



```
transition: filter 0.5s;  
margin-left: 20px;  
}  
icon-right:hover {  
  filter: brightness(1.2);  
}
```



PEMROGRAMAN WEB

Indra Rianto

PEMROGRAMAN WEB

Indra Rianto



Tahta Media Group

UU No 28 tahun 2014 tentang Hak Cipta

Fungsi dan sifat hak cipta Pasal 4

Hak Cipta sebagaimana dimaksud dalam Pasal 3 huruf a merupakan hak eksklusif yang terdiri atas hak moral dan hak ekonomi.

Pembatasan Pelindungan Pasal 26

Ketentuan sebagaimana dimaksud dalam Pasal 23, Pasal 24, dan Pasal 25 tidak berlaku terhadap:

- i. penggunaan kutipan singkat Ciptaan dan/atau produk Hak Terkait untuk pelaporan peristiwa aktual yang ditujukan hanya untuk keperluan penyediaan informasi aktual;
- ii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk kepentingan penelitian ilmu pengetahuan;
- iii. Penggandaan Ciptaan dan/atau produk Hak Terkait hanya untuk keperluan pengajaran, kecuali pertunjukan dan Fonogram yang telah dilakukan Pengumuman sebagai bahan ajar; dan
- iv. penggunaan untuk kepentingan pendidikan dan pengembangan ilmu pengetahuan yang memungkinkan suatu Ciptaan dan/atau produk Hak Terkait dapat digunakan tanpa izin Pelaku Pertunjukan, Produser Fonogram, atau Lembaga Penyiaran.

Sanksi Pelanggaran Pasal 113

1. Setiap Orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak Rp100.000.000 (seratus juta rupiah).
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak Rp500.000.000,00 (lima ratus juta rupiah).

PEMROGRAMAN WEB

Penulis:
Indra Rianto

Desain Cover:
Tahta Media

Editor:
Jentelino Silvester Langitan

Proofreader:
Tahta Media

Ukuran:
vii, 86, Uk: 15,5 x 23 cm

ISBN : 978-623-147-981-5

Cetakan Pertama:
September 2025

Hak Cipta 2025, Pada Penulis

Isi diluar tanggung jawab percetakan

Copyright © 2025 by Tahta Media Group
All Right Reserved

Hak cipta dilindungi undang-undang
Dilarang keras menerjemahkan, memfotokopi, atau
memperbanyak sebagian atau seluruh isi buku ini
tanpa izin tertulis dari Penerbit.

PENERBIT TAHTA MEDIA GROUP
(Grup Penerbitan CV TAHTA MEDIA GROUP)
Anggota IKAPI (216/JTE/2021)

KATA PENGANTAR

Selamat datang di dunia pemrograman web! Sebuah dunia di mana kreativitas bertemu dengan logika untuk menciptakan karya-karya yang dapat diakses oleh jutaan orang di seluruh dunia. Kita hidup di era digital di mana halaman web dan aplikasi telah menjadi bagian tak terpisahkan dari kehidupan sehari-hari, mulai dari cara kita berkomunikasi, belajar, hingga berbisnis. Kemampuan untuk membangun dan memahami teknologi di balik web bukan lagi sekadar keahlian teknis, melainkan sebuah literasi fundamental di abad ke-21.

Buku ajar yang ada di tangan Anda saat ini disusun dengan satu tujuan utama yaitu menjadikan perjalanan Anda mempelajari pemrograman web semudah dan semenarik mungkin. Materi di dalamnya dirancang secara sistematis, dimulai dari fondasi paling dasar seperti HTML untuk membangun kerangka, CSS untuk mempercantik tampilan, hingga JavaScript untuk menambahkan interaktivitas. Kita tidak akan berhenti di situ, buku ini juga akan mengajak Anda menyelami dunia *backend* dan basis data, memberikan pemahaman yang utuh tentang bagaimana sebuah aplikasi web dinamis bekerja.

Saya percaya bahwa cara terbaik untuk belajar pemrograman adalah dengan langsung mempraktikkannya. Oleh karena itu, setiap bab dalam buku ini dilengkapi dengan penjelasan teori yang ringkas, contoh-contoh kode yang relevan, dan diakhiri dengan soal latihan untuk mengasah pemahaman dan keterampilan Anda. Jangan takut untuk membuat kesalahan, karena dari sanalah proses belajar yang sesungguhnya terjadi.

Buku ini saya persembahkan untuk para mahasiswa, khususnya di Jurusan Pendidikan Teknologi Informasi dan Komunikasi, yang memiliki semangat untuk berkarya dan berinovasi. Semoga buku ini dapat menjadi teman setia yang membimbing Anda, dari baris kode pertama hingga proyek web pertama Anda yang berhasil Anda luncurkan.

Selamat belajar, selamat berkarya, dan selamat menjelajahi dunia pemrograman web yang penuh kemungkinan.

DAFTAR ISI

Kata Pengantar.....	iv
Daftar Isi.....	v
BAB 1 Pengantar Dunia Pemrograman Web	1
A. Definisi dan Sejarah World Wide Web (WWW)	1
B. Cara Kerja Website - Client, Server, HTTP/HTTPS, Domain, dan Hosting	2
C. Mengenal Tiga Pilar Utama Web: HTML, CSS, dan JavaScript	5
D. Peralatan yang Dibutuhkan: Browser, Text Editor, dan Local Server.....	6
E. Praktikum.....	7
Latihan.....	10
BAB 2 HTML - Kerangka Dasar Halaman Web	12
A. Anatomi Dokumen HTML	12
B. Konsep Inti: Elemen, Tag, dan Atribut.....	13
C. Elemen Teks Fundamental	13
D. Strukturisasi Konten dengan Daftar (List).....	14
E. Menampilkan Data Tabular dengan Tabel.....	15
F. Mengumpulkan Input Pengguna dengan Formulir	16
G. Membangun Layout Semantik dengan HTML5	17
1. Praktikum.....	18
Latihan.....	22
BAB 3 CSS - Menghias Tampilan Halaman Web	25
A. Apa itu CSS dan Mengapa Penting?.....	25
B. Tiga cara menyisipkan CSS: Inline, Internal, dan External.....	26
C. Sintaks Dasar CSS	28
D. <i>Selector</i> Dasar.....	28
E. Mengatur Tampilan Teks (Tipografi).....	29
F. <i>The Box Model: Konsep Fundamental Spasi</i>	29
G. Mengatur Warna dan Latar Belakang (background-color, background-image).....	30
Praktikum	31
Latihan.....	33

BAB 4	Layouting dengan CSS Tingkat Lanjut	35
A.	Konsep Properti display	35
B.	Layouting Satu Dimensi dengan Flexbox.....	36
C.	Layouting Dua Dimensi dengan CSS Grid.....	37
D.	Desain Responsif dengan Media Queries.	38
E.	Pengenalan framework CSS sebagai perbandingan.....	39
F.	Praktikum	41
	Latihan.....	44
BAB 5	JavaScript - Membuat Halaman Web Interaktif.....	47
A.	Pengenalan JavaScript	47
B.	Menyisipkan JavaScript ke dalam HTML	48
C.	Dasar-Dasar Pemrograman Java Script	51
D.	Pengenalan DOM (Document Object Model)	54
E.	Manipulasi DOM.....	55
F.	Event Handling (Menangani Aksi Pengguna)	57
G.	Praktikum.....	58
	Latihan.....	60
BAB 6	Pengenalan Basis Data dan Backend.....	62
A.	Konsep Basis Data Relasional (RDBMS).	62
B.	Pengenalan SQL (Structured Query Language)	64
C.	Arsitektur Client-Server untuk Aplikasi Web Dinamis	65
D.	Pengenalan Bahasa Pemrograman Sisi Server (PHP).....	66
E.	Sintaks Dasar PHP.....	66
F.	Menghubungkan PHP ke Basis Data MySQL/MariaDB.....	67
G.	Praktikum.....	69
	Latihan.....	74
BAB 7	Studi Kasus - Membangun Proyek Akhir.....	76
A.	Merancang Proyek: Dari Ide ke Perencanaan.....	76
B.	Konsep CRUD (Create, Read, Update, Delete).....	77
C.	Keamanan Dasar: Mencegah SQL Injection dengan Prepared Statements.....	77
D.	Pengenalan Session untuk Autentikasi Pengguna	78
E.	Praktikum: Membuat Aplikasi “Blog Sederhana”	78
	Latihan.....	80

Daftar Pustaka	82
Glosarium	83

BAB 1

PENGANTAR DUNIA

PEMROGRAMAN WEB

A. Definisi dan Sejarah World Wide Web (WWW)

Seringkali Internet dan World Wide Web dianggap sama, namun keduanya adalah entitas yang berbeda. Internet adalah infrastruktur global yang terdiri dari jaringan komputer yang saling terhubung di seluruh dunia. Ia adalah "jalan raya"-nya. Sementara itu, World Wide Web (WWW), atau yang sering kita sebut "Web", adalah sistem informasi di mana dokumen dan sumber daya web lainnya diidentifikasi oleh *Uniform Resource Locators* (URL), yang dapat dihubungkan melalui *hyperlink*, dan diakses melalui Internet. Web adalah salah satu layanan yang berjalan "di atas" Internet, sama seperti email atau FTP.

Web diciptakan oleh seorang ilmuwan Inggris, Tim Berners-Lee, saat bekerja di CERN (Organisasi Eropa untuk Riset Nuklir) pada tahun 1989. Tujuannya adalah untuk memenuhi permintaan akan sistem berbagi informasi yang otomatis antara para ilmuwan di universitas dan institut di seluruh dunia. Untuk mewujudkan visinya, Tim Berners-Lee menciptakan tiga teknologi inti:

1. HTML (HyperText Markup Language): Bahasa *markup* standar untuk membuat dan menyusun halaman web beserta isinya.
2. URL (Uniform Resource Locator): Alamat unik yang digunakan untuk mengidentifikasi setiap sumber daya di web, seperti halaman HTML, gambar, atau video.
3. HTTP (Hypertext Transfer Protocol): Protokol atau seperangkat aturan yang digunakan untuk mentransfer data (seperti dokumen HTML) dari server ke browser.

Sejak pengembangan Web terus menerus mengalami evolusi sampai ke tahap sekarang ini, evolusi tersebut dapat dijabarkan sebagai berikut:

1. Pada Tahun 1991, Proyek WWW diumumkan kepada publik.
2. Pada Tahun 1993, CERN merilis teknologi WWW ke domain publik tanpa biaya royalti. Pada tahun yang sama, browser Mosaic dirilis, yang menjadi sangat populer karena kemampuannya menampilkan gambar di dalam teks, bukan di jendela terpisah. Ini memicu ledakan popularitas web.
3. Pada Pertengahan 1990-an, terjadi "Perang Browser" pertama antara Netscape Navigator dan Microsoft Internet Explorer, yang mendorong inovasi pesat.
4. Pada Awal 2000-an, mulai munculnya era Web 2.0, yang menekankan pada konten buatan pengguna (*user-generated content*), interaktivitas, dan media sosial (lahirnya Wikipedia, Facebook, YouTube).
5. Sekarang ini kita berada di era web mobile, di mana akses internet lebih banyak dilakukan melalui perangkat seluler, serta berkembangnya teknologi *cloud computing* dan konsep-konsep baru seperti Web3.

B. Cara Kerja Website - Client, Server, HTTP/HTTPS, Domain, dan Hosting

Analogi sederhana dari cara kerja website yaitu bayangkan Anda sedang berada di sebuah restoran. Anda (Client) duduk dan melihat menu. Ketika Anda memesan makanan kepada pelayan, pelayan akan meneruskan pesanan Anda ke dapur (Server). Dapur kemudian menyiapkan makanan pesanan Anda dan memberikannya kembali kepada pelayan untuk diantarkan ke meja Anda. Proses inilah yang disebut siklus Request (memesan) dan Response (makanan diantar). Cara kerja website sangat mirip dengan proses ini.

Komponen Utama dari Website

1. Client (Klien): Ini adalah perangkat yang Anda gunakan untuk mengakses internet, seperti laptop, smartphone, atau tablet, beserta perangkat lunak di dalamnya, yaitu web browser (contoh: Google Chrome, Mozilla Firefox, Safari). Browser bertugas sebagai "pelayan" yang membuat permintaan dan menampilkan hasilnya kepada Anda.

2. Server: Ini adalah komputer super kuat yang menyala 24/7, terhubung ke internet, dan dirancang khusus untuk menyimpan file-file website (HTML, CSS, JavaScript, gambar, video, basis data) dan "melayani" permintaan dari *client*.
3. Domain Name (Nama Domain): Ini adalah alamat website yang mudah diingat oleh manusia, misalnya www.google.com atau www.kemdikbud.go.id. Ini seperti nama restoran yang mudah Anda hafal.
4. IP Address (Alamat IP): Setiap server di internet memiliki alamat unik yang sesungguhnya berupa serangkaian angka (misalnya 142.250.204.78). Ini adalah alamat "koordinat GPS" asli dari server.
5. DNS (Domain Name System): Ini adalah "buku telepon" raksasa di internet. Tugasnya adalah menerjemahkan Nama Domain yang mudah diingat manusia menjadi Alamat IP yang dimengerti oleh komputer.
6. Hosting: Ini adalah layanan penyewaan tempat pada sebuah Server untuk menyimpan semua file website Anda agar bisa diakses secara online.
7. HTTP/HTTPS (Hypertext Transfer Protocol/Secure): Ini adalah bahasa atau protokol yang digunakan oleh *client* dan *server* untuk berkomunikasi. HTTPS adalah versi aman dari HTTP, di mana data yang dikirimkan dienkripsi, ditandai dengan ikon gembok di address bar browser.

Alur proses website langkah-demi-langkah adalah sebagai berikut:

1. Pengguna Memasukkan Alamat: Anda mengetikkan nama domain (misal: www.contoh.com) di *address bar* browser Anda dan menekan Enter.
2. Browser Bertanya ke DNS: Browser Anda tidak tahu di mana lokasi www.contoh.com. Maka, browser akan menghubungi server DNS. "Hai DNS, di mana alamat IP untuk www.contoh.com?"
3. DNS Menjawab: Server DNS mencari di dalam catatannya dan menjawab, "Alamat IP untuk www.contoh.com adalah 93.184.216.34."
4. Browser Mengirim HTTP Request: Setelah mengetahui alamat IP server, browser Anda mengirimkan sebuah permintaan HTTP Request ke server tersebut. Isi permintaannya kira-kira, "Tolong kirimkan saya file halaman utama dari www.contoh.com."

5. Server Memproses Request: Server dengan alamat IP 93.184.216.34 menerima permintaan, mencari file-file yang diminta (misalnya `index.html`, `style.css`, dll), dan mengumpulkannya.
6. Server Mengirim HTTP Response: Server mengirimkan kembali file-file tersebut kepada browser Anda dalam sebuah paket HTTP Response.
7. Browser Merender Halaman: Browser Anda menerima file-file tersebut. Ia akan membaca file HTML untuk membangun struktur, file CSS untuk mengatur tampilan, dan menjalankan file JavaScript untuk fungsionalitas. Proses ini disebut rendering, dan hasilnya adalah halaman web lengkap yang Anda lihat di layar Anda.

Website berdasarkan jenisnya dibedakan menjadi dua bagian yaitu Website Statis dan Website Dinamis. Bayangkan website statis seperti sebuah brosur cetak. Informasi di dalamnya tetap, tidak berubah, dan setiap orang yang menerima brosur itu akan melihat konten yang sama persis. Untuk mengubah isinya, Anda harus mendesain dan mencetak ulang seluruh brosur. Sebaliknya, bayangkan website dinamis seperti linimasa media sosial Anda. Kontennya terus berubah, disesuaikan untuk Anda (berdasarkan siapa yang Anda ikuti), dan Anda bisa beraksi dengannya (memberi suka, komentar). Konten yang ditampilkan kepada Anda akan berbeda dengan yang ditampilkan kepada teman Anda.

Website Statis (Static Website) adalah sebuah website yang kontennya bersifat tetap dan tidak berubah, kecuali *file source code*-nya diubah secara manual oleh developer. Setiap pengunjung akan melihat halaman yang sama persis. Cara kerja dari Website Statis yaitu server hanya perlu mengirimkan file HTML, CSS, dan JavaScript yang sudah jadi, apa adanya, ke browser. Tidak ada pemrosesan data di sisi server. Teknologi untuk membuat Website Statis utamanya menggunakan HTML dan CSS. JavaScript bisa digunakan untuk animasi atau interaksi sederhana di sisi klien, namun tidak untuk mengubah konten utama dari server. Kelebihannya yaitu cepat, aman (karena tidak ada basis data atau pemrosesan sisi server yang bisa dieksploitasi), dan murah untuk di-hosting. Kelemahannya yaitu tampilannya hanya seperti itu dan jika ingin dirubah, *file source code*-nya harus diubah secara manual oleh developer. Contoh dari Website Statis yaitu halaman portofolio pribadi, website profil perusahaan sederhana, halaman dokumentasi, website *landing page* produk.

Website Dinamis (Dynamic Website) adalah sebuah website yang kontennya dapat berubah-ubah secara otomatis dan *real-time* berdasarkan interaksi pengguna, waktu, atau faktor lainnya. Kontennya "dirakit" terlebih dahulu di server sebelum dikirim ke browser. Cara Kerja dari Website Dinamis yaitu ketika browser meminta sebuah halaman, server tidak langsung mengirim file HTML. Sebaliknya, server menjalankan sebuah program (misalnya skrip PHP), mengambil data dari basis data (misalnya MySQL), menggabungkan data tersebut ke dalam sebuah *template* HTML, baru kemudian mengirimkan hasil akhirnya sebagai halaman HTML lengkap ke browser. Teknologi yang digunakan adalah teknologi *frontend* (HTML, CSS, JavaScript) ditambah dengan bahasa pemrograman sisi server (*backend*) seperti PHP, Python, atau Node.js, dan sebuah sistem basis data seperti MySQL, PostgreSQL, atau MongoDB. Kelebihan dari Website Dinamis adalah interaktif, konten mudah diperbarui (cukup ubah data di basis data, tidak perlu edit kode), dan dapat dipersonalisasi untuk setiap pengguna. Contohnya yaitu toko online (e-commerce), portal berita, media sosial (Facebook, Instagram), aplikasi perbankan online, sistem manajemen pembelajaran (LMS).

C. Mengetahui Tiga Pilar Utama Web: HTML, CSS, dan JavaScript

Untuk memahami peran ketiganya, mari kita gunakan analogi membangun sebuah rumah. Setiap rumah membutuhkan kerangka, dekorasi, dan fungsionalitas agar bisa dihuni dengan nyaman. HTML (HyperText Markup Language) adalah fondasi dan kerangka dari rumah Anda. Ia mendefinisikan struktur dasar: di mana letak ruang tamu, berapa jumlah kamar tidur, di mana posisi pintu dan jendela. Di dalam web, HTML digunakan untuk menempatkan elemen-elemen seperti judul, paragraf, gambar, dan formulir. Tanpa HTML, tidak ada struktur, hanya ada konten yang berantakan. CSS (Cascading Style Sheets) adalah arsitek interior dan eksterior rumah Anda. Setelah kerangka (HTML) jadi, CSS bertugas untuk memindahkannya. Ia menentukan warna cat dinding, jenis lantai yang digunakan, model furnitur, dan tata letak taman. Di dalam web, CSS mengontrol semua aspek visual: warna teks, jenis huruf, ukuran elemen, gambar latar belakang, dan posisi elemen di halaman. Tanpa CSS, halaman web Anda akan terlihat sangat polos dan tidak menarik. JavaScript (JS) adalah sistem kelistrikan, perpipaan, dan

semua perangkat elektronik di rumah Anda. Ia membuat rumah menjadi "hidup" dan interaktif. Anda bisa menyalakan lampu dengan sakelar, membuka gerbang otomatis, atau menyalakan AC. Di dalam web, JavaScript menambahkan interaktivitas. Ketika Anda mengklik sebuah tombol, melihat *slideshow* gambar yang bergerak, atau mengisi formulir yang menampilkan pesan error secara langsung, itu semua adalah pekerjaan JavaScript. Tanpa JavaScript, halaman web Anda akan menjadi "bisu" dan tidak bisa merespons aksi pengguna.

Untuk memvisualisasikan bagaimana ketiganya saling melengkapi, bayangkan mereka sebagai lapisan-lapisan yang membangun sebuah halaman web. HTML adalah lapisan paling dasar yang wajib ada. CSS bersifat opsional namun sangat direkomendasikan untuk membuat tampilan menarik. CSS tidak bisa berdiri sendiri tanpa HTML. JavaScript juga bersifat opsional, digunakan saat kita membutuhkan interaksi. JavaScript biasanya memanipulasi elemen HTML yang sudah ada. Ketiga teknologi ini bekerja bersama-sama untuk menciptakan pengalaman web yang kita kenal saat ini. HTML menyediakan struktur, CSS menangani tampilan visual, dan JavaScript memberikan interaktivitas. Dalam buku ini, kita akan mempelajari ketiganya secara mendalam.

D. Peralatan yang Dibutuhkan: Browser, Text Editor, dan Local Server

Sebelum memulai petualangan *coding*, ada beberapa peralatan dasar yang perlu kita siapkan di komputer kita. Semua perangkat lunak yang direkomendasikan di sini bersifat gratis dan dapat diunduh oleh siapa saja.

1. Web Browser (Peramban Web)

Ini adalah alat utama kita untuk melihat hasil kerja kita. Browser menerjemahkan kode HTML, CSS, dan JavaScript menjadi halaman web visual yang bisa kita lihat dan interaksikan. Contoh *web browser* adalah Google Chrome atau Mozilla Firefox. Keduanya adalah browser modern yang sangat populer di kalangan developer karena memiliki Developer Tools yang sangat lengkap. Fitur ini memungkinkan kita untuk "mengintip" kode sebuah website, melakukan *debugging* (mencari kesalahan), dan menguji coba perubahan CSS secara langsung. Contoh web browser lainnya adalah Microsoft Edge dan Safari (untuk pengguna macOS).

2. Text Editor (Editor Teks)

Ini adalah tempat kita akan menulis semua kode. Meskipun kita bisa menggunakan aplikasi teks biasa seperti Notepad, *text editor* modern menawarkan banyak fitur yang sangat membantu proses *coding*. Fitur Unggulan dari *text editor* modern adalah *Syntax Highlighting* (memberi warna berbeda pada bagian kode agar mudah dibaca), *Auto-completion* (menyelesaikan kode secara otomatis), dan kemampuan untuk memasang *Extensions* (alat bantu tambahan). Rekomendasi *Text editor* yang bisa digunakan adalah Visual Studio Code (VS Code). Saat ini, VS Code adalah editor teks paling populer di dunia untuk pengembangan web karena ringan, cepat, dan didukung oleh ribuan ekstensi yang sangat berguna. *Text editor* lainnya yang bisa dipakai adalah Sublime Text, Atom, dan lain-lain.

3. Local Server Environment (Lingkungan Server Lokal):

Diperlukan nanti saat kita mulai belajar pengembangan *backend* (Bab 6), yaitu PHP dan basis data. Perangkat lunak ini menginstalasi paket server web (seperti Apache), bahasa pemrograman (PHP), dan server basis data (MySQL/MariaDB) di komputer kita. Ini memungkinkan kita untuk menjalankan dan menguji website dinamis di komputer sendiri seolah-olah sudah online di internet. Aplikasi yang bisa dipakai yaitu XAMPP. XAMPP adalah singkatan dari X (Cross-Platform, artinya bisa berjalan di Windows, macOS, dan Linux), Apache, MariaDB, PHP, dan Perl. Ini adalah paket instalasi yang paling mudah dan populer untuk pemula. Aplikasi lainnya adalah WAMP (untuk Windows), MAMP (untuk macOS), LAMP (untuk Linux).

E. Praktikum

1. Instalasi dan Konfigurasi Text Editor (Visual Studio Code)

a. Mengunduh VS Code

Buka browser Anda dan kunjungi situs web resmi Visual Studio Code di <https://code.visualstudio.com/>. Situs web akan secara otomatis mendeteksi sistem operasi Anda (Windows, macOS, atau Linux). Klik tombol unduh berwarna biru untuk mendapatkan versi yang sesuai.

- b. Menginstal VS Code

Setelah unduhan selesai, jalankan file installer. Ikuti petunjuk instalasi. Disarankan untuk mencentang semua opsi pada bagian "Other", terutama "Add to PATH" (agar VS Code mudah diakses dari terminal) dan "Add 'Open with Code' action" (untuk membuka folder langsung dari file explorer).
 - c. Instalasi Ekstensi Penting

Buka aplikasi Visual Studio Code. Cari ikon Extensions di bilah samping kiri (terlihat seperti 4 kotak, salah satunya terpisah). Di kotak pencarian, ketik nama ekstensi berikut dan klik Install pada masing-masing ekstensi:

 - 1) Live Server: Ekstensi ini sangat berguna. Ia akan membuat server lokal sederhana dan secara otomatis me-*refresh* halaman browser setiap kali Anda menyimpan perubahan pada file HTML atau CSS Anda. Ini akan sangat mempercepat proses development.
 - 2) Prettier - Code formatter: Ekstensi ini akan merapikan kode Anda secara otomatis setiap kali Anda menyimpan file, membuat kode lebih konsisten dan mudah dibaca.
2. Membuat File HTML Pertama ("Hello, World!")
- a. Membuka Folder Proyek

Di VS Code, klik menu File > Open Folder. Arahkan ke folder Latihan-Pemrograman-Web/BAB-01 yang telah Anda buat sebelumnya, lalu klik "Select Folder".
 - b. Membuat File Baru

Di panel Explorer di sebelah kiri, klik ikon "New File" (selembar kertas dengan tanda plus). Beri nama file tersebut index.html dan tekan Enter.
 - c. Menulis Kode HTML

Di dalam file index.html yang masih kosong, ketikkan tanda seru (!) lalu tekan tombol Tab atau Enter. VS Code akan secara otomatis membuatkan struktur dasar HTML untuk Anda

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>

</body>
</html>
```

Ubah `<title>Document</title>` menjadi `<title>Latihan Pertama Saya</title>`. Di dalam tag `<body>`, tambahkan baris berikut: `<h1>Hello, World!</h1>`. Simpan file Anda dengan menekan Ctrl + S (atau Cmd + S di macOS).

d. Melihat Hasil di Browser

Klik kanan pada file `index.html` di panel Explorer VS Code, lalu pilih "Open with Live Server". Browser default Anda akan terbuka secara otomatis dan menampilkan halaman web dengan tulisan "Hello, World!". Coba ubah teks di dalam `<h1>`, simpan, dan lihat browser Anda akan me-*refresh* secara otomatis.

3. Menggunakan Developer Tools di Browser

a. Membuka Developer Tools

Buka website apa pun yang Anda sukai di Google Chrome atau Firefox. Klik kanan di mana saja pada halaman dan pilih "Inspect". Alternatifnya, tekan tombol F12 pada keyboard Anda.

b. Menjelajahi Panel Elements

Panel Developer Tools akan muncul. Perhatikan tab "Elements". Di sini Anda bisa melihat seluruh struktur HTML dari halaman tersebut. Anda bisa mengklik panah kecil di samping setiap elemen untuk membuka dan melihat elemen-elemen di dalamnya.

c. Menginspeksi Elemen Spesifik

Di pojok kiri atas panel Developer Tools, klik ikon "Select an element in the page to inspect it" (biasanya berbentuk kotak dengan panah). Gerakkan mouse Anda ke atas halaman web. Anda akan melihat setiap elemen yang Anda sorot akan ditandai dengan warna biru. Klik pada sebuah judul berita atau gambar. Perhatikan bahwa

di panel "Elements", baris kode HTML yang sesuai untuk elemen tersebut akan langsung disorot.

d. Melakukan Perubahan Sementara

Dengan baris kode HTML yang tersorot, klik dua kali pada teks di dalamnya. Anda bisa mengedit teks tersebut secara langsung. Tekan Enter, dan lihat perubahannya di halaman web. Di sisi kanan panel Developer Tools (biasanya di bawah tab "Styles"), Anda bisa melihat semua aturan CSS yang diterapkan pada elemen tersebut. Coba hilangkan centang pada salah satu properti (misalnya `color` atau `background-color`) dan lihat efeknya. **Ingat:** Semua perubahan ini hanya sementara dan akan hilang jika Anda me-*refresh* halaman. Ini adalah alat yang sangat aman untuk bereksperimen.

Latihan

1. Soal Teori (Esai Singkat)

- a. Jelaskan dengan kata-kata Anda sendiri perbedaan mendasar antara Internet dan World Wide Web (WWW). Berikan satu contoh layanan yang berjalan di atas Internet selain WWW.
- b. Gambarkan proses *request-response cycle* yang terjadi saat Anda membuka website `www.google.com` di browser Anda. Sebutkan setidaknya 4 komponen utama yang terlibat (misalnya: Client, DNS, Server, dll.) dan jelaskan peran masing-masing.
- c. Sebutkan tiga contoh website statis dan tiga contoh website dinamis yang sering Anda kunjungi. Jelaskan secara singkat mengapa Anda mengkategorikannya demikian.
- d. Jelaskan kembali analogi "Membangun Rumah" untuk HTML, CSS, dan JavaScript menggunakan gaya bahasa Anda sendiri. Apa yang akan terjadi jika sebuah website hanya dibuat menggunakan HTML tanpa CSS?

2. Tugas Praktik

a. Persiapan Lingkungan Kerja

Pastikan Anda telah berhasil menginstal Visual Studio Code di komputer Anda. Buat sebuah folder baru di komputer Anda dengan nama `Latihan-Pemrograman-Web`. Di dalam folder tersebut, buat

lagi folder baru bernama BAB-01. Semua file untuk latihan bab ini akan disimpan di sini.

b. Halaman Biodata Diri

- 1) Di dalam folder BAB-01, buat sebuah file baru bernama `biodata.html`.
- 2) Isi file tersebut dengan struktur dasar HTML.
- 3) Di dalam `<body>`, buatlah konten yang menampilkan biodata diri Anda, yang harus mencakup:
 - a) Judul utama (`<h1>`) berisi nama lengkap Anda.
 - b) Sub-judul (`<h2>`) bertuliskan "Biodata Diri".
 - c) Beberapa paragraf (`<p>`) yang berisi: Nomor Induk Mahasiswa (NIM), Jurusan, dan Hobi Anda.
 - d) Sebuah tautan/link (`<a>`) yang mengarah ke website favorit Anda (misalnya, <https://www.google.com>).
 - e) Simpan file tersebut dan buka menggunakan browser (Google Chrome/Firefox) untuk melihat hasilnya.

c. Investigasi dengan Developer Tools

- 1) Buka salah satu portal berita favorit Anda (misalnya, Detik.com, Kompas.com).
- 2) Klik kanan di mana saja pada halaman tersebut dan pilih "Inspect" atau "Inspect Element" untuk membuka Developer Tools.
- 3) Cari dan aktifkan *tool* "select element" (biasanya ikon panah di pojok kiri atas panel Developer Tools).
- 4) Gunakan *tool* tersebut untuk mengklik judul berita utama di halaman tersebut. Perhatikan, baris kode HTML yang sesuai akan disorot di panel "Elements".
- 5) Klik dua kali pada teks judul berita di dalam panel "Elements" dan ubah isinya menjadi nama Anda. Perhatikan perubahan yang terjadi secara langsung di halaman web. (Catatan: Perubahan ini hanya bersifat sementara dan akan hilang jika halaman di-*refresh*).

BAB 2

HTML - KERANGKA DASAR HALAMAN WEB

A. Anatomi Dokumen HTML

Setiap halaman web yang valid mengikuti sebuah struktur standar. Struktur ini, sering disebut sebagai boilerplate, adalah fondasi wajib yang memastikan browser dapat menginterpretasikan konten Anda dengan benar.

```
<!DOCTYPE html>
<html lang="id">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Judul Halaman Web</title>
</head>
<body>
  <!-- Semua konten yang terlihat oleh pengguna diletakkan di sini -->
</body>
</html>
```

Bagian dari Dokumen HTML adalah sebagai berikut:

1. `<!DOCTYPE html>` merupakan deklarasi Tipe Dokumen. Ini bukan tag HTML, melainkan sebuah instruksi untuk browser tentang versi HTML yang digunakan. `<!DOCTYPE html>` menandakan bahwa halaman ini menggunakan standar HTML5, standar terbaru dan yang paling umum saat ini.
2. `<html lang="id">` merupakan elemen akar (root). Semua elemen lain harus berada di dalam tag ini. Atribut `lang="id"` adalah praktik terbaik untuk aksesibilitas dan SEO, yang memberitahu browser bahwa bahasa utama konten di halaman ini adalah Bahasa Indonesia.
3. `<head>` merupakan Bagian "Kepala". Konten di dalam `<head>` tidak ditampilkan di jendela utama browser. Bagian ini berisi metadata atau informasi tentang dokumen HTML itu sendiri.

4. `<meta charset="UTF-8">` merupakan deklarasi set karakter yang digunakan. UTF-8 adalah standar universal yang mencakup hampir semua karakter dan simbol dari berbagai bahasa di dunia.
5. `<meta name="viewport" ...>` merupakan pengaturan yang sangat penting untuk desain web responsif. Ini memberitahu browser untuk mengatur lebar halaman sesuai dengan lebar layar perangkat dan menetapkan tingkat zoom awal ke 100%.
6. `<title>` digunakan untuk menentukan judul halaman yang akan muncul di tab browser, pada hasil pencarian Google, dan saat halaman di-bookmark.
7. `<body>` merupakan bagian "Badan". Di sinilah semua konten visual halaman web—seperti teks, gambar, tautan, tabel, dan formulir—ditempatkan.

B. Konsep Inti: Elemen, Tag, dan Atribut

Tag adalah penanda awal dan akhir dari sebuah elemen. Tag ditulis dengan kurung sudut. Contoh: `<h1>` adalah tag pembuka dan `</h1>` adalah tag penutup. Terdapat juga tag kosong atau self-closing tag yang tidak memerlukan tag penutup, seperti `` dan `
`.

Elemen merupakan unit pembangun halaman web. Sebuah elemen terdiri dari tag pembuka, konten, dan tag penutup. Contoh: `<h1>Selamat Datang</h1>` adalah sebuah elemen heading.

Atribut memberikan informasi tambahan atau konfigurasi pada sebuah elemen. Atribut ditulis di dalam tag pembuka dengan format `nama="nilai"`. Contoh: ``. Elemen `` ini memiliki dua atribut: `src` untuk menentukan sumber gambar dan `alt` untuk teks alternatif.

C. Elemen Teks Fundamental

Ada beberapa elemen dalam membentuk teks pada HTML yang sering dipakai, yaitu:

1. Heading (`<h1>` - `<h6>`) digunakan untuk mendefinisikan judul. Penting untuk menggunakan heading secara hierarkis (`<h1>` untuk judul utama, `<h2>` untuk sub-judul, dst.) demi struktur dokumen yang baik dan SEO.

DAFTAR PUSTAKA

- Duckett, J. (2011). HTML and CSS: Design and Build Websites. John Wiley & Sons.
- Duckett, J. (2014). JavaScript and JQuery: Interactive Front-End Web Development. John Wiley & Sons.
- Haverbeke, M. (2018). Eloquent JavaScript (3rd ed.). No Starch Press.
- MDN Web Docs (2025). Web technology for developers. Mozilla Developer Network. Diakses dari <https://developer.mozilla.org/>
- Nixon, R. (2021). Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5 (6th ed.). O'Reilly Media.
- PHP.net (2025). PHP Manual. The PHP Group. Diakses dari <https://www.php.net/manual/en/>
- W3Schools (2025). Web Development Tutorials. Diakses dari <https://www.w3schools.com/>

GLOSARIUM

API (Application Programming Interface): Sekumpulan aturan dan protokol yang memungkinkan berbagai aplikasi perangkat lunak berkomunikasi satu sama lain.

Atribut: Informasi tambahan yang diberikan pada tag HTML untuk memodifikasi atau memberikan fungsionalitas pada elemen.

Backend: Bagian dari aplikasi web yang berjalan di server, bertanggung jawab atas logika bisnis, pemrosesan data, dan interaksi dengan basis data.

Basis Data (Database): Kumpulan data terstruktur yang disimpan secara elektronik.

Box Model: Konsep di CSS yang menganggap setiap elemen HTML sebagai sebuah kotak dengan properti: content, padding, border, dan margin.

Client: Perangkat lunak atau perangkat keras yang mengakses layanan yang disediakan oleh server (misalnya, browser web).

CRUD: Singkatan dari Create, Read, Update, Delete; empat operasi dasar yang dapat dilakukan pada data dalam basis data.

CSS (Cascading Style Sheets): Bahasa yang digunakan untuk mendeskripsikan tampilan dan format dari dokumen yang ditulis dalam bahasa markup seperti HTML.

DNS (Domain Name System): Sistem yang menerjemahkan nama domain yang mudah dibaca manusia (misal: www.google.com) menjadi alamat IP yang dimengerti komputer.

DOM (Document Object Model): Representasi struktur hierarkis dari dokumen HTML dalam bentuk objek, yang memungkinkan JavaScript untuk mengakses dan memanipulasi elemen halaman.

Elemen: Komponen individual dari halaman HTML, yang terdiri dari tag pembuka, konten, dan tag penutup.

Event: Aksi atau kejadian yang terjadi di dalam browser, seperti klik mouse, penekanan tombol, atau selesainya pemuatan halaman, yang dapat dideteksi dan ditanggapi oleh JavaScript.

Flexbox: Model layout di CSS yang dirancang untuk menata elemen dalam satu dimensi (baris atau kolom).

Framework CSS: Kumpulan file CSS siap pakai yang menyediakan komponen dan utilitas untuk mempercepat pengembangan web (contoh: Bootstrap, Tailwind CSS).

Frontend: Bagian dari aplikasi web yang berinteraksi langsung dengan pengguna, yang dirender di browser (HTML, CSS, JavaScript).

Fungsi: Blok kode yang dapat digunakan kembali yang dirancang untuk melakukan tugas tertentu dalam JavaScript atau PHP.

Grid: Model layout di CSS yang dirancang untuk menata elemen dalam dua dimensi (baris dan kolom).

Hosting: Layanan yang menyediakan ruang pada server untuk menyimpan file-file website agar dapat diakses melalui internet.

HTML (HyperText Markup Language): Bahasa markup standar yang digunakan untuk membuat struktur dan konten halaman web.

HTTP (Hypertext Transfer Protocol): Protokol yang mengatur komunikasi antara client (browser) dan server.

JavaScript (JS): Bahasa pemrograman sisi klien yang digunakan untuk membuat halaman web menjadi interaktif.

Media Query: Fitur CSS yang memungkinkan penerapan gaya yang berbeda berdasarkan karakteristik perangkat, seperti lebar layar, untuk menciptakan desain yang responsif.

PHP (Hypertext Preprocessor): Bahasa pemrograman sisi server yang populer digunakan untuk membuat halaman web dinamis dan berinteraksi dengan basis data.

Primary Key (Kunci Utama): Kolom dalam tabel basis data yang secara unik mengidentifikasi setiap baris.

RDBMS (Relational Database Management System): Sistem manajemen basis data yang mengorganisir data ke dalam tabel-tabel yang saling berhubungan.

Responsif (Desain Web): Pendekatan desain yang membuat tampilan website dapat beradaptasi dengan baik di berbagai ukuran layar perangkat.

Selector: Pola dalam CSS yang digunakan untuk memilih dan menargetkan elemen HTML yang ingin diberi gaya.

Semantik (HTML): Penggunaan tag HTML sesuai dengan maknanya (misalnya, <nav> untuk navigasi) untuk meningkatkan aksesibilitas dan SEO.

Server: Komputer atau sistem komputer yang menyediakan data, layanan, atau sumber daya untuk komputer lain (client) melalui jaringan.

Session: Mekanisme yang digunakan di sisi server untuk menyimpan informasi tentang pengguna di beberapa permintaan, memungkinkan fungsionalitas seperti login.

SQL (Structured Query Language): Bahasa standar yang digunakan untuk mengelola dan memanipulasi data dalam RDBMS.

SQL Injection: Jenis serangan keamanan di mana penyerang menyisipkan kode SQL berbahaya ke dalam input untuk memanipulasi query basis data.

Tag: Penanda dalam HTML yang ditulis dalam kurung sudut (<>) untuk mendefinisikan elemen.

URL (Uniform Resource Locator): Alamat unik yang digunakan untuk mengidentifikasi sumber daya di internet.

Variabel: Nama simbolis yang digunakan untuk menyimpan nilai data dalam bahasa pemrograman.

Di era digital yang serba terhubung, kemampuan untuk membangun dan memahami teknologi di balik situs web adalah sebuah literasi fundamental. Buku ajar "Pemrograman Web: Konsep & Praktik" ini dirancang secara sistematis untuk membimbing mahasiswa dari nol hingga mampu membangun aplikasi web dinamis yang fungsional.

Perjalanan Anda akan dimulai dari fondasi paling dasar di Bab 1, dengan pengenalan komprehensif tentang cara kerja internet dan World Wide Web. Selanjutnya, buku ini akan membedah tiga pilar utama pengembangan *front-end*:

- **HTML:** Pelajari cara membangun kerangka dan struktur konten halaman web Anda.
- **CSS:** Kuasai seni mendesain dan mempercantik tampilan visual, mulai dari tipografi, warna, hingga *layout* modern menggunakan Flexbox dan Grid.
- **JavaScript:** Hidupkan halaman web Anda dengan menambahkan interaktivitas, memanipulasi elemen secara dinamis, dan merespons aksi pengguna melalui DOM.

Setelah menguasai sisi klien (*client-side*), buku ini akan membawa Anda menyelami dunia *backend* (*server-side*). Anda akan diperkenalkan dengan konsep basis data relasional (RDBMS), belajar berkomunikasi dengan basis data menggunakan SQL, dan memahami arsitektur di balik aplikasi web dinamis. Dengan PHP sebagai bahasa pilihan, Anda akan belajar memproses data formulir dan menghubungkan aplikasi Anda ke basis data MySQL/MariaDB.

Setiap bab disusun dengan pendekatan praktis yang mencakup teori, contoh kode yang relevan, sesi praktik langsung, dan soal latihan untuk menguji pemahaman. Puncaknya adalah proyek studi kasus di Bab 7, di mana Anda akan mengintegrasikan semua pengetahuan yang telah dipelajari untuk membangun sebuah aplikasi blog sederhana dari awal, lengkap dengan fungsionalitas CRUD (Create, Read, Update, Delete) dan sistem autentikasi dasar.

Baik Anda seorang pemula yang baru memulai atau mahasiswa yang ingin memperkuat fondasi pemrograman web, buku ini adalah panduan lengkap yang akan membekali Anda dengan keterampilan esensial untuk berkarya di dunia digital.



IKAPI

CV. Tahta Media Group

Surakarta, Jawa Tengah

Web : www.tahtamedia.com

Ig : tahtamedia group

Telp/WA : +62 896-5427-3996

