



UNIVERSITAS BHAYANGKARA JAKARTA RAYA
FAKULTAS ILMU KOMPUTER

Kampus I: Jl. Harsono RM No. 67, Ragunan, Pasar Minggu, Jakarta Selatan, 12550
Telepon: (021) 27808121 – 27808882
Kampus II: Jl. Raya Perjuangan, Marga Mulya, Bekasi Utara, Jawa Barat, 17142
Telepon: (021) 88955882, Fax.: (021) 88955871
Web: fasilkom.ubharajaya.ac.id, E-mail: fasilkom@ubharajaya.ac.id

SURAT TUGAS

Nomor: ST/312/IV/2025/FASILKOM-UBJ

Menimbang : Bahwa dalam rangka mewujudkan Tri Dharma Perguruan Tinggi untuk Dosen di Universitas Bhayangkara Jakarta Raya maka dihimbau untuk melakukan Penelitian. Sehubungan dengan hal tersebut, dipandang perlu mengeluarkan Surat Tugas.

Mengingat : 1. Undang-Undang No. 14 Tahun 2005 tentang Guru dan Dosen;
2. Permendikbud No. 3 Tahun 2020 tentang Standar Nasional Pendidikan Tinggi;
3. Kalender Akademik Universitas Bhayangkara Jakarta Raya Tahun Akademik 2024/2025.

DITUGASKAN

Kepada : Personil yang namanya tercantum dalam Surat Tugas ini.

NO.	NAMA	NIDN	JABATAN
1.	Fried Sinlae, ST., M.Kom..	0318039303	Dosen Tetap Prodi Informatika
2.	Muhammad Yasir, S.Si., M.Kom.	0317129002	Dosen Tetap Prodi Informatika

Untuk : 1. Membuat Buku Ajar dengan judul **“MAHIR PEMROGRAMAN WEB Teori dan Praktik Langsung”** yang diterbitkan oleh Lingkar Edukasi Indonesia, Cetakan Pertama: Mei 2025, ISBN: 978-634-7159-85-4.
2. Melaporkan hasil pelaksanaan kepada Dekan Fakultas Ilmu Komputer Universitas Bhayangkara Jakarta Raya.
3. Melaksanakan tugas ini dengan penuh tanggung jawab.

Ditetapkan di : Jakarta
Pada tanggal : 21 April 2025

DEKAN FAKULTAS ILMU KOMPUTER
UNIVERSITAS BHAYANGKARA JAKARTA RAYA





Lingkar Edukasi
Indonesia

CV. LINGKAR EDUKASI INDONESIA

PENERBIT BUKU DAN JURNAL INDONESIA

No sertifikat : 363/V/SP/LEI/2025

Sertifikat

SERTIFIKAT INI DENGAN BANGGA DIBERIKAN KEPADA

Fried Sinlae, S.T., M.Kom

PENULIS LINGKAR EDUKASI INDONESIA

Buku dengan judul : **MAHIR PEMROGRAMAN WEB Teori dan Praktik
Langsung**

No ISBN :978-634-7159-85-4



LIRA MUHARDI S.P

Lingkar Edukasi
Indonesia

DIREKTUR UTAMA

MAHIR

Pemrograman web

Teori dan Praktik Langsung

MAHIR PEMROGRAMAN WEB

Teori dan Praktik Langsung

Penulis:

Fried Sinlae, S.T., M.Kom

Muhammad Yasir, S.Si., M.Kom



Lingkar Edukasi
Indonesia

LINGKAR EDUKASI INDONESIA

MAHIR PEMROGRAMAN WEB
Teori dan Praktik Langsung

Penulis:

Fried Sinlae, S.T., M.Kom
Muhammad Yasir, S.Si., M.Kom

Editor: Weni Yuliani, S.Si., M.M., C.Ed

Penyunting: Fera Malta, S.TP

Desain Sampul dan Tata Letak: Neza Sartika

Diterbitkan oleh:

Lingkar Edukasi Indonesia
Anggota IKAPI No. 058/SBA/2024
Kolam Janiah, Nagari Kudu Ganting
Kec. V Koto Timur, Kabupaten Padang Pariaman
Email: lingkaredukasiindonesia.id@gmail.com
Website: www.lingkaredukasiindonesia.com

ISBN: 978-634-7159-85-4

Cetakan pertama, Mei 2025

© Hak cipta dilindungi undang-undang.

Dilarang keras memperbanyak, memfotokopi, Sebagian atau
seluruh isi buku tanpa izin tertulis dari penerbit.

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa atas rahmat dan karunia-Nya, sehingga buku berjudul *Mahir Pemrograman Web: Teori dan Praktik Langsung* ini dapat disusun dan diselesaikan dengan baik. Buku ini hadir sebagai jawaban atas kebutuhan pembelajaran pemrograman web yang terstruktur, mudah dipahami, dan aplikatif, baik untuk pemula maupun pembaca yang ingin memperdalam kemampuan teknis di bidang pengembangan web.

Di dalam buku ini, pembaca akan disajikan materi dasar hingga lanjutan yang mencakup teori penting seputar teknologi web, dilengkapi dengan praktik langsung untuk memperkuat pemahaman konsep. Setiap bab dirancang agar pembaca tidak hanya menguasai sintaks dan struktur kode, tetapi juga memahami logika pengembangan aplikasi web secara menyeluruh. Dengan pendekatan *learning by doing*, buku ini diharapkan dapat menjadi panduan praktis yang efektif dalam proses belajar mandiri maupun dalam kegiatan pembelajaran formal.

Penulis menyadari bahwa penyusunan buku ini tidak lepas dari keterbatasan. Oleh karena itu, saran dan kritik yang membangun sangat diharapkan untuk perbaikan pada edisi selanjutnya. Akhir kata, semoga buku ini dapat memberikan manfaat bagi semua pembaca, khususnya mahasiswa, dosen, pengembang pemula, maupun siapa saja yang ingin menguasai keterampilan pemrograman web secara bertahap dan terarah.

Jakarta, April 2025

Penulis

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
DAFTAR GAMBAR	iii
DAFTAR TABEL	v
BAB 1 MENGENAL PEMROGRAMAN WEB DAN PENGELOLAAN DATA DI ERA DIGITAL.....	1
A. Pemrograman Web: Bahasa Pemrograman, Sistem, dan Pengelolaan Data.....	1
B. Rootmap Pemrograman Web: Panduan Pembelajaran Bertahap	3
BAB 2 PEMROGRAMAN WEB: MEMAHAMI HTML, SEJARAH, DAN PERKEMBANGANNYA	5
A. HTML	5
B. Sejarah HTML	7
BAB 3 TEKNIK IMPLEMENTASI PEMROGRAMAN WEB..	9
A. Perintah Dasar HTML.....	9
B. CSS	32
C. JavaScript	40
D. MySQL	92
E. <i>Create Read Update dan Delete</i>	99
BAB 4 PENUTUP	110
DAFTAR PUSTAKA	112
BIODATA PENULIS.....	114

DAFTAR GAMBAR

Gambar 1.1 Root Map Pemrograman Web.....	3
Gambar 3.1 Output Belajar Html.html	10
Gambar 3.2 Output br.html	11
Gambar 3.3 Output paragraph.html.....	12
Gambar 3.4 Output hr.html	13
Gambar 3.5 Output font.html.....	14
Gambar 3.6 Output font2.html	15
Gambar 3.7 Output heading.html.....	17
Gambar 3.8 Output tebal.html.....	18
Gambar 3.9 Output miring.html	19
Gambar 3.10 Output garisbawah.html.....	20
Gambar 3.11 Output gambar.html.....	21
Gambar 3.12 Output tabel.html	23
Gambar 3.13 Output link.html	24
Gambar 3.14 Output div.html.....	26
Gambar 3.15 Output span.html.....	27
Gambar 3.16 Output ol.html	28
Gambar 3.17 Output ul.html	29
Gambar 3.18 Output form-lengkap.html.....	32
Gambar 3.19 Output css1.html	34
Gambar 3. 20 Output css2.html	35
Gambar 3.21 Output css3.html	36
Gambar 3.22 Output css4.html	37
Gambar 3.23 Output css5.html	40
Gambar 3.24 Output hello_world.html	41
Gambar 3.25 Output hello_world.html	42
Gambar 3.26 Output hello_world1.html.....	43
Gambar 3.27 Output js2.html	44
Gambar 3.28 Output js3.html	45
Gambar 3.29 XAMPP	46
Gambar 3.30 Output hello.php.....	48

Gambar 3.31 Output variabel.php	50
Gambar 3.32 Output tipe.php.....	51
Gambar 3.33 Output sambung.php	53
Gambar 3.34 Output konstanta.php.....	54
Gambar 3.35 Output operator.php.....	59
Gambar 3.36 Output komentar.php.....	60
Gambar 3.37 Output if.php	63
Gambar 3.38 Output if_else.php	64
Gambar 3.39 Output if_else2.php.....	65
Gambar 3.40 Output if_var.php.....	66
Gambar 3.41 Output switch.php	67
Gambar 3.42 Output for.php.....	71
Gambar 3.43 Output while.php.....	72
Gambar 3.44 Output dowhile.php	73
Gambar 3.45 Output break.php	74
Gambar 3.46 Output input1.php	77
Gambar 3.47 Output input2.php	79
Gambar 3.48 Output aksi_input2.php	79
Gambar 3.49 Output input3.php	81
Gambar 3.50 Output aksi_input3.php	81
Gambar 3.51 Output form.php.....	86
Gambar 3.52 Output aksi_form.php.....	87
Gambar 3.53 Output form_upload.php	91
Gambar 3.54 Output aksi_upload.php.....	91
Gambar 3.55 Berhasil Upload Gambar	91
Gambar 3.56 Output tamu.php	102
Gambar 3.57 Output tamu_input.php.....	104
Gambar 3.58 Output aksi_tamu_input.php.....	104
Gambar 3.59 Output tamu_edit.php.....	106
Gambar 3.60 Output aksi_tamu_edit.php.....	107
Gambar 3.61 Output aksi_tamu_hapus.php	107
Gambar 3.62 Output tamu.php pencarian kata 'sin'	109
Gambar 3.63 Output tamu_cari.php hasil pencarian kata 'sin'	109

DAFTAR TABEL

Tabel 2.1 Versi HTML	8
Tabel 3.1 Variabel	49
Tabel 3.2 Tipe Data.....	50
Tabel 3.3 PHP Arithmetic Operators	54
Tabel 3.4 PHP Assignment Operators.....	55
Tabel 3.5 PHP Comparison Operators	55
Tabel 3.6 PHP Logical Operators.....	56
Tabel 3.7 PHP String Operators.....	56
Tabel 3.8 PHP Array Operators	57
Tabel 3.9 PHP Conditional Assignment Operators	57

BAB 1

MENGENAL PEMROGRAMAN WEB DAN PENGELOLAAN DATA DI ERA DIGITAL

A. Pemrograman Web: Bahasa Pemrograman, Sistem, dan Pengelolaan Data

Pemrograman komputer adalah kegiatan interaktif yang mencakup penulisan dan pengeditan kode sumber (*source code*) untuk membentuk sebuah program. Proses penyuntingan ini mencakup pengujian, analisis, perbaikan kesalahan, pengoptimalan algoritma, dan penyesuaian struktur kode agar lebih efisien (Saragih, 2018). Tujuan dari kegiatan pemrograman adalah menghasilkan program yang mampu menjalankan instruksi sesuai perintah yang telah ditentukan. Program yang dihasilkan bisa berupa aplikasi, situs web, atau perangkat lunak khusus. Salah satu contoh implementasinya adalah website yang menawarkan komponen seperti halaman berita, informasi alumni yang mudah diakses, serta informasi karier (Sinlae & Yasir, 2024). Seiring dengan kemajuan pesat di era digital saat ini, keberadaan website telah menjadi sarana penting bagi individu maupun organisasi untuk memperluas jangkauan, meningkatkan eksistensi, serta menyampaikan informasi secara efektif kepada publik. Melalui website, suatu entitas bisa memperkenalkan identitasnya, menjelaskan produk atau layanan yang dimiliki, dan berinteraksi langsung dengan pengunjung (Sinlae et al., 2024).

Bahasa pemrograman adalah serangkaian perintah standar yang berfungsi untuk mengarahkan komputer dalam menjalankan tugas atau fungsi tertentu. Terdapat berbagai jenis bahasa pemrograman, dan masing-masing memiliki kelebihan serta keterbatasan tersendiri. Beberapa contoh bahasa pemrograman yang umum dipakai antara lain PHP, JavaScript, Java, dan sebagainya (Saragih, 2018).

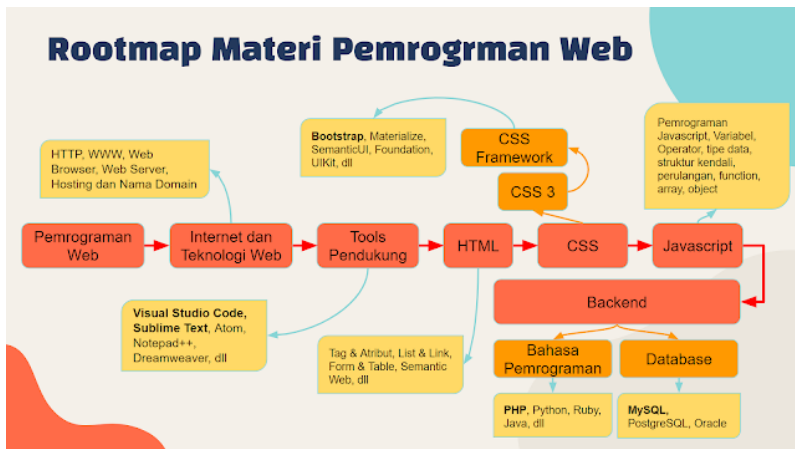
PHP (*Hypertext Preprocessor*) digunakan sebagai bahasa pemrograman server-side yang bekerja bersama HTML dalam pembuatan halaman web. PHP berperan dalam membangun halaman web yang bersifat dinamis, yaitu halaman yang dibuat secara otomatis ketika diminta oleh pengguna, sehingga konten yang ditampilkan selalu diperbarui. Semua perintah PHP diproses di server tempat website disimpan (Anhar, 2010). PHP dapat dijalankan di berbagai sistem operasi, seperti Linux, Windows, dan Unix, karena memiliki kemampuan lintas platform. Salah satu kegunaannya adalah untuk memenuhi kebutuhan dasar dalam pembuatan aplikasi berbasis web. Selain itu, PHP juga mempunyai banyak sumber referensi dalam bentuk bahasa pemrograman lainnya.

Database adalah tempat untuk menyimpan data yang memiliki keterkaitan satu sama lain dan disusun secara rapi tanpa duplikasi data yang tidak diperlukan. Data dalam database disimpan, diatur, dan diolah agar menghasilkan informasi yang akurat dan efisien dalam penggunaan ruang penyimpanan (Kuswayatno, 2007). Database juga bisa diartikan sebagai kumpulan tabel data yang saling terhubung dan mampu menyusun data secara cepat dan teratur. MySQL merupakan bagian dari sistem manajemen database yang cukup populer dan memakai perintah-perintah utama dalam SQL (*Structured Query Language*).

MySQL bersifat open source dan tersedia dalam dua jenis lisensi, yaitu gratis (*Free Software*) dan berbagi pakai (*Shareware*). Karena menggunakan lisensi *General Public License* (GPL), MySQL dapat digunakan secara bebas tanpa harus membayar. Menurut Yusril Helmi Setyawan (2020) dan Rusli (2019), MySQL umum dimanfaatkan sebagai media penyimpanan seluruh data dalam aplikasi berbasis web yang sedang dikaji.

B. Rootmap Pemrograman Web: Panduan Pembelajaran Bertahap

Urutan materi dalam rootmap pemrograman web ini disusun dengan baik dan akan lebih efektif jika dipelajari sesuai dengan alur yang telah ditentukan. Selanjutnya, silakan perhatikan gambar berikut:



Gambar 1.1 Root Map Pemrograman Web

Setelah melihat dan memperhatikan gambar Rootmap Materi Pemrograman Web di atas, silakan coba ikuti langkah-langkah pembelajaran dengan mempelajarinya

secara berurutan. Pendekatan bertahap ini dirancang untuk membantu memahami konsep pemrograman web secara sistematis dan menyeluruh.

Dimulai dari Internet dan Teknologi Web, pembelajar akan dikenalkan pada dasar-dasar cara kerja internet dan bagaimana web berfungsi. Selanjutnya, Tools Pendukung akan membahas perangkat atau software yang dibutuhkan dalam proses pengembangan web.

Setelah itu, HTML akan dipelajari sebagai fondasi utama dalam membuat struktur halaman web, dilanjutkan dengan CSS untuk mempercantik tampilan dan desain halaman tersebut.

Tahapan berikutnya adalah JavaScript, yang berguna untuk menambahkan interaktivitas pada website. Kemudian, dilanjutkan dengan PHP, yaitu bahasa pemrograman sisi server yang memungkinkan pembuatan halaman dinamis.

MySQL, yang merupakan sistem manajemen basis data populer untuk pengolahan dan penyimpanan data dalam aplikasi web, akan dipelajari pada tahap akhir pembelajaran.

Dengan mengikuti alur ini secara berurutan, pemahaman dalam membangun dan mengelola website akan menjadi lebih kuat dan terstruktur.

BAB 2

PEMROGRAMAN WEB: MEMAHAMI HTML, SEJARAH, DAN PERKEMBANGANNYA

A. HTML

Untuk menyusun dan merancang tampilan halaman situs, digunakan HTML (*Hypertext Markup Language*) sebagai bahasa markup utama. Bahasa ini berisi beragam kode yang membentuk struktur dari sebuah situs web. HTML tersusun dari gabungan teks dan tanda khusus (symbol) yang terdapat pada dokumen. Dalam pembuatan file HTML, terdapat aturan atau struktur tertentu yang perlu diikuti, yang telah tercantum sesuai dengan standar kode internasional, yaitu ASCII (American Standard Code for Information Interchange) (Yusuf, 2022). HTML memungkinkan pengguna untuk mengatur berbagai elemen seperti heading, paragraf, gambar, dan link, yang dapat ditampilkan kepada banyak orang melalui halaman web. Browser seperti Internet Explorer, Chrome, dan Mozilla Firefox digunakan agar halaman web dapat diakses secara luas oleh pengguna. Untuk menguasai pemrograman web secara menyeluruh, ada tiga unsur utama yang perlu dipelajari dalam pembuatan halaman web, yaitu:

1. Untuk membangun struktur konten dalam laman web, digunakan HTML sebagai bahasa markup dasarnya. Melalui HTML, kita dapat menentukan berbagai elemen seperti teks, gambar, tabel, link, dan form, serta

mengorganisasikan konten dalam hierarki yang mudah dimengerti oleh browser. Tanpa HTML, halaman web tidak akan memiliki struktur yang jelas atau bisa dipahami.

2. Untuk membuat tampilan halaman web menjadi lebih menarik, CSS digunakan setelah struktur dasarnya dibentuk menggunakan HTML. Dengan CSS, kita dapat mengontrol komponen visual termasuk warna, ukuran, tata letak, jenis huruf, dan komponen desain lainnya. CSS memungkinkan pemisahan antara konten dan presentasi, yang membuat pengelolaan desain halaman menjadi lebih fleksibel dan efisien. Dengan CSS, kita dapat membuat tampilan web yang responsif dan menarik, berdasarkan keinginan pengguna.
3. Untuk menciptakan interaksi pada halaman web, digunakan bahasa pemrograman bernama JavaScript. Menggunakan JavaScript, kita bisa membuat halaman web yang tidak hanya statis, tetapi juga dinamis. JavaScript digunakan untuk menambahkan elemen interaktif seperti form validation, animasi, efek tampilan, serta pengelolaan konten secara real-time tanpa harus memuat ulang halaman. JavaScript sangat penting dalam menciptakan pengalaman pengguna yang lebih kaya dan menarik di situs web.

HTML dapat diibaratkan sebagai struktur dasar yang membentuk kerangka utama dari halaman web, seperti tulang dan sendi yang membangun tubuh. Sementara itu, CSS berfungsi untuk memberikan tampilan luar halaman web, seperti kulit, rambut, atau aksesoris, yang menyelimuti struktur tersebut, termasuk elemen-elemen desain seperti pakaian, sepatu, dan topi. Di sisi lain, JavaScript berperan sebagai penggerak dari elemen-elemen

tersebut, layaknya motor yang menggerakkan sendi-sendi tubuh. Jika kita menginginkan elemen interaktif, seperti popup yang muncul ketika tombol diklik, countdown waktu, slider gambar, zoom gambar, dan lainnya, maka JavaScript lah yang menangani tugas tersebut, memberikan dinamika dan interaktivitas pada halaman web.

B. Sejarah HTML

Tim Berners-Lee adalah ilmuwan yang menciptakan HTML pada tahun 1991, dengan tujuan awal untuk mempermudah para ilmuwan dalam berbagi dokumen. Setahun setelahnya, HTML mulai dapat digunakan oleh masyarakat umum. Keberhasilan ini tidak dapat dipisahkan dari peran penting World Wide Web (WWW), yang bekerja bersama HTML untuk memungkinkan penyebaran informasi secara luas. Tidak lama kemudian, HTML berkembang pesat dan kini dimanfaatkan oleh lebih dari 92,3% situs web di dunia. Perjalanan HTML yang panjang ini penuh dengan evolusi yang menarik dapat dilihat pada tabel berikut.

Tabel 2.1 Versi HTML

Tahun	Versi	Deskripsi Perkembangan
1991	HTML 1	Dirancang untuk mempermudah para ilmuwan dalam mengakses dan berbagi dokumen antar komputer yang terhubung.
1995	HTML 2	Mendukung tampilan data dan formulir yang bisa langsung diedit, menjadi awal dari web interaktif.
1997	HTML 3	Menambahkan elemen <i>figure</i> untuk mendukung penyematan gambar di halaman web.
1998–1999	HTML 4.0	Menyediakan elemen-elemen dasar seperti gambar, tabel, teks, tautan, dan formulir.
1999–2000	HTML 4.1	Sudah dipublikasikan secara internasional dan kompatibel dengan berbagai peramban.
2014	HTML 5	Menggabungkan HTML, CSS, dan JavaScript untuk tampilan dinamis serta mendukung media seperti video/audio.

Hingga kini, HTML5 merupakan versi terbaru dengan fitur yang paling mutakhir. Penggunaannya pun berkembang sangat pesat. Saat ini, tercatat sekitar 87,4% situs web telah mengadopsi HTML5 sebagai standarnya.

BAB 3

TEKNIK IMPLEMENTASI PEMROGRAMAN WEB

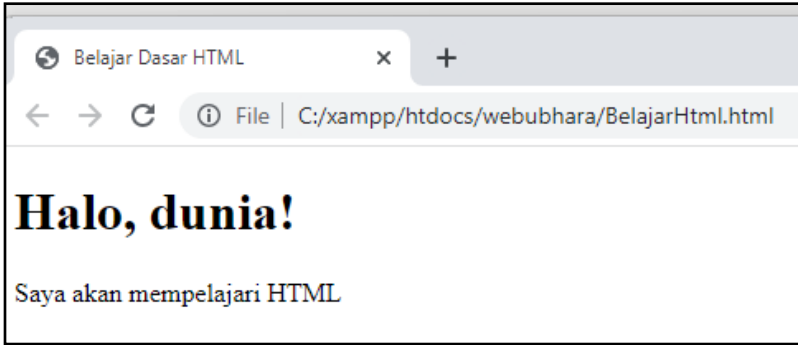
A. Perintah Dasar HTML

Buatlah sebuah folder baru dengan nama “BelajarHtml” di drive D: atau E: sebagai tempat menyimpan seluruh file latihan pemrograman HTML.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Belajar Dasar HTML</title>
</head>
<body>
  <h1>Halo, dunia!</h1>
  <p>Saya akan mempelajari HTML</p>
</body>
</html>
```

Simpan dan beri nama file BelajarHtml.html

Setelah itu, tampilkan hasilnya di browser sebagaimana ditunjukkan dalam gambar berikut.



Gambar 3.1 Output Belajar Html.html

`<br>` digunakan dalam membentuk baris baru, sedangkan spasi yang dimasukkan lebih dari satu tidak akan ditampilkan sebagaimana adanya.

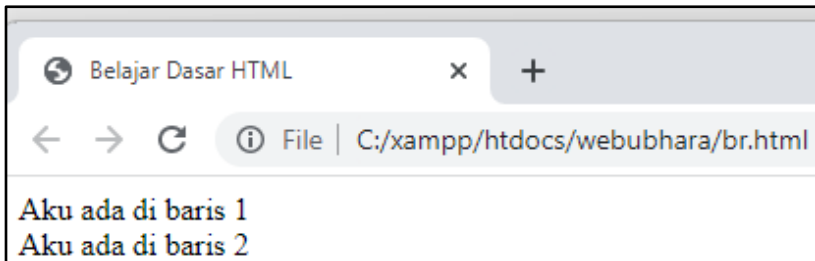
Contoh:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Belajar Dasar HTML</title>
</head>
<body>
  Aku ada di baris 1 <br>
```

```
Aku ada di baris 2
</body>
</html>
```

Simpan dengan nama file br.html

Selanjutnya, periksa tampilan hasilnya di browser sesuai dengan gambar yang ditampilkan di bawah ini.



Gambar 3.2 Output br.html

`<P></P>` berfungsi agar dapat membuat sebuah paragraf. Jika tidak menggunakan tag penutup, hasilnya akan terlihat seperti penggunaan `<br>` sebanyak dua kali. Tag ini juga dapat menggunakan atribut *align*.

Contoh:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Paragraph</title>
</head>
```

```
<body>
  <p align="left">Rata ke Kiri</p>
  <p align="right">Rata ke Kanan</p>
  <p align="center">Rata ke Tengah</p>
  <p align="justify">Rata ke Kiri Kanan</p>
</body>
</html>
```

Simpan dan beri nama file paragraph.html



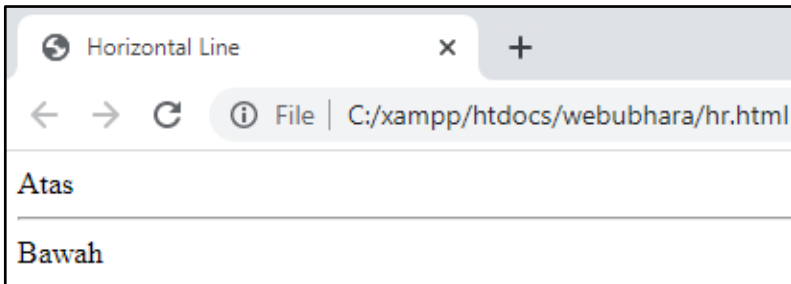
Gambar 3.3 Output paragraph.html

<hr> Horizontal Line (Garis horizontal)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Horizontal Line</title>
</head>
<body>
  Atas
```

```
<hr>
Bawah
</body>
</html>
```

Simpan dengan nama file hr.html



Gambar 3.4 Output hr.html

`<font></font>` digunakan untuk mengatur tampilan teks, seperti ukuran huruf, warna, dan jenis font. Atribut yang bisa dipakai antara lain: `size`, `color`, dan `face`.

Contoh:

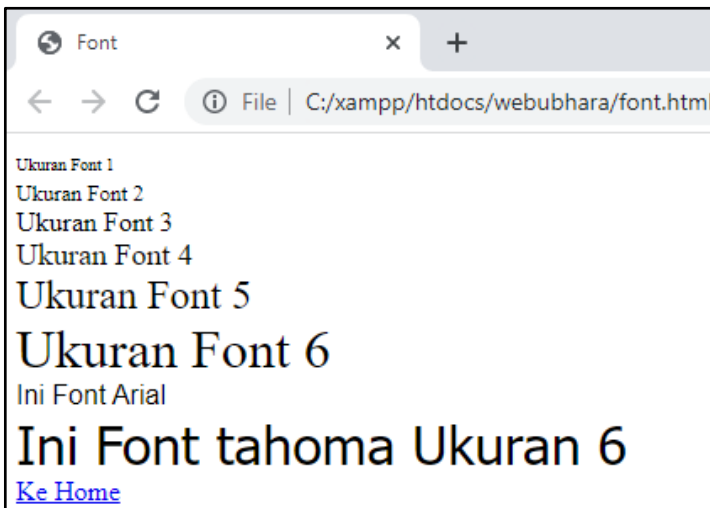
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Font</title>
</head>
<body>
```

```

<font size="1">Ukuran Font 1</font><br>
<font size="2">Ukuran Font 2</font><br>
<font size="3">Ukuran Font 3</font><br>
<font size="4">Ukuran Font 4</font><br>
<font size="5">Ukuran Font 5</font><br>
<font size="6">Ukuran Font 6</font><br>
<font face="Arial">Ini Font Arial</font><br>
  <font face="Tahoma" size="6">Ini Font tahoma Ukuran
6</font><br>
  <a href="http://localhost/webubhara/">Ke Home</a><br>
</body>
</html>

```

Simpan dan beri nama file font.html



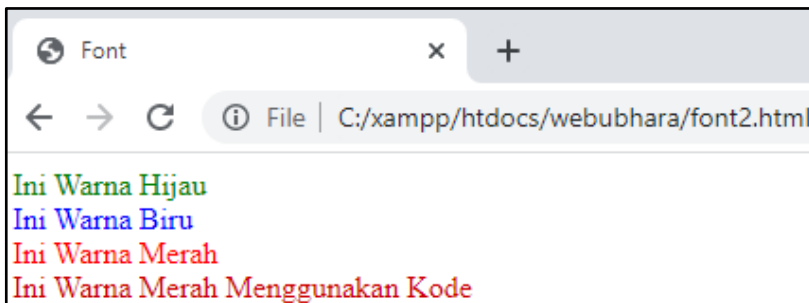
Gambar 3.5 Output font.html

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Font</title>
</head>
<body>
  <font color="green">Ini Warna Hijau</font><br>
  <font color="blue">Ini Warna Biru</font><br>
  <font color="red">Ini Warna Merah</font><br>
  <font color="#CC0000">Ini Warna Merah Menggunakan Kode</font><br>
</body>
</html>

```

Simpan dengan nama file font2.html



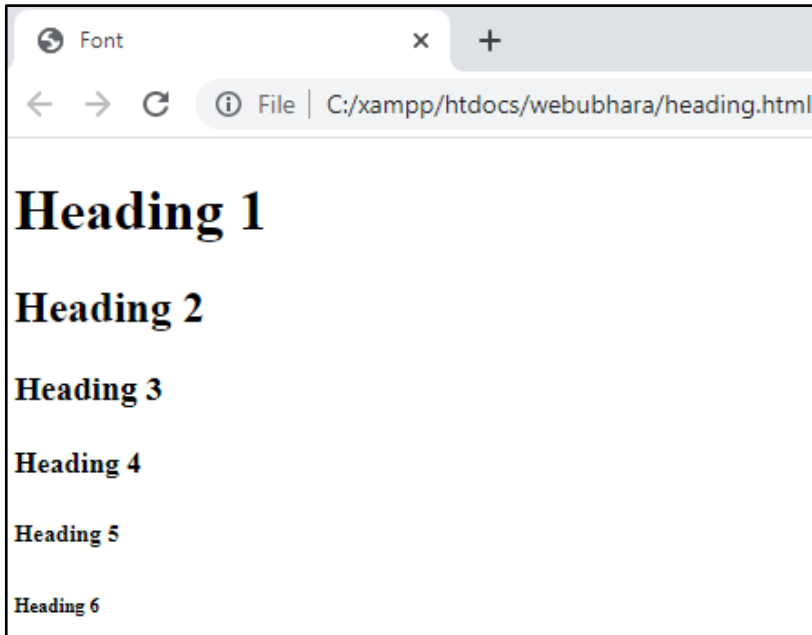
Gambar 3.6 Output font2.html

`<h1></h1>` `<h6></h6>` berfungsi untuk membuat judul atau heading, yang secara otomatis akan menghasilkan baris baru pada tampilan halaman.

Contoh:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Font</title>
</head>
<body>
  <h1>Heading 1</h1>
  <h2>Heading 2</h2>
  <h3>Heading 3</h3>
  <h4>Heading 4</h4>
  <h5>Heading 5</h5>
  <h6>Heading 6</h6>
</body>
</html>
```

Simpan dan beri nama file heading.html



Gambar 3.7 Output heading.html

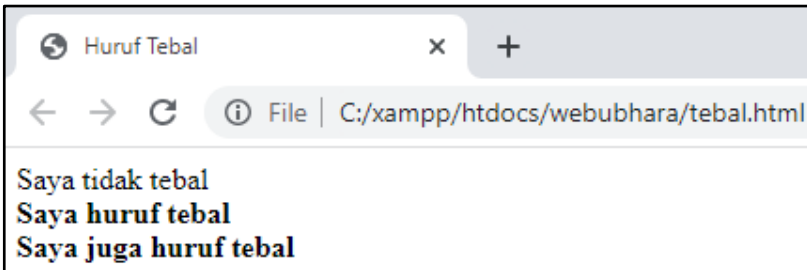
`<b></b>` atau `<strong></strong>` dipakai untuk menebalkan teks.

Contoh :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Huruf Tebal</title>
```

```
</head>
<body>
  Saya tidak tebal <br>
  <b>Saya huruf tebal</b><br>
  <strong>Saya juga huruf tebal</strong>
</body>
</html>
```

Simpan dengan nama file tebal.html



Gambar 3.8 Output tebal.html

`<i></i>` atau `<em></em>` digunakan agar teks menjadi miring.

Contoh:

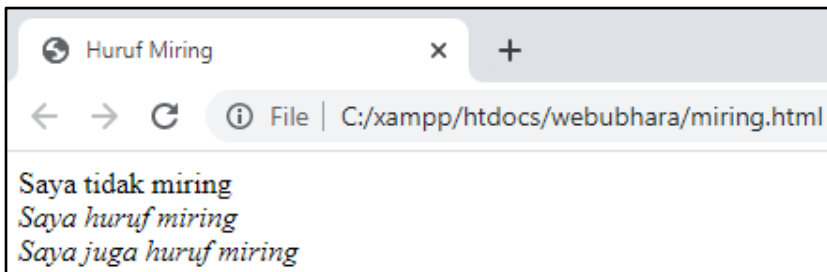
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Huruf Miring</title>
```

```

</head>
<body>
  Saya tidak miring <br>
  <i>Saya huruf miring</i><br>
  <em>Saya juga huruf miring</em>
</body>
</html>

```

Simpan dengan nama file miring.html



Gambar 3.9 Output miring.html

<U></U> dipakai untuk menampilkan teks dengan garis bawah.

Contoh:

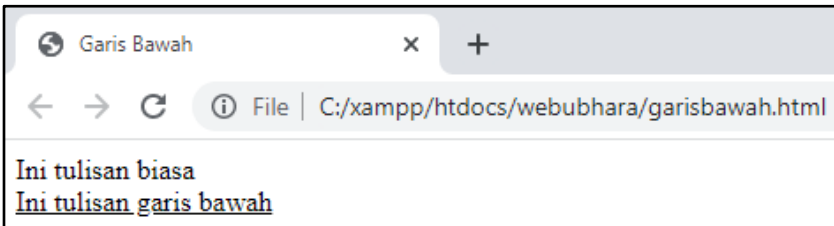
```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Garis Bawah</title>

```

```
</head>
<body>
  Ini tulisan biasa <br>
  <u>Ini tulisan garis bawah</u>
</body>
</html>
```

Simpan dengan nama file garisbawah.html



Gambar 3.10 Output garisbawah.html

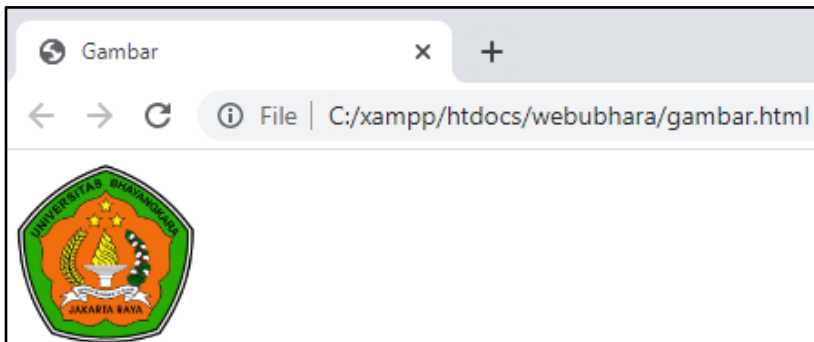
Untuk menampilkan gambar dalam dokumen HTML, digunakan tag `<img>`, yang dapat disertai sejumlah atribut berikut:

1. `src` (lokasi file gambar)
2. `align` (penataan posisi gambar)
3. `width` (lebar gambar dalam persen atau pixel)
4. `height` (tinggi gambar dalam persen atau pixel)
5. `alt` (deskripsi gambar)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
```

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Gambar</title>
</head>
<body>
  <br>
</body>
</html>
```

Simpan dan beri nama file gambar.html



Gambar 3.11 Output gambar.html

`<table></table>` digunakan untuk membuat tabel. Beberapa atribut yang dapat diterapkan pada tag `<table></table>` adalah:

1. border
2. align(perataan table)
3. width
4. height

Atribut yang dapat digunakan pada tag <tr></tr> adalah:

1. align
2. bgcolor
3. width
4. height

Atribut yang dapat diterapkan pada tag <td></td> adalah:

1. align
2. bgcolor
3. width
4. height

Contoh :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Tabel</title>
</head>
<body>
  <table border="1" width="100%">
    <tr>
      <th>No.</th>
      <th>NPM</th>
      <th>Nama</th>
    </tr>
```

```

<tr align="center">
  <td>1.</td>
  <td>201510225255</td>
  <td>Fried Sinlae</td>
</tr>
<tr align="center">
  <td>2.</td>
  <td>201510225001</td>
  <td>Awal Nama</td>
</tr>
<tr align="center">
  <td>3.</td>
  <td>201510225999</td>
  <td>Akhir Nama</td>
</tr>
</table>
</body>
</html>

```

Simpan dengan nama file table.html

No.	NPM	Nama
1.	201510225255	Fried Sinlae
2.	201510225001	Awal Nama
3.	201510225999	Akhir Nama

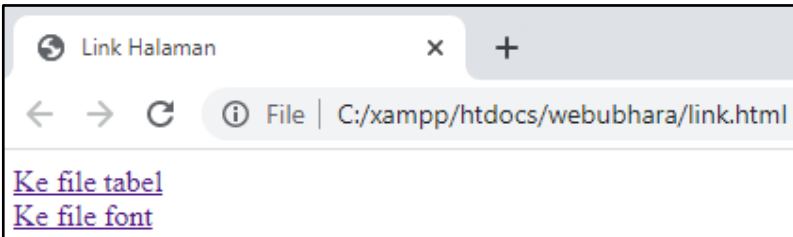
Gambar 3.12 Output tabel.html

`<a></a>` digunakan untuk membuat tautan. Atribut yang dapat digunakan adalah href dan name.

Contoh :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Link Halaman</title>
</head>
<body>
  <a href="table.html">Ke file tabel</a><br>
  <a href="font.html">Ke file font</a><br>
</body>
</html>
```

Simpan dan beri nama file link.html



Gambar 3.13 Output link.html

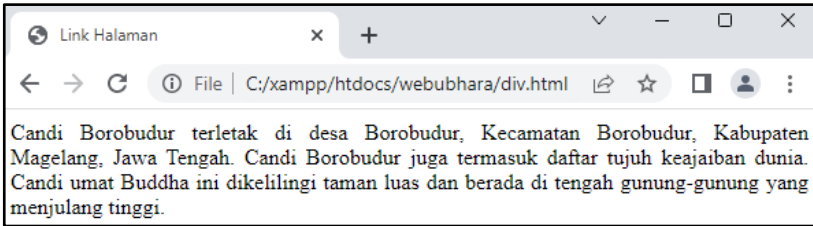
Sebagai pengganti <p>, tag <div></div> sering digunakan dalam pemrograman CSS untuk membentuk paragraf dan menyisipkan baris baru di akhir. Atribut yang dapat

diterapkan adalah align (untuk penataan paragraf), dengan nilai seperti left, right, center, dan justify.

Contoh :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Link Halaman</title>
</head>
<body>
  <div align="justify">
    Candi Borobudur terletak di desa Borobudur, Kecamatan
    Borobudur, Kabupaten Magelang, Jawa Tengah. Candi Borobudur
    juga termasuk daftar tujuh keajaiban dunia. Candi umat Buddha ini
    dikelilingi taman luas dan berada di tengah gunung-gunung yang
    menjulang tinggi.
  </div>
</body>
</html>
```

Simpan dan beri nama file div.html



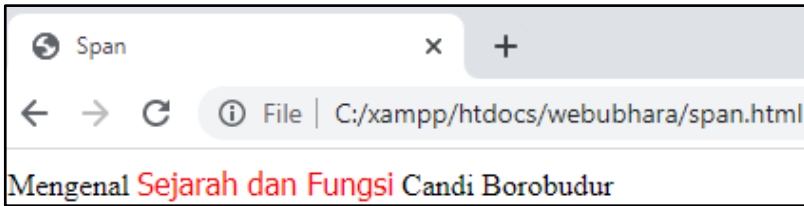
Gambar 3.14 Output div.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Span</title>
</head>
<body>
  Mengenal <span style="font-family: Tahoma; color: red;">Sejarah
dan Fungsi</span> Candi Borobudur <br>
</body>
</html>
```

Simpan dengan nama file span.html

Dalam css, elemen `<span></span>` dimanfaatkan untuk memberi gaya pada teks yang dipilih tanpa memengaruhi struktur baris.

Contoh :



Gambar 3.15 Output span.html

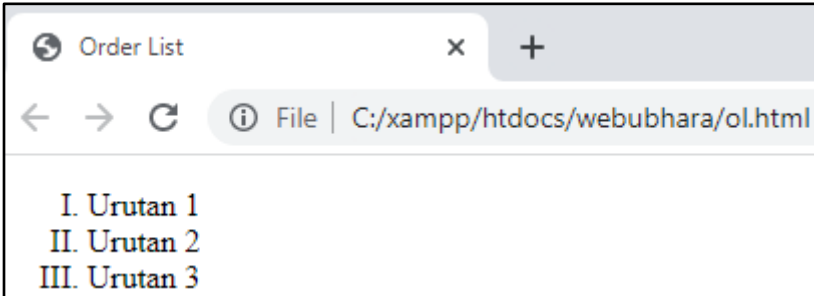
Untuk membuat daftar yang terstruktur dengan penomoran, digunakan tag `<ol></ol>` yang bisa dikustomisasi dengan atribut `type` (misalnya 1, A, a, I, atau i).

Contoh :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Order List</title>
</head>
<body>
  <ol type="I">
    <li>Urutan 1</li>
    <li>Urutan 2</li>
    <li>Urutan 3</li>
  </ol>
```

```
</body>
</html>
```

Simpan dengan nama file ol.html



Gambar 3.16 Output ol.html

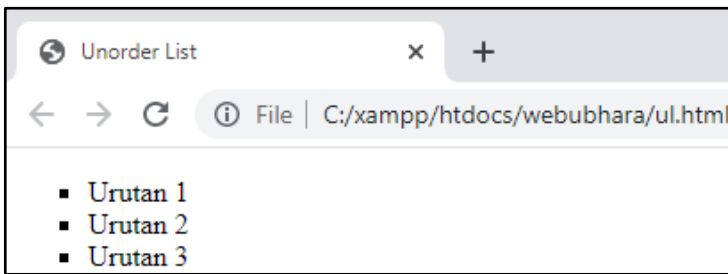
`<ul></ul>` digunakan untuk membuat daftar tidak terurut, yang sering disebut dengan bullet. Atribut yang tersedia adalah `type` (untuk menentukan jenis bullet), dengan nilai seperti `disc`, `square`, dan `circle`.

Contoh:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Unorder List</title>
</head>
<body>
  <ul type="square">
```

```
<li>Urutan 1</li>
<li>Urutan 2</li>
<li>Urutan 3</li>
</ul>
</body>
</html>
```

Simpan dengan nama file ul.html



Gambar 3.17 Output ul.html

<FORM></FORM> digunakan untuk membuat blok elemen formulir, seperti input teks dan area teks. Atribut yang digunakan meliputi:

1. name (nama formulir)
2. action (file tujuan tempat formulir dikirim, digunakan di sisi server)
3. Metode pengiriman data (method) yang digunakan di sisi server ini memiliki dua nilai, yaitu post dan get.

ELEMENT PADA FORM

Tag <INPUT> berada di dalam blok tag formulir. Atribut yang dimiliki antara lain:

1. Untuk menentukan jenis input, atribut type memiliki nilai seperti text, checkbox, radio, hidden, button, submit, reset, file, dan image.
2. name adalah nama dari elemen tersebut.
3. value adalah nilai yang diberikan atau diisi pada elemen.

Contoh: File form-lengkap.html

```
<!DOCTYPE html>
<html>
<head>
<title>Form Lengkap</title>
</head>
<body>
<h2>Form Input Mahasiswa</h2>
<form method='POST' action='form-aksi.php'>
<table border='1' width='100%'>
<tr><td>NIM</td><td><input type='text' name='nim'></td></tr>
<tr><td>Nama</td><td><input type='text' name='nama_mahasiswa'></td></tr>
<tr><td>Tanggal Lahir</td><td><input type='date' name='tgl_lahir'></td></tr>
<tr><td>Jenis Kelamin</td>
<td>
<input type='radio' name='jk' value='L'> Laki-Laki
<input type='radio' name='jk' value='P'> Perempuan
</td>
</tr>
</table>
</form>
</body>
</html>
```

```

</tr>
<tr><td>Agama</td>
<td>
<select name='agama'>
<option value='Islam'>Islam</option>
<option value='Katolik'>Katolik</option>
<option value='Protestan'>Protestan</option>
<option value='Hindu'>Hindu</option>
<option value='Budha'>Budha</option>
<option value='Konghucu'>Konghucu</option>
</select>
</td>
</tr>
<tr><td>Alamat</td><td><textarea
                                name='alamat'
                                cols='70'></textarea></td></tr>
<tr><td>Hobi</td>
<td>
<input type='checkbox' name='hobi1' value='Membaca'>
Membaca<br>
<input type='checkbox' name='hobi2' value='Olahraga'>
Olahraga<br>
<input type='checkbox' name='hobi3' value='Hiking'> Hiking<br>
</td>
</tr>
<tr><td colspan='2'><input
                                type='submit'
                                value='Simpan'></td></tr>
</table>

```

```
</body>
</html>
```

Simpan dengan nama file form-lengkap.html

The screenshot shows a web browser window with the title 'Tabel'. The address bar shows the file path 'C:/xampp/htdocs/webubhara/form-lengkap.html'. The form is titled 'Form Input Mahasiswa' and contains the following fields:

- NIM: Text input field
- Nama: Text input field
- Tanggal Lahir: Date input field with a calendar icon
- Jenis Kelamin: Radio buttons for 'Laki-Laki' and 'Perempuan'
- Agama: Dropdown menu with 'Islam' selected
- Alamat: Text input field
- Hobi: Checkboxes for 'Membaca', 'Olahraga', and 'Hiking'
- Simpan: Submit button

Gambar 3.18 Output form-lengkap.html

B. CSS

Sebagai bahasa yang digunakan untuk penataan, CSS (*Cascading Style Sheets*) bertugas untuk mengelola desain dan tampilan halaman web. Dengan memanfaatkan CSS, kita bisa menentukan jenis huruf, warna teks, serta latar belakang halaman. Biasanya, CSS dipadukan dengan bahasa markup seperti HTML dan XML untuk menciptakan situs web yang menarik dan fungsional.

CSS hadir untuk mengatasi keterbatasan HTML dalam mengatur tata letak dan desain. Misalnya, jika sebuah website memiliki banyak halaman dan hanya menggunakan HTML, maka pengaturan elemen harus ditulis berulang

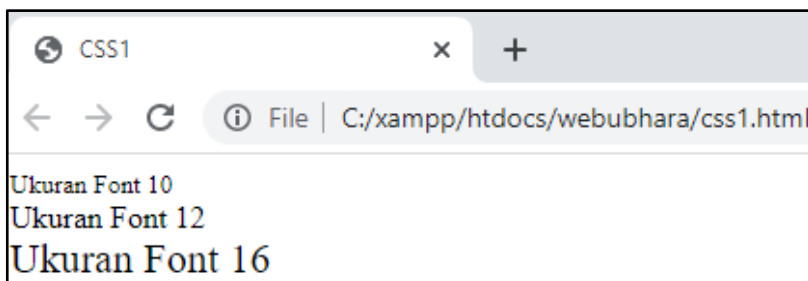
pada setiap halaman. Namun dengan CSS, cukup menuliskan aturan gaya satu kali saja, dan aturan tersebut bisa diterapkan ke seluruh halaman. Saat perlu melakukan perubahan, Anda cukup mengubahnya di satu tempat saja lebih efisien dan praktis.

Meskipun sering dianggap sebagai bahasa pemrograman, CSS sebenarnya lebih tepat disebut sebagai bahasa *style sheet* yang biasa dipakai bersama JavaScript untuk memperkaya tampilan web.

1. Penulisan pada Tag dengan Atribut Style

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>CSS1</title>
</head>
<body>
  <div style="font-size: 10pt;">Ukuran Font 10</div>
  <div style="font-size: 12pt;">Ukuran Font 12</div>
  <div style="font-size: 16pt;">Ukuran Font 16</div>
</body>
</html>
```

Simpan dengan nama file css1.html

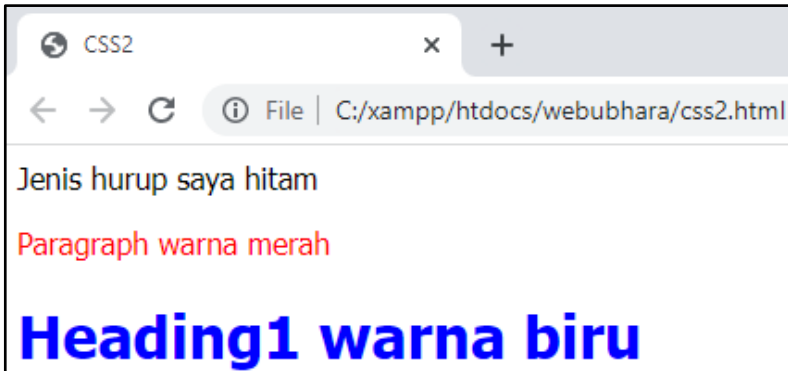


Gambar 3.19 Output css1.html

2. Penulisan pada Head

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>CSS2</title>
  <style type="text/css">
    body {color: Black; font-family: Tahoma;}
    p {color: Red;}
    h1,h2{color: blue;}
  </style>
</head>
<body>
  Jenis hurup saya hitam <br>
  <p>Paragraph warna merah</p>
  <h1>Heading warna biru</h1>
</body>
</html>
```

Simpan dengan nama file css2.html



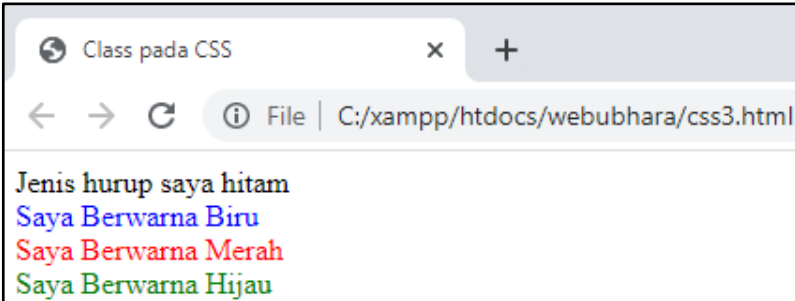
Gambar 3. 20 Output css2.html

3. Penulisan dengan Class

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Class pada CSS</title>
  <style type="text/css">
    div.biru{color: blue;}
    div.merah{color: red;}
    .hijau{color: green;}
  </style>
</head>
<body>
  Jenis hurup saya hitam <br>
  <div class="biru">Saya Berwarna Biru</div>
  <div class="merah">Saya Berwarna Merah</div>
  <div class="hijau">Saya Berwarna Hijau</div>
```

```
</body>  
</html>
```

Simpan dan beri nama file css3.html



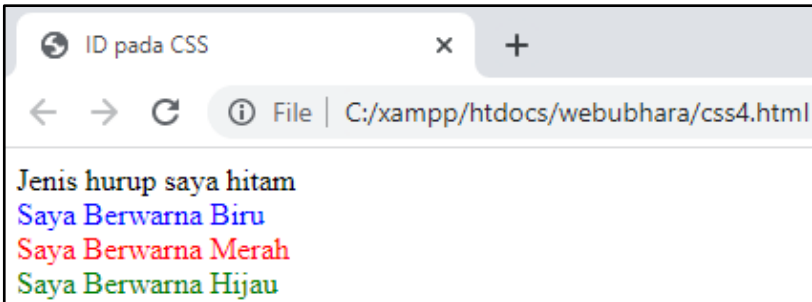
Gambar 3.21 Output css3.html

4. Penulisan dengan id

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <title>ID pada CSS</title>  
  <style type="text/css">  
    div#biru{color: blue;}  
    div#merah{color: red;}  
    #hijau{color: green;}  
  </style>  
</head>  
<body>  
  Jenis hurup saya hitam <br>  
  <div id="biru">Saya Berwarna Biru</div>  
  <div id="merah">Saya Berwarna Merah</div>
```

```
<div id="hijau">Saya Berwarna Hijau</div>
</body>
</html>
```

Simpan dan beri nama file css4.html



Gambar 3.22 Output css4.html

5. CSS Eksternal

Sebagai contoh, buat sebuah file dengan nama css5.html, lalu masukkan kode berikut ke dalamnya:

```
<!DOCTYPE HTML>
<html>
<head>
<title>CSS External</title>
<link href="Style.css" rel="stylesheet">
</head>
<body>
<div>
<h1>All-time Home Run Record</h1>
<table>
<tr id="trHeader">
```

```

<th>Player</th>
<th>Home Runs</th>
<th>Team</th>
</tr>
<tr>
<td>Barry Bonds</td>
<td class="centerCell">762</td>
<td>Giants</td>
</tr>
<tr>
<td>Hank Aaron</td>
<td class="centerCell">755</td>
<td>Braves</td>
</tr>
<tr>
<td>Babe Ruth</td>
<td class="centerCell">714</td>
<td>Yankees</td>
</tr>
<tr>
<td>Willie Mays</td>
<td class="centerCell">660</td>
<td>Giants</td>
</tr>
</table>
</div>

```

```
</body>  
</html>
```

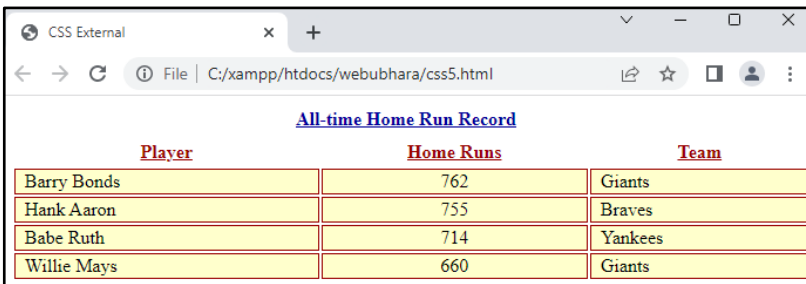
Simpan dengan nama file css5.html

```
h1 {  
  text-align: center;  
  font-size: 1em;  
  color: #009;  
  margin-bottom: .3em;  
  text-decoration: underline;  
}  
table {  
  margin: .3em;  
  width: 100%;  
}  
th {  
  padding: .2em;  
}  
td {  
  padding-left: .5em;  
  padding-right: .5em;  
  border: 1px solid #900;  
  background-color: #ffc;  
}  
#trHeader {  
  text-decoration: underline;
```

```
color: #900;
}
.centerCell {
text-align: center;
}
```

Simpan dan beri nama file style.css

Buka file css5.html di browser kemudian amati hasilnya.



<u>All-time Home Run Record</u>		
<u>Player</u>	<u>Home Runs</u>	<u>Team</u>
Barry Bonds	762	Giants
Hank Aaron	755	Braves
Babe Ruth	714	Yankees
Willie Mays	660	Giants

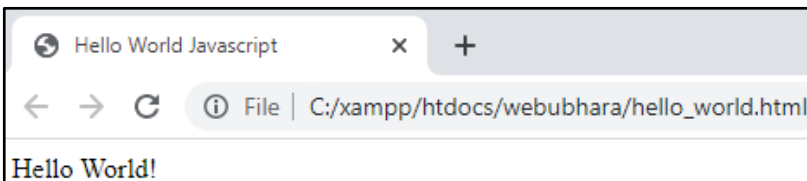
Gambar 3.23 Output css5.html

C. JavaScript

Pada awalnya, JavaScript dikembangkan sebagai bahasa pemrograman yang dapat berjalan di dalam browser. Namun, seiring berjalannya waktu, penggunaannya meluas hingga mencakup sisi server, pengembangan game, Internet of Things (IoT), aplikasi desktop, dan lainnya. Pada mulanya, bahasa ini diberi nama Mocha, kemudian berganti menjadi LiveScript ketika versi beta Netscape Navigator 2.0 dirilis pada September 1995. Tidak lama setelah itu, namanya diubah menjadi JavaScript. Berikut ini adalah contoh kode program JavaScript:

```
<!DOCTYPE html>
<html>
<head>
<title>Hello World Javascript</title>
</head>
<body>
<script>
console.log("Saya belajar Javascript");
document.write("Hello World!");
</script>
</body>
</html>
```

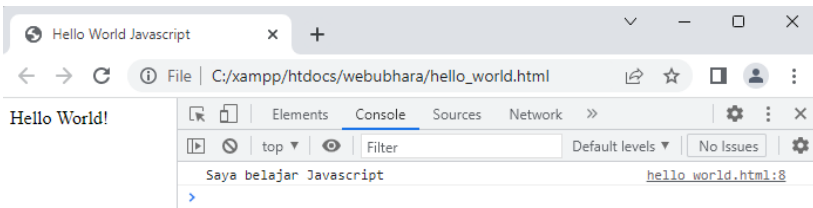
Simpan file dengan nama `hello_world.html`, buka file tersebut menggunakan browser web. Setelah itu, tampilan hasilnya akan terlihat seperti berikut:



Gambar 3.24 Output `hello_world.html`

Mengapa teks “Saya belajar Javascript” tidak muncul di halaman?

Hal ini disebabkan karena fungsi `console.log()` dipakai untuk menunjukkan pesan ke console JavaScript, bukan ke tampilan halaman web. Sementara itu, `document.write()` berfungsi untuk menampilkan teks langsung ke dalam dokumen HTML.



Gambar 3.25 Output hello_world.html

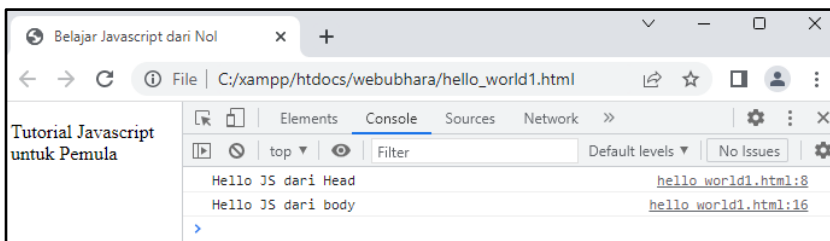
1. Menulis Kode javascript menggunakan Metode Embed

Kode JavaScript disisipkan ke dalam HTML dengan menggunakan tag `<script>`, yang dapat diletakkan di dalam tag `<head>` atau `<body>`.

```
<!DOCTYPE html>
<html>
<head>
<title>Belajar Javascript dari Nol</title>
<script>
// ini adalah penulisan kode javascript
// di dalam tag <head>
console.log("Hello JS dari Head");
</script>
</head>
<body>
<p>Tutorial Javascript untuk Pemula</p>
<script>
// ini adalah penulisan kode javascript
// di dalam tag <body>
```

```
console.log("Hello JS dari body");
</script>
</body>
</html>
```

Simpanlah file dan beri nama `hello_world1.html`, setelah itu, akses file tersebut menggunakan browser. Hasilnya akan terlihat seperti berikut:



Gambar 3.26 Output `hello_world1.html`

Sebaiknya kode `<body>` ditempatkan di bagian bawah halaman, karena cara ini biasanya dapat mempercepat proses pemuatan halaman secara keseluruhan.

2. Menulis Kode Javascript Inline

Kode JavaScript ditulis di dalam atribut HTML menggunakan metode ini. Pendekatan ini umumnya dipakai untuk menjalankan fungsi tertentu ketika terjadi sebuah peristiwa, seperti saat sebuah tautan diklik.

Contoh:

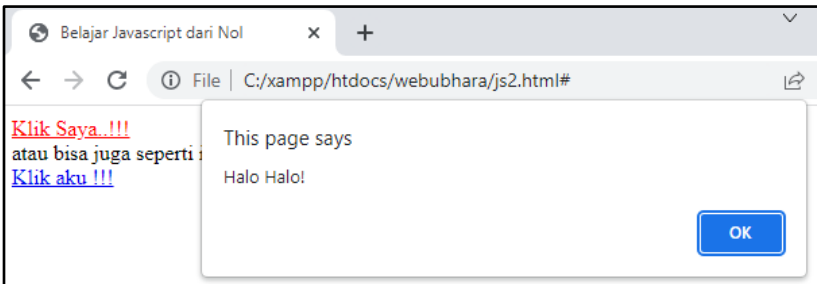
```
<!DOCTYPE html>
<html>
<head>
<title>Belajar Javascript dari Nol</title>
```

```

</head>
<body>
<a href="#" onclick="alert('Halo Halo!')">Klik Saya...!!!</a>
<br>atau bisa juga seperti ini: <br>
<a href="javascript:alert('Hei...!')">Klik aku !!!</a>
</body>
</html>

```

Simpan file dengan nama js2.html, lalu buka menggunakan peramban web untuk melihat hasilnya.



Gambar 3.27 Output js2.html

3. Penulisan Kode JavaScript Eksternal

Dalam metode ini, kode JavaScript ditulis terpisah dari file HTML. Pendekatan ini umumnya dipakai dalam proyek berskala besar karena memudahkan pengelolaan kode secara keseluruhan. Sebagai contoh, buat dua file yang terletak dalam folder yang sama: satu file JavaScript dan satu file HTML, misalnya kode-program.js dan js3.html.

Isi dari file kode-program.js:

```

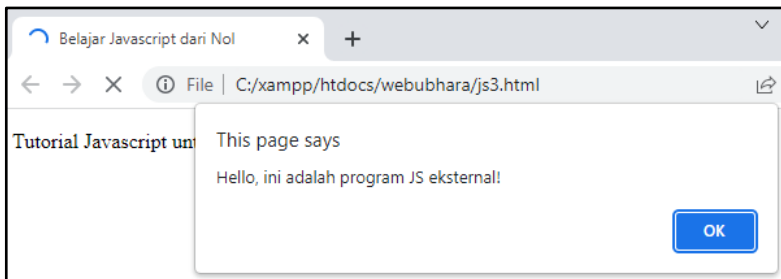
alert("Hello, ini adalah program JS eksternal!");

```

Isi dari file js3.html:

```
<!DOCTYPE html>
<html>
<head>
<title>Belajar Javascript dari Nol</title>
</head>
<body>
<p>Tutorial Javascript untuk Pemula</p>
<!-- Menyisipkan kode js eksternal -->
<script src="kode-program.js"></script>
</body>
</html>
```

Hasilnya:



Gambar 3.28 Output js3.html

Pada contoh di atas, JavaScript ditulis terpisah dari HTML.

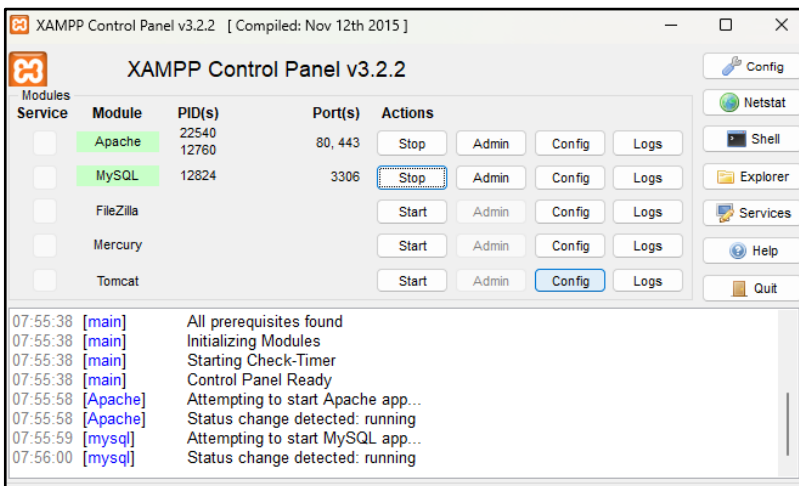
4. PHP

Berbeda dengan JavaScript yang dijalankan di sisi klien melalui browser, PHP (PHP: *Hypertext Preprocessor*) adalah bahasa pemrograman yang berfungsi di sisi server (server-

side). Bahasa pemrograman PHP dibangun menggunakan C dan dipengaruhi oleh bahasa seperti C, Java, dan Perl. Fungsi utama PHP adalah untuk melakukan proses awal dalam membentuk halaman HTML yang bersifat dinamis dan disesuaikan dengan masing-masing pengguna.

Sebelum membuat situs web dengan PHP, diperlukan web server sebagai lingkungan eksekusi, serta sistem basis data untuk menyimpan informasi pengguna secara terstruktur. Secara umum, PHP bekerja bersama web server Apache dan MySQL sebagai basis data. Kombinasi ini sering dikenal dengan istilah AMP.

Beberapa contoh paket AMP yang umum digunakan antara lain XAMPP, WAMPServer, VertrigoServ, LAMP, UniServer (Uniform Server), dan Server2Go. Gambar dibawah ini memperlihatkan proses menyalakan Web Server melalui XAMPP.



Gambar 3.29 XAMPP

File tersebut akan diberi ekstensi .php dan disimpan dalam folder /htdocs/BelajarPHP/.

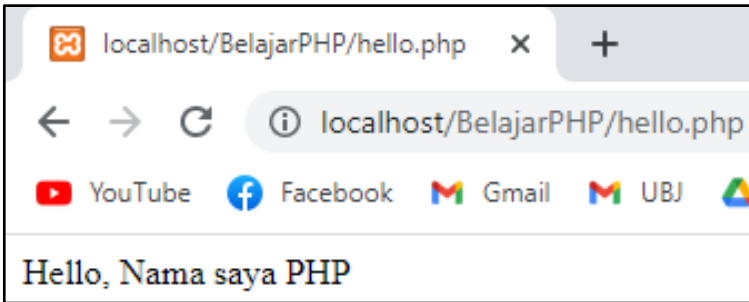
a. Struktur Dasar PHP

- 1) Kode PHP harus ditempatkan di antara tag `<?php` dan `?>`
- 2) Setiap perintah (*statement*) umumnya diakhiri dengan tanda titik-koma (;).
- 3) Dalam PHP, nama identifier yang dibuat oleh pengguna (seperti variabel, konstanta, dan fungsi) bersifat CASE SENSITIVE, namun untuk identifier bawaan PHP, sifatnya tidak CASE SENSITIVE. Contoh:
 - a) `$nama` $\neq$ `$Nama` $\neq$ `$NAMA`
 - b) `hitungLuas()` $\neq$ `HitungLuas()`
 - c) `echo` $\neq$ `ECHO`
 - d) `while` $\neq$ `WHILE`

Say Hello to PHP Ketikkan perintah di bawah ini menggunakan editor: Nama File: hello.php Deskripsi: Program dasar untuk menampilkan teks ke layar.

```
<?php
echo "Hello, Nama saya PHP ";
?>
```

Simpan file yang telah dibuat dengan nama "hello.php" ke dalam folder htdocs/belajarPHP/. Kemudian, jalankan browser dan buka alamat `http://localhost/BelajarPHP/hello.php`. Hasilnya akan muncul di layar browser seperti pada gambar berikut ini.



Gambar 3.30 Output hello.php

b. Variabel

Variabel berfungsi sebagai tempat penyimpanan nilai, data, atau informasi. Beberapa ketentuan dalam penulisan variabel antara lain:

- 1) Penulisan variabel dimulai dengan simbol \$.
- 2) Tidak ada batasan panjang untuk nama variabel.
- 3) Setelah tanda \$, variabel harus diberi nama yang diawali huruf atau garis bawah (_), dan karakter berikutnya dapat berupa huruf, angka, atau karakter yang diperbolehkan (dalam rentang ASCII 127–255).
- 4) Variabel bersifat *case-sensitive*, artinya huruf besar dan kecil memiliki makna berbeda.
- 5) Variabel dapat langsung digunakan tanpa deklarasi sebelumnya.
- 6) Variabel harus diberi nama tanpa menggunakan spasi
- 7) Contoh :

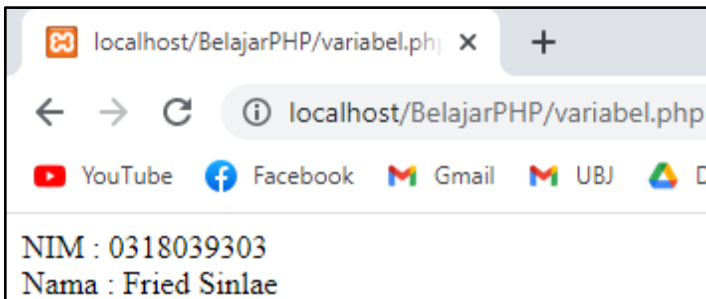
Tabel 3.1 Variabel

Benar	Salah
<code>\$_name</code>	<code>\$3name</code>
<code>\$first_name</code>	<code>\$name?</code>
<code>\$name3</code>	<code>\$first+name</code>
<code>\$name_3</code>	<code>\$first.name</code>
	<code>\$first name</code>

Nama File : variabel.php

Penjelasan: Program ini berfungsi untuk memberikan nilai pada variabel dan menampilkannya di layar.

```
<?php
$nim = "0318039303";
$nama = 'Fried Sinlae';
echo "NIM : $nim <br>";
echo "Nama : $nama";
?>
```



Gambar 3.31 Output variabel.php

c. Tipe Data

Dalam PHP, tipe data pada variabel tidak perlu ditentukan secara langsung oleh pembuat program karena akan dikenali secara otomatis oleh sistem interpreter PHP. Meskipun begitu, PHP tetap menyediakan delapan jenis tipe data dasar yang dapat dipakai.

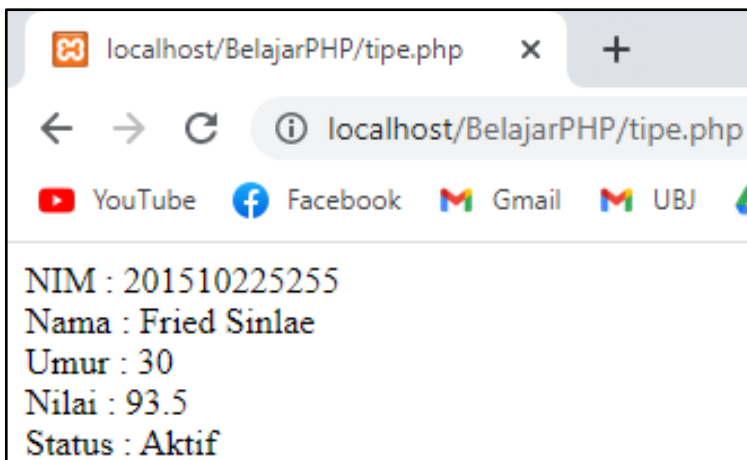
Tabel 3.2 Tipe Data

Boolean	Integer
Float	String
Array	Object
Resource	Null

Nama File: tipe.php

Penjelasan: Contoh program ini menunjukkan cara menangani berbagai tipe data dasar dalam PHP.

```
<?php
$nim = "201510225255";
$nama = "Fried Sinlae";
$umur = 30;
$nilai = 93.5;
$status = "Aktif";
echo "NIM : $nim <br>";
echo "Nama : $nama <br>";
echo "Umur : $umur <br>";
echo "Nilai : $nilai <br>";
echo "Status : $status";
?>
```



Gambar 3.32 Output tipe.php

d. Tanda Sambung

Untuk menggabungkan antar variabel atau menghubungkan teks, gunakan simbol titik (.).

Nama File : sambung.php

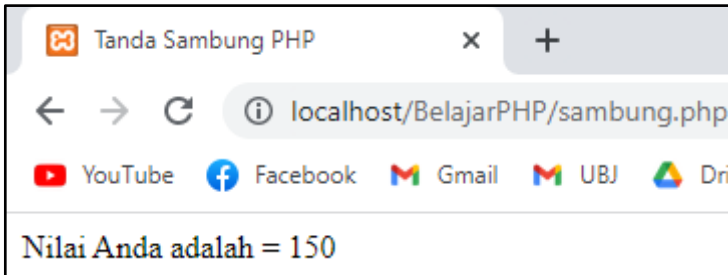
Penjelasan: Program ini merupakan contoh penerapan beberapa jenis tipe data dasar dalam PHP.

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Tanda Sambung PHP</title>
</head>

<body>
  <?php
    $nilai = 150;
    echo "Nilai Anda adalah = " . $nilai;
  ?>
</body>

</html>
```



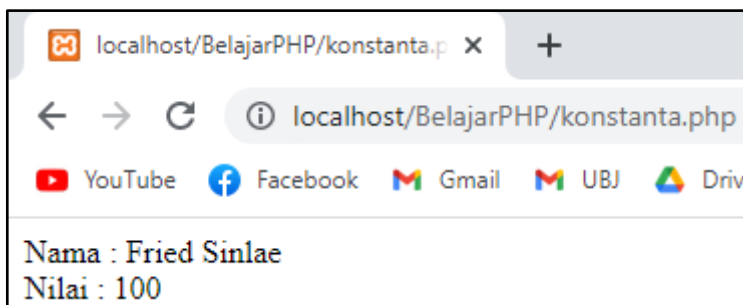
Gambar 3.33 Output sambung.php

Konstanta adalah nilai tetap yang tidak dapat diubah. Untuk mendeklarasikan konstanta di PHP, digunakan fungsi `define()`.

Nama File : konstanta.php

Penjelasan: Program yang menunjukkan penerapan konstanta dalam pemrograman.

```
<?php
define('nama', "Fried Sinlae");
define('nilai', 100);
echo "Nama : " . nama . "<br>";
echo "Nilai : " . nilai;
?>
```



Gambar 3.34 Output konstanta.php

e. Operator Dalam PHP

Tabel 3.3 PHP Arithmetic Operators

PHP Arithmetic Operators			
Operator	Name	Example	Result
+	Addition	$\$x + \y	Sum of $\$x$ and $\$y$
-	Subtraction	$\$x - \y	Difference of $\$x$ and $\$y$
*	Multiplication	$\$x * \y	Product of $\$x$ and $\$y$
/	Division	$\$x / \y	Quotient of $\$x$ and $\$y$
%	Modulus	$\$x \% \y	Remainder of $\$x$ divided by $\$y$
**	Exponentiation	$\$x ** \y	Result of raising $\$x$ to the $\$y$ 'th power

Tabel 3.4 PHP Assignment Operators

PHP Assignment Operators		
Assignment	Same as...	Description
<code>x = y</code>	<code>x = y</code>	The left operand gets set to the value of the expression on the right
<code>x += y</code>	<code>x = x + y</code>	Addition
<code>x -= y</code>	<code>x = x - y</code>	Subtraction
<code>x *= y</code>	<code>x = x * y</code>	Multiplication
<code>x /= y</code>	<code>x = x / y</code>	Division
<code>x %= y</code>	<code>x = x % y</code>	Modulus

Tabel 3.5 PHP Comparison Operators

PHP Comparison Operators			
Operator	Name	Example	Result
<code>==</code>	Equal	<code>\$x == \$y</code>	Returns true if \$x is equal to \$y
<code>===</code>	Identical	<code>\$x === \$y</code>	Returns true if \$x is equal to \$y, and they are of the same type
<code>!=</code>	Not equal	<code>\$x != \$y</code>	Returns true if \$x is not equal to \$y
<code><></code>	Not equal	<code>\$x <> \$y</code>	Returns true if \$x is not equal to \$y
<code>!==</code>	Not identical	<code>\$x !== \$y</code>	Returns true if \$x is not equal to \$y, or they are not of the same type
<code>></code>	Greater than	<code>\$x > \$y</code>	Returns true if \$x is greater than \$y
<code><</code>	Less than	<code>\$x < \$y</code>	Returns true if \$x is less than \$y
<code>>=</code>	Greater than or equal to	<code>\$x >= \$y</code>	Returns true if \$x is greater than or equal to \$y
<code><=</code>	Less than or equal to	<code>\$x <= \$y</code>	Returns true if \$x is less than or equal to \$y
<code><=></code>	Spaceship	<code>\$x <=> \$y</code>	Returns an integer less than, equal to, or greater than zero, depending on if \$x is less than, equal to, or greater than \$y. Introduced in PHP 7.

Tabel 3.6 PHP Logical Operators

PHP Logical Operators			
Operator	Name	Example	Result
and	And	\$x and \$y	True if both \$x and \$y are true
or	Or	\$x or \$y	True if either \$x or \$y is true
xor	Xor	\$x xor \$y	True if either \$x or \$y is true, but not both
&&	And	\$x && \$y	True if both \$x and \$y are true
	Or	\$x \$y	True if either \$x or \$y is true
!	Not	!\$x	True if \$x is not true

Tabel 3.7 PHP String Operators

PHP String Operators			
Operator	Name	Example	Result
.	Concatenation	<u>\$txt1.</u> \$txt2	Concatenation of \$txt1 and \$txt2
.=	Concatenation assignment	\$txt1 .= \$txt2	Appends \$txt2 to \$txt1

Tabel 3.8 PHP Array Operators

PHP Array Operators			
Operator	Name	Example	Result
+	Union	<code>\$x + \$y</code>	Union of <code>\$x</code> and <code>\$y</code>
<code>==</code>	Equality	<code>\$x == \$y</code>	Returns true if <code>\$x</code> and <code>\$y</code> have the same key/value pairs
<code>===</code>	Identity	<code>\$x === \$y</code>	Returns true if <code>\$x</code> and <code>\$y</code> have the same key/value pairs in the same order and of the same types
<code>!=</code>	Inequality	<code>\$x != \$y</code>	Returns true if <code>\$x</code> is not equal to <code>\$y</code>
<code><></code>	Inequality	<code>\$x <> \$y</code>	Returns true if <code>\$x</code> is not equal to <code>\$y</code>
<code>!==</code>	Non-identity	<code>\$x !== \$y</code>	Returns true if <code>\$x</code> is not identical to <code>\$y</code>

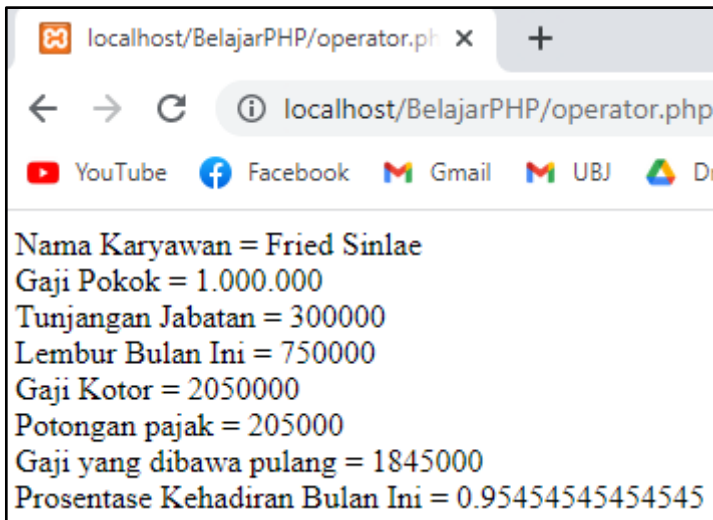
Tabel 3.9 PHP Conditional Assignment Operators

PHP Conditional Assignment Operators			
Operator	Name	Example	Result
<code>?:</code>	Ternary	<code>\$x = <u><code>expr1</code></u> ? <code>expr2</code> : <code>expr3</code></code>	Returns the value of <code>\$x</code> .
			The value of <code>\$x</code> is <code>expr2</code> if <code>expr1</code> = TRUE.
			The value of <code>\$x</code> is <code>expr3</code> if <code>expr1</code> = FALSE
<code>??</code>	Null coalescing	<code>\$x = <u><code>expr1</code></u> ?? <code>expr2</code></code>	Returns the value of <code>\$x</code> .
			The value of <code>\$x</code> is <code>expr1</code> if <code>expr1</code> exists, and is not NULL.
			If <code>expr1</code> does not exist, or is NULL, the value of <code>\$x</code> is <code>expr2</code> .
			Introduced in PHP 7

Nama File : operator.php

Deskripsi: Program yang menunjukkan penggunaan berbagai operator aritmatika yang digunakan dalam pemrograman.

```
<?php
$nama = "Fried Sinlae";
$gajipokok = 1000000;
$tunjangan = 300000;
$lembur = 750000;
$gajikotor = $gajipokok + $tunjangan + $lembur;
$pajak = 0.1;
$potonganpajak = $gajikotor * $pajak;
$thp = $gajikotor - $potonganpajak;
$harikerja = 22;
$harimasuk = 21;
$prosentasekehaditan = $harimasuk / $harikerja;
echo "Nama Karyawan = $nama <br>";
echo "Gaji Pokok = " . number_format($gajipokok, 0, ",", ".") . " <br>";
echo "Tunjangan Jabatan = $tunjangan <br>";
echo "Lembur Bulan Ini = $lembur <br>";
echo "Gaji Kotor = $gajikotor <br>";
echo "Potongan pajak = $potonganpajak <br>";
echo "Gaji yang dibawa pulang = $thp <br>";
echo "Prosentase Kehadiran Bulan Ini = $prosentasekehaditan";
?>
```



Gambar 3.35 Output operator.php

Untuk menuliskan komentar PHP, dapat digunakan:

/* dan */ // dan # #

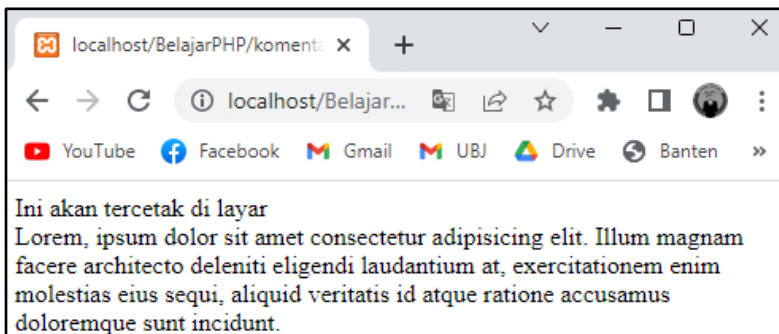
Nama File : komentar.php

Deskripsi: Contoh program yang menunjukkan teknik penulisan komentar di dalam kode PHP.

```
<?php
/* Ini komentar tidak akan tercetak di layar
yang ini juga komentar termasuk dalam banyak baris
*/
// Ini komentar satu baris
# ini juga komentar
echo "Ini akan tercetak di layar";

echo "<br>";

echo "Lorem, ipsum dolor sit amet consectetur adipisicing elit. Illum magnam facere architecto deleniti eligendi laudantium at, exercitationem enim molestias eius sequi, aliquid veritatis id atque ratione accusamus doloremque sunt incidunt.";
?>
```



Gambar 3.36 Output komentar.php

4. Struktur Kondisi

Struktur Kondisi If

```
if (kondisi) {  
    statement-jika-kondisi-true;  
}
```

Keterangan :

// kondisi merujuk pada pernyataan atau variabel yang dievaluasi untuk menghasilkan nilai TRUE atau FALSE-nya.

Struktur Kondisi If ... Else

```
if (kondisi) {  
    statement-jika-kondisi-true;  
}  
else {  
    statement-jika-kondisi-false;  
}
```

Struktur Kondisi If ... ElseIf..Else

```
if (kondisi) {  
    statement-jika-kondisi-true;  
} elseif (kondisi) {  
    statement-jika-kondisi-false;  
}  
else {  
    statement-jika-kondisi-false;  
}
```

Keterangan :

Kondisi adalah pernyataan atau variabel yang akan dievaluasi untuk menentukan apakah bernilai benar (*TRUE*) atau salah (*FALSE*). Jika hasil evaluasi menunjukkan nilai *TRUE*, maka perintah di dalam blok *if* akan dijalankan. Namun, apabila nilainya *FALSE*, maka perintah dalam blok *else* yang akan dilaksanakan.

Kondisi Case :

Kondisi *case* adalah pernyataan atau variabel yang dievaluasi untuk menentukan apakah bernilai *TRUE* atau *FALSE*. Baik pada blok kondisi benar maupun salah, hanya diperbolehkan satu pernyataan yang dijalankan. Struktur ini dikenal sebagai *Switch ... Case* dalam pengendalian alur program.

```
switch ($var) {  
  case '1' : statement-1; break;  
  case '2' : statement-2; break;  
  ....  
}
```

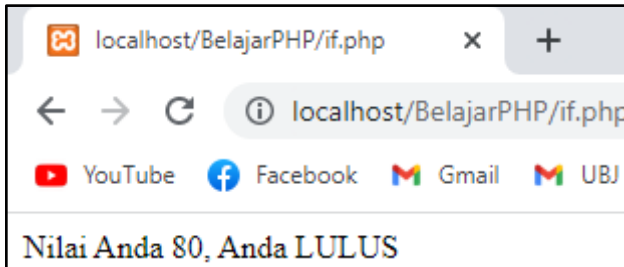
Keterangan :

- `$var` adalah elemen yang nilainya akan dievaluasi atau dibandingkan. Jenis data yang digunakan pada variabel ini bersifat fleksibel dan tidak dibatasi tipe tertentu.
- Sementara itu, nilai pada setiap *case* dapat berupa string, bilangan bulat (*integer*), nilai logika (*boolean*), bahkan ekspresi bersyarat. Penulisan nilai tersebut dapat menggunakan tanda kutip tunggal maupun ganda.

Nama File : if.php

Deskripsi : Program Struktur Kondisi If.

```
<?php  
$nilai = 80;  
if ($nilai >= 80) {  
    echo "Nilai Anda $nilai, Anda LULUS";  
} ?>
```



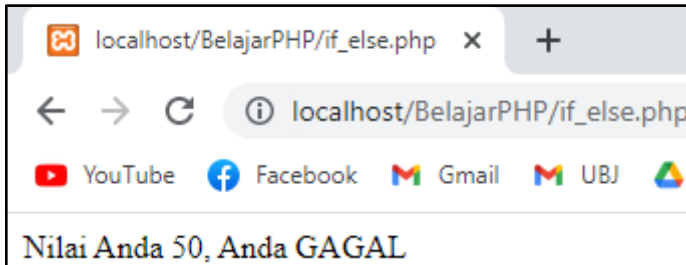
Gambar 3.37 Output if.php

Penjelasan Program: Pada kode tersebut, sistem melakukan pengecekan terhadap nilai yang disimpan dalam variabel `$nilai` pada baris ketiga. Jika nilai tersebut lebih dari 60, maka kondisi akan bernilai TRUE dan perintah di baris keempat akan dijalankan. Sebaliknya, apabila kondisi tidak terpenuhi (FALSE), maka perintah pada baris keempat tidak akan dijalankan.

Nama File : if_else.php

Penjelasan: Contoh program yang menerapkan struktur logika kondisional If..Else.

```
<?php
$nilai = 50;
if ($nilai >= 60) {
echo "Nilai Anda $nilai, Anda LULUS";
} else {
echo "Nilai Anda $nilai, Anda GAGAL";
}
?>
```



Gambar 3.38 Output if_else.php

Penjelasan Program:

Pada program tersebut, dilakukan evaluasi terhadap kondisi apakah nilai dalam variabel `$nilai` melebihi angka 60 (pada baris ke-3). Jika kondisi tersebut menghasilkan nilai *TRUE* (artinya isi variabel lebih dari 60), maka perintah pada baris ke-4 akan dijalankan. Namun, jika hasilnya *FALSE*, maka instruksi yang berada di baris ke-6 akan dijalankan. Pada contoh program ini, kondisi di baris ke-3 menghasilkan nilai *FALSE* karena variabel `$nilai` berisi 50.

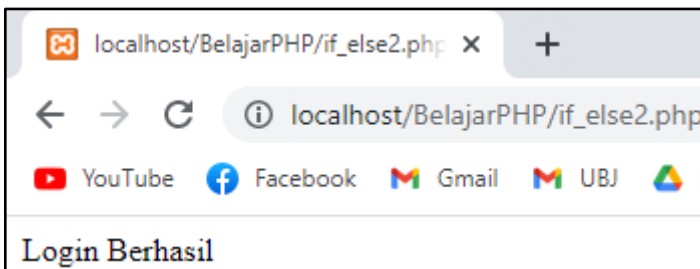
Nama File : if_else2.php

Penjelasan: Kode program untuk mengecek username dan password menggunakan pernyataan If..Else.

```
<?php
$user = "fried";
$pass = "123";
if ($user == "fried" and $pass == "123") {
    echo "Login Berhasil";
} else {
    echo "Login Gagal";
}
?>
```

Penjelasan Program:

Dalam program tersebut, dilakukan pengecekan terhadap dua kondisi sekaligus: apakah variabel \$nilai berisi nilai "fried" dan apakah variabel \$pass memiliki nilai "123" (pada baris ke-4). Apabila kedua kondisi tersebut terpenuhi atau bernilai *TRUE*, maka perintah di baris ke-5 akan dijalankan. Namun, jika salah satu atau kedua kondisi bernilai *FALSE*, maka instruksi yang ada di baris ke-7 akan dijalankan.



Gambar 3.39 Output if_else2.php

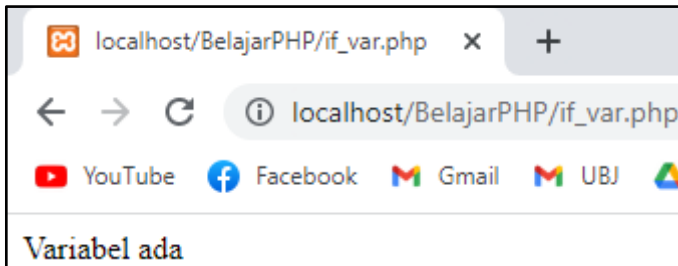
Nama File : if_var.php

Deskripsi: Program yang menentukan apakah variabel tersedia atau tidak

```
<?php
$user = "";
if (!isset($user)) {
    echo "Variabel tidak ada/belum terbentuk";
} else {
    echo "Variabel ada";
}
?>
```

Penjelasan Program :

Dalam program tersebut, digunakan fungsi `isset()` yang berfungsi untuk mengecek apakah sebuah variabel telah didefinisikan atau belum. Jika variabel tersebut tersedia, maka fungsi ini akan memberikan hasil *TRUE*. Oleh karena itu, apabila baris ke-2 dalam program dihapus atau diberi tanda komentar, maka instruksi yang berada di baris ke-4 akan dijalankan.



Gambar 3.40 Output if_var.php

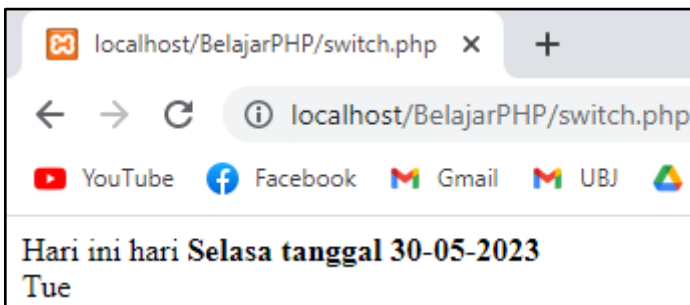
Nama File : switch.php

Deskripsi : Program menggunakan struktur Switch..Case bertujuan untuk menampilkan hari dalam format bahasa Indonesia

```

<?php
$day = date("D");
switch ($day) {
    case 'Sun': $hari = "Minggu"; break;
    case 'Mon': $hari = "Senin"; break;
    case 'Tue': $hari = "Selasa"; break;
    case 'Wed': $hari = "Rabu"; break;
    case 'Thu': $hari = "Kamis"; break;
    case 'Fri': $hari = "Jum'at"; break;
    case 'Sat': $hari = "Sabtu"; break;
    default: $hari = "Hari-hari Swalayan";
}
$tanggal = date('d-m-Y');
echo "Hari ini hari <b>$hari tanggal $tanggal</b>";
echo "<br>" . date('D') . " ";
?>

```



Gambar 3.41 Output switch.php

Penjelasan Program:

Pada program tersebut, variabel \$day (pada baris ke-2)

akan menyimpan tiga huruf pertama dari nama hari dalam bahasa Inggris. Dengan menggunakan fungsi `date()`, kita dapat mendapatkan informasi mengenai tanggal, hari, dan waktu saat ini. Fungsi ini dapat dibahas secara rinci pada pertemuan berikutnya. Selanjutnya, nilai dalam variabel `$day` akan dievaluasi menggunakan pernyataan `switch` (baris ke-3). Jika nilainya adalah 'Sun', maka variabel `$hari` akan diisi dengan "Minggu", dan begitu selanjutnya untuk hari-hari lainnya.

5. Pengulangan

1) Struktur Perulangan For

```
for (init_awal, kondisi, counter) {  
    statement-yang-diulang;  
}
```

Keterangan :

- `init_awal` adalah proses pemberian nilai awal pada sebuah variabel.
- Kondisi adalah pernyataan logika yang digunakan untuk menentukan batas berlangsungnya perulangan.
- *Counter* berfungsi untuk menambah atau mengurangi nilai variabel agar proses perulangan dapat terus berlangsung hingga kondisi terpenuhi.

2) Struktur Perulangan While

```
init_awal;  
while (kondisi) {  
    statement-yang-diulang;  
    counter;  
}
```

Keterangan :

- *init_awal* adalah proses pemberian nilai awal pada sebuah variabel.
- *Kondisi* adalah pernyataan logika yang digunakan untuk menentukan batas berlangsungnya perulangan.
- *Counter* berfungsi untuk menambah atau mengurangi nilai variabel agar proses perulangan dapat terus berlangsung hingga kondisi terpenuhi.

3) Struktur Perulangan Do ... while

```
init_awal;  
do {  
statement-yang-diulang;  
counter;  
} while (kondisi);
```

Keterangan :

- *init_awal* adalah pengaturan atau penetapan nilai awal untuk variabel.
- *Kondisi* adalah pernyataan yang menentukan batasan dalam perulangan.
- *Counter* adalah penambahan atau pengurangan nilai dari variabel yang memastikan perulangan tetap berlangsung.
- Dalam struktur *do...while*, pengecekan kondisi dilakukan setelah eksekusi, sehingga pernyataan dalam blok *do...while* akan dijalankan setidaknya satu kali.

4) Struktur Perulangan Foreach

```
foreach (array_expression as $value)  
statement;
```

```
foreach (array_expression as $key => $value)
statement;
```

Keterangan :

Struktur foreach umumnya dipakai untuk melakukan perulangan pada setiap elemen array. Perulangan akan selesai ketika seluruh elemen dalam array telah diproses.

Nama File : for.php

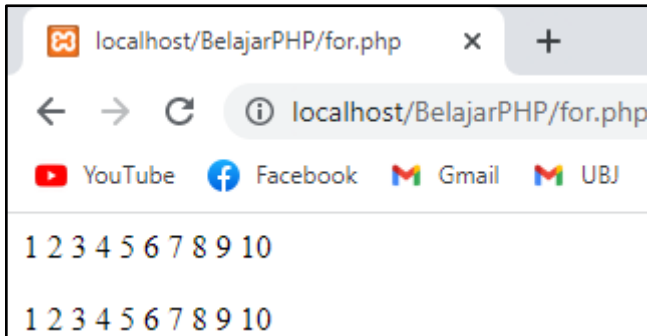
Deskripsi: Program yang menggunakan struktur perulangan For beserta variasi-variasinya.

```
<?php
// contoh 1
for ($i = 1; $i <= 10; $i++) {
    echo "$i ";
}
echo "<br><br>";
// contoh 2
for ($i = 1;; $i++) {
    if ($i > 10) {
        break;
    } // akan dieksekusi secara berkala sampai kondisinya tidak
    terpenuhi
    echo "$i ";
}
?>
```

Penjelasan Program:

Program di atas menunjukkan berbagai jenis perulangan yang menggunakan struktur for. Contoh pertama (baris 4-

6) adalah jenis yang paling sering digunakan. Sedangkan pada contoh kedua, batas akhir perulangan tidak ditentukan di dalam for, melainkan dikendalikan dengan if dan break (baris 11-13).



Gambar 3.42 Output for.php

Nama File : while.php

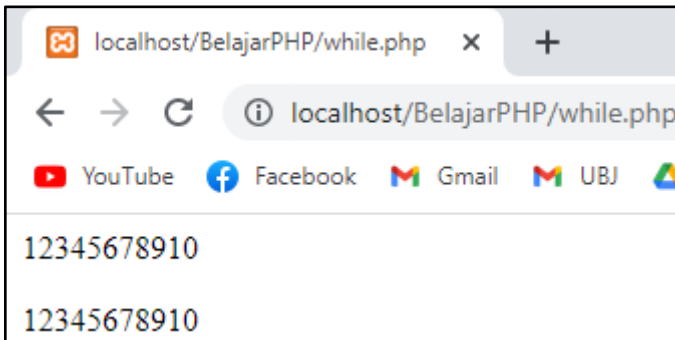
Deskripsi: Program yang menggunakan struktur perulangan While beserta variasi-variasinya.

```
<?php
// contoh 1
$i = 1;
while ($i <= 10) {
    echo $i++;
}
echo "<br><br>";
// contoh 2
$i = 1;
while ($i <= 10) :
    echo "$i";
```

```
$i++;  
endwhile;  
?>
```

Penjelasan Program:

Program di atas menunjukkan berbagai variasi perulangan menggunakan while. Contoh pertama (baris 3-6) adalah jenis yang paling sering digunakan. Blok perintah (baris 5) akan dijalankan selama kondisi di baris ke-4 terpenuhi dengan nilai TRUE. Sedangkan contoh kedua adalah bentuk lain dari while, yang jarang dipakai.



Gambar 3.43 Output while.php

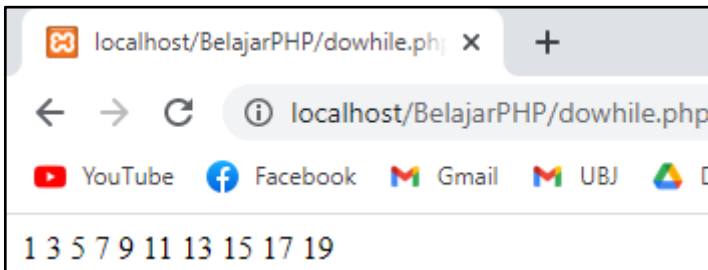
Nama File : dowhile.php

Deskripsi : Program yang menggunakan struktur Perulangan Do...While.

```
<?php
$i = 1;
do {
    echo "$i ";
    $i+=2;
} while ($i <= 20);
?>
```

Penjelasan Program

Program tersebut menampilkan angka ganjil dari 1 hingga 20 dengan menggunakan struktur perulangan do...while. Dalam struktur perulangan ini, pengecekan kondisi dilakukan setelah dijalankan.



Gambar 3.44 Output dowhile.php

Struktur Break dan Continue

Nama File : break.php

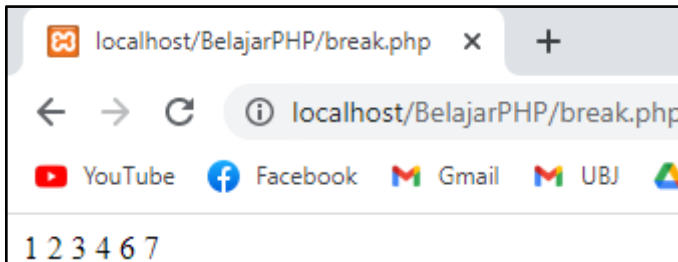
Deskripsi : Program yang menggunakan struktur control Break dan Continue.

```
<?php
for ($i=1; $i<10; $i++) {
    if ($i == 5)
        continue;
```

```
if ($i == 8)
break;
echo "$i ";
}
?>
```

Penjelasan Program:

Program tersebut menunjukkan bahwa perintah continue akan mengabaikan perintah setelahnya dalam satu siklus perulangan dan langsung melanjutkan ke perulangan selanjutnya. Dengan demikian, perintah pada baris 6-8 akan dilewati (angka 5 tidak akan ditampilkan). Sementara itu, perintah break berfungsi untuk menghentikan perulangan dan keluar dari blok perulangan secara langsung.



Gambar 3.45 Output break.php

5. Penanganan Form

Formulir input dibuat menggunakan tag-tag HTML. Halaman yang hanya berisi form (tanpa script PHP) tidak perlu disimpan dalam format PHP, bisa disimpan dalam format HTML. Untuk membuat sebuah formulir input, terdapat minimal tiga poin penting yang perlu diperhatikan, yaitu:

a. METHOD

Method pada sebuah formulir berfungsi untuk menentukan bagaimana data yang diinput dikirimkan ke server. Terdapat dua metode utama, yaitu GET dan POST. Keduanya menentukan proses pengiriman dan penanganan data oleh PHP.

b. ACTION

Action pada formulir menentukan lokasi untuk pemrosesan data inputan. Jika atribut action tidak diisi, maka data akan diproses pada halaman yang sama. Dengan demikian, halaman formulir dan halaman pemrosesan bisa digabungkan atau dipisahkan.

c. SUBMIT BUTTON

Tombol Submit berfungsi untuk mengirimkan data dari formulir input. Saat tombol ini diklik, data dari formulir akan dikirimkan dan diproses pada halaman yang ditentukan dalam atribut action.

Berbagai Cara Penanganan Form

Cara 1 : Menyatukan Form antara Form dan Proses

Pengolahan formulir dilakukan pada halaman yang sama dengan formulir input. Jika pemrosesan formulir berada dalam satu halaman, atribut action pada tag form dapat dibiarkan kosong.

Nama File : input01.php

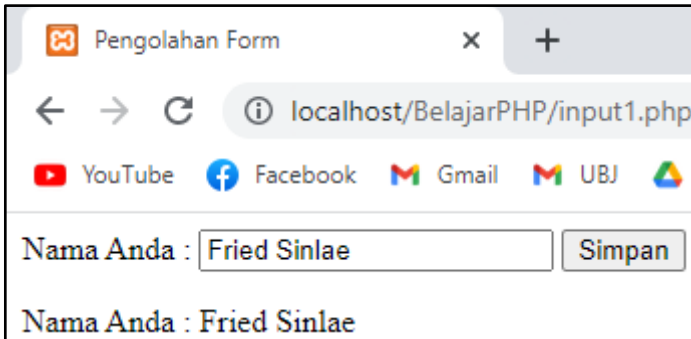
Deskripsi: Program yang menunjukkan contoh pengolahan formulir, di mana formulir input dan proses pengolahan data berada dalam halaman yang sama.

```
<html>

<head>
  <title>Pengolahan Form</title>
</head>

<body>
  <FORM ACTION="" METHOD="POST">
    Nama Anda : <input type="text" name="nama">
    <input type="submit" name=" kirim" value="Simpan">
  </FORM>
</body>

</html>
<?php
if (isset($_POST[' kirim'])) {
  $nama = $_POST['nama'];
  echo "Nama Anda : $nama";
}
?>
</body>
```



Gambar 3.46 Output input1.php

Penjelasan Program:

Hal-hal yang perlu diperhatikan dalam program ini, antara lain nama setiap komponen form, yang akan menjadi indeks array dalam PHP. Dalam program tersebut, atribut action pada tag form kosong (baris 4), yang berarti pemrosesan formulir terjadi di halaman yang sama. Di baris 4, metode yang digunakan untuk memproses formulir adalah POST, yang lebih disarankan untuk menangani data inputan. Di baris 11 hingga 16, terdapat skrip PHP yang memproses data dari formulir. Lokasi pemrosesan bisa diletakkan sebelum atau setelah formulir, tergantung pada kebutuhan.

Pada baris ke-12, skrip memeriksa apakah tombol yang bernama "Input" (dengan mencocokkan \$_POST dan nilai name tombol submit di form) telah ditekan oleh pengguna. Pada baris ke-13, nilai yang dimasukkan di formulir akan diambil melalui akses ke array \$_POST sesuai dengan komponen yang bersangkutan. Pada baris 13, nilai dari input dengan nama "nama" diambil dan disimpan dalam variabel \$nama. Indeks \$_POST di baris 13 harus sesuai dengan atribut name di baris ke-5.

Cara 2 : Memisahkan antara Form dan Proses

Proses pengolahan formulir dilakukan di halaman terpisah. Jika pemrosesan dilakukan di halaman lain, atribut action pada tag form harus mencantumkan alamat halaman tersebut.

Nama File : input2.php

Deskripsi: Program menyajikan form input menggunakan metode POST.

```
<html>

<head>
  <title>Pengolahan Form</title>
</head>

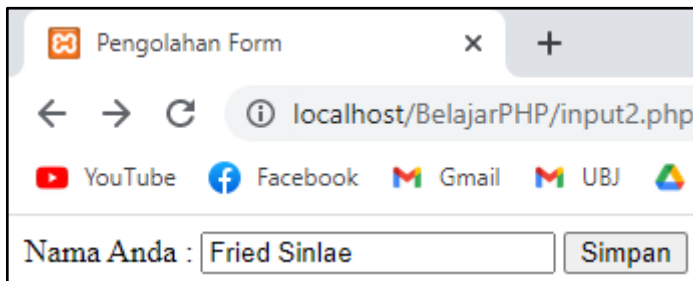
<body>
  <FORM ACTION="aksi_input2.php" METHOD="POST">
    <input type="hidden" name="takterlihat" value="Kiriman Tak Terlihat">
    Nama Anda : <input type="text" name="namanya">
    <input type="submit" name="submit" value="Simpan">
  </FORM>
</body>

</html>
```

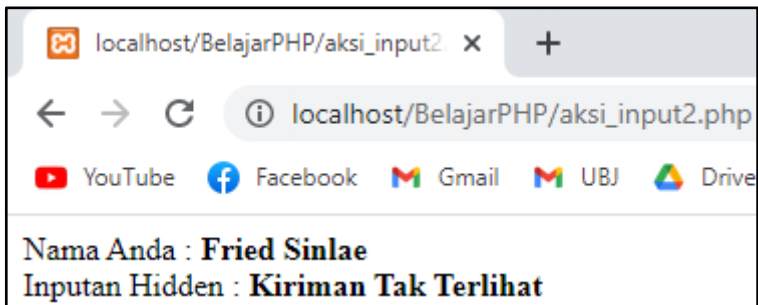
Nama File : aksi_input2.php

Deskripsi: Program pemrosesan form input pada input2.php.

```
<?php
$nama = $_POST['namanya'];
$gakeliatan = $_POST['takterlihat'];
echo "Nama Anda : <b>$nama</b><br>";
echo "Inputan Hidden : <b>$gakeliatan</b>";
?>
```



Gambar 3.47 Output input2.php



Gambar 3.48 Output aksi_input2.php

Nama File : input3.php

Deskripsi: Program menyajikan form input dengan metode GET.

```
<html>

<head>
  <title>Pengolahan Form</title>
</head>

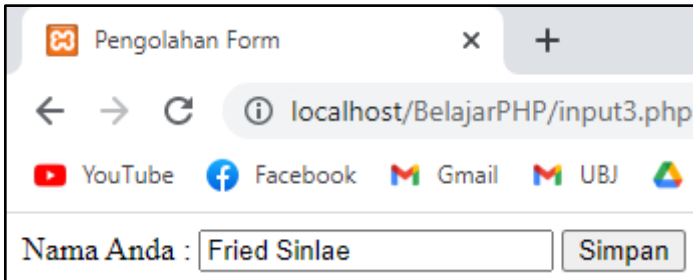
<body>
  <FORM ACTION="aksi_input3.php" METHOD="GET">
    Nama Anda : <input type="text" name="namanya">
    <input type="submit" name="submit" value="Simpan">
  </FORM>
</body>

</html>
```

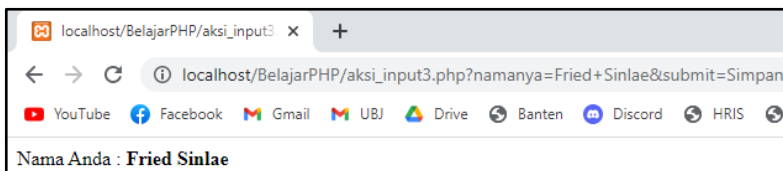
Nama File : aksi_input3.php

Deskripsi: Program pemrosesan form input di input3.php.

```
<?php
$nama = $_GET['namanya'];
echo "Nama Anda : <b>$nama</b>";
?>
```



Gambar 3.49 Output input3.php



Gambar 3.50 Output aksi_input3.php

Untuk method “GET” variabel yang diinput terlihat di URL.

Nama File : form.php

Deskripsi: Program menyajikan form input secara lengkap menggunakan metode POST.

```
<html>

<head>
  <title>Form Lengkap</title>
</head>

<body>
  <h2>Form Input Mahasiswa</h2>
  <form method='POST' action='aksi_form.php'>
```

```

<table border='0' width='100%>
  <tr>
    <td>NIM</td>
    <td><input type='text' name='nim'></td>
  </tr>
  <tr>
    <td>Nama</td>
    <td><input type='text' name='nama_mahasiswa'>
  </tr>
  <tr>
    <td>Tanggal Lahir</td>
    <td><input type='date' name='tgl_lahir'>
  </tr>
  <tr>
    <td>Jenis Kelamin</td>
    <td>
      <input type='radio' name='jk' value='L'> Laki-Laki
      <input type='radio' name='jk' value='P'> Perempuan
    </td>
  </tr>
  <tr>
    <td>Agama</td>
    <td>
      <select name='agama'>
        <option value='Islam'>Islam</option>
        <option value='Katolik'>Katolik</option>
      </select>
    </td>
  </tr>

```

```

        <option value='Protestan'>Protestan</option>
        <option value='Hindu'>Hindu</option>
        <option value='Budha'>Budha</option>
        <option value='Konghucu'>Konghucu</option>
    </select>
</td>
</tr>
<tr>
    <td>Alamat</td>
    <td><textarea name='alamat' cols='70'></textarea></td>
</tr>
<tr>
    <td>Hobi</td>
    <td>
        <input type='checkbox' name='hobi1' value='Membaca'>
            Membaca<br>
        <input type='checkbox' name='hobi2' value='Olahraga'>
            Olahraga<br>
        <input type='checkbox' name='hobi3' value='Hiking'>
            Hiking<br>
    </td>
</tr>
<tr>
    <td>Password</td>
    <td><input type='password' name='sandi'></td>
</tr>

```

```

        <tr>
                                <td colspan='2'><input type='submit'
value='Simpan'></td>
        </tr>
    </table>
</body>

</html>

```

Nama File : aksi_form.php

Deskripsi : Program menampilkan hasil imputan dari file form.php.

```

<?php
$nim = $_POST['nim'];
$nama_mahasiswa = $_POST['nama_mahasiswa'];
$ttl_lahir = $_POST['ttl_lahir'];
$jk = $_POST['jk'];
$agama = $_POST['agama'];
$alamat = $_POST['alamat'];

if (isset($_POST['hobi1'])) {
    $hobi1 = $_POST['hobi1']; //melakukan aksi dengan variabel
    $hobi1
} else {
    $hobi1 = ""; //variabel hobi1 tidak ditemukan, lakukan aksi
    alternatif jika diperlukan
}

```

```

if (isset($_POST['hobi2'])) {
    $hobi2 = $_POST['hobi2']; //melakukan aksi dengan variabel $hobi2
} else {
    $hobi2 = ""; //variabel hobi2 tidak ditemukan, lakukan aksi alternatif jika diperlukan
}

if (isset($_POST['hobi3'])) {
    $hobi3 = $_POST['hobi3']; //melakukan aksi dengan variabel $hobi3
} else {
    $hobi3 = ""; //variabel hobi3 tidak ditemukan, lakukan aksi alternatif jika diperlukan
}

$sandi = $_POST['sandi'];
echo "
NIM : $nim <br>
Nama Mahasiswa : $nama_mahasiswa <br>
Tanggal Lahir : $tgl_lahir <br>
Jenis Kelamin : $jk <br>
Agama : $agama <br>";
if ($hobi1 != "") {
    echo "Hobi 1 : $hobi1 <br>";
}
if ($hobi2 != "") {

```

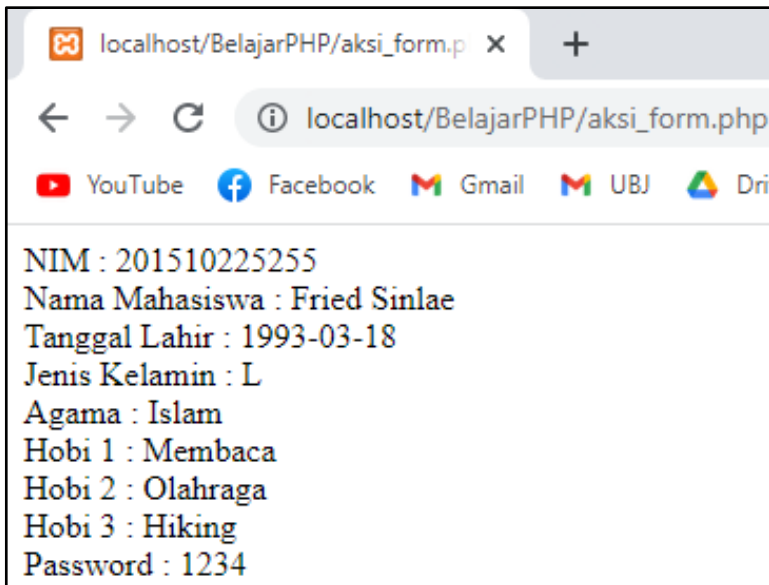
```
echo "Hobi 2 : $hobi2 <br>";  
}  
if ($hobi3 != "") {  
    echo "Hobi 3 : $hobi3 <br>";  
}  
echo "Password : $sandi <br>";  
?>
```

The screenshot shows a web browser window with the title 'Form Lengkap'. The address bar displays 'localhost/BelajarPHP/form.php'. The browser's bookmark bar includes links to YouTube, Facebook, Gmail, UBJ, Drive, Banten, Discord, HRIS, HRIS Local, and PhpMyAdmin. The main content area is titled 'Form Input Mahasiswa' and contains the following form fields:

- NIM: Text input with value '201510225255'
- Nama: Text input with value 'Fried Sinlae'
- Tanggal Lahir: Date picker showing '18-Mar-1993'
- Jenis Kelamin: Radio buttons for 'Laki-Laki' (selected) and 'Perempuan'
- Agama: Dropdown menu showing 'Islam'
- Alamat: Text area with value 'Perum. Pejuang Pratama Blok G No 17. Kel. Pejuang Kec. Medan Satria'
- Hobi: Checkboxes for 'Membaca', 'Olahraga', and 'Hiking' (all selected)
- Password: Password input field with masked characters '....'

A 'Simpan' button is located at the bottom left of the form.

Gambar 3.51 Output form.php



Gambar 3.52 Output aksi_form.php

Upload File

PHP, yang berfungsi sebagai *scripting* sisi server, mendukung proses unggah file ke server. Berikut beberapa hal yang perlu diperhatikan dalam proses ini:

- Form HTML perlu menambahkan atribut `ENCTYPE="multipart/form-data"`.
- Input form untuk mengunggah file dapat menggunakan tag `<input>` dengan atribut `TYPE="FILE"`.
- Untuk menangani input, PHP menyediakan variabel array global berikut:
 - `$_FILES`. Index dari variabel ini antara lain :
 - `$_FILES['file']['name']` : Nama asli dari file yang diupload

- 3) \$_FILES['file']['tmp_name'] : Nama temporary file yang diupload
- 4) \$_FILES['file']['size'] : Ukuran file asli (dalam byte)
- 5) \$_FILES['file']['type'] : MIME type file yang diupload
- 6) Folder tujuan untuk upload file harus dapat diakses dan ditulis, biasanya dengan permission 777 atau 775 di Linux.

Nama File : form_upload.php

Deskripsi: Program menampilkan form untuk unggah file.

```
<html>

<head>
  <title>Upload File</title>
</head>

<body>
  <FORM ACTION="aksi_upload.php" METHOD="POST"
  ENCTYPE="multipart/form-data">
    Upload File : <input type="file" name="fupload"><br>
    <input type="submit" name="upload" value="Upload">
  </FORM>
</body>

</html>
```

Nama File : aksi_upload.php

Deskripsi: Program yang memproses upload file. Sebelum

membuat program ini pastikan anda membuat folder baru dengan nama gambar di folder latihan ini (saya menggunakan folder BelajarPHP).

```
<?php
$target_dir = "uploads/"; // folder untuk menyimpan file yang
diunggah
$target_file = $target_dir . basename($_FILES["fupload"]["name"]);
// path lengkap file yang diunggah
$uploadOk = 1;
$imageFileType = strtolower(pathinfo($target_file,
PATHINFO_EXTENSION));

if (isset($_POST["submit"])) { // Periksa apakah file sudah diunggah
atau belum
    $check = getimagesize($_FILES["fupload"]["tmp_name"]);
    if ($check !== false) {
        echo "File adalah gambar - " . $check["mime"] . ".";
        $uploadOk = 1;
    } else {
        echo "File bukan gambar.";
        $uploadOk = 0;
    }
}

if (file_exists($target_file)) { // Periksa apakah file sudah ada di
server
    echo "Maaf, file sudah ada.";
```

```

    $uploadOk = 0;
}

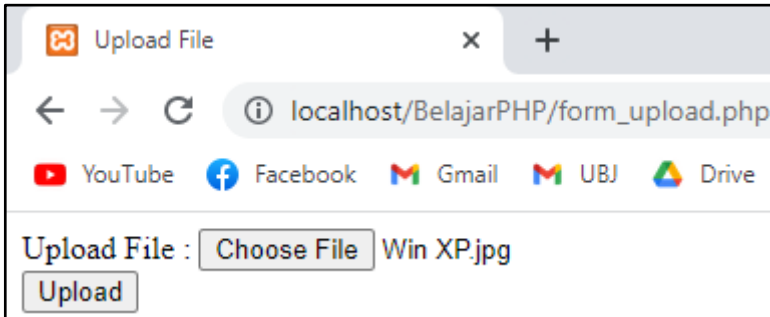
if ($_FILES["fupload"]["size"] > 500000) { // Periksa ukuran file
    echo "Maaf, file terlalu besar.";
    $uploadOk = 0;
}

if (
    $imageFileType != "jpg" && $imageFileType != "png" &&
    $imageFileType != "jpeg"
    && $imageFileType != "gif"
) { // Izinkan hanya beberapa jenis file
    echo "Maaf, hanya file JPG, JPEG, PNG & GIF yang diizinkan.";
    $uploadOk = 0;
}

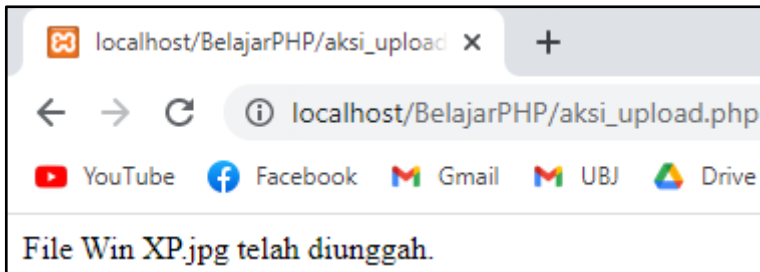
if ($uploadOk == 0) { // Periksa apakah $uploadOk = 0, artinya
    terdapat error
    echo "Maaf, file tidak dapat diunggah.";
} else { // Jika semua pemeriksaan sukses, maka file akan diunggah
    if (move_uploaded_file($_FILES["fupload"]["tmp_name"],
    $target_file)) {
        echo "File " . basename($_FILES["fupload"]["name"]) . " telah
        diunggah.";
    } else {
        echo "Maaf, terdapat kesalahan saat mengunggah file.";
    }
}

```

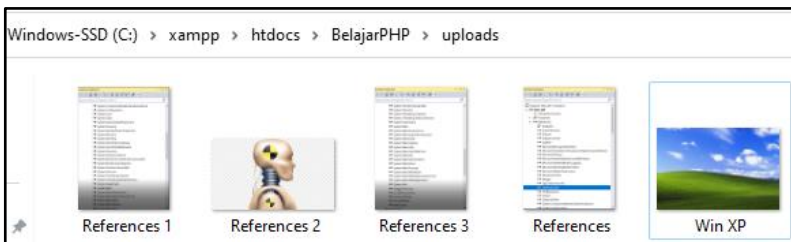
```
}  
  
}
```



Gambar 3.53 Output form_upload.php



Gambar 3.54 Output aksi_upload.php



Gambar 3.55 Berhasil Upload Gambar

D. MySQL

MySQL adalah DBMS freeware yang sangat populer di kalangan pengguna database. MySQL tidak hanya gratis, tetapi juga merupakan DBMS yang ringan dan mudah digunakan. Beberapa perangkat lunak yang mendukung layanan MySQL antara lain WAMP, PhpTriad, dan XAMPP. Namun, XAMPP menjadi yang paling populer saat ini karena pembaruan perangkat lunaknya yang konsisten.

XAMPP adalah paket perangkat lunak yang menggabungkan beberapa aplikasi, termasuk MySQL, Apache, FileZilla, Mercury, dan Tomcat. Dengan XAMPP, pengguna tidak perlu menginstal Apache dan MySQL secara terpisah. Dalam buku ini tidak dibahas mengenai pengenalan database MySQL secara mendalam dikarenakan, seharusnya bagi yang ingin mempelajari pemrograman web sudah terlebih dahulu mempelajari dasar database. Berikut adalah perintah dasar untuk memanipulasi data yang sering digunakan. Buat database baru bernama DBBelajarPHP di phpMyAdmin, kemudian buat tabel baru bernama buku tamu dengan struktur sebagai berikut:

```
CREATE TABLE 'bukutamu' (  
  'id' int(11) NOT NULL,  
  'nama' varchar(50) NOT NULL,  
  'email' varchar(50) NOT NULL,  
  'kota' varchar(50) NOT NULL  
)  
INSERT INTO 'bukutamu' ('id', 'nama', 'email', 'kota') VALUES  
(1, 'Fried Sinlae', 'sinlae@gmail.com', 'Bekasi'),
```

```
(2, 'Ikbal Eka Saputra', 'ikbal@gmail.com', 'Jakarta'),
(3, 'Linda Fitriyani', 'linda@gmail.com', 'Bekasi'),
(4, 'Kaka Irawan', 'kaka@gmail.com', 'Medan'),
(5, 'Rifqi Hafidz', 'rifqi@gmail.com', 'Makasar');
```

1. Perintah Select

Fungsi ini digunakan untuk menampilkan data dari tabel, seperti `SELECT * FROM [nama_table]` untuk menampilkan semua data, atau `SELECT nama, email FROM [nama_table]` untuk menampilkan kolom tertentu, seperti nama dan email.

2. Select Distinct

Fungsi ini digunakan untuk memilih data unik (menghapus duplikasi) dari sebuah tabel database.

```
SELECT DISTINCT [nama_kolom] FROM [nama_table]
```

contoh `SELECT DISTINCT kota FROM bukutamu`
`WHERE` `CLAUSE`

Fungsi ini digunakan untuk menyaring data berdasarkan kondisi yang diperlukan.

```
SELECT [nama_kolom] FROM [nama_table] WHERE
column_name [operator_value]
```

Contoh `SELECT * FROM bukutamu WHERE`
`kota='YOGYAKARTA' ORDER BY`

Digunakan untuk mengurutkan data berdasarkan kolom tertentu, dengan urutan default menaik (*ascending*). Untuk mengurutkan secara menurun (*descending*), tambahkan perintah `DESC`.

```
SELECT * FROM [nama_table] ORDER BY
[nama_kolom] ASC/DESC
```

Contoh `SELECT * FROM bukutamu ORDER BY nama DESC LIKE %%`

Digunakan bersama perintah `WHERE` untuk menemukan data sesuai dengan kriteria tertentu.

`SELECT * FROM [nama_table] WHERE [nama_kolom] LIKE pattern`

Contoh `SELECT * FROM bukutamu WHERE nama LIKE 'a%'`

Contoh di atas mencari data berdasarkan kolom nama yang diawali dengan huruf "a".

2. IN

Digunakan untuk mencari data dengan menerapkan beberapa kondisi filter pada perintah `WHERE`.

`SELECT [nama_kolom] FROM [nama_table] WHERE [nama_kolom] IN (value1,value2,...)`

contoh `SELECT * FROM bukutamu WHERE kota IN ('Yogyakarta','Jakarta')`

Between digunakan untuk menetapkan rentang nilai dalam pencarian data.

`SELECT [nama_kolom] FROM [nama_table] WHERE [nama_kolom] BETWEEN value1 AND value2`

Contoh `SELECT * FROM bukutamu WHERE id BETWEEN 5 and 15`

Contoh di atas mencari data dengan nomor ID yang berada di antara 5 dan 15.

3. Insert Into

Fungsi ini digunakan untuk menambahkan data baru ke dalam tabel database.

```
INSERT INTO table_name VALUES
(value1,value2,value3,...)
```

Contoh menambahkan 1 record

```
INSERT INTO bukutamu VALUES
(1,'Arini','arini@mail.com','Yogyakarta')
```

Contoh menambahkan banyak record sekaligus:

```
CREATE TABLE kategoriaset (
'id_kategoriaset' int(11) NOT NULL,
'kd_kategoriaset' varchar(15) DEFAULT NULL,
'nama_kategoriaset' varchar(45) DEFAULT NULL
)
INSERT INTO kategoriaset ('id_kategoriaset', 'kd_kategoriaset',
'nama_kategoriaset') VALUES
(1, '01', 'Mesin'),
(2, '02', 'Tools'),
(3, '03', 'Kendaraan');
```

3. Update

Digunakan untuk memperbarui data dalam tabel. Perintah Update biasanya diikuti oleh klausa Where sebagai penentu kondisi.

```
UPDATE table_name SET
column1=value,column2=value,... WHERE some_column =
some_value
```


bukutamu LEFT JOIN order ON bukutamu.id=order.id
ORDER BY bukutamu.nama RIGHT JOIN

Digunakan untuk menampilkan data dari tabel kiri yang tidak cocok dengan data di tabel kanan.

```
SELECT column_name(s) FROM table_name1 RIGHT
JOIN                table_name2                ON
table_name1.column_name=table_name2 column_name
```

Contoh SELECT
bukutamu.nama,bukutamu.email,order.no_order FROM
bukutamu RIGHT JOIN order ON bukutamu.id=order.i
ORDER BY bukutamu.nama FULL JOIN

Digunakan untuk menampilkan data jika terdapat kesamaan pada salah satu tabel.

```
SELECT column_name(s) FROM table_name1 FULL
JOIN                table_name2                ON
table_name1.column_name=table_name2 column_name;
```

Contoh SELECT
bukutamu.nama,bukutamu.email,order.no_order FROM
bukutamu FULL JOIN order ON bukutamu.id=order.id
ORDER BY bukutamu.nama

6. Union

Dipakai untuk mengombinasikan hasil dari beberapa pernyataan SELECT.

```
SELECT column_name(s)FROM table_name1 UNION
column_name(s) FROM table_name2
```

Contoh SELECT nama FROM mhs_kampus1 UNION
SELECT nama FROM mhs_kampus

7. Avg() (Average)

Dipakai untuk menghitung nilai rata-rata dari data yang tersedia.

```
SELECT AVG (column_name) FROM table_name
Contoh SELECT AVG(harga) AS Harga_rata2 FROM order
```

8. Count()

Dipakai untuk mengetahui total jumlah data yang ada.

```
SELECT COUNT (column_name) FROM table_name
Contoh SELECT COUNT(id) AS Jumlah_tamu FROM
bukutamu
```

9. Max()

Dipakai untuk menemukan nilai terbesar dari data yang ada.

```
SELECT MAX (column_name) FROM table_name
Contoh SELECT MAX(harga) AS Harga_termahal FROM
order
```

10.Min()

Dipakai untuk mencari nilai terkecil dari data yang ada.

```
SELECT MIN (column_name) FROM table_name
Contoh SELECT MIN(harga) AS Harga_termurah
FROM order SUM()
```

Dipakai untuk menjumlahkan semua data yang tersedia.

```
SELECT SUM (column_name) FROM table_name
Contoh SELECT SUM(harga) AS Harga_total FROM order
```

11. Group By()

Dipakai untuk mengkategorikan data sesuai dengan kriteria tertentu.

```
SELECT  
column_name, aggregate_function(column_name) FROM  
table_name WHERE column_name operator value GROUP  
BY column_name
```

Contoh `SELECT nama_customer, SUM(harga) FROM order GROUP BY nama_customer`

12. Round()

Dipakai untuk menghasilkan angka pecahan.

```
SELECT ROUND (column_name, decimals) FROM  
table_name
```

Contoh `SELECT no_mhs, ROUND (nilai, 0) as nilai_bulat FROM tnilai`

E. *Create Read Update dan Delete*

Dalam proses belajar pemrograman, penting untuk memahami cara membuat program yang dapat mengelola data, seperti menambahkan, menampilkan, mengubah, dan menghapus informasi yang dikenal dengan istilah CRUD (*Create, Read, Update, Delete*). Pada bagian ini akan dijelaskan langkah-langkah untuk membuat program CRUD menggunakan sebuah tabel data yang telah tersedia sebelumnya.

1. Membuat File Koneksi

Buatlah file baru dengan nama koneksi.php Tuliskan kode sebagai berikut.

```
<?php
$server = "localhost";
$username = "fried";
$password = "Ubj2023";
$database = "dbbelajarphp";
$koneksi = mysqli_connect($server, $username, $password,
$database)
    or die("Database tidak bisa dibuka");
?>
```

2. Membuat file tampil buku tamu

Buatlah file baru dengan nama tamu.php Tuliskan kode sebagai berikut.

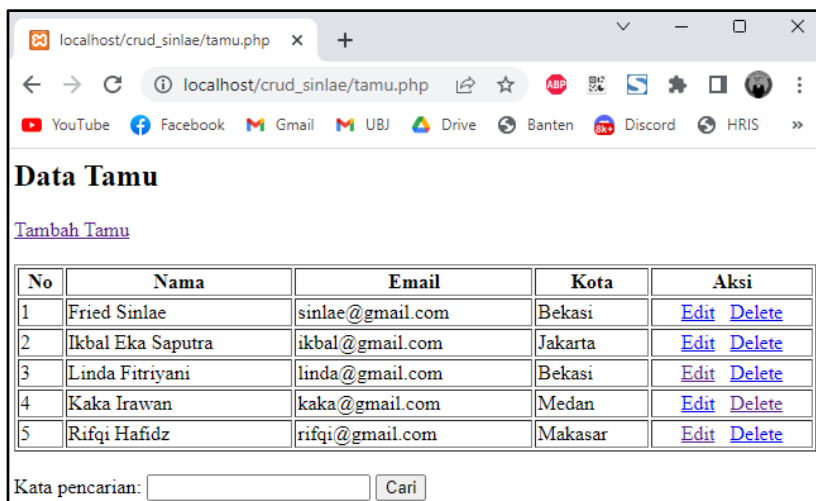
```
<?php
include "koneksi.php";
echo "
<h2>Data Tamu</h2>
<a href='tamu_input.php'>Tambah Tamu</a>
<br><br>
<table border='1' width='100%'>
<tr>
<th>No</th><th>Nama</th><th>Email</th><th>Kota</th><th>A
ksi</th>
</tr>";
$stampil = mysqli_query($koneksi, "SELECT * FROM bukutamu");
$no = 1;
```

```

while ($r = mysqli_fetch_array($tampil)) {
    echo "
<tr>
<td>$no</td>
<td>$r[nama]</td>
<td>$r[email] </td>
<td>$r[kota]</td>
<td align='center'>
<a href='tamu_edit.php?id=$r[id]'>Edit</a> &nbsp;
<a href='aksi_tamu_delete.php?id=$r[id]'>Delete</a>
</td>
</tr>";
    $no = $no + 1;
}
echo "
</table><br>
<form method='POST' action='tamu_cari.php'>
Kata pencarian: <input type='text' name='keyword'>
<input type='submit' value='Cari'>
</form>";
?>

```

Jalankan program aplikasi tamu.php dengan tampilan seperti terlihat pada gambar di bawah:



Gambar 3.56 Output tamu.php

3. Membuat program input tamu

Buatlah file baru dengan nama tamu_input.php
Tuliskan kode sebagai berikut :

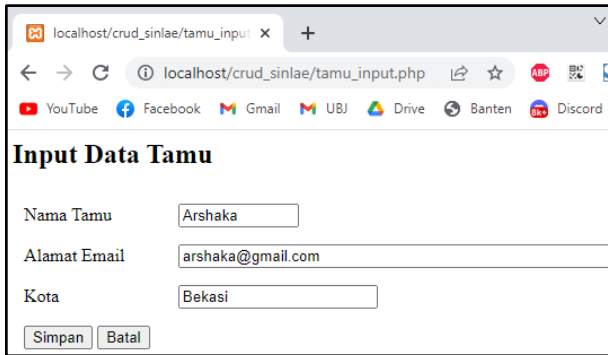
```
<?php
include "koneksi.php";
echo "
<h2>Input Data Tamu</h2>
<form method='POST' action='aksi_tamu_input.php'
enctype='multipart/form-data'>
<table border='0' width='100%' cellpadding='5'>
<tr><td>Nama Tamu</td><td><input type='text' name='nama'
size='10'></td></tr>
<tr><td>Alamat Email</td><td><input type='text' name='email'
size='50'></td></tr>
<tr><td>Kota</td><td><input type='text' name='kota'></td></tr>
```

```
<tr>
<td colspan=2>
<input type='submit' value='Simpan'>
<input type='reset' value='Batal'>
</td>
</tr>
</table>
</form>";
?>
```

Buatlah file baru dengan nama aksi_tamu_input.php yang berisi perintah untuk menyimpan data inputan dari form tamu_input.php. Tuliskan kode sebagai berikut :

```
<?php
include "koneksi.php";
mysqli_query($koneksi,"INSERT INTO bukutamu(nama,email,kota)
VALUES('$_POST[nama]','$_POST[email]','$_POST[kota]')");
echo "<a href='tamu.php'>Data Sudah Masuk...!</a>";
?>
```

Jalankan program siswa_input melalui program induknya yaitu tamu.php, kemudian klik link 'Tambah Tamu'. Berikut adalah tampilan program tamu_input.php.



localhost/crud_sinlae/tamu_input x +

localhost/crud_sinlae/tamu_input.php

YouTube Facebook Gmail UBJ Drive Banten Discord

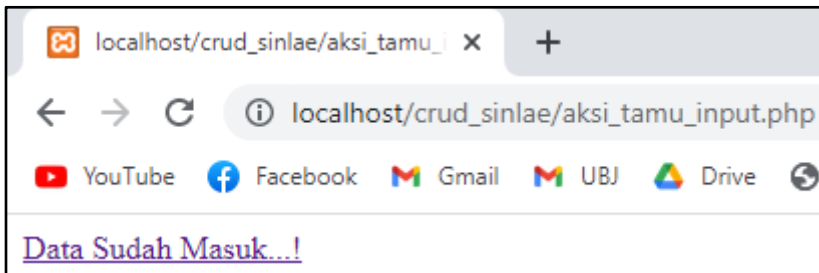
Input Data Tamu

Nama Tamu

Alamat Email

Kota

Gambar 3.57 Output tamu_input.php



Gambar 3.58 Output aksi_tamu_input.php

4. Membuat program Edit Tamu

Pada prinsipnya program tamu_edit hampir sama dengan program tamu_input, hanya saja pada program edit kita perlu menampilkan data lamanya saja, sehingga diperlukan query untuk memanggil data tersebut berdasarkan field kunci. Buatlan file baru dengan tamu_edit.php Tuliskan kode sebagai berikut :

```
<?php  
include "koneksi.php";
```

```

$stampil=mysqli_query($koneksi,"SELECT * FROM buktamu where
id='$_GET[id]' ");
$r=mysqli_fetch_array($stampil);
echo "
<h2>Edit Data Tamu</h2>
<table border='0' width='100%' cellpadding='5' cellspacing='5'>
<form method='POST' action='aksi_tamu_edit.php'
enctype='multipart/form-data'>
<input type='hidden' name='id' value='$r[id]'>
<tr><td>Nama Tamu</td><td><input type='text' name='nama'
value='$r[nama]'></td></tr>
<tr><td>Alamat Email</td><td><input type='text' name='email'
value='$r[email]'></td></tr>
<tr><td>Kota</td><td><input type='text' name='kota'
value='$r[kota]'></td></tr>
<tr>
<td colspan=2>
<input type='submit' value='Update'>
<input type='reset' value='Reset'>
</td>
</tr>
</form>
</table>";
?>

```

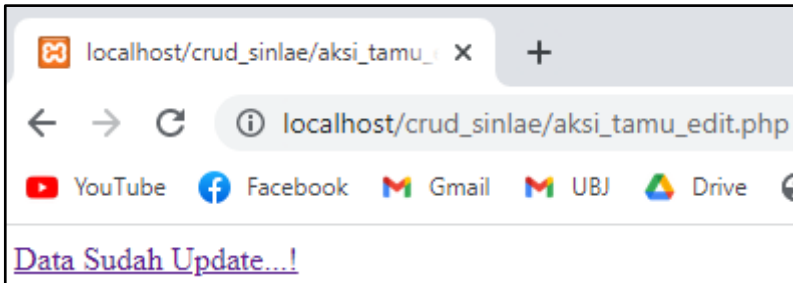
Buatlah file baru dengan nama aksi_tamu_edit.php yang berisi perintah untuk merubah data inputan dari form tamu_edit.php. Tuliskan kode sebagai berikut :

```
<?php
include "koneksi.php";
$id = $_POST['id'];
mysqli_query($koneksi, "UPDATE buku tamu SET
nama = '$_POST[nama]',
email = '$_POST[email]',
kota = '$_POST[kota]'
WHERE id = '$id' ");
echo "<a href='tamu.php'>Data Sudah Update...!</a>";
?>
```

Jalankan program tamu_edit melalui program induknya yaitu tamu.php, kemudian klik link 'Edit' pada data yang ingin dirubah. Berikut adalah tampilan program tamu_edit.php.

The screenshot shows a web browser window with the title 'localhost/crud_sinlae/tamu_edit.php'. The address bar contains 'localhost/crud_sinlae/tamu_edit.php?id=8'. Below the browser window, there is a form titled 'Edit Data Tamu'. The form contains three input fields: 'Nama Tamu' with the value 'Arshaka Sinlae', 'Alamat Email' with the value 'arshaka@gmail.com', and 'Kota' with the value 'Bekasi'. At the bottom of the form, there are two buttons: 'Update' and 'Reset'.

Gambar 3.59 Output tamu_edit.php

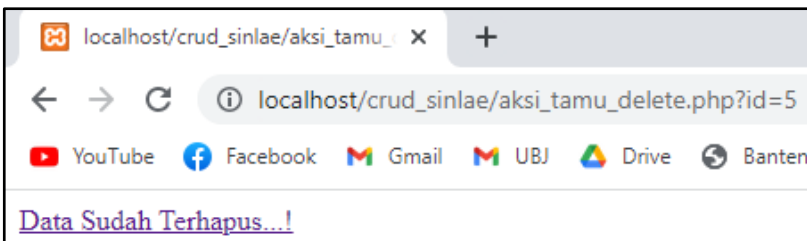


Gambar 3.60 Output aksi_tamu_edit.php

5. Membuat program Hapus Tamu

Buatlan file baru dengan aksi_tamu_delete.php Tuliskan kode sebagai berikut:

```
<?php
include "koneksi.php";
$id = $_GET['id'];
mysqli_query($koneksi, "DELETE FROM buktutamu WHERE id = $id");
echo "<a href='tamu.php'>Data Sudah Terhapus...!</a>";
?>
```



Gambar 3.61 Output aksi_tamu_hapus.php

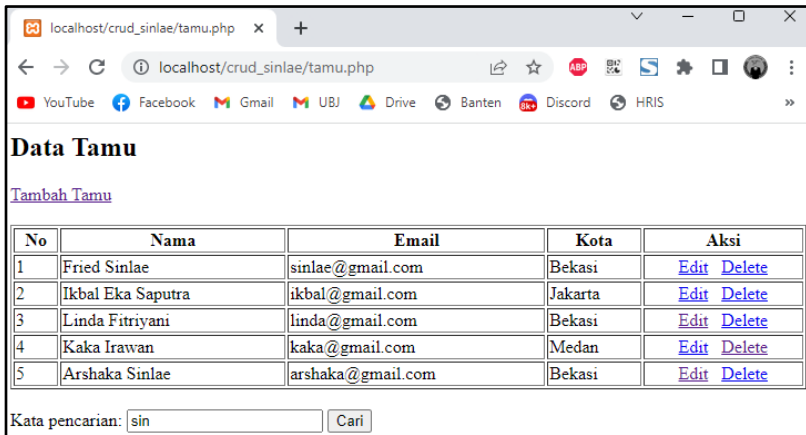
6. Membuat program Cari Tamu

Buatlan file baru dengan siswa_cari.php Tuliskan kode sebagai berikut :

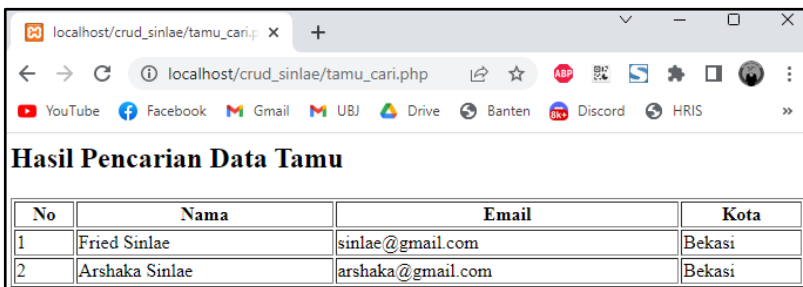
```
<?php
include "koneksi.php";
$keyword = $_POST['keyword'];
echo "
<h2>Hasil Pencarian Data Tamu</h2>
<table border='1' width='100%'>
<tr>
<th>No</th><th>Nama</th><th>Email</th><th>Kota</th>
</tr>";
$stampil = mysqli_query($koneksi, "SELECT * FROM bukutamu
where nama like '%$keyword%' OR kota like '%$keyword%'");
$no = 1;
while ($r = mysqli_fetch_array($stampil)) {
    echo "
<tr>
<td>$no</td>
<td>$r[nama]</td>
<td>$r[email] </td>
<td>$r[kota]</td>
</tr>";
    $no = $no + 1;
}
echo "</table>";
```

```
?>
```

Jalankan program dengan cara menulis kata yang akan dicari, misalnya pada contoh kita tulis 'sin'.



Gambar 3.62 Output tamu.php pencarian kata 'sin'



Gambar 3.63 Output tamu_cari.php hasil pencarian kata 'sin'

BAB 4

PENUTUP

Pemrograman web merupakan salah satu keterampilan yang sangat vital di era digital saat ini. Kemampuan ini tidak hanya sebatas pada penulisan kode semata, tetapi juga mencakup pemahaman menyeluruh terhadap berbagai bahasa pemrograman, struktur halaman web, pengelolaan data secara efisien, serta penerapan teknologi web secara fungsional dan terintegrasi. Dalam proses pengembangan web, terdapat sejumlah komponen penting yang harus dikuasai agar sebuah situs dapat berjalan optimal dan memenuhi kebutuhan pengguna.

HTML (*Hypertext Markup Language*), CSS (*Cascading Style Sheets*), dan JavaScript merupakan fondasi utama dalam membangun tampilan dan interaktivitas suatu situs web. HTML digunakan untuk membentuk kerangka dasar dari halaman web, seperti menentukan heading, paragraf, gambar, dan link. CSS bertugas mempercantik tampilan dengan memberikan sentuhan estetika pada elemen-elemen HTML, seperti warna, tata letak, dan jenis huruf. Sementara itu, JavaScript berfungsi menambahkan dinamika dan interaktivitas, seperti validasi formulir, animasi, dan manipulasi elemen secara real-time.

Di sisi lain, untuk menangani proses yang lebih kompleks seperti penyimpanan dan pengolahan data pengguna, dibutuhkan bahasa pemrograman sisi server seperti PHP. PHP berperan penting dalam membangun halaman web dinamis, di mana konten dapat berubah

sesuai dengan input atau kondisi tertentu. Dalam mendukung kerja PHP, MySQL digunakan sebagai sistem manajemen basis data yang menyimpan berbagai informasi penting seperti data login, konten artikel, atau data transaksi.

Lebih lanjut, kemampuan dalam menangani form input, melakukan validasi data, serta mengelola proses upload file menjadi bagian krusial dalam pengembangan web modern. Aspek-aspek ini tidak hanya penting untuk memastikan fungsionalitas web, tetapi juga menyangkut aspek keamanan, efisiensi, dan kenyamanan pengguna.

Dengan mengikuti alur pembelajaran yang sistematis serta melakukan praktik secara langsung, setiap pembelajar dapat mengembangkan keterampilan dalam membangun aplikasi web yang efisien, aman, menarik, dan mudah digunakan. Oleh karena itu, pemrograman web menjadi salah satu kompetensi strategis yang sangat relevan dalam berbagai bidang, mulai dari pendidikan, bisnis, pemerintahan, hingga industri kreatif. Dalam konteks yang lebih luas, pemrograman web juga berperan dalam merancang solusi digital yang adaptif, dapat diakses secara global, dan mampu menjawab tantangan kebutuhan informasi serta komunikasi di era serba cepat seperti sekarang ini.

DAFTAR PUSTAKA

- Anhar. (2010). *Panduan menguasai PHP dan MySQL secara otodidak*. Mediakita.
- Kuswayatno, L. (2007). *MAHIR DAN TERAMPIL BERKOMPUTER KELAS XI JILID 2*. Bandung: Grafindo Media Pratama.
- Rusli, A. S. (2019). *Pemrograman Website dengan PHP-MySQL untuk Pemula*. Takalar: Yayasan Ahmar Cendekia Indonesia.
- Saragih, R. R. (den 24 12 2018). *Pemrograman dan Bahasa Pemrograman*. Hämtat från https://www.researchgate.net/profile/Richy-Saragih/publication/329885312_PEMROGRAMAN_DAN_BAHASA_PEMROGRAMAN/links/5c209488458515a4c7f4b123/PEMROGRAMAN-DAN-BAHASA-PEMROGRAMAN.pdf den 06 11 2022
- Sinlae, F., & Yasir, M. (2024). Pembuatan Website Menggunakan CMS Wordpress di IKA Ubhara Jaya. *Jurnal Pengabdian kepada Masyarakat Nusantara (JPkMN)*, 6(1), 196-204.
- Sinlae, F., Bowono, P., Christian, H. R., & Elfan, A. (2024). Pelatihan Dalam Desain Website Menggunakan HTML dan CSS. *JIM : Jurnal Ilmu Multidisiplin*, 3(2), 173-183.
- Yusril Helmi Setyawan, D. A. (2020). *Membuat Sistem Informasi Gadai Online Menggunakan Codeigniter Serta Kelola Proses Pemberitahuannya*. Bandung: Kreatif Industri Nusantara.

Yusuf, D. (den 16 January 2022). *Buku Seri Pemrograman Web PHP Native dan MySQL Seri 1*. Hämtat från https://www.academia.edu/95217854/BUKU_SERI_PEMROGRAMAN_WEB_PHP_NATIVE_DAN_MYSQL_SERI_1 den 31 May 2023

BIODATA PENULIS



Fried Sinlae, S.T., M.Kom.

Dosen Program Studi Informatika
Fakultas Ilmu Komputer, Universitas Bhayangkara Jakarta
Raya

Penulis lahir di Jakarta pada 18 Maret 1993. Beliau adalah seorang akademisi dan praktisi di bidang Ilmu Komputer. Beliau menyelesaikan pendidikan Sarjana Teknik di bidang Teknik Informatika dan melanjutkan pendidikan Magister Ilmu Komputer. Saat ini, beliau merupakan Dosen Tetap di Fakultas Ilmu Komputer Universitas Bhayangkara Jakarta Raya dan juga aktif sebagai Senior Software Engineer di PT. Gerbang Cahaya Utama. Dengan pengalaman lebih dari 10 tahun sebagai Software Engineer, Fried Sinlae memiliki keahlian dalam pengembangan perangkat lunak, arsitektur sistem, dan teknologi cloud. Pengalaman kerjanya mencakup

pengembangan berbagai sistem dan aplikasi yang digunakan di sektor industri maupun akademis. Selain itu, beliau juga berkontribusi dalam penelitian yang berfokus pada inovasi teknologi di bidang perangkat lunak dan kecerdasan buatan. Sebagai pengajar, Fried Sinlae aktif dalam mengembangkan kurikulum yang berorientasi pada kebutuhan industri 4.0 dan sering membimbing mahasiswa dalam berbagai proyek pengembangan perangkat lunak. Keterlibatan beliau dalam dunia industri dan akademik memberikan kontribusi signifikan pada pengembangan SDM yang kompeten di bidang teknologi informasi.

BIODATA PENULIS



Muhammad Yasir S.Si., M.Kom

Dosen Program Studi Informatika
Fakultas Ilmu Komputer, Universitas Bhayangkara Jakarta
Raya

Penulis lahir di Jakarta pada 17 Desember 1990. Beliau adalah seorang dosen Informatika dengan fokus khusus pada bidang jaringan komputer dan keamanan informasi. Dengan latar belakang pendidikan S2 dan pengalaman selama 6,5 tahun sebagai NOC Monitoring di salah satu ISP terkemuka, yaitu Indonet, beliau memiliki pemahaman yang mendalam tentang infrastruktur jaringan dan aspek-aspek kritis dalam keamanan data.

Saat ini, beliau mengabdikan diri sebagai pengajar di Universitas Bhayangkara Jakarta Raya, di mana beliau tidak hanya mengajarkan konsep-konsep dasar tetapi juga membimbing mahasiswa dalam memahami penerapan praktis dari teori yang diajarkan dengan minat yang

mendalam pada teknologi informasi, khususnya dalam bidang network dan security, Penulis terus berusaha untuk memberikan kontribusi signifikan dalam dunia pendidikan melalui pengembangan materi ajar yang bermanfaat dan dapat diterapkan dalam dunia nyata.

Melalui e-book ini, beliau berharap dapat membagikan pengetahuannya kepada pembaca, baik sebagai bahan ajar maupun referensi praktis bagi para profesional dan mahasiswa yang tertarik dalam bidang jaringan komputer dan keamanan informasi.