. Apakah ada informasi privat seseorang yang kamu lihat hari ini?
2. Apakah kamu menjamin koneksimu akan aman, apabila terhubung dengan Wi-Fi publik, yang tidak membutuhkan *password* (kata sandi)?
3. Apakah kamu atau temanmu pernah menjadi korban serangan siber? Bagaimana hal tersebut bisa terjadi?
4. Apa saja cara untuk melindungi akun kita dari serangan siber?

5. Bagaimana cara menjadi pengguna teknologi yang bijak, aman, dan bertanggung jawab?

Coba renungkan pertanyaan-pertanyaan ini. Tuliskan pendapatmu secara singkat di buku catatanmu, ya.

A. Jenis Informasi

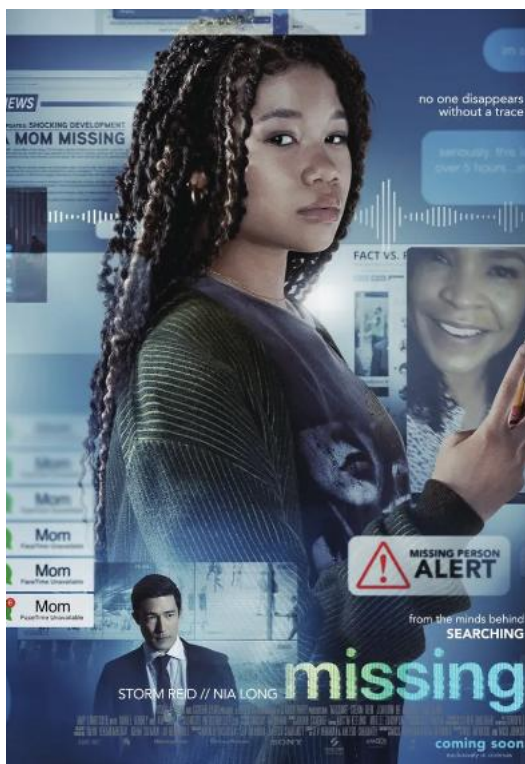
Bayangkan kamu sedang memegang dua amplop. Amplop pertama adalah amplop publik yang bisa dibuka oleh siapa saja dan isinya bisa dilihat oleh semua orang. Amplop kedua adalah amplop privat yang bersifat rahasia dan hanya boleh dibuka oleh kamu atau orang-orang yang kamu izinkan. Jadi, amplop ini diibaratkan sebagai informasi, sementara isi amplop adalah data. Untuk memahami lebih jauh tentang data, kita bisa melihat *puzzle*. Asumsikan setiap potongan *puzzle* dapat berupa angka, teks, gambar, atau fakta-fakta kecil lainnya. Namun, potongan-potongan ini belum bisa memberikan gambaran yang jelas. Nah, informasi adalah hasil akhir ketika semua potongan *puzzle* disatukan dan diurutkan, sehingga orang lain bisa memahami makna dari kumpulan data tersebut. Dunia digital memungkinkan informasi tersebar dengan cepat dan luas, menghilangkan batas ruang dan waktu.

Keakuratan informasi di dunia digital menjadi tantangan besar, karena mudahnya konten diproduksi dan disebar oleh siapapun tanpa verifikasi dari berbagai sumber, baik melalui media sosial, *game online*, lokapasar, dan lainnya. Oleh karena itu, kemampuan dalam menyaring data dan informasi menjadi tantangan tersendiri. Tidak hanya itu, kita juga harus memiliki kecakapan dalam mengamankan data digital kita.

Dalam beraktivitas di dunia digital, kita terbiasa menyimpan data melalui berbagai *platform* seperti penyimpanan *cloud*, akun *email*, media sosial, dan lainnya. Sebagian data dapat diakses oleh siapa saja, dan sebagian lainnya hanya bisa diakses oleh pihak tertentu. Nah, di sinilah pentingnya membedakan antara informasi publik dan informasi privat. Informasi yang aman untuk dibagikan kepada khalayak umum disebut **informasi publik**, sementara informasi yang bersifat rahasia dan sangat riskan apabila diketahui umum karena berpotensi disalahgunakan oleh pihak yang tidak bertanggung jawab disebut sebagai **informasi privat**.

Berbeda dengan data publik yang bisa diakses oleh siapapun, data privat sangat penting untuk dijaga. Data privat bukan untuk dikonsumsi oleh publik dan dapat mengganggu kita bahkan orang lain, baik di lingkungan keluarga, teman, atau lingkungan sekolah. Oleh karena itu, kemampuan dalam melindungi data digital menjadi sangat penting.

Pada film *Missing* (2023) digambarkan bagaimana jejak digital seperti riwayat pencarian, lokasi GPS, dan rekaman CCTV bisa disalahgunakan untuk memata-matai orang lain, sehingga menegaskan betapa rentannya data pribadi di era teknologi. Film ini juga menampilkan adegan penyadapan akun dengan menebak informasi saat proses verifikasi lupa *password*. Penyadap berhasil masuk menggunakan informasi privat pemilik akun asli yang tersebar di media sosialnya dan sistem keamanan akun hanya menggunakan metode standar tanpa autentikasi dua faktor. Film tersebut juga menunjukkan bagaimana informasi yang terlihat sepele dapat digunakan secara kreatif oleh penjahat siber untuk mencapai tujuan mereka. Meskipun film ini adalah karya fiksi, namun sangat mencerminkan situasi nyata tentang kurangnya kepedulian terhadap keamanan digital. Dalam konteks perlindungan data, Indonesia telah memiliki payung hukum yang jelas, yaitu Undang-Undang Nomor 27 Tahun 2022 tentang Perlindungan Data Pribadi. Undang-undang ini mengklasifikasikan data pribadi menjadi dua jenis, yaitu data pribadi umum dan data pribadi spesifik. Berikut adalah detailnya.



Gambar 4.3 Poster Film Missing (2023)

Sumber: SonyPictures (2023)

1 Data Pribadi Umum

- a. Nama lengkap,
- b. Jenis kelamin,
- c. Kewarganegaraan,
- d. Agama,
- e. Status perkawinan, dan
- f. Data pribadi yang dikombinasikan untuk mengidentifikasi seseorang.

2. Data Pribadi Spesifik

a. Data dan Informasi Kesehatan

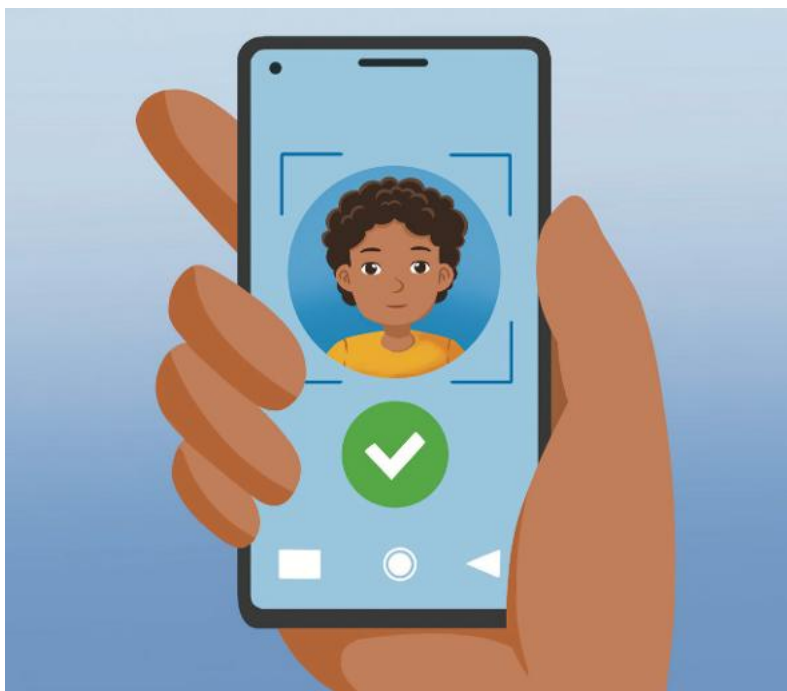


Gambar 4.4
Ilustrasi Dokter
Memberikan
Map Data
Medis pada Pasien

Rekam medis dan data kesehatan wajib dijaga kerahasiaannya karena mengandung informasi sangat pribadi seperti riwayat penyakit, obat-obatan, atau hasil laboratorium yang berpotensi disalahgunakan jika bocor. Kebocoran data semacam ini bisa menyebabkan diskriminasi, stigma sosial, bahkan penipuan. Oleh karena itu, rumah sakit dan aplikasi yang berkaitan dengan data kesehatan seseorang harus memiliki sistem keamanan yang ketat untuk melindungi data pasien.

b. Data Biometrik

Data biometrik dan rekam medis adalah aset pribadi yang sangat sensitif dan krusial. Data biometrik merujuk pada ciri khas biologis yang melekat pada fisik setiap manusia. Oleh karena sifatnya yang unik (berbeda setiap orang), maka data biometrik ini sangatlah penting. Contoh data biometrik mencakup sidik jari, pemindaian iris mata, atau data wajah yang bersifat permanen dan unik. Berbeda dengan kata sandi yang dapat diganti, data biometrik bersifat tetap. Jika terjadi kebocoran, data ini tidak dapat diperbarui dan berpotensi dimanfaatkan untuk tindakan kriminal.



Gambar 4.5 Ilustrasi Buka Kunci Layar dengan Pengenalan Wajah

c. Data Genetika

Data genetika menyimpan kode unik tubuh kita yang tidak bisa diubah (permanen), seperti cetak biru (*blueprint*) diri manusia. Informasi ini sangat sensitif karena berpotensi disalahgunakan untuk kloning ilegal atau manipulasi biologis. Peretas dapat menjual data genetik seseorang untuk meraup keuntungan pribadi. Tentu hal ini sangat



Gambar 4.6 Ilustrasi DNA

merugikan, terlebih jika data tersebut digunakan secara ilegal. Oleh karena itu, perlindungan hukum dan keamanan ekstra mutlak diperlukan untuk menjaga kerahasiaan dan integritas data genetika.

d. Catatan Kriminal

Catatan kriminal berisi riwayat sensitif yang bisa memengaruhi reputasi, peluang karir seseorang, dan pandangan sosial seseorang di masyarakat. Di Indonesia, catatan kejahatan terlampir pada Surat Keterangan Catatan Kepolisian (SKCK). SKCK biasa dilampirkan saat seseorang melamar pekerjaan atau juga digunakan untuk kenaikan jenjang karir, karena memuat rekam jejak hukum seseorang. Kebocoran data catatan kriminal dapat menimbulkan dampak serius, bahkan setelah masa hukuman berakhir. Hal ini tentu sangat merugikan bagi individu yang bersangkutan.



Gambar 4.7 Ilustrasi Map SKCK

e. Data Anak

Data keluarga berisi informasi pribadi, seperti hubungan kekerabatan, alamat, hingga kebiasaan sehari-hari. Jika data ini bocor, keamanan keluarga dapat terancam. Kebocoran data ini dapat dimanfaatkan pelaku kejahatan untuk melakukan penipuan, penculikan, atau target manipulatif lainnya. Karena itu, kita tidak boleh sembarangan memublikasikan nama maupun data anggota keluarga di ruang publik. Jika ada pihak yang mengatasnamakan keluarga, kita perlu melakukan pengecekan ulang (kroscek) untuk memastikan kebenaran informasi tersebut.



Gambar 4.8 Ilustrasi Nama Anggota Keluarga di Belakang Mobil

f. Data Keterangan Pribadi



Gambar 4.9 Ilustrasi Profil Media Sosial Menyertakan Informasi Privat

Pernahkah kamu membuat biodata? Biodata berisi informasi privat seperti nama lengkap, alamat, tanggal lahir, dan informasi lainnya tentang diri kita. Informasi ini harus dijaga dengan baik dan hanya diberikan kepada pihak yang benar-benar membutuhkan untuk keperluan penting. Jika bocor, data ini dapat disalahgunakan, misalnya untuk pembuatan dokumen palsu,

peminjaman dana, atau bahkan mengakses akun pribadi kita, terlebih apabila kata sandi yang kita gunakan berkaitan dengan data yang tercantum dalam biodata tersebut.

g. Data Lainnya Sesuai dengan Ketentuan Peraturan Perundang-Undangan.

Data lainnya ini adalah data tertentu untuk kepentingan tertentu. Setelah memahami data pribadi, gunakanlah informasi tersebut dengan bijak. Bagikan hanya kepada pihak yang benar-benar membutuhkan. Selain itu, orang lain tidak boleh menggunakan data kita tanpa izin. Setiap individu berhak mengetahui bagaimana datanya digunakan, bahkan berhak meminta data tersebut dihapus jika tidak setuju.

Melindungi data privat adalah tanggung jawab bersama, bukan hanya tugas pemilik data. Sekolah, perusahaan, atau teman sekalipun harus berhati-hati dalam mengelola data orang lain. Jika data disalahgunakan, ada hukum yang mengatur dan pelakunya bisa dikenai denda atau hukuman penjara. Oleh karena itu, kita semua harus lebih berhati-hati dan bijak dalam menjaga dan memberikan data pribadi.



Aktivitas
IH-KD-A01-I

▶ **Aktivitas Individu**

Berbagi Cerita Kebocoran Informasi

- ✓ Tujuan: Pada aktivitas ini, kamu akan berbagi informasi dan pengalaman mengenai kebocoran data dengan temanmu.
- ✓ Syarat pengetahuan: Data pribadi atau privat.

Aktivitas 1



Guru akan memulai permainan melempar bola kertas (*snowball throwing*) ke arah belakang guru dengan ketentuan:

1. Siswa yang pertama kali terkena lemparan bola (Siswa A), berbagi cerita atau pengalaman kebocoran informasi privat dan dampak kebocoran informasi tersebut.
2. Siswa A melempar bola lagi kepada teman yang lain. Siswa yang terkena lemparan bola berikutnya (siswa B) memberikan tanggapan berupa

solusi yang bisa dilakukan, agar kejadian tersebut tidak terulang lagi di kemudian hari.

3. Setelah mengomentari jawaban siswa A, siswa B boleh melempar ke arah siswa lain dan yang terkena lemparan berikutnya (siswa C) memaparkan pengalamannya seperti siswa A.
4. Hal ini dilakukan berulang, sampai 10 siswa memaparkan cerita dan 10 siswa memberikan tanggapan.

Aktivitas 2



Carilah tiga artikel atau berita mengenai kasus kebocoran data privat atau pribadi, baik dari media cetak maupun digital. Tulislah hasil temuanmu pada tabel berikut.

Tabel 4.1 Aktivitas Analisis Kasus Kebocoran Data

No	Nama Media Pers dan Judul Berita/Artikel	Data Privat yang Bocor	Dampak Kebocoran Data	Penanganan Kasus
1.
2.
3.

Pilihlah satu artikel atau berita yang memiliki penanganan kasus paling efektif. Kamu dapat membuat infografis sederhana berdasarkan artikel tersebut dan memajangnya di mading sekolah atau dinding kelas agar dapat dipelajari oleh teman-teman lainnya.

Ayo, Refleksi!

Nama :

Kelas :

Tanggal :

1. Apa informasi baru yang kamu dapatkan dari aktivitas berbagi tentang kebocoran data?

.....

.....

2. Bagaimana pendapatmu mengenai pentingnya menjaga keamanan data pribadi?

.....

.....

3. Bagaimana perasaanmu setelah mendengar pengalaman teman-teman tentang kebocoran data?

- a. Optimis
- b. Khawatir
- c. Lebih waspada
- d. Biasa saja
- e. Lainnya:

Jelaskan alasanmu.

.....

.....

4. Sebutkan langkah terbaik yang akan kamu lakukan untuk melindungi data pribadimu setelah aktivitas ini.

.....

.....

B. Serangan Siber

Setiap orang pasti memiliki data atau arsip yang penting. Andaikan data tersebut adalah barang berharga, tentu kita akan menyimpannya dengan baik di dalam rumah. Sebagai pemilik rumah, kita harus bisa mengontrol siapa saja yang diizinkan untuk keluar-masuk rumah demi alasan keamanan. Ibarat rumah, perangkat yang kita gunakan juga perlu dilindungi. Kita harus mengetahui dan membatasi siapa saja yang dapat mengakses data kita. Namun, sebaik apapun kita memberikan batasan, tetap ada risiko penyusup yang bisa masuk ke gawai kita untuk menyadap atau bahkan mencuri data. Lalu, apa yang harus kita lakukan? Gangguan semacam ini kerap disebut dengan istilah *malware*.

1 Jenis Serangan Siber

Di dunia digital, terdapat gangguan yang dapat membahayakan data dan privasi kita kapan saja. Gangguan ini disebut serangan siber, yaitu pemanfaatan teknologi untuk mencuri data pribadi, merusak perangkat, atau mengganggu ketersediaan layanan. Oleh karena itu, penting bagi kita untuk mengenali jenis-jenis serangan siber. Dengan memahami hal tersebut, kita dapat meningkatkan kewaspadaan dan mengetahui cara melindungi data serta privasi. Berikut adalah beberapa serangan siber yang sering terjadi. Ayo kita bahas satu per satu.

a. *Malware*

Malware adalah akronim dari *malicious software* yaitu program komputer yang dibuat untuk merusak, mengubah, atau mencuri data pada komputer atau jaringan komputer. Gangguan *malware* yang terjadi bisa terlihat atau bahkan tidak terlihat. Namun, meskipun dampaknya tidak selalu terlihat secara langsung, *malware* tetap dapat merugikan kita. Tentu, kita tidak rela jika data kita diretas atau bahkan dicuri oleh pihak yang tidak berwenang, bukan? Begitu menjengkelkannya *malware*! Sebelum mempelajari bagaimana cara mengatasi serangan *malware*, ayo kenali terlebih dahulu kategori serangan *malware* berikut ini.

- **Tidak Merusak**

Jenis serangan ini biasanya dilakukan untuk kesenangan atau kepuasan pribadi pelaku. Keberhasilan serangan memberi rasa puas, karena pelaku merasa mampu dan merasa dirinya dapat melakukan hal lebih. *Malware*

jenis ini umumnya tidak merusak *file* atau perangkat secara langsung, namun dapat mengganggu atau mencuri informasi. Contohnya, seperti *adware* yang menampilkan iklan yang tidak diinginkan. Ada juga *spyware* yang berfungsi menyadap dan mengumpulkan informasi pengguna. Meskipun tidak merusak data secara langsung, keberadaan *malware* ini bisa menjadi tanda bahwa sistem keamanan perangkat telah terbobol. Oleh karena itu, penting untuk tetap waspada dan menjaga perangkat agar tetap aman dari serangan *malware*.

- **Merusak**

Serangan jenis ini lebih berbahaya, karena dapat menghapus, mengubah, atau bahkan merusak *file* sehingga tidak dapat dioperasikan sebagaimana mestinya. Dampaknya tidak hanya pada sistem, namun juga berpotensi merusak perangkat keras. *Malware* jenis ini memang dirancang untuk merusak, menghancurkan, atau mengambil alih perangkat dan data. *Malware* perusak sering kali masuk ke dalam sistem melalui unduhan, *phishing*, atau eksploitasi celah keamanan perangkat lunak. Setelah terinstal, *malware* ini dapat merusak *file*, sistem operasi, bahkan mengganggu kinerja perangkat keras. Contoh kerusakan yang diakibatkan oleh *malware* yaitu dapat memberikan akses *remote* kepada peretas dan memungkinkan mereka untuk mengontrol atau mengawasi perangkat secara diam-diam.

Hal lain yang sering dilakukan peretas melalui *malware* adalah pemerasan. Modus ini dilakukan dengan menyusupkan *malware* ke dalam data sampai data target terkunci menggunakan enkripsi yang dibuat oleh peretas. Untuk membuka (mendekripsi) data tersebut, korban harus memperoleh kode yang hanya dimiliki oleh peretas. Selain itu, ada juga peretas yang membuat virus khusus yang hanya bisa dihapus dengan antivirus buatan mereka. Dengan cara ini, mereka memaksa banyak pihak yang terinfeksi untuk membeli antivirus tersebut, sehingga menambah pundi-pundi uang bagi pembuatnya. Beragamnya motif penyebaran *malware* menunjukkan pentingnya memahami jenis *malware*, cara penyebarannya, dan dampak serangan *malware*. Secara umum, kamu telah mempelajari bahaya *malware*. Selanjutnya, kamu akan mempelajari berbagai jenis *malware* dan bagaimana cara *malware* bisa memengaruhi komputer.



1) Virus

Seperti penyakit pada makhluk hidup, virus komputer dapat menyebar dengan cara menyisipkan dirinya sendiri ke dalam data, *file*, atau bahkan sistem operasi. Virus adalah potongan kode yang dapat mereplikasi diri dan menempel pada program lain. Dalam penyebarannya, biasanya virus memerlukan interaksi pengguna komputer, seperti menjalankan *file* yang terinfeksi. Virus tidak berbahaya saat komputer dalam kondisi luring (*offline*). Namun, ketika komputer terhubung ke jaringan, virus dapat menyebar lebih cepat ke perangkat lain, baik secara sengaja maupun tidak. Dampak serangan virus sangat bervariasi, diantaranya dapat merusak *file* sehingga pengguna tidak dapat menggunakan *file* tersebut. Virus juga bisa merusak sistem operasi, yang menyebabkan fitur komputer tidak dapat berfungsi sebagaimana mestinya.



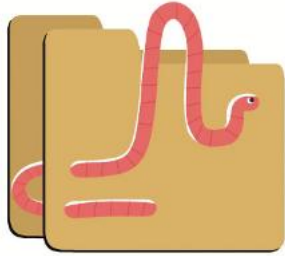
Gambar 4.10 Ilustrasi Virus

Efek negatif virus komputer adalah kemampuannya dalam memperbanyak diri, yang menyebabkan sumber daya komputer (seperti penggunaan memori) berkurang secara signifikan. Hampir 95% virus komputer menargetkan sistem operasi Windows, sementara sisanya menyerang Linux/GNU, Mac, FreeBSD, dan sistem operasi lainnya. Virus yang ganas tidak hanya merusak sistem operasi, namun dapat menyebabkan kerusakan pada perangkat keras.

2) Worm

Worm adalah perangkat lunak berbahaya yang dapat menggandakan diri dan menyebar secara otomatis tanpa memerlukan interaksi pengguna. *Worm* sering kali menyebar melalui *email* atau *file* yang dikirimkan. *Worm* berupa potongan kode, mampu mereplikasi dirinya sendiri dan menyebar melalui jaringan. *Worm* dapat menghasilkan *file* acak tidak berguna pada komputer.

Karena kemampuan duplikasi *worm* sangat cepat, mengakibatkan jumlah *file* yang tidak diperlukan semakin banyak. Hal ini meningkatkan



Gambar 4.11 Ilustrasi *Worm*

proses yang berjalan pada komputer, sehingga kinerja komputer melambat dan bahkan dapat menyebabkan *Denial of Service* (DoS). DoS merupakan jenis serangan yang bertujuan untuk membuat layanan atau sistem komputer tidak dapat diakses oleh pengguna. DoS adalah upaya untuk menghalangi akses ke suatu layanan atau situs web, sehingga orang lain tidak

bisa menggunakan layanan seperti biasa. Selain dapat menyebar dalam sebuah sistem, *worm* juga dapat menyebar ke banyak sistem lainnya melalui jaringan yang terhubung dengan sistem yang telah terinfeksi. Untuk menanganinya, beberapa aplikasi utilitas dan antivirus biasanya menyediakan fitur untuk menghapus *file* duplikat, sehingga dapat membantu membersihkan dampak serangan *worm*.

3) **Ransomware**

Ransomware adalah jenis *malware* yang dirancang untuk mengunci atau mengenkripsi data di komputer, lalu meminta uang agar pengguna bisa kembali mengakses data tersebut. Bayangkan jika seseorang mengunci brankas yang berisi surat dan dokumen berhargamu, lalu meminta tebusan sebagai syarat untuk mendapatkan kuncinya kembali. Sederhananya, *ransomware* akan mencegah pengguna mengakses komputer sampai mereka membayar uang kepada peretas. *Ransomware* merupakan jenis *malware* yang populer karena biasanya menyerang data penting atau rahasia, sehingga sering menarik perhatian banyak pihak.



Gambar 4.12 Ilustrasi *Ransomware*

4) **Backdoor**

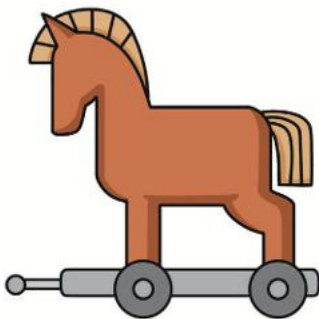
Backdoor adalah pintu belakang, yaitu metode yang memungkinkan akses tersembunyi ke sistem komputer, sering kali tanpa sepengetahuan pemilik sistem. Melalui “pintu belakang” ini, *backdoor* akan memberikan akses kepada admin *malware* untuk masuk ke dalam sistem, lalu melakukan *troubleshooting* atau mengubah pengaturan sistem dari luar.



Gambar 4.13 Ilustrasi *Backdoor*

Backdoor memungkinkan penyerang melewati kontrol keamanan normal pada sistem, sehingga mendapatkan akses sesuai ketentuan penyerang. *Backdoor* dapat berupa *file* atau data yang menyamar dan bersembunyi untuk disisipkan ke dalam sistem, sehingga pelaku dapat mengendalikannya dari jarak jauh. Berbeda dengan *trojan horse* yang menyerang secara langsung, *backdoor* justru bertindak saat menemukan momen yang tepat untuk melakukan serangan jarak jauh.

5) **Trojan Horse**



Gambar 4.14 Ilustrasi Kuda Troya

Trojan horse atau kuda troya adalah salah satu cerita mitologi Yunani, dimana sebuah patung kuda besar diberikan sebagai hadiah kepada sebuah kota. Namun, di dalam patung tersebut tersembunyi prajurit elit yang keluar saat penduduk lengah untuk menghancurkan kota. Karena kemiripan pola penyerangannya dengan cerita dalam mitologi Yunani tersebut, gangguan ini dinamakan sebagai *trojan horse* atau kuda troya.

Jenis perangkat lunak berbahaya ini menyamar sebagai *file* biasa, yang terlihat berguna atau tidak berbahaya, padahal sebenarnya menyembunyikan fungsi jahat atau berbahaya. Perbedaan antara kuda troya dengan *backdoor* terletak pada cara penyerangannya. Kuda troya menyerang secara langsung dari dalam sistem, namun

backdoor melakukan penyerangan dari jarak jauh. Kuda troya sering kali mengunduh dan menginstal versi baru dari program berbahaya lainnya pada komputer korban dan biasanya menargetkan komputer yang telah terinfeksi sebelumnya.

6) **Spyware**

Spyware adalah perangkat lunak berbahaya yang memata-matai pengguna untuk mencuri informasi seperti nama pengguna dan kata sandi. Program ini menyusup ke perangkat komputer untuk memantau kebiasaan pengguna serta mencuri informasi penting, baik data dari penyimpanan lokal maupun selama beraktivitas di internet.



Gambar 4.15 Ilustrasi *Spyware*

Spyware dapat menyebar melalui jaringan internet maupun secara *offline* dengan cara menyalin *file* yang ada *spyware* di dalamnya, lalu merekam aktivitas penggunaan komputer. Akan sangat berbahaya, jika pengguna memasukkan informasi akun (*username* dan *password*) tanpa verifikasi lanjutan. Oleh karena itu, verifikasi berlapis seperti autentikasi dua faktor harus tetap dilakukan pada saat masuk ke akun privat.

b. **Phishing**

Phishing adalah jenis serangan dunia maya yang menggunakan *email*, pesan teks, panggilan telepon, atau situs web palsu untuk mengelabui orang agar membagikan data sensitif, mengunduh perangkat lunak perusak, atau mengekspos diri mereka pada kejahatan dunia maya. Peretas akan mengarahkan calon korban untuk mengklik tautan atau mengunduh lampiran berbahaya yang dirancang menyerupai aslinya (misalnya berupa logo, pencatutan nama, atau alamat *email* yang hampir mirip dengan aslinya, namun identitas pengirim tetap disamarkan).

Phishing tidak menyerang secara langsung karena serangan akan bereaksi apabila korban memberikan respon terhadap pancingan yang dibuat oleh peretas. Trik pancingan ini juga dapat dilakukan dengan berpura-pura menjadi

seseorang atau institusi yang dipercaya korban (seperti keluarga, teman dekat, sekolah, bank, atau instansi resmi lainnya) untuk mencapai tujuannya.

Untuk menghindari serangan ini, kamu harus selalu waspada apabila ada pihak yang meminta mengirimkan data pribadi atau informasi terkait akun digitalmu. Jangan asal mengklik tautan mencurigakan. Apabila ada *email* atau notifikasi yang tidak kamu percayai, sebaiknya tidak perlu dibuka. Selalu periksa alamat situs *website* tersebut dan pastikan berasal dari domain resmi.

c. *Distributed Denial of Service (DDoS)*

Bagaimana jika sebuah toko didatangi ribuan orang palsu yang berpura-pura ingin berbelanja, tetapi hanya memenuhi ruangan? Hal tersebut tentu akan merugikan pedagang dan pelanggan asli yang benar-benar berniat berbelanja karena tidak dapat masuk. Itulah analogi serangan DDoS. Serangan DDoS menyebabkan sebuah situs atau aplikasi menjadi lumpuh tidak dapat digunakan. Cara kerjanya adalah dengan membanjiri server, sehingga kinerja server menurun akibat melebihi batas maksimal kemampuannya dalam menerima permintaan. Dampaknya, pengguna asli tidak dapat mengakses situs atau aplikasi tersebut.

Bagi suatu instansi atau perusahaan, serangan ini dapat menyebabkan kerugian finansial yang besar karena transaksi terhenti dan pelanggan beralih ke pesaing. Reputasi serta kepercayaan pelanggan terhadap keamanan dan keandalan layanan pun menurun. Selain itu, instansi atau perusahaan harus mengeluarkan biaya tambahan untuk pemulihan sistem. Pada tingkat yang lebih luas, jika serangan menasar lembaga pemerintah atau infrastruktur penting, hal ini dapat mengganggu pelayanan publik bahkan menimbulkan keresahan di masyarakat.

Salah satu kasus serangan DDoS pernah menimpa layanan DeepSeek. Perusahaan tersebut saat itu sedang naik daun dalam industri kecerdasan artifisial. DeepSeek menghadapi serangan DDoS berskala besar sejak model penalarannya dirilis pada 20 Januari 2025. Serangan itu terjadi tepat ketika DeepSeek mengalami lonjakan popularitas, bahkan menyalip ChatGPT dari OpenAI sebagai aplikasi gratis yang paling banyak diunduh di App Store Apple. Akibatnya, situs www.deepseek.com sempat tidak dapat diakses oleh pengguna baru.

d. *Man In The Middle (MitM)*

Serangan *Man-in-the-Middle* (MitM) terjadi ketika pihak ketiga menyadap komunikasi antara dua pihak tanpa sepengetahuan mereka, terutama pada jaringan Wi-Fi publik. Dalam situasi ini, penyerang menempatkan diri ‘di tengah’ komunikasi antara pengirim dan penerima. Dampaknya, data yang dikirim dapat disadap, diubah, atau dicuri oleh peretas sebelum sampai ke tujuan, tanpa disadari oleh pengguna. Akibatnya, data yang diterima merupakan data yang telah dimodifikasi. Serangan MitM umumnya terjadi pada jaringan publik. Penyerang sering menggunakan trik pengalihan jaringan ke halaman tiruan untuk memaksa pengguna *login* sebelum mengakses Wi-Fi. Hal ini membuat pengguna mengira sedang terhubung langsung ke layanan asli, padahal sebenarnya semua data telah melalui tangan penyerang. Oleh karena itu, saat terhubung ke Wi-Fi gratis di area publik, disarankan menggunakan *Virtual Private Network* (VPN) untuk mengamankan koneksi.

Jaringan Wi-Fi publik umumnya memiliki keamanan yang lemah, sehingga informasi yang kita kirim berisiko disadap oleh penyerang MitM. Untuk mengatasi hal ini, VPN berfungsi sebagai perisai. VPN membuat saluran khusus yang terlindungi, seperti lorong rahasia di dunia maya, sehingga seluruh data yang kita akses atau bagikan akan terenkripsi dan tidak dapat dibaca oleh pihak luar. Dengan cara ini, data sensitif seperti kata sandi, pesan, atau nomor kartu tetap aman meskipun ada penyusup. Selain itu, VPN juga menyamarkan alamat IP asli pengguna sehingga identitas serta lokasi lebih terjaga. Oleh karena itu, menggunakan VPN saat menggunakan Wi-Fi publik menjadi langkah sederhana yang sangat penting untuk melindungi privasi dan keamanan digital.

e. *Password Attack*

Password attack adalah serangan dengan cara mencoba menebak kata sandi untuk masuk ke akun atau perangkat korban. Karena kata sandi merupakan kunci pertama dalam dunia digital, serangan ini sangat sering dilakukan. Terdapat beberapa jenis *password attack*, antara lain:

1) *Brute Force Attack*

Peretas akan mencoba semua kemungkinan kombinasi huruf, angka, dan simbol hingga menemukan *password* yang benar.

2) **Dictionary Attack**

Peretas menggunakan daftar kata (*dictionary*) yang berisi ribuan hingga jutaan kata umum, misalnya “*password*”, “*qwerty*”, atau “*iloveyou*”. Selain itu, peretas juga akan mencoba kata yang identik dengan diri korban, seperti hobi, klub bola kesukaan, nama publik figur favorit, atau yang lainnya.

3) **Credential Stuffing**

Peretas akan mencoba menggunakan kata sandi yang telah bocor dari akun tertentu, lalu mencobanya pada akun korban yang lain. Misalnya, apabila kata sandi akun Instagram telah bocor, maka peretas akan mencoba menggunakan kata sandi tersebut untuk masuk ke akun lain milik korban.

4) **Keylogger Attack**

Peretas memasang aplikasi atau program berbahaya (*keylogger*) di perangkat korban untuk merekam setiap aktivitas pada *keyboard*. Dengan demikian, apapun yang diketikkan korban akan terekam oleh aplikasi *keylogger*.

2. Dampak Pada Teknologi Modern

Saat ini banyak perangkat dan aplikasi yang terhubung ke internet. Teknologi yang saat ini masif dikembangkan yaitu *Internet of Things* (IoT), Kecerdasan Artifisial (KA), dan *Cloud*. Namun, ketiganya tetap memiliki risiko terkena dampak serangan *malware*.

a. **Internet of Things (IoT)**

IoT adalah teknologi yang memungkinkan benda di sekitar kita terhubung ke internet dan saling berkomunikasi. Contohnya seperti jam tangan pintar yang dapat memantau kesehatan untuk dibuat rekap data kesehatan kita, CCTV yang bisa dipantau melalui ponsel sekaligus diatur tindakannya ketika terdeteksi objek asing, atau lampu rumah yang dapat dikendalikan dari jauh. Namun, perangkat pintar tersebut juga berisiko terkena *malware*. Misalnya, *malware* dapat meretas CCTV sehingga orang asing bisa melihat isi rumah, atau membuat jam tangan pintar menjadi lambat karena diam-diam digunakan untuk mengirim data kepada peretas. Dampaknya, privasi

keluarga dapat hilang dan perangkat sehari-hari yang seharusnya membantu justru membahayakan pemiliknya.

b. Kecerdasan Artifisial (KA)

Malware bisa merusak cara kerja sistem cerdas. Contohnya, sebuah aplikasi KA yang digunakan untuk merekomendasikan film bisa dimanipulasi dengan data palsu, sehingga hasilnya tidak akurat. Bahkan, *malware* dapat dipakai untuk membuat pemalsuan konten, seperti gambar atau video palsu yang sangat mirip aslinya. Dampaknya, KA dapat memberikan informasi yang salah, menyebarkan kebohongan, atau menurunkan kepercayaan masyarakat terhadap teknologi.

c. Cloud

Cloud merupakan media menyimpan data di internet. *Malware* dapat menyerang *cloud* dengan cara mengunci file menggunakan *ransomware* (meminta tebusan agar data dapat dibuka kembali) atau mencuri data penting dari akun seseorang. Bayangkan jika semua tugas sekolah, foto, atau dokumen penting disimpan di *cloud* lalu terkunci dan tidak dapat dibuka lagi. Dampaknya bisa sangat besar, baik bagi individu maupun perusahaan, karena data merupakan aset yang sangat berharga.

3. Penanganan

Dalam menjaga kesehatan tubuh, mencegah penyakit lebih baik daripada mengobati. Prinsip serupa berlaku untuk mencegah penyebaran *malware*. Dengan menginstal perlindungan yang sesuai, ancaman *malware* dapat langsung terdeteksi dan ditangani dengan cepat. Langkah termudah mengatasi *malware* adalah kontrol diri yaitu menghindari perilaku berisiko yang memungkinkan kontaminasi *malware*. Beberapa cara yang dapat digunakan antara lain:

a. Antivirus

Di dunia digital, ada berbagai macam penjahat siber yang bisa membahayakan komputer. Mereka sering disebut *malware*, seperti yang telah dipelajari sebelumnya. *Malware* bisa mencuri data, merusak sistem, atau bahkan membuat perangkat keras komputer tidak bisa digunakan. Dengan menggunakan antivirus, kita seperti memiliki tameng pelindung yang dapat mendeteksi dan mengatasi ancaman ini sebelum mereka bisa merusak

komputer kita. Perangkat lunak antivirus dirancang untuk mencegah infeksi komputer dengan mendeteksi data dan perangkat lunak berbahaya. Jika diperlukan, antivirus juga akan memberikan rekomendasi untuk menghapus *malware* tersebut dan bahkan mengkarantina *file* tertentu, sehingga *file* tersebut tidak dapat digunakan sementara waktu sampai ada tindakan berikutnya dari pengguna.



Gambar 4.16 Ilustrasi Ikon Antivirus

Sebagian besar antivirus memiliki kemampuan yang sama. Antivirus bekerja dengan mencari pola tertentu yang biasa digunakan oleh virus. Hal ini karena pembuat *malware* sering menggunakan kode yang identik saat menyebarkan virus ke banyak target. Kode identik tersebut membentuk pola dalam memori penyimpanan yang dapat dipindai dengan cepat dan efisien oleh antivirus. Berikut adalah beberapa kemampuan atau fitur yang umumnya tersedia pada antivirus.

1. Kemampuan untuk memindai *file* terkompresi dan paket *executable* seperti *file* winrar, winzip, ISO, dan *file* kompresi lainnya.
2. Alat untuk melakukan pemindaian *file* atau direktori secara langsung.
3. *Driver* perlindungan diri untuk mencegah *malware* menyerang antivirus itu sendiri.
4. Fungsi pemeriksaan *firewall* dan jaringan.

Antivirus tidak hanya melakukan pemindaian secara *offline*, tetapi juga diperlukan saat terhubung ke internet. Fitur ini dikenal dengan *Web Security* yang bisa terintegrasi dengan *browser* sebagai ekstensi (fitur tambahan pada aplikasi web browser). Banyak produk antivirus menyediakan fitur ini untuk browser populer guna memeriksa reputasi situs web, alamat URL, dan bahkan konten *file* yang diunduh untuk menentukan tingkat keamanannya. Keamanan pada *website* sangatlah penting karena data yang kita akses secara *online* ibarat seperti pintu gerbang pertukaran informasi antara perangkat komputer kita dengan dunia digital.

b. Firewall

Salah satu faktor kuatnya keamanan komputer adalah *firewall*. *Firewall* ibarat sebuah benteng kokoh dengan dinding tinggi dan juga pintu gerbang yang dijaga ketat. “Penjaga” ini bertugas untuk menyaring lalu lintas data yang diperbolehkan untuk keluar masuk jaringan, sehingga semua sistem di dalamnya tetap aman. *Firewall* umumnya telah terintegrasi dalam sistem operasi. Kita dapat menyesuaikan pengaturan *firewall* secara lebih rinci sesuai dengan kebutuhan keamanan komputer.



Gambar 4.17 Ilustrasi Ikon Firewall

c. Backup File

Bayangkan kamu memiliki sebuah buku catatan mata pelajaran yang berisi semua hal penting seperti catatan untuk ujian, tugas, hingga catatan dari guru. Jika buku itu hilang atau rusak, tentu akan sangat merepotkan, bukan? Nah, inilah alasan pentingnya melakukan *backup file*. *Backup file* berarti membuat salinan data sebelum rusak atau hilang karena serangan siber atau sebab lainnya.

Dengan begitu, jika terjadi sesuatu yang tidak bisa kita prediksi, kamu masih memiliki cadangan yang aman. *Backup file* dapat dilakukan melalui berbagai media penyimpanan seperti flashdisk, hardisk, SSD, CD, atau DVD. Kamu juga bisa menggunakan layanan penyimpanan *online (cloud)* seperti Google Drive, One Drive, atau iCloud. Dengan cara ini, *file* cadangan bisa disimpan dan diakses dari mana saja selama terhubung dengan internet.



Gambar 4.18 Ilustrasi Ikon Backup File



Aktivitas IX-KD-B01-K



Aktivitas Kelompok

Bermain Peran Serangan *Malware* dan Penanganannya

- ✔ Tujuan: Pada aktivitas kali ini, kalian akan bermain peran dengan karakter yang berbeda untuk lebih memahami serangan *malware*.
- ✔ Syarat pengetahuan: *Malware*.

Aktivitas 1



Bagilah kelas menjadi beberapa kelompok. Masing-masing kelompok akan mendapatkan topik nama *malware* yang berbeda dengan kelompok lainnya. Setiap kelompok beranggotakan tiga orang dengan pembagian peran sebagai berikut.

- Satu orang berperan sebagai penyerang (*hacker*), yang merencanakan dan meluncurkan serangan *malware* (misalnya *ransomware*, *backdoors*, atau yang lainnya).
- Dua orang berperan sebagai tim *Information Technology* (IT), yang mencari tahu bagaimana mengatasi serangan atau dampak yang ditimbulkan dari *malware*.

Setelah kamu mendapatkan peran, lakukan kegiatan berikut ini.

1. Siapkan skenario dan materi simulasi serangan *malware*. Tentukan media penyisipan yang akan digunakan, seperti *email*, *pop-up window*, atau media lainnya).
2. Setiap anggota kelompok harus memainkan peran sesuai dengan skenario yang telah ditetapkan.
3. Setiap peran dalam kelompok harus menyampaikan tugas sesuai dengan tanggung jawabnya masing-masing.
4. Peragakan simulasi serangan dan penanganan *malware* di depan kelas sesuai skenario.
5. Setiap kelompok saling menilai simulasi berdasarkan kesesuaian peran, efektivitas tindakan, dan optimalitas keputusan yang diambil.
6. Setelah simulasi, adakan diskusi tentang apa yang terjadi, bagaimana setiap peran berfungsi, dan apa yang bisa dipelajari dari aktivitas tersebut.



Aktivitas 2



Pada aktivitas ini, setiap kelompok mencari satu buah kasus serangan *malware*, yang dapat diambil dari berita *online*, jurnal, atau sumber tepercaya lainnya. Setelah itu, jawablah pertanyaan berikut.

1. Jelaskan dampak serangan *malware* terhadap korban (individu, organisasi, atau sistem).
2. Coba kalian analisis, bagaimana upaya pencegahan yang dapat dilakukan agar kasus serupa tidak terulang lagi?

Setelah berhasil menganalisis kasus tersebut, lakukan langkah berikut.

1. Tukar kasus kelompok kalian dengan kasus dari kelompok lain. Lakukan hal yang sama yaitu menuliskan dampak dan upaya pencegahan untuk kasus kedua ini.
2. Bandingkan solusi penanganan yang diusulkan kelompokmu dengan kelompok yang lain.



Ayo, Refleksi!

- Nama :
- Kelompok :
- Tanggal :

Peran dalam simulasi (centang salah satu):

- Penyerang (*hacker*)
- Tim IT 1
- Tim IT 2

1. Apa yang kamu lakukan dalam peranmu saat simulasi?
.....
2. Bagian mana yang paling menantang dalam menjalankan peranmu?
.....
3. Dari simulasi yang dilakukan, apa yang kamu pelajari tentang cara kerja *malware*?
.....
4. Menurut kamu, apa saja upaya pencegahan dan penanganan *malware* yang paling efektif?
.....
5. Perasaan dan sikap
- a. Tertantang
 - b. Lebih paham tentang *malware*
 - c. Lebih waspada
 - d. Biasa saja
 - e. Lainnya:
- Jelaskan alasannya:
.....
6. Sebutkan 2-3 langkah nyata yang akan kamu lakukan untuk mencegah atau menghadapi ancaman *malware* di kehidupan sehari-hari:
- a.
 - b.
 - c.

C. Keamanan Jaringan Nirkabel di Ruang Publik

Siapa yang pernah mencari jaringan Wi-Fi gratis di tempat umum agar tetap bisa terhubung ke internet? Meskipun mungkin kamu enggan mengakuinya, faktanya saat ini hampir di mana pun kita berada dan apa pun perangkat yang digunakan, baik ponsel pintar, laptop, maupun tablet, pastinya terasa kurang lengkap apabila tidak terhubung ke jaringan internet. Namun, tahukah kalian bagaimana cara perangkat ini terhubung? Dan tahukah kamu bahwa saat menggunakan Wi-Fi publik, data pribadimu dapat diintip oleh orang lain?

Sebagai pelajar abad 21 yang tengah mempersiapkan diri menghadapi era Revolusi Industri 5.0, kamu tidak hanya dituntut menjadi pengguna teknologi, tetapi juga pengguna yang paham, cermat, dan bijak dalam memanfaatkan teknologi tersebut. Untuk itu, mari kita pelajari bagaimana perangkat digital dapat terhubung ke jaringan lokal (LAN) dan internet, serta mengidentifikasi risiko keamanan yang mungkin terjadi saat menggunakan jaringan nirkabel. Dengan memahami ini, kamu dapat menggunakan internet dengan lebih aman dan memahami cara kerja teknologi dalam kehidupan sehari-hari.

1. Jaringan Nirkabel

Jaringan nirkabel adalah jaringan komputer yang menghubungkan perangkat tanpa menggunakan kabel fisik, melainkan melalui gelombang radio. Pada jenjang sebelumnya, kamu telah mempelajari dasar-dasar jaringan nirkabel. Sekarang, mari kita ulas kembali secara singkat beberapa jenis jaringan nirkabel yang umum ditemui dalam kehidupan sehari-hari.

- a. *Wireless Fidelity* (Wi-Fi), digunakan di rumah, sekolah dan kantor. Wi-Fi tidak hanya digunakan untuk mengakses internet, tetapi juga dapat digunakan untuk mengirimkan *file* dalam jaringan lokal, misalnya dengan menggunakan aplikasi Share-IT.
- b. Bluetooth, menghubungkan perangkat dalam jarak dekat, seperti *smartphone*, mikrofon, *keyboard*, TWS, *speaker*, dan lainnya. Hampir semua gawai dan aksesoris saat ini dilengkapi dengan bluetooth yang bisa digunakan untuk mentransfer *file* dalam jarak dekat. Bluetooth juga bisa digunakan untuk melakukan *tethering*, agar kita dapat berbagi jaringan internet dengan perangkat lainnya.

- c. Infrared atau infra merah, biasanya digunakan pada perangkat yang terhubung melalui *remote control*, seperti TV, AC, kipas angin, atau perangkat elektronik lainnya.
- d. *Near Field Communication* (NFC), digunakan untuk pembayaran digital. Contohnya saat menggunakan kartu pembayaran jalan tol (e-toll). Cara penggunaannya adalah dengan menempelkan kartu pada perangkat untuk melakukan transaksi. NFC pada ponsel juga dapat dimanfaatkan untuk mengisi ulang saldo kartu pembayaran digital tertentu.

2. Dampak Pengabaian Keamanan Jaringan

Setelah memahami berbagai jenis jaringan nirkabel seperti Wi-Fi, Bluetooth, Infrared, dan NFC, kita juga perlu mengetahui bahwa setiap jenis jaringan tersebut memiliki risiko keamanan tersendiri. Oleh karena itu, kita perlu menerapkan langkah-langkah keamanan agar jaringan yang digunakan tetap aman. Berikut adalah tabel yang menunjukkan berbagai upaya keamanan jaringan yang bisa dilakukan, serta dampak yang mungkin terjadi jika langkah-langkah tersebut diabaikan.

Tabel 4.2 Dampak Pengabaian Keamanan Jaringan

No	Keamanan Jaringan yang Dilakukan	Kemungkinan Dampak Apabila Diabaikan
1.	Mengatur kata sandi Wi-Fi yang kuat dan tidak mudah ditebak	Wi-Fi dapat diretas orang lain, digunakan tanpa izin, dan memperlambat koneksi
2.	Mengaktifkan enkripsi WPA2 atau WPA3 di router	Data dapat disadap dengan mudah oleh orang yang tidak bertanggung jawab
3.	Menyembunyikan SSID (nama Wi-Fi) agar tidak tampil otomatis	Siapa pun dapat melihat jaringan dan mencoba menyusup
4.	Memisahkan antara jaringan Wi-Fi publik dan pribadi untuk pembatasan kecepatan akses	Apabila kata sandi SSID publik telah bocor, setidaknya tidak terlalu berpengaruh terhadap kecepatan SSIS utama yang digunakan pemilik Wi-Fi
5.	Rutin mengganti kata sandi Wi-Fi	Kata sandi dapat tersebar dan disalahgunakan tanpa kamu sadari

No	Keamanan Jaringan yang Dilakukan	Kemungkinan Dampak Apabila Diabaikan
6.	Mengaktifkan <i>firewall</i> .	Perangkat tidak terlindungi dari serangan dari luar jaringan.
7.	Filter alamat kartu jaringan (<i>MAC Address</i>).	Hanya perangkat dengan alamat MAC yang telah didaftarkan sebelumnya, yang dapat terhubung ke jaringan. Kita juga bisa membatasi akses berdasarkan alamat MAC tertentu melalui pengaturan pada <i>router</i> .

3. Risiko Pengiriman Data di Jaringan Publik

Internet telah menjadi bagian dari kehidupan sehari-hari, baik di sekolah, rumah, saat berkumpul dengan teman, komunitas, kerja kelompok, bahkan di tempat wisata. Kita dapat dengan mudah terhubung ke internet baik melalui jaringan seluler maupun Wi-Fi. Saat menggunakan internet, perangkatmu akan menerima data (*download*) dan mengirim data (*upload*). Tetapi, pernahkah kamu mempertanyakan keamanan aktivitas ini jika dilakukan melalui jaringan Wi-Fi publik?

Nah, pada materi berikut, kita akan membahas dampak yang mungkin terjadi ketika menggunakan jaringan publik tanpa perlindungan, serta bagaimana cara melindungi diri agar tetap aman saat *online* di ruang publik. Di era digital, ruang publik tidak hanya merujuk pada tempat fisik, melainkan juga mencakup jaringan yang terhubung secara digital. Jaringan internet di ruang publik cenderung lebih rentan disalahgunakan. Berikut beberapa risiko yang bisa terjadi.

a. Pencurian Data Pribadi

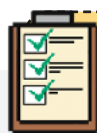
Salah satu risiko terbesar adalah pencurian data pribadi. Misalnya, saat login ke akun media sosial atau *email* melalui Wi-Fi publik tanpa perlindungan, data seperti *username* dan *password* dapat disadap oleh pihak tidak bertanggung jawab. Pelaku dapat mencatat data tersebut dan menggunakannya untuk membobol akunmu. Dalam kasus yang lebih parah, informasi pribadi seperti alamat rumah, nomor telepon, atau bahkan data tugas sekolah dapat disalahgunakan atau disebarluaskan tanpa izin.

b. Penyadapan

Ketika kamu menggunakan jaringan publik tanpa perlindungan keamanan tambahan, aktivitas *online* seperti situs yang diakses, kata-kata yang diketik, dan seluruh aktivitas *browsing* dapat disadap atau direkam oleh pihak tidak bertanggung jawab. Hal ini merupakan pelanggaran terhadap privasi digital. Informasi pribadi seperti minat, kebiasaan menonton, hingga situs yang sering diakses dapat dianalisis dan dimanfaatkan untuk tujuan tertentu tanpa sepengetahuan kamu.

c. Penyebaran *Malware*

Jaringan publik dapat menjadi tempat penyebaran *malware*. Tidak menutup kemungkinan, bahwa pengguna lain yang terhubung pada Wi-Fi yang sama, dapat menyebarkan *file* berbahaya secara sengaja di jaringan tersebut. Ketika kita mengunduh *file* dari internet atau mengklik tautan mencurigakan, *malware* dapat masuk ke perangkat tanpa disadari. *Malware* ini bisa menyebabkan kerusakan pada *file*, memperlambat perangkat, atau bahkan mengakses data yang tersimpan di dalamnya.



Aktivitas IX-KD-C01-I



Aktivitas Individu

Menganalisis Kasus Keamanan Jaringan

- ✔ Tujuan: Pada aktivitas kali ini, kamu diharapkan dapat menganalisis kasus keamanan jaringan nirkabel pada ruang publik.
- ✔ Syarat pengetahuan: Keamanan jaringan nirkabel.

Carilah satu contoh berita di internet yang membahas kasus pencurian data melalui Wi-Fi publik. Kemudian, buatlah ringkasan yang berisi:

1. Deskripsi singkat kejadian.
2. Korban dan dampak kerugian.
3. Upaya pencegahan dan penanganan yang dilakukan.

Setiap peserta didik menuliskan upaya terbaik untuk mencegah pencurian data pada jaringan publik dan saling mengomentari satu sama lain pada *sticky notes*.



Aktivitas IX-KD-C02-K



Aktivitas Individu

Adu Argumen Penggunaan Wi-Fi Publik

- ✔ Tujuan: Pada aktivitas kali ini, kalian diharapkan belajar bersama-sama tentang penggunaan Wi-Fi publik dari dua sudut pandang yang berbeda dan menarik kesimpulan dari kedua perspektif tersebut.
- ✔ Syarat pengetahuan: Keamanan jaringan nirkabel.

Ikuti langkah-langkah aktivitas berikut ini.

1. Guru membagi peserta didik dalam beberapa kelompok yang mewakili tim pro dan kontra.
2. Setiap kelompok menyiapkan beberapa argumen dan bukti sebagai penguat argumen pro atau kontra mengenai penggunaan Wi-Fi publik.
3. Guru akan memilih secara acak kelompok pro dan kontra untuk berdebat secara terbuka.
4. Diskusi akan dipandu oleh guru sebagai moderator.
5. Kelompok lain dapat menanggapi atau menambahkan argumen setelah memperoleh izin dari guru.



Ayo, Refleksi!

Nama :
 Kelas :
 Tanggal :

1. Apa informasi baru yang kamu peroleh setelah membaca berita tentang pencurian data melalui Wi-Fi publik?

2. Menurutmu, mengapa pencurian data melalui Wi-Fi publik berbahaya?

3. Bagaimana perasaanmu setelah mengetahui risiko penggunaan Wi-Fi publik?

- a. Lebih waspada
- b. Cemas
- c. Termotivasi untuk belajar lebih jauh
- d. Biasa saja
- e. Lainnya:

Jelaskan alasanmu:

4. Tuliskan tiga langkah nyata yang akan kamu lakukan untuk melindungi data pribadimu saat menggunakan jaringan publik!

- a.
- b.
- c.

D. Perlindungan Data Pribadi

Perlindungan data pribadi menjadi prioritas yang tidak dapat diabaikan. Data pribadi, mulai dari identitas, alamat, riwayat transaksi, hingga informasi kesehatan, memiliki nilai kerahasiaan yang sangat tinggi dan rentan disalahgunakan. Kebocoran data dapat berdampak serius, mulai dari pencurian identitas, penipuan, hingga kerugian finansial. Oleh karena itu, pemahaman tentang cara melindungi data pribadi merupakan langkah awal yang penting. Berikut ini beberapa teknik yang dapat digunakan untuk melindungi data kita dari peretasan.

1. Kata Sandi yang Kuat

Di era digital saat ini, kata sandi memainkan peran yang sangat penting dalam menjaga keamanan informasi pribadi. Kata sandi merupakan kunci dan langkah pertama, setiap kali kita mengakses akun *online*, seperti *email*, media sosial, atau layanan pada situs tertentu. Karena fungsinya yang krusial, penting untuk memahami bagaimana membuat kata sandi yang kuat. Semakin kuat kata sandi, semakin sulit pula orang lain membukanya.

Kata sandi yang kuat merupakan pertahanan pertama terhadap serangan siber. Jika kata sandi lemah, peretas dapat dengan mudah menggunakan metode *brute force* untuk memperoleh akses ke akunmu. *Brute force* adalah metode serangan siber, di mana peretas mencoba berbagai kombinasi kata sandi atau kunci enkripsi secara sistematis hingga berhasil. Berikut ini merupakan saran dalam pembuatan kata sandi yang kuat.

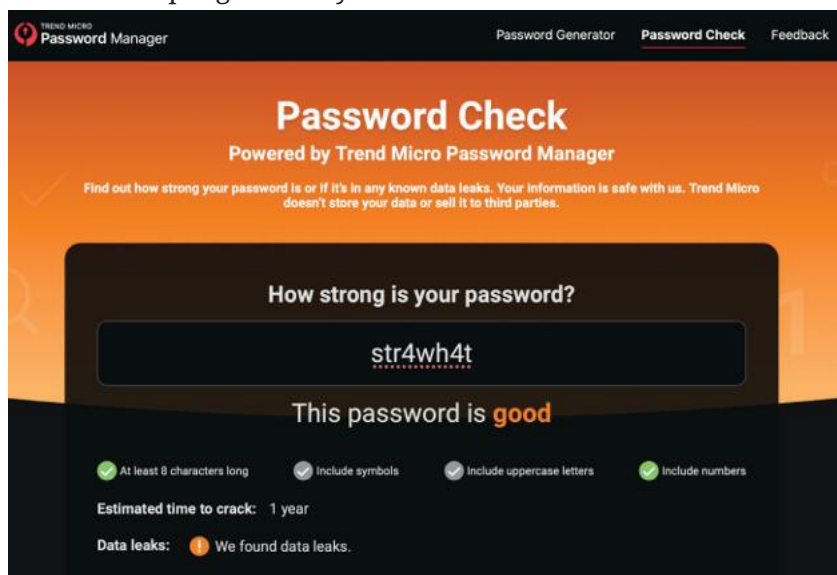
- a. **Panjang karakter kata sandi.** Gunakan kata sandi yang memiliki setidaknya 12 karakter. Semakin panjang jumlah karakter kata sandi, maka akan semakin sulit untuk dipecahkan.
- b. **Kombinasi karakter.** Buatlah kata sandi yang menggabungkan antara huruf kapital (A-Z), huruf kecil (a-z), angka (0-9), dan simbol (@, #, \$, dll.).
Contoh: **S3nj4&&m4Lam.**
- c. **Hindari informasi privat.** Hindari penggunaan nama, tanggal lahir, atau informasi privat lain yang mudah ditebak.
Contoh : hindari kata sandi seperti **17082013** yang menggunakan nama bulan atau tahun lahir asli pengguna.

- d. **Gunakan frasa atau gabungan kata.** Ciptakan kata sandi dari gabungan beberapa kata yang tidak terkait satu dengan yang lain.
Contoh: **#K3rt45Tr4mpol1n.**
- e. **Gunakan kata sandi yang berbeda untuk setiap akun.** Gunakan kata sandi yang berbeda untuk setiap akun.
- f. **Perbarui secara berkala.** Gantilah kata sandi kamu secara rutin, terutama jika kamu merasa ada kemungkinan kata sandi kamu sudah bocor.
- g. **Gunakan manajer kata sandi.** Jika kesulitan mengingat banyak kata sandi, pertimbangkan untuk menggunakan manajer kata sandi yang terpercaya untuk dapat menyimpan kata sandi dengan aman.

Setelah membuat kata sandi, kamu dapat mengecek kekuatannya melalui situs :

- <https://www.passwordmonster.com/>
- <https://password.kaspersky.com>
- <https://www.experte.com/password-check>
- <https://www.security.org/how-secure-is-my-password/>
- <https://pwm.trendmicro.com/password-check>

Setelah membuat kata sandi, kamu dapat mengecek kekuatannya melalui salah satu situs, contohnya <https://pwm.trendmicro.com/password-check>. Berikut contoh hasil pengecekannya.



Gambar 4.19 Tampilan Hasil Pengecekan Kekuatan Kata Sandi

Sumber: TREND MICRO Password Manager (2025)

Pada hasil pengecekan tersebut terlihat bahwa dari kata sandi **str4wh4t** yaitu:

1. Terdapat 8 karakter,
2. Tidak terdapat simbol,
3. Tidak terdapat huruf kapital, dan
4. Terdapat angka.

Estimasi peretasan membutuhkan waktu 1 tahun dan tidak ditemukan data/istilah yang berkaitan dengan kata sandi tersebut. Semakin banyak variasi dan jumlah karakter semakin kuat kata sandi. Jadi, coba refleksikan ulang, apakah kata sandi pada akunmu termasuk sangat kuat, kuat, sedang, atau bahkan lemah? Yuk, lindungi akunmu dengan kata sandi yang sulit ditebak orang lain.

Autentikasi Dua Faktor

Agar perlindungan datamu semakin kuat, aktifkan **Autentikasi Dua Faktor (2FA)** pada akun media sosialmu sebagai lapisan keamanan tambahan. Sistem keamanan ini membutuhkan dua langkah verifikasi berbeda, misalnya dengan memasukkan kata sandi dan kode *One-Time Password (OTP)* yang dikirimkan melalui SMS, sehingga lebih aman dari upaya peretasan. Dengan 2FA, meskipun kata sandi diketahui orang lain, akun tetap terlindungi karena peretas juga harus memasukkan kode verifikasi seperti kode OTP atau sidik jari yang terdaftar di *smartphone*. Kode atau data biometrik tersebut hanya dapat diakses oleh pemilik akun yang sah. Berikut cara kerja 2FA untuk mengamankan data kamu.

- a. Masukkan kata sandi atau pin akun.
- b. Setelah kata sandi diterima, kamu akan diminta untuk memasukkan kode verifikasi. Kode ini bisa dikirimkan langsung ke *smartphone*, melalui SMS atau pesan via aplikasi *Whatsapp* yang dihasilkan oleh aplikasi autentikator (seperti *Google Authenticator*). Kode ini juga bisa dikirimkan melalui *email*, bahkan bisa melalui panggilan telepon.
- c. Ketikkan kode verifikasi yang kalian terima di halaman login.

Contohnya, setelah memasukkan kata sandi untuk mengakses akun email di perangkat baru, kamu akan diminta untuk memasukkan kode verifikasi yang dikirimkan ke *smartphone* kamu. Hanya dengan kode tersebut, kamu dapat masuk ke akunmu. Dengan demikian, meskipun seseorang mengetahui kata sandimu, mereka tetap memerlukan kode verifikasi dari *smartphone* kamu untuk mengakses akun tersebut.

Selain 2FA, terdapat juga *Multi-Factor Authentication* (MFA). Keduanya saling berhubungan. 2FA merupakan bagian dari MFA. Artinya, setiap 2FA termasuk MFA, tetapi tidak semua MFA hanya sebatas 2FA. MFA adalah istilah yang lebih luas, yaitu sistem keamanan yang menggunakan lebih dari dua faktor untuk verifikasi identitas. Sementara itu, 2FA hanya menggunakan dua faktor saja. Contoh, penerapan 2FA adalah kombinasi kata sandi dan kode OTP yang dikirim melalui SMS. Jika ditambahkan faktor lain, seperti sidik jari atau token keamanan, maka sistem tersebut disebut MFA, bukan lagi 2FA.

Contoh penerapan *Multi-Factor Authentication* (MFA) dapat dilihat saat login ke akun dompet digital. Pertama, pengguna diminta memverifikasi sidik jari sebagai autentikasi biometrik. Jika berhasil, tahap selanjutnya adalah memasukkan kata sandi. Setelah kata sandi benar, apabila pengguna ingin melakukan perubahan pengaturan akun, aplikasi akan mengirimkan kode OTP sebagai lapisan keamanan tambahan. Berikut manfaat penggunaan keamanan ganda.

- a. Keamanan lebih tinggi karena menambah lapisan perlindungan ekstra.
- b. Mencegah akses tidak sah. Meskipun kata sandi bocor, kode verifikasi tetap akan melindungi akun.
- c. Pengaturan mudah diterapkan.

Perlu diingat, meskipun 2FA meningkatkan keamanan, kamu harus tetap menggunakan kata sandi yang kuat dan unik untuk menjaga informasi pribadi dengan baik.

3. Enkripsi dan Dekripsi

Dalam dunia intelijen, enkripsi digunakan untuk melindungi pesan agar tidak bisa dibaca oleh peretas atau pihak yang tidak berwenang. Prinsip serupa sering kita temui dalam kegiatan kepramukaan saat berlatih menggunakan sandi. Misalnya, ketika seorang agen intelijen ingin mengirimkan informasi rahasia, mereka akan mengubah pesan tersebut supaya pesan hanya bisa dipahami oleh pihak tertentu. Apabila pesan tersebut diretas, peretas hanya bisa melihat tulisan acak yang tidak bisa dibaca.

Proses pengubahan pesan ini disebut **enkripsi**. Pesan acak (teks terenkripsi) hanya bisa dibuka oleh pihak yang memiliki kunci untuk mengembalikannya menjadi informasi awal. Proses pembukaan kode ini disebut **dekripsi**. Enkripsi

dan dekripsi adalah teknik yang digunakan untuk menjaga kerahasiaan informasi, seperti yang dilakukan oleh agen rahasia.

Dengan mempelajari enkripsi dan dekripsi, kamu akan memahami dasar-dasar keamanan informasi serta cara melindungi data penting di dunia digital. Penerapan metode enkripsi dan dekripsi sangat penting diterapkan di era teknologi saat ini. Sudah siapkah kamu untuk memahami, bagaimana agen rahasia digital menerapkan teknik ini? Ayo, kita mulai!

a. Tahapan Enkripsi dan Dekripsi

Dalam implementasinya, terdapat tiga tahapan proses enkripsi dan dekripsi, antara lain:

1) Pengacakan Pesan

Pesan asli biasa disebut dengan *plain text*. Sebelum mengirimkan pesan, agen intelijen menggunakan enkripsi untuk mengubah *plain text*. Misalnya, mereka bisa menggunakan algoritma seperti *Advanced Encryption Standard* (AES) untuk mengubah *plain text* menjadi bentuk kode yang tidak dapat dimengerti tanpa kunci yang biasa disebut *ciphertext*.

2) Pengiriman Pesan

Ciphertext yang telah dienkripsi kemudian dikirim melalui saluran komunikasi seperti radio, *email*, atau sistem komunikasi yang dirasa aman.

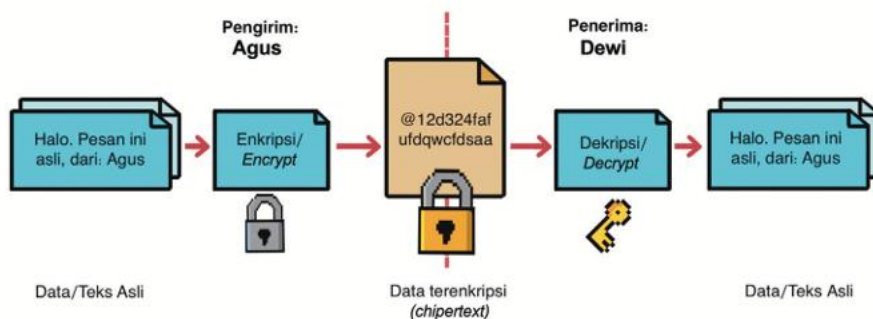
3) Pengembalian ke Bentuk Asli

Setelah *ciphertext* tiba di tujuan, penerima menggunakan kunci dekripsi untuk mengembalikan *ciphertext* tersebut ke bentuk aslinya sehingga mereka dapat membacanya dan mengambil tindakan yang diperlukan.

Contoh penerapan enkripsi dapat dilihat pada ilustrasi di bawah ini (Gambar 4.20). Agus mengirimkan teks kepada temannya, Dewi. Berikut adalah tahapan proses enkripsi dan dekripsi yang terjadi dalam komunikasi antara Agus dan Dewi.

- a) Data asli (*plaintext*) dari Agus dienkripsi menggunakan fungsi tertentu (*cipher*).
- b) Hasil enkripsi menghasilkan *ciphertext*, yaitu teks yang telah terenkripsi.

- c) *Ciphertext* tersebut dikirimkan kepada Dewi melalui jaringan internet/komputer.
- d) Dewi menerima *ciphertext* dan dilakukan proses dekripsi.
- e) Pesan berhasil dikembalikan ke dalam bentuk asli (*plaintext*), sehingga dapat dibaca oleh Dewi.



Gambar 4.20 Ilustrasi Proses Enkripsi dan Dekripsi

b. Kasus-Kasus Terkenal Penerapan Enkripsi

Sejarah mencatat sejumlah kasus terkenal yang menunjukkan peran krusial enkripsi dalam menjaga kerahasiaan informasi. Salah satu contoh yang paling ikonik terjadi pada masa Perang Dunia II, ketika Jerman menggunakan mesin Enigma untuk mengenkripsi pesan-pesan militer mereka. Mesin ini bekerja dengan mengubah setiap huruf dalam pesan asli menjadi kombinasi huruf yang sangat rumit dan acak, sehingga mustahil dibaca oleh pihak musuh yang berhasil menyadapnya tanpa memiliki mesin dan kunci yang tepat. Namun, para ahli kode dari Inggris, termasuk Alan Turing, bekerja keras untuk memecahkan kode Enigma dan membaca pesan-pesan rahasia yang dikirim oleh Jerman. Upaya ini sangat penting dalam memenangkan perang sehingga Enigma berhasil dipecahkan.

c. Caesar's Cipher

Enkripsi bekerja dengan menggunakan algoritma tertentu yang mengubah data asli (*plaintext*) menjadi data terenkripsi (*ciphertext*). Algoritma ini dikenal dengan istilah *Cipher*. Salah satu *Cipher* yang sederhana adalah *Caesar's Cipher*, yang beroperasi dengan cara menggeser setiap huruf dalam alfabet sejumlah langkah tertentu. Algoritma ini menggunakan nama Caesar karena terinspirasi dari Julius Caesar, seorang Kaisar Romawi yang konon

menggunakan algoritma ini untuk melindungi pesan-pesan rahasia militernya saat berkorespondensi. Contoh pergeseran alfabet dengan *Caesar's Cipher* sebanyak sekali ke kanan yaitu huruf A akan setara huruf Z, huruf B setara huruf A, huruf C setara huruf B dan seterusnya. Kamu dapat mencoba simulasi enkripsi ini secara interaktif melalui situs <https://studio.code.org/s/hoc-encryption/lessons/1/levels/1>.



Gambar 4.21 Ilustrasi *Caesar's Cipher*

Sumber: [Code.org](https://code.org) (2025)

Sehingga teks:

- “selamat ulang tahun” akan menjadi “rdkzls tkzmf szgtm”.
- “ini adalah teks biasa” akan menjadi “hnh zczkzg sdjr ahzrz”.

Lalu, bisakah kalian menebak teks asli

- “zjt zczkzg zmzj fdlazkz”? “hmenqlzshjzrlo”?

Mengubah kembali teks yang telah terenkripsi menjadi teks asli disebut **dekripsi**. Proses ini merupakan kebalikan dari enkripsi, yang mengubah pesan rahasia (*ciphertext*) kembali menjadi pesan asli (*plaintext*). *Caesar's Cipher* dengan metode pergeseran alfabet merupakan salah satu teknik enkripsi yang sederhana. Ada metode enkripsi yang lebih sulit daripada sekadar menggeser huruf, misalnya dengan mengganti setiap huruf dengan huruf yang lain secara acak atau berdasarkan aturan yang telah kita tetapkan sendiri. Contoh “aku bisa lebih hebat” dapat diubah menjadi “vaya jafm lebmi iebat”, menggunakan aturan penggantian huruf yang ditetapkan sendiri. Setelah naik ke level 4, kamu akan mempelajari teknik enkripsi acak dan bagaimana melakukan kostumisasi aturan enkripsi.



Gambar 4.22 Ilustrasi Proses Enkripsi *Caesar's Cipher* Secara Acak

Sumber: [Code.org](https://code.org) (2025)

4. Update Software

Update software merupakan proses memperbarui sistem operasi atau aplikasi untuk meningkatkan keamanan dan kinerja. Perangkat digital butuh diperbarui secara berkala untuk meminimalisir celah yang dapat dimanfaatkan oleh peretas. Perusahaan pengembang aplikasi biasanya menemukan kelemahan atau kesalahan (*bug*) pada produknya, lalu memperbaikinya melalui *update software*. Dengan demikian, melakukan *update* sama dengan menutup jalan masuk bagi peretas serta menjaga perangkat tetap aman dan optimal.

Jika kamu tidak melakukan *update*, risiko keamanan akan semakin besar. Peretas bisa masuk melalui celah lama yang belum diperbaiki, lalu mencuri data, atau bahkan merusak perangkat. Misalnya, jika kita tidak pernah *update* aplikasi *WhatsApp*, varian virus baru yang menyusup melalui *file* yang dikirim mungkin tidak terdeteksi oleh sistem keamanan versi lama. Hal ini berbahaya jika pengguna mengunduh pesan tersebut. Demikian pula, jika laptop tidak pernah *update* sistem operasi *Windows*, mungkin ada program jahat yang bisa masuk tanpa disadari. Oleh karena itu, *update software* bukan sekadar menambah fitur baru, namun utamanya untuk melindungi data pribadi dan keamanan data kita.



Aktivitas IX-KD-D01-K



Aktivitas Individu

Membuat Pesan Rahasia

- ✔ Tujuan: Pada aktivitas kali ini, kalian akan merefleksikan pemahaman metode enkripsi dan dekripsi dengan *Caesar's Cipher*.
- ✔ Syarat pengetahuan: *Caesar's Cipher*.

Guru membagi kelas menjadi beberapa kelompok kecil (4–5 orang). Setiap kelompok akan berperan sebagai pengirim pesan rahasia dan pemecah kode.

1. Setiap kelompok diminta membuat sebuah kata atau sekumpulan karakter alfabet yang panjangnya sesuai dengan jumlah anggota kelompok. Panjang kata atau karakter akan ditentukan oleh guru. Misalnya, guru menentukan 10 karakter, kelompok dapat menyusun kata dengan 10 karakter seperti **KRIPTOGRAF**.
2. Karakter tersebut harus dienkripsi oleh kelompok lain menggunakan *Caesar's Cipher* dengan kunci geser yang ditentukan oleh guru, misalnya geser 4 huruf ke kanan.
3. Setiap kelompok berlomba memecahkan kode enkripsi dengan cepat dan menuliskan hasil deskripsinya di papan tulis.
4. Guru dan kelompok lawan akan menilai keakuratan jawaban setiap tim.



Ayo, Refleksi!

Nama :

Kelompok :

Tanggal :

1. Mengapa enkripsi penting dalam kehidupan sehari-hari, terutama di dunia digital?

.....

.....

2. Apa perasaanmu setelah mengikuti aktivitas ini?

- a. Senang
- b. Tertantang
- c. Lebih paham tentang enkripsi
- d. Biasa saja
- e. Lainnya:

Jelaskan alasannya:

.....

.....

3. Bagaimana kamu bisa menerapkan pengetahuan tentang enkripsi atau kode rahasia ini dalam kehidupan sehari-hari?

.....

.....

E. Perilaku dan Kesadaran Penuh (*Mindfulness*) pada Dunia Digital

Cepatnya arus informasi pada saat ini baik melalui media cetak maupun elektronik dapat memberikan dampak yang kompleks, baik secara langsung maupun tidak langsung. Terlebih ketika kita menggunakan perangkat untuk berkomunikasi dengan orang lain secara digital, hal ini dapat memengaruhi kondisi fisik dan psikologis, bahkan pada lingkungan sekitar.

Misalnya, jika kamu menjadi pegiat media sosial seperti *content creator*, mungkin kamu pernah menerima lebih banyak hujatan dibanding pujian, lebih banyak *dislike* dibandingkan *like* atau bahkan mendapatkan ujaran kebencian saat tayangan siaran langsung atau *streaming*. Bahkan bisa saja konten yang kamu buat menjadi viral karena kesalahpahaman orang lain, dalam mencerna isi konten yang kamu sampaikan. Hal-hal tersebut dapat membuat kamu merasa tidak nyaman atau *badmood*, bukan? Kondisi *badmood* ini bisa terbawa ke dunia nyata, seperti saat kamu berbincang dengan keluarga, teman di sekolah atau lingkungan sosial lainnya. Kamu mungkin akan menjadi lebih mudah cemas, curiga atau bahkan *overthinking*. Namun, hal ini dapat diatasi jika kita melatih diri untuk tetap sadar dan tenang, atau *mindful* dalam keseharian terutama sebelum masuk ke dalam dunia digital.



Gambar 4.23 Ilustrasi Kecemasan Pengguna Media Sosial

Mindfulness adalah kemampuan untuk memperhatikan dan memahami apa yang terjadi di dalam diri maupun sekitar kita, dengan penuh kesadaran dan perhatian. Dalam konteks digital, *mindfulness* berarti mampu berpikir jernih dan tidak tergesa-gesa saat berinteraksi dengan teknologi. *Mindfulness* dapat diterapkan dengan mengevaluasi isi konten atau media yang akan dibuat. Misalnya, dengan meminta saran dari orang lain atau mengkaji ulang materi yang ingin diangkat. Dengan begitu, kita bisa memperbaiki kekurangan sebelum membagikan konten kepada orang lain.

Kesadaran penuh merupakan bentuk kesiapan dan kematangan, dalam memanfaatkan teknologi serta media digital secara bertanggung jawab. *Mindfulness* dalam dunia digital berarti menerapkan prinsip kesadaran penuh untuk mengelola dan berinteraksi dengan teknologi secara sadar dan efektif, terutama dalam mengamankan data privat kita yang tersimpan di dalamnya.

1. Perilaku Digital

Di era sekarang, hampir semua orang hidup berdampingan dengan dunia digital. Setiap hari, kita menggunakan internet untuk belajar, bermain, berkomunikasi, belanja, dan mencari hiburan. Segala tindakan kita di dunia digital disebut sebagai perilaku digital. Hal ini mencakup cara kita menggunakan media sosial, membagikan informasi, menonton konten, memberi komentar, *like*, serta berbagai bentuk interaksi lainnya di ruang digital.

Perilaku digital sangat penting untuk dipahami, karena apa yang kita lakukan di dunia digital dapat berdampak besar bagi diri sendiri maupun orang lain. Perilaku yang baik dapat menyebarkan kebaikan, membangun citra positif, bahkan membuka peluang masa depan. Sebaliknya, perilaku yang buruk bisa menyakiti orang lain, merusak reputasi, dan berdampak negatif pada kehidupan nyata.

Oleh karena itu, ayo kita belajar mengenali, menilai, dan mengevaluasi perilaku digital, baik yang kita lakukan sendiri maupun yang dilakukan oleh orang-orang di sekitar kita. Dengan begitu, kita bisa menjadi masyarakat digital yang cerdas, bijak, dan bertanggung jawab.

a. Konten Positif

Konten positif adalah segala bentuk informasi yang kita akses atau bagikan di internet seperti media sosial, chat, video, atau blog. Konten positif dapat memberikan manfaat, menyebarkan kebaikan, pengetahuan, dan nilai yang positif. Berikut beberapa jenis konten positif yang sering kita temui di dunia digital.

1) Konten Edukasi

Konten edukasi adalah jenis konten yang bertujuan untuk memberikan informasi atau pengetahuan kepada orang lain. Konten ini biasanya dibuat untuk membantu proses belajar atau memberi pemahaman tentang suatu topik. Misalnya, video pelajaran Matematika, tips belajar efektif, atau informasi tentang beasiswa dan lomba pelajar. Konten ini sangat bermanfaat karena bisa menambah wawasan siapapun yang melihatnya.

2) Konten Inspiratif

Konten inspiratif adalah konten yang dapat membangkitkan semangat, motivasi, atau membuat orang merasa lebih kuat dalam menghadapi tantangan. Contohnya adalah cerita perjuangan seseorang yang sukses walau berasal dari keluarga sederhana atau kutipan motivasi yang menyentuh hati. Konten seperti ini biasanya dibagikan untuk menyemangati orang lain agar tidak mudah menyerah.

3) Konten Kreatif dan Produktif

Konten kreatif dan produktif adalah konten yang menampilkan hasil karya atau kemampuan seseorang. Misalnya, menggambar digital, membuat puisi, menyanyi, menulis cerpen, atau membuat vlog yang mendidik. Konten seperti ini membuktikan bahwa dunia digital bisa menjadi tempat untuk mengekspresikan bakat dan ide-ide positif, bukan hanya untuk hiburan semata.

4) Konten Sosial dan Empati

Konten sosial dan empati adalah konten yang menunjukkan kepedulian terhadap lingkungan dan sesama. Misalnya, kampanye peduli sampah plastik, penggalangan dana untuk orang yang membutuhkan, atau ajakan



untuk saling menghargai dan membantu. Konten ini bisa menggerakkan orang lain untuk ikut berbuat baik dan lebih peka terhadap kondisi sekitar.

5) Konten Religi dan Moral

Konten ini biasanya berisi nilai-nilai agama atau moral yang mendorong orang untuk berbuat baik dan menghindari hal buruk. Misalnya, kutipan dari kitab suci, video ceramah yang menyejukkan, atau cerita pendek yang mengajarkan kejujuran dan kasih sayang. Konten ini penting, karena bisa memperkuat karakter dan memperbaiki perilaku.

b. Dampak Perilaku Digital

Setelah memahami pentingnya konten positif, kini saatnya kita menyadari bahwa segala aktivitas digital seperti mengunggah, membagikan, dan berkomentar merupakan bagian dari perilaku digital. Perilaku ini bisa berdampak positif atau negatif, tergantung bagaimana kita menggunakannya. Karena itu, mari pelajari dampaknya, cara mengevaluasinya, dan tips agar kita tetap bijak dan bertanggung jawab di dunia digital.

1) Dampak Positif

- a) Menambah wawasan. Internet memudahkan kita mencari informasi dan belajar hal baru.
- b) Menjalin relasi positif. Kita bisa berteman dan berkomunikasi dengan siapa pun.
- c) Menyalurkan kreativitas. *Platform* digital bisa menjadi wadah untuk menunjukkan bakat.
- d) Mendukung karir atau pendidikan. Profil digital yang baik bisa membantu masa depan.

2) Dampak Negatif

- a) *Cyberbullying* (perundungan *online*). Komentar negatif, hinaan, atau serangan verbal di internet.
- b) Membagikan terlalu banyak informasi pribadi yang dapat disalahgunakan.
- c) Kecanduan digital. Terlalu lama menggunakan *smartphone*, dapat membuat waktu belajar dan sosialisasi terganggu.

- d) Penyebaran hoaks. Membagikan informasi tanpa mengecek kebenarannya, dapat merugikan banyak orang karena berpotensi penyebaran hoaks.

c. Cara Mengevaluasi Perilaku Digital

1) Evaluasi Diri Sendiri

- a) Apakah yang saya bagikan bermanfaat bagi orang lain?
- b) Apakah konten saya bisa menyakiti orang lain atau hanya sekedar melampiaskan emosi pribadi?
- c) Apakah saya selalu mengecek fakta sebelum menyebarkan informasi?
- d) Apakah saya menghargai pendapat orang lain yang berbeda pandangan dengan saya?

2) Evaluasi Lingkungan Sekitar

- a) Amati bagaimana temanmu menggunakan media sosial. Apakah mereka sering menyebarkan hoaks, ujaran kebencian, atau menyebarkan semangat positif?
- b) Apakah interaksi di dunia digital membuatmu merasa aman, nyaman, dan dihargai?

3) Tips Meningkatkan Perilaku Digital Positif

- a) Berpikir dahulu sebelum memberikan komentar atau reaksi di dunia digital, khususnya di media sosial.
- b) Selalu verifikasi informasi sebelum membagikannya.
- c) Gunakan media sosial untuk hal-hal produktif seperti belajar, berkarya, mencari inspirasi, atau bergabung dengan komunitas yang positif.
- d) Jaga etika digital dengan menghormati orang lain walaupun berbeda pendapat.
- e) Gunakan teknologi sesuai kebutuhan dan tidak berlebihan.

2. Mindfulness Dunia Digital

Mindfulness merupakan kemampuan dasar manusia untuk hadir sepenuhnya, sadar di mana kita berada, apa yang kita lakukan, dan tidak terlalu reaktif

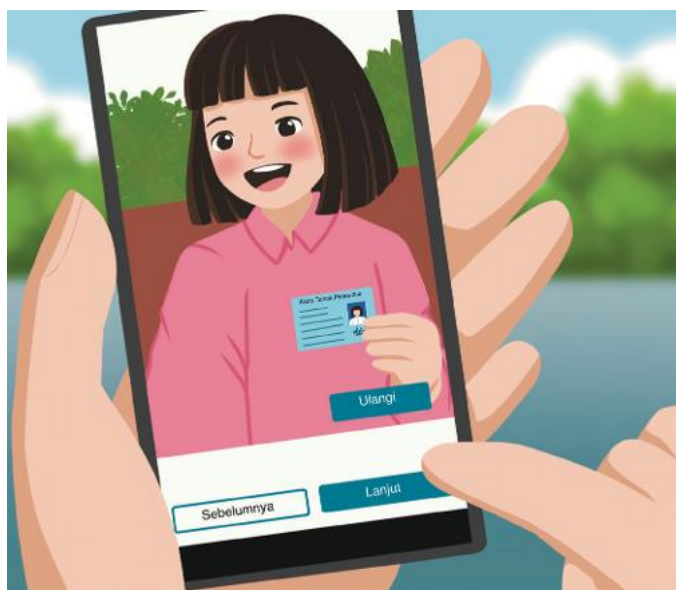
terhadap sesuatu yang terjadi di sekitar kita. *Mindfulness* atau kesadaran penuh juga membantu kita mengelola stres akibat penggunaan teknologi. Dengan menyadari kebiasaan digital sehari-hari, kita dapat lebih bijak dalam menggunakan gawai dan media sosial. Praktik seperti meditasi dan detoks digital, dapat menciptakan ruang untuk melepaskan diri dari tekanan. Dengan *mindfulness* atau kesadaran penuh, kita dapat menjadi lebih sadar dan proaktif dalam melindungi data privat. Pada bab 3, kita telah mempelajari penerapan *mindfulness* dalam berbagai bidang. Kini, pembelajaran difokuskan pada keamanan dunia digital. Berikut adalah beberapa cara yang dapat kita lakukan sebagai bentuk penerapan *mindfulness* di dunia digital.

a. Atur Opsi Privasi pada Pengaturan Akun

Periksa dan sesuaikan pengaturan privasi di setiap *platform* media sosial yang kamu gunakan. Pastikan hanya orang-orang yang kamu kenal atau izinkan yang dapat melihat informasi pribadi kamu. Misalnya, kamu bisa mengatur siapa yang bisa melihat *story* kamu di Instagram, contohnya hanya teman terdekat atau daftar teman tertentu.

b. Batasi Informasi

Kesadaran untuk membatasi informasi sangat penting karena risiko keamanan, terlebih apabila ada informasi privat di sana. Membagikan data sensitif seperti lokasi, alamat rumah, dokumen pribadi, atau kebiasaan sehari-hari dapat membuka celah bagi peretasan, penipuan, hingga eksploitasi data privat oleh pihak yang tidak bertanggung jawab. Kamu dapat menerapkan langkah sederhana seperti mengatur privasi akun dan menghindari berbagi informasi secara berlebihan. Semakin sedikit informasi pribadi yang dibagikan, semakin kecil risiko yang dihadapi.



Gambar 4.24 Ilustrasi Membagikan Data Pribadi

c. Perhatikan Penanda Lokasi yang Dipublikasi

Hindari menandai lokasi kamu secara langsung atau memberikan detail spesifik tentang tempat dimana kamu berada. Hal ini dapat membahayakan keamanan pribadi, terutama jika kamu sering memposting lokasi saat kamu tidak berada di rumah atau lingkungan tertentu. Jika lokasi seperti alamat rumah, sekolah, tempat bermain, tempat kerja, atau rutinitas harian lainnya terlalu sering dipublikasi, akan memudahkan pelaku kejahatan siber menandai aktivitas keseharian kita.



Gambar 4.25 Ilustrasi Penanda Lokasi pada Unggahan di Media Sosial

d. Periksa Kembali Konten yang Akan Dipublikasi

Periksa kembali postingan kamu sebelum dipublikasi. Sesuaikan pengaturan privasi jika diperlukan, seperti pembatasan siapa saja yang bisa melihat postingan kita. Coba periksa apakah ada informasi privat yang tidak seharusnya dibagikan, baik dalam gambar, video, audio, maupun keterangan (*caption*).



Gambar 4.26 Ilustrasi Berpikir Ulang Sebelum Membagikan Konten

e. Selektif Menerima Permintaan Pertemanan

Pastikan kamu menerima permintaan pertemanan hanya dari akun yang kamu kenal. Penipu sering menggunakan akun palsu untuk mengumpulkan informasi pribadi yang mungkin berkaitan dengan data privat bahkan untuk verifikasi akun sehingga membahayakan keamanan data kita.



Gambar 4.27 Ilustrasi Menerima Permintaan Pertemanan

f. Hindari Mengklik Tautan atau Lampiran Mencurigakan

Pernahkah kamu mendapatkan tautan (*link*) mencurigakan dari orang yang tidak kamu kenal? Terkadang peretas mengirimkan tautan untuk mengiming-imingi kita hadiah atau hal menarik lainnya dengan menggunakan kata-kata ajakan yang sangat menggiurkan. Waspadai tautan atau lampiran yang dikirimkan melalui pesan pribadi. Tautan ini bisa jadi upaya penipuan daring atau biasa disebut *phishing*, untuk mencuri informasi pribadi kamu. Dengan mengklik tautan tersebut, tanpa sadar kamu dapat membuka pintu masuk bagi peretas untuk merugikanmu, bahkan orang lain.



Gambar 4.28 Ilustrasi *Phishing*

g. Berhati-Hati dengan Informasi dalam Bio dan Profil

Tampilkan informasi seperlunya saja pada bio atau profil akunmu. Hindari mencantumkan detail pribadi seperti alamat rumah, nama sekolah, atau pekerjaan orang tua. Informasi semacam ini dapat dimanfaatkan oleh pihak

yang tidak bertanggung jawab untuk melakukan penipuan atau tindak kejahatan lainnya.

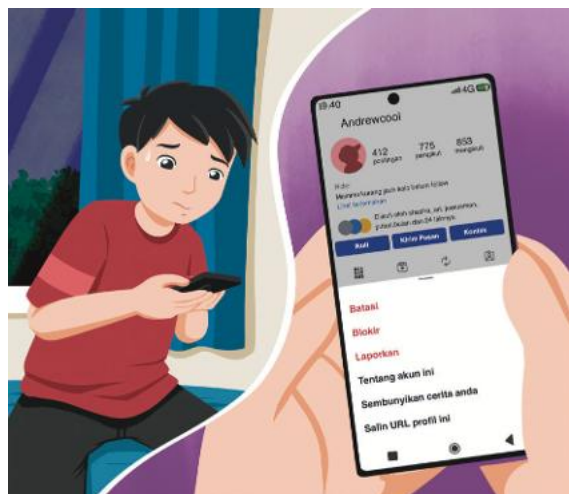
Hapus postingan lama yang tidak relevan dan tidak diperlukan. Periksa kembali unggahan lama, yang berisi informasi pribadi seperti alamat, nomor telepon, atau data pribadi lainnya. Jika sudah tidak relevan, sebaiknya dihapus agar tidak digunakan oleh pihak lain. Misalnya, untuk tujuan penipuan atau pencurian identitas. Dengan menghapus postingan lama, kamu telah mengurangi risiko data pribadi jatuh ke tangan yang salah.

h. Edukasi Diri

Pastikan kita tetap *up-to-date* terhadap informasi terbaru tentang perkembangan keamanan digital. Ketahui potensi ancaman digital terbaru yang banyak dibahas di media digital. Pelajari kasus terbaru tentang dampak, pencegahan, dan penanganan serangan agar dapat lebih mawas diri dalam menjaga data. Biasanya sistem keamanan pada sistem operasi juga akan mendapat pembaruan secara berkala. Oleh karena itu, biasakan untuk selalu memperbarui *security patch* pada perangkat digital kita.

i. Laporkan Aktivitas Mencurigakan

Kita harus selalu waspada terhadap aktivitas mencurigakan di dunia digital. Ancaman pencurian identitas dapat muncul melalui *link* tidak dikenal, pesan mencurigakan, atau permintaan data pribadi yang mengatasnamakan institusi resmi. Kecerobohan sekecil apapun, seperti serangan *phishing* dengan mengklik tautan asing atau membagikan kode OTP, dapat membuka pintu bagi peretas untuk mengakses akun pribadi, mencuri data sensitif, atau kerugian lainnya. Tidak hanya itu, jika menemukan akun palsu yang mencoba menghubungi atau mengaku sebagai diri kita, segera laporkan akun tersebut ke *platform* media sosial



Gambar 4.29 Ilustrasi Melaporkan Akun

terkait. Begitu juga jika menemukan konten mencurigakan, merugikan, atau berbahaya, segera laporkan kepada *platform* atau pihak berwenang untuk penanganan lebih lanjut.



Aktivitas
IX-KD-E01-I

**Aktivitas Individu**

Evaluasi Pengamanan Data Privasi

- ✔ Tujuan: Pada aktivitas kali ini, kamu akan menganalisis media sosial dan beberapa situs berita mengenai perlindungan data.
- ✔ Syarat pengetahuan: *Mindfulness*.

Diskusikan dengan temanmu: bagaimana media sosial mengelola data pribadi.

1. Pilihlah salah satu *platform* media sosial (misalnya WhatsApp, Instagram, TikTok, Telegram, Facebook, atau media sosial lainnya).
2. Tuliskan pengaturan privasi yang kamu terapkan pada media sosialmu.
3. Tuliskan data privasi apa saja yang telah kamu bagikan dan terlihat di profil kamu. Baik berupa postingan, bio, *story* atau yang lainnya, yang dapat dilihat publik, privat, atau bahkan hanya orang tertentu saja.
4. Buatlah rencana atau panduan pribadi untuk menjaga privasi di media sosial. Contohnya: mengelola keamanan dengan autentikasi dua faktor, jenis informasi apa saja yang tidak akan dibagikan, dan rencana lainnya.
5. Presentasikan rencana perlindungan data kamu di depan kelas dan sampaikan tips kepada teman-teman bagaimana cara melindungi data pribadi secara *online*.

3. Pola Hidup Sehat Mendukung Kesehatan Mental

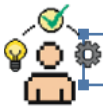
Pernahkah kamu merasa *overthinking*? Biasanya hal ini terjadi karena terlalu banyak informasi yang masuk ke otak, sehingga membuat kita kewalahan. Akibatnya, kita mudah menyerah pada keadaan atau bereaksi berlebihan terhadap suatu hal. Kita pun menjadi sulit membuat keputusan, sulit berpikir jernih, bahkan dapat merasa stres atau lelah secara mental. Salah satu cara untuk mengatasinya adalah dengan melatih *mindfulness*. *Mindfulness* membantu kita berhenti sejenak dari derasnya arus informasi, memberi waktu otak untuk beristirahat dan menolong kita agar bisa lebih tenang dan bijak saat menghadapi informasi yang datang terus-menerus di dunia digital ini. Untuk memulai praktik *mindfulness* dasar, terdapat tiga langkah utama yang dikenal sebagai *Intention*, *Attention*, dan *Attitude* (IAA).

- a. *Intention* (intensi) yaitu niat diri untuk menjadi lebih baik dan lebih bijak.
- b. *Attention* (atensi) adalah melatih kemampuan untuk hadir sepenuhnya, baik secara fisik dan pikiran di momen saat ini. Hal ini melatih kamu untuk fokus menghadapi apa yang sedang kamu kerjakan atau alami, tanpa terganggu oleh pikiran lain yang tidak relevan.
- c. *Attitude* (sikap). Sikap seperti keterbukaan, rasa ingin tahu, dan penerimaan diri akan membantu *mindfulness* menjadi lebih dari sekadar latihan sementara, melainkan gaya hidup yang terus berlanjut.

Apakah kamu pernah jalan-jalan ke tempat wisata yang sedang viral dan ramai dikunjungi? Coba kamu ingat kembali. Biasanya banyak orang sibuk dengan *smartphone* mereka untuk mengambil dokumentasi, lalu membagikannya ke media sosial. Tindakan ini tentu tidak salah dan sangat umum terjadi, karena ada banyak cara untuk menikmati lingkungan sekitar. Namun, jika kamu menerapkan prinsip IAA, kamu tidak hanya fokus untuk mendokumentasikan momen, namun kamu juga bisa benar-benar merasakan segarnya udara, mencium aroma alam, merasa bersyukur atas keindahan ciptaan Tuhan, dan sungguh-sungguh menikmati momen tersebut secara utuh. Dengan melatih diri untuk menerapkan *mindfulness* setiap saat, kita belajar untuk berhenti sejenak, mengenali apa yang kita rasakan dan mengatur respons kita terhadap berbagai situasi. Latihan ini sangat membantu menjaga kesehatan mental dan membuat kita lebih tenang menghadapi dunia digital yang berkembang sangat cepat.

Ayo, Refleksi!

- Tulis pengalaman kamu di buku catatan dengan beberapa pertanyaan refleksi berikut ini.
- 1. Aktivitas apa yang sering membuat kamu menikmati momen secara *mindfulness*? Kamu boleh menyebutkan hobi, aktivitas, atau hal tertentu.
- 2. Apa saja cara atau langkah-langkah terbaik untuk menikmati momen dan hadir sepenuhnya pada situasi tersebut?
- 3. Apakah langkah-langkah pada soal nomor 2 dapat diterapkan pada aktivitas lainnya?
- 4. Menurut kamu, apakah *mindfulness* dapat membuat kesehatan mental kita semakin baik? Jelaskan alasannya.



Uji Kompetensi

1. Tentukan apakah jenis informasi berikut termasuk informasi privat atau publik.

Tabel 4.3 Asesmen Informasi Privat atau Publik

No	Data	Jenis Informasi	Potensi Ancaman Keamanan Digital
1.	Nomor Induk Siswa Nasional (NISN)		
2.	Nomor HP orang tua		
3.	ID media sosial		
4.	Nomor ijazah		
5.	Hobi		
6.	Nilai rapor		
7.	Foto		

No	Data	Jenis Informasi	Potensi Ancaman Keamanan Digital
8.	Nomor Induk Kependudukan (NIK)		
9.	Nama lengkap orang tua		
10.	Kode atau rumus sidik jari		

2. Cocokkan dan hubungkan dengan garis antara jenis *malware* dengan jenis gangguannya berikut ini.

KUDA TROYA	Replikasi <i>file</i> /data dengan cepat
SPYWARE	Mengkripsi <i>file</i> dan meminta tebusan untuk kunci membuka enkripsi
RANSOMWARE	Masuk dan mengontrol <i>file</i> /sistem secara diam-diam
WORM	Mengubah <i>file</i> bahkan sistem setelah ada interaksi pengguna dengan <i>file</i> yang terinfeksi
BACKDOOR	Menyusup ke <i>file</i> dengan aman, saat di waktu tertentu akan menyerang <i>file</i> /data
VIRUS	Mencuri informasi untuk kepentingan tertentu

- Keamanan jaringan merupakan hal yang sangat penting, terutama pada konektivitas jaringan publik. Apabila kamu sedang berada di kafe atau tempat umum yang menyediakan layanan Wi-Fi gratis tanpa kata sandi, berikan 3 saran yang dapat kamu lakukan agar tetap aman saat mengakses internet melalui Wi-Fi publik, sehingga data pribadimu tidak terancam.
- Enkripsikan kalimat berikut dengan ketentuan *cipher* sesuai pada tabel.

Tabel 4.4 Asesmen *Caesar's Cipher*

Teks Asli (<i>Plaintext</i>)	<i>Cipher</i>	Hasil Enkripsi
CINTA INDONESIA	Geser alfabet dari A ke B, B ke C, C ke D, dst.... (geser satu langkah ke kiri)
TEKNOLOGI SIBER	Geser alfabet dari A ke D, B ke E, C ke F, dst... (geser tiga langkah ke kiri)
BIJAK BERMEDIA SOSIAL	Geser alfabet dari A ke B, B ke C, C ke D, dst.... (geser satu langkah ke kanan)
SIAP UPDATE TEKNOLOGI	Geser alfabet dari A ke D, B ke E, C ke F, dst... (geser tiga langkah ke kanan)	

- Berikan tanda centang (✓) pada kolom “Benar” atau “Salah” untuk setiap pernyataan berikut.

Tabel 4.5 Asesmen Persoalan *Mindfulness*

No	Persoalan	Benar	Salah
1.	Menggunakan media sosial selama berjam-jam tanpa tujuan jelas merupakan salah satu contoh penggunaan teknologi secara <i>mindful</i> .		
2.	<i>Mindfulness</i> berarti kesadaran penuh terhadap apa yang sedang kita lakukan, termasuk saat menggunakan teknologi.		

No	Persoalan	Benar	Salah
3.	Notifikasi yang terus muncul tidak mengganggu fokus dan tidak perlu dikendalikan.		
4.	Mengatur waktu penggunaan <i>smartphone</i> setiap hari dapat membantu kita lebih bijak menggunakan teknologi.		
5.	<i>Multitasking</i> dengan banyak aplikasi sekaligus, justru membuat kita lebih produktif dan tenang.		
6.	Merasa cemas jika tidak membuka media sosial dalam waktu singkat, bisa menjadi tanda kecanduan teknologi.		
7.	Teknologi tidak ada hubungannya dengan kesehatan mental, jadi tidak perlu diatur penggunaannya.		
8.	Salah satu cara menerapkan <i>mindfulness</i> adalah dengan mengambil jeda sejenak dari layar gawai saat merasa lelah.		



Refleksi

Selamat! Kamu telah berhasil menyelesaikan bab ini. Ayo, kita periksa apa saja yang sudah kamu pelajari. Salinlah tabel berikut ke dalam buku tugasmu, lalu isilah sesuai dengan pengalaman yang kamu rasakan.

Tabel 4.6 Refleksi Pemahaman Belajar Bab IV Keamanan Digital

No	Materi yang Saya Pahami di Bab IV	Materi yang Belum Saya Pahami di Bab IV	Usaha Saya untuk Memahami Materi yang Belum Saya Pahami di Bab IV



Pengayaan

Kunjungi tautan <https://bukupusbuk.id/s/if9p4> atau pindai kode QR di samping!





Glosarium

- Algoritma (*Algorithm*)** : langkah-langkah dari proses untuk mencapai tujuan tertentu
- antivirus** : program yang mendeteksi, mencegah, dan menghapus virus pada komputer, serta memeriksa *file* yang masuk agar bebas dari virus
- backdoor*** : celah tersembunyi dalam sistem komputer atau perangkat lunak yang memungkinkan pihak tidak sah mengakses sistem tanpa diketahui pemiliknya
- Bahasa Pemrograman (*Programming Language*)** : bahasa buatan yang digunakan untuk menuliskan algoritma atau program komputer dengan aturan sintaksis dan semantik yang ketat, sehingga dapat dipahami dan dijalankan oleh komputer
- badmood*** : perasaan kesal atau suasana hati yang buruk, biasanya terjadi karena suatu hal yang mengganggu atau mengecewakan
- Berpikir komputasional (*Computational thinking*)** : kemampuan untuk memformulasikan masalah dan merancang solusi dalam bentuk langkah-langkah komputasional atau algoritma yang dapat dieksekusi komputer. Berpikir komputasional mencakup pemahaman kemampuan komputer, perumusan masalah, serta perancangan algoritma, dan dapat diterapkan tidak hanya pada bidang informatika, tetapi juga pada sains, teknologi, rekayasa, matematika (STEM), seni, dan ilmu sosial
- blog** : jurnal daring yang dikelola di web oleh individu atau kelompok, berisi tulisan dengan topik yang beragam sesuai kewenangan penulis, serta dapat diakses publik
- blogger** : seseorang yang menulis untuk dan mengelola blog
- brute force attack*** : metode untuk menebak kata sandi atau kunci enkripsi dengan mencoba semua kemungkinan kombinasi hingga menemukan yang benar



<i>caption</i>	: teks singkat yang menjelaskan atau memberi konteks pada gambar, video, atau unggahan di media sosial
<i>central processing unit (CPU)</i>	: peralatan dalam komputer yang mengeksekusi instruksi
Caesar's cipher	: teknik enkripsi klasik yang menggeser setiap huruf dalam pesan sejumlah langkah tertentu di alfabet
<i>content creator</i>	: seseorang yang membuat dan membagikan konten digital, seperti video, tulisan, atau gambar, biasanya melalui <i>platform</i> media sosial
<i>credential stuffing</i>	: serangan yang memanfaatkan kebiasaan orang memakai kata sandi sama di banyak akun, dengan mencoba kredensial bocor dari satu layanan ke layanan lain
<i>cyberbullying</i>	: perundungan atau tindakan menyakiti orang lain melalui media digital, seperti menghina, mengancam, memfitnah atau menyebarkan informasi palsu secara <i>online</i>
data	: informasi yang dikumpulkan dalam bentuk digital maupun non-digital, seperti angka, teks, gambar, suara, atau video, untuk dijadikan referensi atau analisis
<i>decode</i>	: proses mengubah pesan terenkripsi (<i>ciphertext</i>) kembali menjadi bentuk aslinya (<i>plaintext</i>) agar dapat dipahami
Dekomposisi (<i>Decomposition</i>)	: proses memecah suatu masalah atau sistem menjadi beberapa komponen yang lebih sederhana
<i>dictionary attack</i>	: serangan yang menggunakan daftar kata (<i>dictionary</i>) berisi kemungkinan kata sandi umum untuk menebak kata sandi akun korban
<i>dislike</i>	: tindakan atau tombol pada <i>platform</i> digital untuk menunjukkan bahwa seseorang tidak menyukai suatu konten
<i>distributed denial of service (DDOS)</i>	: serangan siber yang membuat layanan atau server tidak dapat diakses dengan membanjirinya permintaan palsu dari banyak perangkat secara bersamaan
<i>email</i>	: sistem pengiriman pesan digital melalui jaringan komputer, terutama internet

<i>encode</i>	: proses mengubah informasi ke dalam format tertentu, sering kali untuk tujuan penyimpanan atau pengiriman
enkripsi	: proses mengubah data agar tidak dapat dibaca tanpa kunci tertentu, untuk menjaga keamanan informasi
<i>file</i>	: dokumen digital yang disimpan di komputer atau perangkat lain, berisi data seperti teks, gambar, video, atau audio
<i>firewall</i>	: sistem keamanan digital yang melindungi perangkat atau jaringan dari akses yang tidak diinginkan
Fungsi (<i>Function</i>)	: unit program yang menerima input, melakukan perhitungan, lalu menghasilkan nilai keluaran, baik berupa fungsi standar seperti $\sin(x)$, $\cos(x)$, $\exp(x)$ atau fungsi yang didefinisikan pengguna
Google Authenticator	: aplikasi yang menghasilkan kode verifikasi sekali pakai untuk login dua faktor (2FA), menambah lapisan keamanan saat masuk ke akun digital
<i>graph</i>	: cara menyusun data dalam bentuk titik-titik (disebut simpul atau <i>node</i>) yang saling terhubung oleh garis (disebut sisi atau <i>edge</i>)
<i>hacker</i>	: seseorang/kelompok yang meretas dengan cara mengakses atau memodifikasi sistem komputer dengan cara ilegal
internet	: jaringan komputer global yang koneksinya menggunakan protokol bersama (dalam hal struktur dan bahasa untuk permintaan <i>file</i> antara klien dan server) untuk berkomunikasi
Kecerdasan Artifisial (<i>Artificial Intelligence</i>)	: kemampuan sistem komputer atau mesin untuk meniru kecerdasan manusia, seperti berpikir, belajar, mengambil keputusan, dan memecahkan masalah
Kecerdasan Artifisial Generatif (<i>Generative Artificial Intelligence</i>)	: teknologi kecerdasan artifisial yang mampu menghasilkan sesuatu yang baru, seperti teks, gambar, suara, video, atau kode program, berdasarkan data yang telah dipelajarinya



Keluaran (output)	: informasi yang dihasilkan dan dikirim dari perangkat komputasi, misalnya tampilan pada layar monitor atau hasil cetak dokumen
keylogger attack	: program berbahaya yang merekam aktivitas setiap tombol yang ditekan pada <i>keyboard</i> korban
Kode (code)	: kumpulan instruksi yang ditulis dalam bahasa pemrograman
Koding (coding)	: aksi untuk menulis program komputer dengan menggunakan bahasa pemrograman
Komputasi (computation)	: setiap aktivitas berorientasi tujuan yang membutuhkan, memanfaatkan, atau menciptakan proses algoritmik
Komputasional (computational)	: pendekatan atau metode yang berhubungan dengan komputer
Komputer (computer)	: mesin atau perangkat yang menjalankan proses, kalkulasi, dan operasi berdasarkan instruksi yang diberikan oleh program perangkat lunak atau perangkat keras
like	: tindakan atau tombol di media sosial untuk menyatakan bahwa seseorang menyukai konten yang dibagikan
literasi digital	: kemampuan untuk menggunakan, memahami, mengevaluasi, dan menciptakan informasi melalui media digital, alat komunikasi, atau jaringan secara bijak
malware	: program berbahaya yang dapat mengganggu, merusak sistem komputer, bahkan mencuri data
man-in-the-middle attack (MitM)	: serangan di mana penyerang menyusup ke jalur komunikasi antara dua pihak untuk menyadap, mengubah, atau mencuri data
Masukan (input)	: sinyal, data, atau instruksi yang dikirim ke komputer melalui perangkat masukan, seperti keyboard, mouse, mikrofon, <i>touchpad</i> , layar sentuh, atau sensor

memori (<i>memory</i>)	: ruang penyimpanan fisik dalam perangkat komputasi untuk menyimpan data dan instruksi pemrosesan, meliputi RAM (<i>Random Access Memory</i>), ROM (<i>Read Only Memory</i>), dan penyimpanan sekunder seperti <i>hard drive</i> , <i>removable drive</i> , dan <i>cloud storage</i>
mesin pencari kecerdasan artifisial	: sistem pencarian informasi yang menggunakan teknologi kecerdasan artifisial, seperti pemrosesan bahasa alami (NLP), pembelajaran mesin (<i>machine learning</i>), dan pembelajaran mendalam (<i>deep learning</i>) untuk memahami maksud dan konteks pencarian pengguna, serta memberikan jawaban yang lebih relevan, cepat, dan cerdas
mesin pencari tradisional	: sistem perangkat lunak yang menampilkan informasi dari internet berdasarkan kata kunci dengan metode pencocokan kata dan algoritma pencarian berbasis indeks, seperti Google, Bing, atau Yahoo! pada masa awal
<i>mindfulness</i>	: kemampuan dasar manusia untuk hadir sepenuhnya, sadar di mana kita berada dan apa yang kita lakukan, dan tidak terlalu reaktif oleh apa yang terjadi di sekitar kita
Model bahasa besar (<i>large language model</i>)	: model kecerdasan artifisial yang dilatih dengan jumlah teks yang sangat besar (misalnya dari buku, artikel, dan internet) sehingga bisa mengerti, menjawab, dan menulis dalam bahasa manusia secara cerdas
<i>multi-factor authentication</i> (MFA)	: sistem keamanan yang membutuhkan lebih dari satu lapisan verifikasi, misalnya kata sandi ditambah kode OTP atau sidik jari
<i>multitasking</i>	: melakukan beberapa kegiatan secara bersamaan dalam satu waktu
<i>one time password</i> (OTP)	: kode unik yang hanya bisa digunakan sekali untuk login atau verifikasi identitas, biasanya dikirim lewat SMS atau aplikasi otentikasi
<i>overthinking</i>	: kebiasaan berpikir berlebihan tentang suatu hal, sering kali menyebabkan kecemasan atau keraguan yang tidak perlu



deep learning	: metode kecerdasan artifisial yang menggunakan jaringan syaraf buatan untuk belajar dari data secara otomatis dan mendalam, sehingga komputer dapat mengenali wajah, memahami suara, menerjemahkan bahasa, atau membuat gambar
Pembelajaran mesin (machine learning)	: cabang dari kecerdasan artifisial <i>artificial intelligence</i> (AI) yang memungkinkan komputer untuk belajar dari data dan pengalaman, lalu mengambil keputusan atau membuat prediksi tanpa harus diprogram secara langsung untuk setiap tugasnya
Pemrograman tekstual (text-based programming)	: metode penulisan program dengan sintaks dan struktur bahasa pemrograman dalam bentuk teks, di mana instruksi ditulis manual menggunakan keyboard sesuai aturan bahasa pemrograman tertentu seperti Python, Java, C++, atau JavaScript
Pemrograman visual blok (block-based visual programming)	: metode pemrograman dengan antarmuka grafis berupa blok-blok berwarna yang disusun secara <i>drag-and-drop</i> untuk membentuk logika atau instruksi program tanpa menuliskan kode teks langsung
Pemrosesan bahasa alami (natural language processing)	: teknologi kecerdasan artifisial yang membuat komputer bisa memahami, membaca, menulis, dan berbicara dalam bahasa manusia, seperti bahasa Indonesia atau bahasa Inggris
Perangkat keras (hardware)	: komponen fisik yang menyusun sistem komputasi, seperti komputer atau perangkat komputasi, yang berbeda dengan perangkat lunak (<i>software</i>)
Perangkat lunak (software)	: program yang dijalankan pada sistem komputasi, komputer, atau perangkat komputasi lainnya, yang berbeda dengan perangkat keras (<i>hardware</i>)
phishing	: penipuan digital yang mencoba mencuri data pribadi dengan menyamar sebagai pihak terpercaya
plaintext	: teks biasa atau informasi asli yang belum dienkripsi, sehingga bisa dibaca langsung oleh siapa saja
platform	: tempat atau layanan digital agar pengguna bisa berinteraksi dan berbagi di dalamnya

<i>plugged</i>	: kegiatan pembelajaran informatika atau pemrograman yang menggunakan perangkat digital, seperti komputer, laptop, tablet, atau <i>smartphone</i> , dengan dukungan aplikasi atau <i>platform</i> tertentu (Scratch, Blockly, Python, Code.org)
program; memprogram; pemrograman (<i>programming</i>)	: program (kata benda): sekumpulan instruksi yang dijalankan komputer untuk mencapai tujuan tertentu; memprogram (kata kerja): untuk menghasilkan program komputer; pemrograman: proses menganalisis masalah dan merancang, menulis, menguji, dan memelihara program untuk menyelesaikan masalah
<i>prompt</i>	: perintah, instruksi, atau pertanyaan yang di berikan kepada komputer (biasanya ke <i>chatbot</i> seperti ChatGPT atau asisten AI) agar komputer bisa memberikan jawaban atau melakukan sesuatu
Prosedur (<i>Procedure</i>)	: bagian dari program yang menjalankan serangkaian operasi pada data melalui parameter, dapat dipanggil dari mana saja dalam program dengan nilai parameter yang berbeda
<i>ransomware</i>	: jenis <i>malware</i> yang mengunci atau mengenkripsi data korban, lalu meminta tebusan agar data bisa diakses kembali
<i>security patch</i>	: pembaruan perangkat lunak untuk memperbaiki celah keamanan atau <i>bug</i> pada sistem
<i>slide presentasi</i>	: halaman visual yang digunakan untuk menyampaikan informasi dalam <i>file</i> presentasi
<i>spyware</i>	: program jahat yang diam-diam memantau dan mencuri data pengguna tanpa sepengetahuan mereka
<i>sticky note</i>	: selembar kertas kecil yang memiliki lapisan perekat di salah satu sisinya, sehingga dapat ditempelkan sementara pada berbagai permukaan
<i>streaming</i>	: proses menonton atau mendengarkan konten digital (seperti video atau audio) secara langsung melalui internet



struktur data	: cara menyimpan dan mengatur data agar bisa digunakan dengan lebih efisien dan mudah dipahami
struktur data <i>tree</i>	: cara menyusun data seperti bentuk pohon terbalik, di mana ada satu data utama di atas, lalu di bawahnya ada cabang-cabang data lain
<i>tethering</i>	: menghubungkan satu perangkat ke internet menggunakan koneksi data dari perangkat lain
trojan horse	: <i>malware</i> yang menyamar sebagai program atau <i>file</i> yang sah, tetapi ketika dijalankan memberikan akses ke penyerang
<i>unplugged</i>	: kegiatan pembelajaran informatika atau pemrograman tanpa perangkat digital, menggunakan alat bantu manual seperti kertas, kartu, papan tulis, atau permainan fisik untuk memahami konsep komputasi secara menyenangkan dan konkret
virus	: program berbahaya yang menempel pada <i>file</i> atau program lain, menyebar saat <i>file</i> dibuka, dan dapat merusak sistem atau mencuri data
<i>virtual private network</i> (VPN)	: teknologi yang membuat koneksi internet aman dengan mengenkripsi data dan menyembunyikan alamat IP pengguna
visualisasi data	: cara menyajikan data dalam bentuk gambar atau grafik, supaya lebih mudah dibaca, dimengerti, dan dianalisis
vlog	: singkatan dari video blog atau video log, yaitu jenis konten digital berupa rekaman video yang menampilkan pengalaman, pendapat, aktivitas, atau informasi tertentu yang dibagikan secara <i>online</i> , biasanya melalui <i>platform</i> seperti YouTube, TikTok, atau media sosial lainnya
<i>website</i>	: sekumpulan halaman digital yang saling terhubung dan dapat diakses melalui internet dengan menggunakan alamat tertentu yang disebut URL (<i>Uniform Resource Locator</i>), berisi teks, gambar, video, dan elemen interaktif yang ditampilkan oleh aplikasi peramban
<i>worm</i>	: <i>malware</i> yang menyebar dengan mereplikasi <i>file</i> untuk memperlambat dan memenuhi penyimpanan pada komputer




Daftar Pustaka

- Adinda, R. *Mindfulness: Mengenal Lebih Dalam tentang Mindfulness*. Gramedia Blog, 2021. Diakses pada 6 Juli 2025. <https://www.gramedia.com/best-seller/mindfulness/?srsltid=AfmBOorLsj6cKDIk44HNHACO5uLjxhltTdjoklx5cwCuGgDGVnUbTqgH>
- Al-Khalifa, Hend S. "An Accessible Multimedia Content Model for E-Learning Systems." *IEEE Multidisciplinary Engineering Education Magazine* 3, no. 3 (2008): 1–6. <https://doi.org/10.1109/MEEM.2008.4579844>
- Al-Taani, Atif T. "An Algorithm for Data Processing and Decision Making." In *2010 2nd International Conference on Computer Engineering and Technology*, vol. 4, V4-425–28. IEEE, 2010. <https://doi.org/10.1109/ICCET.2010.5486194>
- Alzahrani, S. M., N. Salim, and Ajith Abraham. "Understanding Semantic Web and Search Engines Based on Artificial Intelligence Approaches." In *IEEE International Conference on Advanced Computer Science Applications and Technologies (ACSAT)*, 383–87. 2012. <https://doi.org/10.1109/ACSAT.2012.80>
- Armoni, Michal. "On Teaching Abstraction to Computer Science Novices." *Journal of Computers in Mathematics and Science Teaching* 32, no. 3 (2013): 265–84. (Reprinted in IEEE).
- Beekman, George, and Michael Quinn. *Tomorrow's Technology and You: Fundamentals*. 10th ed. Boston, MA: Pearson, 2014.
- Brennan, Karen, and Mitchel Resnick. "New Frameworks for Studying and Assessing the Development of Computational Thinking." In *Proceedings of the 2012 Annual Meeting of the American Educational Research Association (AERA)*. 2012.
- Brin, Sergey, and Lawrence Page. "The Anatomy of a Large-Scale Hypertextual Web Search Engine." *Computer Networks and ISDN Systems* 30, no. 1–7 (1998): 107–17. [https://doi.org/10.1016/S0169-7552\(98\)00110-X](https://doi.org/10.1016/S0169-7552(98)00110-X)
- Castaño, C., J. J. Maquilón, and J. M. Romero. "Use of Flowcharts and Pseudocode to Improve Programming Logic in Secondary Education." In *2020 IEEE Global Engineering Education Conference (EDUCON)*, 1945–50. 2020. <https://doi.org/10.1109/EDUCON45650.2020.9125344>
- Ceruzzi, Paul E. *Computing: A Concise History*. Cambridge, MA: MIT Press, 2012.
- Fadzli, F. H., and M. S. S. Ahamad. "Enhancing Students' Understanding in Learning Programming Using Visual Aids: Flowchart and Pseudocode." In *2016 IEEE 8th International Conference on Engineering Education (ICEED)*, 135–39. 2016. <https://doi.org/10.1109/ICEED.2016.7856066>
- FastNetMon. "DeepSeek DDoS Attacks Explained—What Really Happened?" 5 Februari 2025. Diakses pada 1 September 2025. <https://fastnetmon.com/2025/02/05/deepseek-ddos-attacks-explained-what-really-happened>
- Fleming, W. *Digital Content Creation: A Guide for Students*. New York: Routledge, 2014.
- Gaddis, Tony. *Starting Out with Programming Logic and Design*. 5th ed. Boston, MA: Pearson, 2018.

- Grover, Shuchi, and Roy Pea. "Computational Thinking in K–12: A Review of the State of the Field." *Educational Researcher* 42, no. 1 (2013): 38–43. <https://doi.org/10.3102/0013189X12463051>
- Gulli, Antonio, and Alessio Signorini. "The Indexable Web Is More than 11.5 Billion Pages." *IEEE Internet Computing* 9, no. 1 (2005): 70–73. <https://doi.org/10.1109/MIC.2005.14>
- Gunter, Glenda A., Robert E. Gunter, and Gary B. Shelly. *Teachers Discovering Computers: Integrating Technology in a Connected World*. 7th ed. Boston, MA: Cengage Learning, 2011.
- Hasan, Md. Kamrul, and Md. Shahinur Rahman. "A Beginner's Approach to Understanding Data Processing in Computing Systems." In *2018 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, 964–68. 2018. <https://doi.org/10.1109/TALE.2018.8615372>
- Ilham, Khairul. "Apa Itu Backdoor? Pengertian dan Cara Menghindarinya, Kamu Wajib Tahu!" *Narasi*, 22 Desember 2022. Diakses pada 27 Juli 2025. <https://narasi.tv/read/narasi-daily/apa-itu-backdoor-pengertian-dan-cara-menghindarinya-kamu-wajib-tahu>
- Kalelioglu, Filiz, Yasemin Gülbahar, and Volkan Kukul. "A Framework for Computational Thinking Based on a Systematic Research Review." *Baltic Journal of Modern Computing* 4, no. 3 (2016): 583–96.
- Kaspersky. "Man-in-the-Middle Attack." *Kaspersky IT Encyclopedia*. Diakses pada 1 September 2025. <https://encyclopedia.kaspersky.com/glossary/man-in-the-middle-attack/>
- Khine, Myint Swe, and Darrell L. Fisher. "Infographics: Making Information Visible and Engaging." In *IEEE International Conference on Advanced Learning Technologies*, 105–9. 2003. <https://doi.org/10.1109/ICALT.2003.1215069>
- Koret, Joxean, and Elias Bachaalany. *The Antivirus Hacker's Handbook*. Indianapolis: John Wiley & Sons, 2015.
- Kosinski, Matthew. "What Is Phishing?" IBM. 2024. Diakses pada 1 September 2025. <https://www.ibm.com/think/topics/phishing>
- Kritzinger, Elmarie, and S. H. von Solms. "Cybersecurity for Home Users: A New Way of Protecting against Cyber Threats." *Computer Fraud & Security* 2010, no. 11 (2010): 12–17. [https://doi.org/10.1016/S1361-3723\(10\)70112-5](https://doi.org/10.1016/S1361-3723(10)70112-5)
- Lambert, Kenneth A. *Fundamentals of Python: First Programs*. Boston, MA: Cengage Learning, 2012.
- Lankow, Jason, Josh Ritchie, and Ross Crooks. *Infographics: The Power of Visual Storytelling*. Hoboken, NJ: Wiley, 2012.
- Lee, Irene. "Reclaiming the Roots of CT." *CSTA Voice: The Voice of K–12 Computer Science Education and Its Educators* 12, no. 1 (2016): 3–4.
- Lye, S. Y., and J. H. L. Koh. "Review on Teaching and Learning of Computational Thinking through Programming: What Is Next for K-12?" *IEEE Transactions on Education* 57, no. 4 (2014): 279–83. <https://doi.org/10.1109/TE.2014.2318134>

- Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge: Cambridge University Press, 2008.
- Moreno-León, J., G. Robles, and M. Román-González. “Dr. Scratch: A Web Tool to Automatically Evaluate Scratch Projects.” In *Proceedings of the Workshop in Primary and Secondary Computing Education (WiPSCE)*, 132–33. 2015. <https://doi.org/10.1145/2818314.2818338>
- Russell, Stuart J., and Peter Norvig. *Artificial Intelligence: A Modern Approach*. 4th ed. Hoboken, NJ: Pearson, 2021.
- Schneider, David I., and Judith L. Gersting. *An Invitation to Computer Science*. 8th ed. Cengage Learning, 2019.
- Shailaja, K., and C. Kavitha. “Enhancing Computational Thinking in Middle School Using Block-Based Programming Environments.” In *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 1–5. 2019. <https://doi.org/10.1109/ICECCT.2019.8869366>
- Shelly, Gary B., and Misty E. Vermaat. *Discovering Computers: Fundamentals*. 1st ed. Cengage Learning, 2011.
- Sholekhah, Arifatus. “‘Mindfulness’ di Era Digital: Bagaimana Melatih Otak agar Bisa Tetap Fokus di Dunia yang Serba Cepat?” *The Conversation*, 12 Oktober 2024. Diakses pada 7 Juli 2025. <https://theconversation.com/mindfulness-di-era-digital-bagaimana-melatih-otak-agar-bisa-tetap-fokus-di-dunia-yang-serba-cepat-239395>
- Skoudis, Ed, and Lenny Zeltser. *Malware: Fighting Malicious Code*. New Jersey: Prentice Hall, 2003.
- Smith, Grant. *Everything You Need to Ace Computer Science in One Big Fat Notebook: Science and Coding*. New York: Workman Publishing, 2020.
- Tang, X., Yin, Y., Lin, Q., Hadad, R., & Zhai, X. “Assessing Computational Thinking in K-12: A Systematic Review of Empirical Studies.” *Computers & Education*, 148 (2020): 103798. <https://doi.org/10.1016/j.compedu.2019.103798>
- Tenny, Tjia Markus. “Mindfulness: Menemukan Ketenangan di Tengah Arus Digital.” *Radio Republik Indonesia (RRI)*, 3 November 2024. Diakses pada 6 Juli 2025. <https://www.rri.co.id/lain-lain/1093395/mindfulness-menemukan-ketenangan-di-tengah-arus-digital>
- Weintrop, David, and Uri Wilensky. “Comparing Block-Based and Text-Based Programming in High School Computer Science Classrooms.” *ACM Transactions on Computing Education (TOCE)* 18, no. 1 (2017): 3:1–3:25. <https://doi.org/10.1145/3089799>
- Wibowo, Agus. *Keamanan Sistem Jaringan Komputer*. Semarang: Universitas STEKOM, 2021.
- Wong, W. Y., and C. C. Fung. “Teaching Flowchart and Pseudocode in Problem-Solving.” In *Proceedings of the 2004 IEEE International Conference on Information Technology: Coding and Computing (ITCC)*, vol. 2, 787–93. 2004. <https://doi.org/10.1109/ITCC.2004.1286764>
- Yang, J., and H. Zhang. “Design and Implementation of Digital Media Curriculum for Middle School Students.” In *2016 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, 473–77. 2016. <https://doi.org/10.1109/TALE.2016.7851824>
- Zhong, B., Q. Wang, and J. Chen. “The Impact of a Flipped Classroom on Mathematics Concept Learning in a Middle School.” *Educational Technology & Society* 19, no. 3 (2016): 134–42.



Daftar Sumber Gambar

- Gambar 1.1** diunduh dari <https://transjakarta.co.id/rute> pada 22 Juni 2025 pukul 13.00
- Gambar 1.3** diunduh dari <https://www.bbc.co.uk/bitesize/guides/zp92mp3/revision/1> pada 22 Juni 2025 pukul 13.00
- Gambar 1.16** diunduh dari <https://www.bps.go.id/id/infographic> pada 22 Juni 2025 pukul 13.00
- Gambar 1.17** diunduh dari <https://www.bps.go.id/id/infographic> pada 22 Juni 2025 pukul 13.00
- Gambar 1.18** diunduh dari <https://www.bps.go.id/id/infographic> pada 22 Juni 2025 pukul 13.00
- Gambar 1.19** diunduh dari <https://www.bps.go.id/id/infographic> pada 22 Juni 2025 pukul 13.00
- Gambar 3.8** hasil pindai dari buku Informatika Kelas IX, Hanson Prihantoro Putro, dkk., Jakarta : Pusat Perbukuan, Badan Standar, Kurikulum, dan Asesmen Pendidikan Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi, 2022.
- Gambar 3.9** diunduh dari <https://fileinfo.com/extension/docx> pada 22 Juni 2025 pukul 13:00 WIB
- Gambar 3.10** diunduh dari <https://fileinfo.com/extension/xlsx> pada 22 Juni 2025 pukul 13:30 WIB
- Gambar 3.11** diunduh dari <https://fileinfo.com/extension/pptx> pada 22 Juni 2025 pukul 14:00 WIB
- Gambar 3.12** diunduh dari <https://fileinfo.com/extension/odt> pada 22 Juni 2025 pukul 14:30 WIB
- Gambar 3.13** diunduh dari <https://fileinfo.com/extension/ods> pada 22 Juni 2025 pukul 15:00 WIB
- Gambar 3.14** diunduh dari <https://fileinfo.com/extension/odp> pada 22 Juni 2025 pukul 15:30 WIB
- Gambar 3.15** diunduh dari <https://fileinfo.com/extension/rtf> pada 22 Juni 2025 pukul 16:00 WIB
- Gambar 3.16** diunduh dari <https://fileinfo.com/extension/pdf> pada 22 Juni 2025 pukul 16:30 WIB
- Gambar 3.17** diunduh dari <https://fileinfo.com/extension/csv> pada 22 Juni 2025 pukul 17:00 WIB

- Gambar 3.18** diunduh dari <https://fileinfo.com/extension/txt> pada 22 Juni 2025 pukul 17:30 WIB
- Gambar 3.20** diunduh dari <https://fileinfo.com/extension/jpg> pada 22 Juni 2025 pukul 18:00 WIB
- Gambar 3.21** diunduh dari <https://fileinfo.com/extension/png> pada 22 Juni 2025 pukul 18:30 WIB)
- Gambar 3.22** diunduh dari <https://fileinfo.com/extension/gif> pada 22 Juni 2025 pukul 19:00 WIB
- Gambar 3.23** diunduh dari <https://fileinfo.com/extension/bmp> pada 22 Juni 2025 pukul 19:30 WIB
- Gambar 3.24** diunduh dari <https://fileinfo.com/extension/tiff> pada 22 Juni 2025 pukul 20:00 WIB
- Gambar 3.25** diunduh dari <https://fileinfo.com/extension/svg> pada 22 Juni 2025 pukul 20:30 WIB
- Gambar 3.26** diunduh dari <https://fileinfo.com/extension/raw> pada 22 Juni 2025 pukul 21:00 WIB
- Gambar 3.27** diunduh dari <https://fileinfo.com/extension/heif> pada 22 Juni 2025 pukul 21:30 WIB
- Gambar 3.28** diunduh dari <https://fileinfo.com/extension/psd> pada 22 Juni 2025 pukul 22:00 WIB
- Gambar 3.34** diunduh dari <https://www.capcut.com/> pada 23 Juni 2025 pukul 13:00 WIB
- Gambar 3.35** diunduh dari *Youtube Creators* pada 23 Juni 2025 pukul 13:05 WIB
- Gambar 4.3** diunduh dari <https://www.sonypictures.com/movies/missing> pada 26 September 2025 pukul 12.17 WIB.
- Gambar 4.19** diunduh dari <https://pwm.trendmicro.com/password-check> pada 26 September 2025 pukul 15.40 WIB.
- Gambar 4.21** diunduh dari <https://studio.code.org/courses/hoc-encryption/units/1/lessons/1/levels/1> pada 13 November 2024 pukul 20.00 WIB.
- Gambar 4.22** diunduh dari <https://studio.code.org/courses/hoc-encryption/units/1/lessons/1/levels/4> pada 13 November 2024 pukul 20.15 WIB.



Indeks

A

algoritma 4, 41, 76, 91–92, 94, 98, 106, 110, 219–221
analisis data 2–5, 19–23, 35–36, 150
aplikasi perkantoran 130–131, 142

B

backdoor 198–199
badmood 225
bahasa pemrograman 40–41, 51, 54, 64, 66, 76–79, 85, 90, 92–94, 107
berpikir komputasional 3, 5, 42, 76, 95, 99–100
blog 117, 150, 155–159, 162–163, 170–171, 227
blogger 157
brute force attack 201, 241

C

caption 173, 175, 231
central processing unit (CPU) 120, 242
caesar's cipher 220–223, 242
content creator 170, 225
credential stuffing 202, 242
cyberbullying 228, 242

D

data 2–12, 14–16, 18–25, 27–33, 35–38, 66, 96, 98, 107, 116–117, 119–123, 130, 132–133, 135–139, 142, 144–150, 152–153, 162, 164–169, 177–178, 182–191, 193–205, 209, 211–212, 214–215, 217, 219–220, 222, 226, 230, 232–234, 238
decode 242
dekomposisi 95, 99–100
dictionary attack 202, 242
dislike 225
distributed denial of service (DDOS) 200, 242

E

email 123, 185, 196, 199–200, 211, 215, 217, 219
encode 243
enkripsi 195, 215, 218–221, 223–224

F

fakta 185, 229
file 8, 37, 43, 50, 53, 63, 75, 122–123, 130–152, 154, 169, 176–177, 195–199, 203–205, 209, 212, 222
firewall 204–205
folder 8, 37
fungsi iii, 22, 41, 53, 57, 65–66, 75, 118, 132–133, 152, 198, 219
function 41, 53–54, 56–57, 59, 63, 67, 69–71, 75

G

Google Authenticator 217
graph 2–3, 6, 14–16, 18–19, 37

H

hoaks 170, 173, 229

I

informasi iv, 3, 5, 9, 20, 23, 26–27, 29–33, 36, 116–117, 119–121, 123, 143, 150, 153, 155, 157–158, 161, 164–166, 169–171, 178, 182–187, 190–191, 193, 195, 199–201, 203–204, 211, 214–215, 218–220, 225–233, 235–236
informatika xiii
internet 117, 121–123, 127, 147, 150, 155, 161, 170–171, 174–175, 182–184, 199, 202–205, 209, 211–212, 220, 226–228, 238

J

jaringan komputer 194, 209

K

kecerdasan artifisial 125, 170, 200
kecerdasan artifisial generatif 243
keylogger attack 202, 244
kode 41–42, 48, 64–65, 68, 76–77, 79, 85, 90, 94, 100, 102, 104, 106–110, 112–113, 136, 144, 188, 195–196, 204, 217–220, 224, 233
koding 244
komputasi 2

komputasional 3, 5, 40–42, 76, 95, 99–100
komputer iv, 3, 8, 37, 43, 63, 98, 106,
119–125, 130, 144, 147, 150–153, 163,
178, 182, 194–199, 203–205, 209, 220

L

lembar kerja 96–97, 100–101, 106–107,
110–111, 117, 150, 152–153, 166, 169
like 225–226
literasi digital iv, 117, 119, 170

M

malware 123, 182, 194–195, 197–198,
202–204, 206, 208, 212, 237
man-in-the-middle attack (MitM) 201, 244
masukan 96–98, 173
memori 122, 196, 204
mesin pencari 117
mesin pencari kecerdasan artifisial 245
mesin pencari tradisional 117
mindfulness 125, 128, 182, 226,
230, 235–236
**model bahasa besar (large
language model)** 245
multi-factor authentication (MFA) 218, 245
multitasking 239, 245

O

one time password (OTP) 245
overthinking 127, 225, 235

P

pembelajaran mesin 245, 246
pembelajaran mendalam iv
pemrograman tekstual 40–41, 76–79, 85,
90, 102, 106, 108, 110
pemrograman visual blok 44, 65–66,
76–79, 85, 102, 106, 108, 110
pemrosesan bahasa alami 245, 246
pengenalan pola 4
perangkat keras 66, 122, 150,
195–196, 203
perangkat lunak 66, 116–117, 130, 135,
144, 146, 150–154, 158, 162–164, 166,
172, 175, 195–196, 198–199, 204
perkakas 116, 118–119, 125, 172, 174–175
phishing 195, 232–233
plaintext 219–221

platform 66, 135, 150, 157–160, 173, 175,
185, 230, 234

plugged iv

processing 120, 242, 246

program 40–47, 49–54, 56–58, 61–69,
74–77, 85, 93–97, 99–100, 102, 104,
106–109, 112–113, 120, 123, 143,
151–153, 194, 196, 199, 202, 222

programming 241, 247

prompt 117

prosedur 53, 57, 67, 75

R

ransomware 197, 203

S

scratch viii, xi, 40–41, 43–45, 48–51, 54,
65–66, 79–90, 94, 112–113

security patch 233

slide presentasi 132, 134, 153

sprite 66, 94, 113

spyware 195, 199

streaming 122, 146–147, 149, 159, 225

struktur data 2–3, 5–12, 14–16,
18–19, 37, 152

struktur data tree 2, 7–12, 14–16, 19, 37

T

tethering 209

tree 2–3, 6–12, 14–16, 19, 37

trojan horse 198

U

unplugged 248

V

variabel 3, 25, 32, 45, 51–52, 56, 62, 66, 68,
70–71, 74, 76, 93, 96–98, 107

virtual private network (VPN) 201, 248

virus 123, 195–196, 204, 222

visualisasi data 2, 20, 23–25, 36, 152

vlog 117, 150, 157–161, 227

W

website 21, 170, 200, 204

wireless 209

worm 196–197





Profil Penulis

Erlangga

Email : erlangga.kmoekasan@gmail.com
Instansi : Universitas Pendidikan Indonesia
Alamat Instansi : Jl. Dr. Setiabudhi No. 229 Bandung 40154 Jawa Barat - Indonesia
Bidang Keahlian : Informatika, Sistem Informasi, Rekayasa Perangkat Lunak, *Machine Learning*

Riwayat Pekerjaan / Profesi (10 Tahun Terakhir)

1. Dosen Universitas Pendidikan Indonesia (2018 - Sekarang)
2. Dosen Universitas Terbuka (2022 - Sekarang)
3. LSP P1 Universitas Pendidikan Informatika Skema Senior Programmer (2023 - Sekarang)
4. Tim Penyusun Modul Bimtek ToT Informatika Bagi Guru SMP dan SMA KEMENDIKDASMEN GTK (2022 - Sekarang)
5. Tim Penyusun Modul Bimtek ToT Koding Kecerdasan Artifisial (KKA) Guru SMP dan SMA KEMENDIKDASMEN GTK (2025 - Sekarang)
6. Narasumber Modul Bimtek ToT Informatika Bagi Guru SMP dan SMA KEMENDIKDASMEN GTK (2022 - Sekarang)
7. Narasumber Modul Bimtek ToT Koding Kecerdasan Artifisial (KKA) Guru SMP dan SMA KEMENDIKDASMEN GTK (2025 - Sekarang)
8. Tim Developer proyek Jasa Pengembangan Solusi Registrasi Sosial Ekonomi (Regsosek) Bappenas PT. PRABATECH (2023)
9. Dosen Universitas Komputer Indonesia (UNIKOM) (2010 -2017)

Riwayat Pendidikan dan Tahun Belajar

1. S1 - Universitas Komputer Indonesia (UNIKOM) Bandung - Teknik Informatika (2004 - 2009)
2. S2 - Institut Teknologi Bandung (ITB) - Magister Informatika (2013 - 2015)
3. S3 - Universitas Pendidikan Indonesia (UPI) - Doktorat Pendidikan Teknologi dan Kejuruan (2025 - Sekarang)

Judul Buku dan Tahun Terbit (10 Tahun Terakhir)

1. Modul Bimtek ToT Informatika Bagi Guru SMP dan SMA KEMENDIKDASMEN GTK (2022) Fase D dan E/F
2. Modul Bimtek ToT Informatika Bagi Guru SMP dan SMA KEMENDIKDASMEN GTK (2023) Fase D dan E/F



Profil Penulis

Erna Piantari

Email : erna.piantari14@gmail.com
Instansi : Universitas Pendidikan Indonesi
Alamat Instansi : Jl. Dr. Setiabudhi No. 229 Bandung 40154 Jawa Barat - Indonesia
Bidang Keahlian : Informatika, Business Intelligence, Data Science, *Machine Learning*, Pedagogis

Riwayat Pekerjaan / Profesi (10 Tahun Terakhir)

1. Dosen Universitas Pendidikan Indonesia (2018 - Sekarang)
2. Dosen Universitas Terbuka (2022 - Sekarang)
3. Tim Penyusun Modul Bimtek ToT Informatika Bagi Guru SMP dan SMA KEMENDIKDASMEN GTK (2022)
4. Tim Penyusun Modul Bimtek ToT Koding Kecerdasan Artifisial (KKA) Guru SMP dan SMA KEMENDIKDASMEN GTK (2025)
5. Narasumber Modul Bimtek ToT Informatika Bagi Guru SMP dan SMA KEMENDIKDASMEN GTK (2022)
6. Narasumber Modul Bimtek ToT Koding Kecerdasan Artifisial (KKA) Guru SMP dan SMA KEMENDIKDASMEN GTK (2025)
7. Dosen Universitas Jenderal Ahmad Yani (2017-2018)
8. IT Consultant LAPI DIVUSI ITB (2016-2018)

Riwayat Pendidikan dan Tahun Belajar

1. S1 - IPB University - Ilmu Komputer (2007 - 2011)
2. S2 - Institut Teknologi Bandung (ITB) - Magister Informatika (2013 - 2016)

Judul Buku dan Tahun Terbit (10 Tahun Terakhir)

1. Modul Bimtek ToT Informatika Bagi Guru SMP dan SMA KEMENDIKDASMEN GTK (2022) Fase D dan E/F
2. Modul Bimtek ToT Informatika Bagi Guru SMP dan SMA KEMENDIKDASMEN GTK (2023) Fase D dan E/F
3. Modul Bimtek ToT Informatika Bagi Guru SMP dan SMA KEMENDIKDASMEN GTK (2024) Fase D dan E/F
4. Modul Bimtek ToT Koding Kecerdasan Artifisial (KKA) Guru SMP dan SMA KEMENDIKDASMEN GTK Fase D dan E/F (2025)
5. Sistem Komputer, Penamuda Media 2024
6. Data Science, Penamuda Media 2024

Profil Penulis

Khairur Rosyidi



Email : kacongarul@gmail.com
Instansi : SDN Manyar Sabrangan II/231 Kota Surabaya
Alamat Instansi : Jl. Manyar Tirtomoyo VIII No.1,
Kelurahan Manyar Sabrangan, Kecamatan
Mulyorejo, Kota Surabaya.
Bidang Keahlian : Informatika dan Pedagogis

Riwayat Pekerjaan / Profesi (10 Tahun Terakhir)

1. Guru Teknik Komputer dan Jaringan, SMK Negeri 5 Malang (2016-2019)
2. Mentor Lembaga Kursus Komputer bidang Jaringan Komputer, Alpine Citra Komputer Malang (2016)
3. Guru Mata Pelajaran Informatika, SMP Negeri 52 Surabaya (2019-2025)
4. Narasumber Bimtek Mata Pelajaran Informatika Pada Guru Jenjang SMP (Fase D) : Penerapan HOTS dan TPACK dalam Pembelajaran Informatika; dan Micro/Peer-Teaching, KEMDIKDASMEN (2024).
5. Kepala SDN Manyar Sabrangan II/231 Kota Surabaya (2025-sekarang)

Riwayat Pendidikan dan Tahun Belajar

1. SMAN 1 Sumenep - IPA (2009-2012)
2. S1 - Universitas Negeri Malang (UM) - Pendidikan Teknik Informatika (2012 - 2016)
3. S2 - Universitas Brawijaya (UB) - Ilmu Komputer (2017)

Judul Buku dan Tahun Terbit (10 Tahun Terakhir)

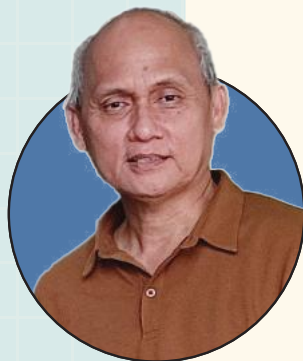
1. Doaku Menanti Matahari (2020)
2. Suluk Pandemi: Kumpulan Esai Mencari Hikmah di Balik Musibah (2020)
3. Teka-Teki Silang Informatika Kelas VII SMP/MTs (2021)

Judul Penelitian dan Tahun Terbit (10 Tahun Terakhir)

1. Perbandingan Pengaruh Model Pembelajaran TGT Dan STAD Terhadap Hasil belajar Mata Pelajaran Pemrograman Web Siswa Kelas X SMK Negeri 12 Malang (2016)
2. *Design and Development Educational Media Diction of Madurese Language Level* (2018)

Profil Penelaah

Budi Permana



Email : budi.permana.dr@gmail.com
Instansi : STMIK LIKMI
Alamat Instansi : Jl. Ir. H. Djuanda No.96 Bandung
Bidang Keahlian : Sistem Informasi

Riwayat Pekerjaan / Profesi (10 Tahun Terakhir)

Dosen Tetap dan sekarang menjabat sebagai Ketua STMIK LIKMI Bandung

Riwayat Pendidikan dan Tahun Belajar

1. Lulus Sarjana (S1) Jurusan Ekonomi Akuntansi dari Universitas Padjadjaran Bandung (UNPAD) pada tahun 1986.
2. Lulus Pasca Sarjana (S2) Jurusan Teknik dan Manajemen Industri dari Institut Teknologi Bandung (ITB) pada tahun 1989.
3. Lulus Pasca Sarjana (S3) Administrasi Pendidikan UPI konsentrasi Sistem Informasi Pendidikan pada tahun 2013.

Judul Buku dan Tahun Terbit (10 Tahun Terakhir)

1. Penulis Buku-buku Komputer di PT Elex Media Komputindo Gramedia yang diterbitkan secara Nasional. Hasil karya yang telah diterbitkan di antaranya:
 - a. Seri Buku 36 Jam Belajar Komputer
 - b. Seri buku Student Guide Series (SGS)
 - c. Seri buku Student Exercise Series (SES) untuk pemakai komputer pemula.
2. Buku berbahasa Melayu yang diterbitkan di Malaysia oleh Synergy Media.
 - a. Student Guide Series - Easy Com "Microsoft Windows XP" ISBN:983-197-678-9
 - b. Student Guide Series - Easy Com "Microsoft PowerPoint" ISBN:983-197-677-9
3. Karya buku yang telah diterbitkan tahun 2020 oleh Direktorat Pembinaan Sekolah Menengah Pertama - Kementerian Pendidikan dan Kebudayaan berjudul "#Bijak dan Kreatif dalam Bermedia Sosial jenjang SMP" ISBN:978-623-95423-5-1
4. Buku Teks Utama Informatika SMA Kelas XII - Pusat Perbukuan, Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi - 2022 ISBN:978-602-244-505-0

Profil Penelaah

Aris Sunawar



Email : arissunawar@unj.ac.id
Instansi : Universitas Negeri Jakarta
Alamat Instansi : Jl. Rawamangun muka 1 Jakarta
Bidang Keahlian : Elektro

Riwayat Pekerjaan / Profesi (10 Tahun Terakhir)

1. (2009-Sekarang) : Staff Pengajar Prodi Pendidikan Teknik Elektro Universitas Negeri Jakarta
2. (2020-Sekarang) : Direktur PT Indoglass Putra Jaya
3. (2022-Sekarang) : Direktur PT Geneus DNA Indonesia

Riwayat Pendidikan dan Tahun Belajar

1. S1 Pendidikan Teknik Elektro - Universitas Negeri Jakarta (2001-2006)
2. S2 Teknik Elektro - Universitas Indonesia (2006-2008)
3. S3 Teknik Elektro - Universitas Indonesia (2012-2020)

Judul Buku dan Tahun Terbit (10 Tahun Terakhir)

1. Teknik Pengerjaan Listrik (2016)

Judul Penelitian dan Tahun Terbit (10 Tahun Terakhir):

1. Inovasi Pembelajaran Ekonomi Keluarga Berbasis Website Di Program Studi Pendidikan Kesejahteraan Keluarga (2018)
2. The characteristics of heat inside a parked car as energy source for thermoelectric generators (2019)
3. Development of prototype e-learning mathematics learning tools using Moodle (2021)
4. Analysis of the Impact of Distance and Temperature in Determining Transfer Power Using Laser Applications (2021)
5. Rancang Bangun Pengecekan Alat Pelindung Diri Menggunakan Algoritma You Only Look Once (Yolo) (2022)
6. The effect of weight, speed, and mileage on the use of electric bicycle batteries (2023)

Informasi Lain dari Penelaah:

<https://scholar.google.co.id/citations?hl=id&user=LhOLO30AAAAJ>

Profil Ilustrator

Prehatin

Email : pristories@gmail.com
Instansi : freelance
Alamat Instansi : Jl. Adam No 34 Sukabumi Utara,
Kebon Jeruk, Jakarta Barat
Bidang Keahlian : Ilustrasi

Riwayat Pekerjaan / Profesi (10 Tahun Terakhir)

1. 2012 - 2019 staff admin MNC Skyvision
2. Ilustrator Freelance

Riwayat Pendidikan dan Tahun Belajar

1. 1997-2003 MI Darul Muqinin
2. 2004-2006 MTs Negeri 12 Jakarta Barat
3. 2007-2009 SMK Negeri 45 Jakarta
4. 2010-2014 STIE Kasih Bangsa

Judul Buku dan Tahun Terbit (10 Tahun Terakhir)

1. Bulu istimewa hege 2025
2. Lizy dan buntutnya 2024
3. Pipit dan beras ketan 2024
4. Aku tau ciri fisik rasulullah 2023

Profil Editor

Misianita Hapsari



Email : misianitahapsari38@guru.smp.belajar.id
Instansi : SMP Negeri 117 Jakarta
Alamat Instansi : Jl. Taruna Jl. Pahlawan Revolusi No.54,
RT.6/RW.4, Pd. Bambu, Kec. Duren Sawit,
Kota Jakarta Timur, Jakarta 13430
Bidang Keahlian : Informatika

Riwayat Pekerjaan / Profesi (10 Tahun Terakhir)

1. Tahun 2015 : Recruitment & Development Staff di PT. MarkPlus, Inc
2. Tahun 2015 - 2020 : Project Controller dan Procurement Staff di PT. Fiberhome Technologies Indonesia
3. Tahun 2021 - sekarang : Guru Informatika di SMP Negeri 117 Jakarta

Riwayat Pendidikan dan Tahun Belajar

1. SMKN 26 Jakarta - Jurusan Teknik Komputer Jaringan (2006-2010)
2. Universitas Pendidikan Indonesia - Pendidikan Ilmu Komputer (2010-2014)

Judul Buku dan Tahun Terbit (10 Tahun Terakhir)

1. Editor Buku Panduan Guru dan Buku Siswa Buku Teks Mata Pelajaran Informatika Kelas XII, Pusat Kurikulum dan Perbukuan (2022)
2. Penulis Naskah Buku Audio Mata Pelajaran Informatika Kelas XII, Pusat Kurikulum dan Perbukuan (2023)
3. Editor Buku Panduan Guru dan Buku Siswa Buku Teks Mata Pelajaran Informatika Kelas VIII Edisi Revisi, Pusat Kurikulum dan Perbukuan (2024)

Informasi Lain dari Editor:

<https://www.youtube.com/@informatikaseru>



Profil Editor Visual

M Rizal Abdi

Email : kotakpesandarimu@gmail.com
Instansi : *Center for Religious and Cross-cultural Studies,*
Sekolah Pascasarjana, Universitas Gadjah Mada
Bidang Keahlian : Editorial desain dan ilustrasi

Riwayat Pekerjaan / Profesi (10 Tahun Terakhir)

1. Desainer. Hocuspocus Rekavasthu (2006—2012)
2. Desainer editorial dan ilustrator beberapa penerbit indie di Yogyakarta, Jakarta, dan California (2015—sekarang)
3. Editor lepas beberapa penerbit indie di Yogyakarta dan Jakarta (2017—sekarang)
4. Staf Pendidikan Publik, *Center for Religious and Cross-cultural Studies,* Sekolah Pascasarjana, Universitas Gadjah Mada (2021—sekarang)

Riwayat Pendidikan dan Tahun Belajar

1. S1 - Ilmu Komunikasi, Fisipol, UGM (2004)
2. S2 - *Center for Religious and Cross-cultural Studies* (CRCS). Sekolah Pascasarjana UGM (2015)

Judul Buku dan Tahun Terbit (10 Tahun Terakhir)

1. *Indonesian Pluralities: a companion book*, 2023 (Boston University)
2. *Menjadi Hawa: Antologi Pengalaman Biologis dan Sosial Perempuan*, 2022. Editor. (Penerbit Gading)
3. *Buddhisme Militan*, 2021. Editor. (Penerbit Gading)
4. *Gerombolan Kucing Bandel*, 2021, Editor. (Penerbit Pocer)
5. *Bangga Jadi MD: 40 Tahun Bertumbuh, Berubah, dan Berinovasi*. 2020. (Magister dan Doktor Ilmu-Ilmu Ekonomi, Fakultas Ekonomika dan Bisnis, Universitas Gadjah Mada)
6. *Kudus: Yerusalem di Tanah Jawa*. 2019. Penulis, bersama Revianto Budi Santosa. (Yayasan Masjid, Menara, dan Makam Sunan Kudus)
7. *Jejak Amnesti Pajak*. Editor. (Direktorat Jenderal Pajak D.I. Yogyakarta, March, 2017)



Profil Desainer

Syndhi Renolarisa

Instansi : Praktisi
Alamat Instansi : Jakarta
Bidang Keahlian : Desain grafis dan ilustrasi

Riwayat Pekerjaan / Profesi (10 Tahun Terakhir)

1. (2014 – sekarang) *Freelance* Ilustrator & Penata Letak/Desainer
2. (2014 – 2016) *Quality Control & Internal Censorship*, Transvision.
3. (2016 – sekarang) *Quality Control & Internal Censorship*, UseeTV - Indihome.
4. (2023 – sekarang) *Digital Advertising Business Operation*, Telkomsel.

Riwayat Pendidikan dan Tahun Belajar

1. S1 Jurusan Desain Komunikasi Visual, Sekolah Tinggi Media Komunikasi Trisakti, Jakarta - 2014

Judul Buku dan Tahun Terbit (10 Tahun Terakhir)

1. Desain isi Buku Panduan Guru dan Buku Siswa Buku Teks Pelajaran di Pusat Kurikulum dan Perbukuan (2014-sekarang)
2. Ilustrator Buku Panduan Guru dan Buku Siswa Buku Teks Pelajaran di Pusat Kurikulum dan Perbukuan (2014)