



INSTITUT BISNIS  
& INFORMATIKA  
KESATUAN



# PEMROGRAMAN

---

# WEB

Septian Cahyadi  
Suci Sutjipto  
Isnain Mulia

Febri Damatraseta Fairuz  
Irvan Rizky Ariansyah  
Raiyana Jan Winata

PROGRAM STUDI TEKNOLOGI INFORMASI  
FAKULTAS PARIWISATA DAN INFORMATIKA  
INSTITUT BISNIS & INFORMATIKA KESATUAN  
Jl. Rangga Gading No.1, RT.02/RW09, Gudang,  
Kecamatan Bogor Tengah, Kota Bogor, Jawa Barat 16123

## **KATA PENGANTAR**

Puji syukur ke hadirat Tuhan Yang Maha Kuasa, yang telah memberikan rahmat-Nya sehingga Modul Praktikum Pemograman Web untuk mahasiswa/i Program Studi Teknologi Informasi Fakultas Informatika dan Pariwisata Institut Bisnis dan Informatika Kesatuan Bogor ini dapat diselesaikan dengan sebaik-baiknya.

Modul praktikum ini dibuat sebagai pedoman dalam melakukan kegiatan praktikum Pemograman Web yang merupakan kegiatan penunjang mata kuliah Pemograman Web pada Program Studi Teknologi Informasi Fakultas Informatika dan Pariwisata Institut Bisnis dan Informatika Kesatuan Bogor. Modul praktikum ini diharapkan dapat membantu mahasiswa/i dalam mempersiapkan dan melaksanakan praktikum dengan lebih baik, terarah, dan terencana. Pada setiap topik telah ditetapkan tujuan pelaksanaan praktikum dan semua kegiatan yang harus dilakukan oleh mahasiswa/i serta teori singkat untuk memperdalam pemahaman mahasiswa/i mengenai materi yang dibahas.

Dalam pembuatan Modul Praktikum Pemograman Web ini, penyusun menyadari masih jauh dari sempurna. Oleh karena itu penyusun mengharapkan kritik dan saran yang membangun guna penyempurnaan modul praktikum ini dimasa yang akan datang. Akhir kata, penyusun mengucapkan banyak terima kasih kepada semua pihak yang telah terlibat baik secara langsung maupun tidak langsung dalam pembuatan modul ini.

Bogor, 10 Februari 2021

Penyusun,

## **TATA TERTIB PRAKTIKUM**

1. Peserta praktikum hadir 10 menit sebelum praktikum dimulai.
2. Keterlambatan maksimal 15 menit (maksimal 3 kali pertemuan), dan setiap keterlambatan maka poin nilai dikurangi 5.
3. Wajib memakai pakaian rapi dan sepatu tertutup.
4. Membawa kartu praktikum.
5. Dilarang makan dan minum selama praktikum berlangsung.
6. Dilarang meninggalkan laboratorium selama praktikum berlangsung, kecuali atas izin asisten atau dosen yang bersangkutan.
7. Pengumpulan tugas dikolektifkan pada komti.
8. Jika mahasiswa mengumpulkan tugas yang sama persis dengan rekannya maka tugas dikembalikan dan harus membuat ulang.
9. Absensi tidak full 100% maka tidak diperkenankan mengikuti UAP.
10. Dilarang Memalsukan tanda tangan Asisten Praktikum. Jika melanggar, absen akan di anggap Nol (tidak pernah hadir).
11. Dilarang memindahkan dan mencabut peralatan lab (mouse, keyboard, dll) tanpa seizin Asisten Praktikum.
12. Jika terlambat atau tidak bisa hadir, silakan hubungi asisten praktikum sebelum praktikum dilaksanakan. Tidak ada keterangan dianggap tidak hadir.
13. Jika tidak dapat hadir pada hari tertentu, silakan ganti dengan hari lain di minggu tersebut tidak hadir. (WAJIB menghubungi asisten yang bersangkutan dan WAJIB mengisi dan membawa kertas bukti pindah jam yang sediakan).

## DAFTAR ISI

KATA PENGANTAR	i
TATA TERTIB PRAKTIKUM	ii
DAFTAR ISI	iii
MODUL 1	1
MENGENAL BAHASA PEMROGRAMAN HTML	1
<b>1.1. Tujuan Praktikum</b>	1
<b>1.2. Dasar Teori</b>	1
<i>1.2.1. Sekilas Mengenai HTML</i>	1
<i>1.2.2. Struktur HTML</i>	3
<i>1.2.3. HTML Documents</i>	4
<i>1.2.4. HTML Elements</i>	6
<b>1.3. Tugas dan Latihan</b>	16
MODUL 2	22
PENGENALAN CASCADING STYLE SHEETS (CSS)	22
<b>2.1. Tujuan Praktikum</b>	22
<b>2.2. Dasar Teori</b>	22
<i>2.2.1 Pengenalan CSS</i>	22
<i>2.2.2. Cara penggunaan CSS</i>	23
<i>2.2.3. Selector</i>	23
<i>2.2.4. CSS Inline</i>	27
<i>2.2.5. CSS Internal</i>	27
<i>2.2.6. CSS Eksternal</i>	28
<i>2.2.7. Component Properties</i>	29
<b>2.3. Tugas dan Latihan</b>	36
MODUL 3	39
PENGENALAN BAHASA PEMROGRAMAN JAVASCRIPT	39
<b>3.1. Tujuan Praktikum</b>	39
<b>3.2. Dasar Teori</b>	39
<i>3.2.1.Pengantar</i>	39
<i>3.2.2. Penulisan Javascript</i>	40
<i>3.2.3. Variables</i>	43
<i>3.2.4. Tipe Data</i>	44

3.2.5. <i>Tipe Numerik</i>	44
3.2.6 <i>Tipe String</i>	45
3.2.7 <i>Tipe Boolean</i>	45
3.2.8 <i>Tipe Null</i>	45
3.2.9. <i>Operator</i>	45
3.2.10 <i>Masukan Data</i>	49
3.2.11. <i>Object Javascript</i>	50
<b>3.3. Kontrol Program</b>	58
3.3.1. <i>Percabangan</i>	58
3.3.2. <i>Perulangan</i>	65
<b>3.4. Tugas dan Latihan</b>	70
MODUL 4	71
JAVASCRIPT 2	71
<b>4.1. Tujuan Praktikum</b>	71
<b>4.2. Dasar Teori</b>	71
4.2.1. <i>Object Array</i>	71
4.2.2. <i>Object Date</i>	72
4.2.3. <i>Object Math</i>	73
4.2.4. <i>Object String</i>	75
4.2.5. <i>Object Document</i>	77
4.2.5. <i>Object Window</i>	79
<b>4.3. Event Handler</b>	82
<b>4.4. Tugas dan Latihan</b>	95
DAFTAR PUSTAKA	96

# **MODUL 1**

## **MENGENAL BAHASA PEMROGRAMAN HTML**

### **1.1. Tujuan Praktikum**

1. Mahasiswa mengetahui sejarah Bahasa HTML
2. Instalasi HTML
3. Mahasiswa memahami struktur program Bahasa HTML
4. Setup dan instalasi git

### **1.2. Dasar Teori**

#### *1.2.1. Sekilas Mengenai HTML*

HTML merupakan standart umum yang digunakan untuk membangun sebuah halaman website. HTML merupakan kepanjangan dari Hyper Text Markup Language, merupakan:

- a. Menggambarkan struktur halaman website.
- b. Terdiri dari serangkaian element, dimana element HTML akan memberitahukan browser untuk menampilkan konten dari berbagai platform komputer manapun.
- c. Mengontrol tampilan dari halaman website dan kontennya. Seperti memberikan label potongan konten dengan menggunakan element HTML seperti “bagian judul”, “bagian paragraf”, “bagian tautan” dan lain sebagainya.

Pada tahun 1980 seorang ahli fisika, Tim Berners-Lee, dan juga seorang kontraktor di CERN (Organisasi Eropa untuk Riset Nuklir) mengusulkan dan menyusun ENQUIRE, sebuah sistem untuk ilmuwan CERN dalam membagi dokumen. Sembilan tahun kemudian, Berners-Lee mengusulkan adanya sistem markah berbasis internet. Berners-Lee menspesifikasikan HTML dan menulis jaringan beserta perangkat lunaknya di akhir 1990. Pada tahun yang sama, Berners-Lee dan Robert Cailliau, insinyur sistem data CERN berkolaborasi dalam sebuah permintaan untuk pendanaan, namun tidak diterima secara resmi oleh CERN. Di catatan pribadinya sejak 1990 dia mendaftar "beberapa dari banyak daerah yang menggunakan hypertext" dan pertama-tama menempatkan sebuah ensiklopedia.

Berikut ini adalah sekilas mengenai perjalanan versi dari HTML yang telah diakui sampai saat ini:

a. HTML 1.0

Penjelasan pertama kali HTML dipublikasikan sebagai bentuk bahasa pemograman untuk sistem markah berbasis internet ialah sebuah dokumen yang disebut "Tanda HTML", pertama kali disebutkan di Internet oleh Tim Berners-Lee pada akhir 1991. Tanda ini menggambarkan 18 elemen awal mula, versi sederhana dari HTML. Kecuali untuk tag hyperlink, yang sangat dipengaruhi oleh SGMLguid, in-house Standard Generalized Markup Language (SGML) berbasis format dokumen di CERN.

b. HTML 2.0

Dikembangkan oleh Kelompok Kerja HTML IETF, yang ditutup pada tahun 1996. Ini menetapkan standar untuk fitur inti HTML berdasarkan praktik saat ini pada tahun 1994.

c. HTML 3.0

Rekomendasi pertama W3C untuk HTML yang mewakili konsensus tentang fitur HTML untuk tahun 1996. HTML 3.2 menambahkan fitur yang digunakan secara luas seperti tabel, applet, aliran teks di sekitar gambar, skrip super, dan subskrip, sekaligus memberikan kompatibilitas mundur dengan standar HTML 2.0 yang ada.

d. HTML 4.01

Versi ini dikenalkan pada tahun 1999 dengan menambahkan dukungan untuk lebih banyak opsi multimedia, bahasa skrip, styles sheets, fasilitas pencetakan yang lebih baik, dan dokumen yang lebih mudah diakses oleh pengguna penyandang disabilitas.

e. XHTML 1.0

Pada tahun 2000 dengan banyak fitur baru, XHTML 1.0 merupakan formulasi ulang HTML 4.01 dalam XHTML, dan menggabungkan kekuatan HTML 4 dengan kekuatan XML.

f. HTML 5.0

dibuat untuk menggantikan HTML 3.2, HTML 4. dan XHTML 1.x, HTML 5 memiliki banyak fitur baru dibandingkan pendahulunya. ini termasuk dukungan penyimpanan media offline, elemen konten yang lebih spesifik (seperti footer, header, navigasi, dll), doctype inline yang lebih sederhana, dukungan penyematkan audio, dan video. Pembaharuan ini diperkenalkan pada tahun 2015.

g. HTML 5.1

Selang satu tahun dari peluncuran HTML 5.0, bahasa pemograman standart berbasis internet ini sudah memiliki pembaharuan seperti kemampuan yang lebih baik terkait pengalaman video, formulir web, aksesibilitas gambar, dan pemeriksaan ejaan dan tata bahasa.

h. HTML 5.2

Peningkatan kebijakan keamanan konten, API permintaan pembayaran untuk situs web eCommerce, aksesibilitas aplikasi rich internet untuk penyandang disabilitas, dan perubahan pada elemen <main> untuk mendukung desain responsif. Versi HTML 5.2 dikenalkan pada tahun 2017 dan masih dipergunakan hingga saat ini sebagai bahasa pemograman standart berbasis internet yang direkomendasikan oleh W3C.

### *1.2.2. Struktur HTML*

Sifat dari HTML ialah client script dimana dokumen tersebut dapat dibuka didalam komputer secara stand alone yang tidak membutuhkan sebuah server untuk dapat menampilkan di dalam browser. Dokument HTML biasanya berekstensi .html atau .htm dimana bahasa HTML tersebut tersusun atas tag atau syntax yang berformat <tagname>...</tagname>. Aturan dalam penulisan HTML sebagai berikut:

- a. Dalam penulisan tag HTML diapit dengan menggunakan dua kurung siku (angle bracket) “<” dan “>”
- b. Tag HTML selalu berpasangan, dimana tag awal dalam suatu syntax disebut dengan tag awal dan tag kedua sebagai tag terakhir. contoh: <p>...</p>
- c. Penulisan tag HTML tidak case sensitive dimana tag dengan huruf kecil sama dengan tag huruf besar, contoh: <P>...</P> sama dengan <p>...</p>Namun berdasarkan aturan pada W3C (HTML5.0 dan XHTML) penulisan tag HTML sebaiknya menggunakan huruf kecil.
- d. Jika memiliki tag nested atau terdapat suatu tag berisi tag lagi, maka penulisan tag akhir tidak boleh bersilang dan harus berurutan. Contoh: <p>Ini contoh <b>Huruf Tebal</b> dan <i>Huruf cetak miring</i> </p>

- e. Penulisan syntax dokumen HTML selalu diawali dengan tag <html> dan diakhiri dengan tag </html>

```
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

*Gambar 1.2.2 Stuktur HTML*

### *1.2.3. HTML Documents*

Semua dokumen HTML harus dimulai dengan deklarasi tipe dokumen: <!DOCTYPE html>. Dokumen HTML itu sendiri diawali dengan <html> dan diakhiri dengan </html>. Bagian yang terlihat dari dokumen HTML adalah antara <body> dan </body>.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

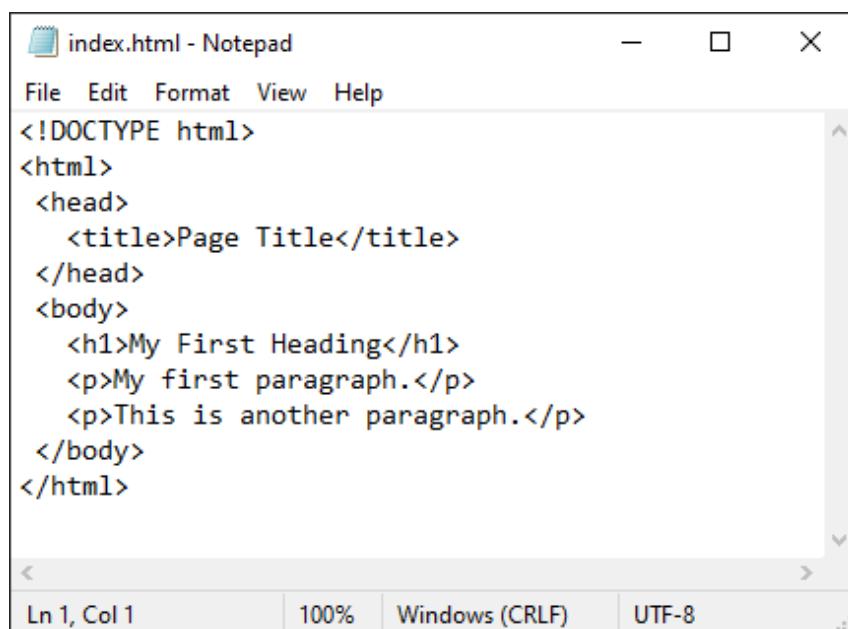
*Gambar 1.2.3.1 Simple Document HTML*

Dari contoh penulisan dokumen HTML diatas dapat diartikan sebagai berikut:

- Deklarasi <!DOCTYPE html> mendefinisikan bahwa dokumen ini adalah dokumen HTML5.
- Elemen <html> adalah elemen root dari halaman HTML.
- Elemen <head> berisi informasi meta tentang halaman HTML.

- d. Elemen <title> menentukan judul untuk halaman HTML (yang ditampilkan di bilah judul browser atau di tab halaman).
- e. Elemen <body> mendefinisikan tubuh dokumen, dan merupakan wadah untuk semua konten yang terlihat, seperti judul, paragraf, gambar, hyperlink, tabel, daftar, dll.
- f. Elemen <h1> mendefinisikan heading besar.
- g. Elemen <p> mendefinisikan paragraf.

Dalam membangun sebuah halaman website dengan menggunakan HTML, text editor yang dapat digunakan cukup sederhana dimana aplikasi text editor ini terdapat disemua operating sistem pada komputer. Jika menggunakan Windows OS dapat menggunakan aplikasi Notepad, sedangkan jika menggunakan UNIX OS dapat menggunakan aplikasiTextEdit.



```

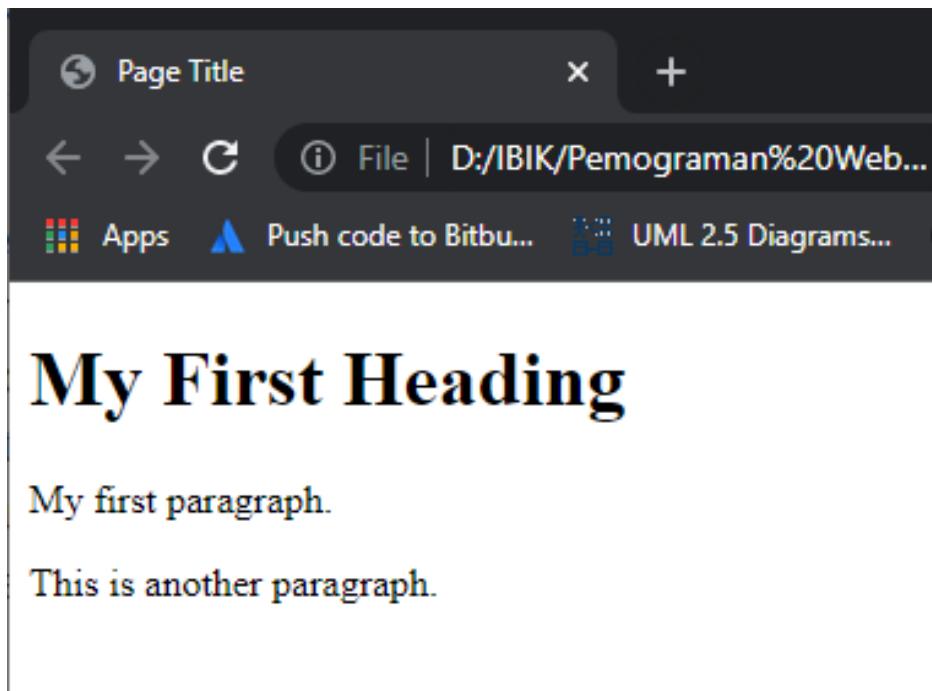
index.html - Notepad
File Edit Format View Help
<!DOCTYPE html>
<html>
<head>
    <title>Page Title</title>
</head>
<body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
    <p>This is another paragraph.</p>
</body>
</html>

```

Ln 1, Col 1      100%      Windows (CRLF)      UTF-8

*Gambar 1.2.3.2 HTML Documents dalam text editor*

Untuk membuat sebuah dokumen HTML tahap pertama ialah buka text editor yang dimiliki seperti Notepad dan masukan script dibawah ini, lalu simpan dengan nama *index.html* atau *index.htm*. Setalah berhasil tersimpan dengan format extension *.html* atau *.htm*, maka file tersebut akan membentuk sebuah file dengan format icon browser. Jika dibuka file tersebut maka akan membuka sebuah aplikasi browser dan menampilkan sebuah tampilan berdasarkan tag HTML.



Gambar 1.2.3.3 Hasil Preview dari HTML Documents

#### 1.2.4. HTML Elements

##### 1.2.4.1. HTML Element Head

Berikut adalah tag atau elemen yang dapat digunakan dibagian <head>...</head> pada sebuah dokumen HTML:

###### 1. Title

Elemen <title>, menentukan judul untuk halaman HTML (yang ditampilkan di bilah judul browser atau di tab halaman)

```
<title>Page Title</title>
```

###### 2. Favicon

Elemen ini diperuntukan untuk menentukan icon pada halaman HTML (yang ditampilkan di bilah judul browser atau di tab halaman). Untuk menggunakan favicon, dikhkususkan memiliki file gambar dengan extension .ico, .svg atau .png dan size yang digunakan ialah 16x16 pixel atau 32x32 pixel.

```
<link rel="icon" type="image/x-icon" href="/path/to/icons/favicon.ico">
<link rel="icon" type="image/png" href="/path/to/icons/favicon.png">
<link rel="icon" type="image/svg" href="/path/to/icons/favicon.svg">
```

### 3. Meta

Elemen <meta> biasanya digunakan untuk menentukan set karakter, deskripsi halaman, kata kunci, penulis dokumen, dan pengaturan *viewport*. Metadata tidak akan ditampilkan di halaman, tetapi digunakan oleh browser (cara menampilkan konten atau memuat ulang halaman), oleh mesin pencari (kata kunci), dan layanan web lainnya.

- Menentukan set karakter

```
<meta charset="UTF-8">
```

- Menentukan kata kunci untuk search engine

```
<meta name="keywords" content="HTML, CSS, JavaScript">
```

- Menentukan deskripsi halaman web

```
<meta name="description" content="Website IBIK – Pemograman Web">
```

- Menentukan penulis halaman

```
<meta name="author" content="Febry D Fairuz">
```

- Merefresh halaman setiap 30 detik

```
<meta http-equiv="refresh" content="30">
```

- Mengatur *viewport* agar situs website terlihat bagus di semua perangkat

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

### 4. Javascript

*JavaScript* membuat halaman HTML lebih dinamis dan interaktif. Tag HTML <script> digunakan untuk mendefinisikan skrip disisi klien (*JavaScript*). Elemen <script> berisi pernyataan skrip, atau menunjuk ke file skrip eksternal melalui atribut src. Penggunaan umum untuk *JavaScript* adalah manipulasi gambar, validasi formulir, dan perubahan konten yang dinamis. Untuk memilih elemen HTML, *JavaScript* paling sering menggunakan metode document.getElementById(). Contoh *JavaScript* ini menulis "Halo *JavaScript*!" menjadi elemen HTML dengan id="demo":

```
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML = "Hello"
```

```
JavaScript!";  
</script>
```

## 5. Styles CSS

*Cascading Style Sheets* (CSS) digunakan untuk memformat tata letak halaman web. Dengan CSS, dapat mengontrol warna, font, ukuran teks, jarak antar elemen, bagaimana elemen diposisikan dan ditata, gambar latar belakang atau warna latar apa yang akan digunakan, tampilan berbeda untuk perangkat dan ukuran layar yang berbeda, dan lebih banyak!

```
<style>  
    body {background-color: powderblue;}  
    h1   {color: blue;}  
    p    {color: red;}  
</style>
```

### 1.2.4.2. *HTML Element Body*

Sedangkan berikut ini adalah tag atau elemen yang bisa digunakan dibagian <body>...</body> pada sebuah dokumen HTML:

- **Elements Contents HTML**

1. Headings

*Headings* adalah judul atau subjudul yang ingin ditampilkan di halaman web. *Headings* didefinisikan dengan tag <h1> hingga <h6>. <h1> mendefinisikan heading yang paling penting. <h6> mendefinisikan heading yang paling tidak penting.

```
<h1>Heading 1</h1>  
<h2>Heading 2</h2>  
<h3>Heading 3</h3>  
<h4>Heading 4</h4>  
<h5>Heading 5</h5>  
<h6>Heading 6</h6>
```

2. Paragraphs

Sebuah paragraf selalu dimulai pada baris baru, dan biasanya berupa blok teks. Elemen HTML <p> mendefinisikan sebuah paragraf. Sebuah paragraf selalu

dimulai pada baris baru, dan browser secara otomatis menambahkan beberapa spasi (margin) sebelum dan sesudah paragraf.

```
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
```

### 3. Formatting

HTML berisi beberapa elemen untuk mendefinisikan teks dengan arti khusus. Elemen pemformatan dirancang untuk menampilkan jenis teks khusus:

Element	Description
<b>	Bold
<strong>	Important
<i>	Italic
<em>	Emphasized
<mark>	Marked
<small>	Smaller
<del>	Deleted text
<sub>	Subscript text
<sup>	Superscript text

### 4. Quotations

Bagian ini kita akan membahas elemen HTML <blockquote>, <q>, <abbr>, <address>, <cite>, dan <bdo>.

Element	Description
<abbr>	singkatan atau akronim
<address>	informasi kontak untuk penulis/pemilik dokumen
<bdo>	arah teks
<blockquote>	bagian yang dikutip dari sumber lain
<cite>	judul sebuah karya
<q>	kutipan inline pendek

## 5. Comments

Untuk menambahkan komentar ke sumber HTML Anda dengan menggunakan sintaks berikut:

```
<!-- Write your comments here -->
```

## 6. List

List HTML memungkinkan programmer untuk mengelompokkan satu set item terkait dalam bentuk list. Daftar tidak berurutan dimulai dengan tag `<ul>` atau `<ol>`. Setiap item list dimulai dengan tag `<li>`.

Syntax	Display
<pre>&lt;ul&gt;   &lt;li&gt;Coffee&lt;/li&gt;   &lt;li&gt;Tea&lt;/li&gt;   &lt;li&gt;Milk&lt;/li&gt; &lt;/ul&gt;</pre>	<ul style="list-style-type: none"><li>Coffee</li><li>Tea</li><li>Milk</li></ul>
<pre>&lt;ol&gt;   &lt;li&gt;Coffee&lt;/li&gt;   &lt;li&gt;Tea&lt;/li&gt;   &lt;li&gt;Milk&lt;/li&gt; &lt;/ol&gt;</pre>	<ol style="list-style-type: none"><li>1. Coffee</li><li>2. Tea</li><li>3. Milk</li></ol>

## 7. Block and Inline

Setiap elemen HTML memiliki nilai tampilan default, tergantung pada jenis elemennya. Ada dua nilai tampilan: block dan inline.

### - Block Element

Elemen level blok selalu dimulai pada baris baru. Elemen level blok selalu menggunakan lebar penuh yang tersedia (membentang ke kiri dan kanan sejauh mungkin). Elemen level blok memiliki margin atas dan bawah, sedangkan elemen inline tidak.

```
<div>Hello World</div>
```

### - Inline Element

Elemen Inline tidak dimulai pada baris baru. Elemen Inline hanya membutuhkan lebar sebanyak yang diperlukan. Ini adalah elemen `<span>` di dalam paragraf.

```
<span>Hello World</span>
```

## 8. Links

Links ditemukan di hampir semua halaman web. Links memungkinkan pengguna mengeklik jalan mereka dari halaman ke halaman lain. Links HTML disebut dengan *hyperlink*. Untuk dapat mengklik tautan dan melompat ke halaman lain. Saat Anda menggerakkan mouse di atas tautan, panah mouse akan berubah menjadi tangan kecil.

Atribut terpenting dari elemen `<a>` adalah atribut `href`, yang menunjukkan tujuan tautan. Teks tautan adalah bagian yang akan terlihat oleh pembaca. Mengklik teks tautan, akan mengirim pembaca ke alamat URL yang ditentukan.

```
<a href="https://www.ibik.ac.id/> Visit IBIK</a>
```

### HTML Links - The target Attribute

Secara default, halaman tertaut akan ditampilkan di jendela browser saat ini. Untuk mengubahnya, Anda harus menentukan target lain untuk tautan tersebut.

Atribut target dapat memiliki salah satu dari nilai berikut:

Target	Description
<code>&lt;_self&gt;</code>	Membuka windows atau tab dalam satu halaman yang sama
<code>&lt;_blank&gt;</code>	Membuka windows atau tab baru
<code>&lt;_parent&gt;</code>	Membuka windows atau tab didalam elemen induk
<code>&lt;_top&gt;</code>	Membuka windows atau tab secara keseluruhan pada body

### Create Bookmarks

Link HTML dapat digunakan untuk membuat bookmark, sehingga pembaca dapat melompat ke bagian tertentu dari halaman web.

```
<h2 id="C4">Chapter 4</h2>
<a href="#C4">Jump to Chapter 4</a>
```

## 9. Images

Gambar dapat meningkatkan desain dan tampilan halaman web. Tag HTML <img> digunakan untuk menyematkan gambar di halaman web. Gambar secara teknis tidak dimasukkan ke dalam halaman web; gambar ditautkan ke halaman web. Tag <img> membuat ruang penyimpanan untuk gambar yang direferensikan. Tag <img> kosong, hanya berisi atribut, dan tidak memiliki tag penutup. Tag <img> memiliki dua atribut yang diperlukan:

- src = Menentukan jalur ke gambar
- alt = Menentukan teks alternatif untuk gambar.
- width = Menentukan panjang
- height = Menentukan tinggi

```

```

## 10. Tables

Tabel HTML memungkinkan pengembang web untuk mengatur data ke dalam baris dan kolom.

### Table Kolom

```
<table>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
</table>
```

Each table cell is defined by a <td> and a </td> tag. Segala sesuatu di antara <td> dan </td> adalah konten sel tabel.

### Table Baris

```
<table>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
  <tr>
```

```

<td>16</td>
<td>14</td>
<td>10</td>
</tr>
</table>

```

Setiap baris tabel dimulai dengan `<tr>` dan diakhiri dengan tag `</tr>`. Berikut ini adalah tag elemen table yang dapat dipergunakan:

Element	Description
<code>&lt;table&gt;</code>	Mendefinisikan tabel
<code>&lt;th&gt;</code>	Mendefinisikan sel header dalam tabel
<code>&lt;tr&gt;</code>	Mendefinisikan baris dalam tabel
<code>&lt;td&gt;</code>	Mendefinisikan sel dalam tabel
<code>&lt;caption&gt;</code>	Mendefinisikan judul tabel
<code>&lt;colgroup&gt;</code>	Menentukan grup dari satu atau lebih kolom dalam tabel untuk pemformatan
<code>&lt;col&gt;</code>	Menentukan properti kolom untuk setiap kolom dalam elemen <code>&lt;colgroup&gt;</code>
<code>&lt;thead&gt;</code>	Mengelompokkan konten header dalam sebuah tabel
<code>&lt;tbody&gt;</code>	Mengelompokkan konten tubuh dalam sebuah tabel
<code>&lt;tfoot&gt;</code>	Mengelompokkan konten footer dalam tabel

## 11. Iframe

Tag HTML `<iframe>` menentukan bingkai sebaris. Bingkai sebaris digunakan untuk menyematkan dokumen lain dalam dokumen HTML saat ini.

```

<iframe src="url" title="description" width="100px"
        height="50px"></iframe>

```

## 12. Audio/Vidio

HTML5 memiliki metode, properti, dan peristiwa untuk elemen `<audio>` dan `<video>`. Atribut kontrol menambahkan kontrol audio, seperti putar, jeda, dan volume. Elemen `<source>` memungkinkan Anda menentukan file audio alternatif yang dapat dipilih oleh browser. Browser akan menggunakan format pertama yang dikenali. Teks di antara tag `<audio>` dan `</audio>` hanya akan ditampilkan di browser yang tidak mendukung elemen `<audio>`.

```

<audio controls>
  <source src="lagu-1.ogg" type="audio/ogg">
  <source src="lagu-2.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>

```

Elemen HTML <video> digunakan untuk menampilkan video di halaman web.

```
<video width="320" height="240" controls>
  <source src="vidio-1.mp4" type="video/mp4">
  <source src="vidio-2.ogg" type="video/ogg">
  Your browser does not support the video tag.
</video>
```

Atribut kontrol menambahkan kontrol video, seperti putar, jeda, dan volume. Sebaiknya selalu sertakan atribut lebar dan tinggi. Jika tinggi dan lebar tidak disetel, halaman mungkin berkedip saat video dimuat. Elemen <source> memungkinkan Anda menentukan file video alternatif yang dapat dipilih oleh browser. Browser akan menggunakan format pertama yang dikenali. Teks antara tag <video> dan </video> hanya akan ditampilkan di browser yang tidak mendukung elemen <video>.

#### 1.2.4.3. HTML Element Content Form

Form HTML digunakan untuk mengumpulkan input data. Input data paling sering dikirim ke server untuk diproses.

Elemen HTML <form> digunakan untuk membuat formulir HTML untuk input pengguna:

First name: <input type="text" value="Febry"/>	<pre>&lt;form&gt;   &lt;label for="fname"&gt;First name:&lt;/label&gt;&lt;br&gt;   &lt;input type="text" id="fname" name="fname"&gt;&lt;br&gt;   &lt;label for="lname"&gt;Last name:&lt;/label&gt;&lt;br&gt;   &lt;input type="text" id="lname" name="lname"&gt; &lt;/form&gt;</pre>
Last name: <input type="text" value="Fairuz"/>	
<input type="button" value="Submit"/>	

##### 1. Form Atributes

List atribut <form> yang dapat digunakan:

Attribute	Description
accept-charset	Menentukan pengkodean karakter yang digunakan untuk pengiriman formulir
action	Menentukan ke mana harus mengirim formulir-data saat formulir dikirimkan

<b>autocomplete</b>	Menentukan apakah formulir harus memiliki pelengkapan otomatis aktif atau nonaktif
<b>enctype</b>	Menentukan bagaimana formulir-data harus dikodekan saat mengirimkannya ke server (hanya untuk metode = "posting")
<b>method</b>	Menentukan metode HTTP yang akan digunakan saat mengirim data formulir
<b>name</b>	Menentukan nama formulir
<b>novalidate</b>	Menentukan bahwa formulir tidak boleh divalidasi saat dikirimkan
<b>rel</b>	Menentukan hubungan antara sumber daya yang ditautkan dan dokumen saat ini
<b>target</b>	Menentukan tempat untuk menampilkan respons yang diterima setelah mengirimkan formulir

## 2. Form Elements

Elemen <form> HTML dapat berisi satu atau lebih elemen form sebagai berikut:

- Elemen <input>

Salah satu elemen form yang paling sering digunakan adalah elemen <input>.

Elemen <input> dapat ditampilkan dalam beberapa cara, tergantung pada atribut type.

```
<label for="fname">First name:</label>
<input type="text" id="fname" name="fname">
```

- Elemen <label>

Elemen <label> mendefinisikan label untuk beberapa elemen form. Elemen <label> berguna untuk pengguna pembaca layar, karena pembaca layar akan membacakan label dengan keras saat pengguna fokus pada elemen input.

Elemen <label> juga membantu pengguna yang mengalami kesulitan mengklik wilayah yang sangat kecil (seperti tombol radio atau kotak centang) - karena ketika pengguna mengklik teks di dalam elemen <label>, itu akan mengaktifkan tombol radio/kotak centang. Atribut for dari tag <label> harus sama dengan atribut id dari elemen <input> untuk mengikatnya bersama.

- Elemen <textarea>

Elemen <textarea> mendefinisikan bidang input multi-baris (area teks):

```
<textarea name="message" rows="10" cols="30">
Type something in here
</textarea>
```

Atribut *rows* menentukan jumlah baris yang terlihat di area teks. Atribut *cols* menentukan lebar yang terlihat dari area teks.

- Elemen <button>

Elemen <button> mendefinisikan tombol yang dapat diklik:

```
<button type="button">Click Me! Butoon</button>
<button type="submit">Click Me! Submit</button>
<button type="reset">Click Me! Reset</button>
```

- Elemen <select>

Elemen <select> mendefinisikan daftar drop-down:

```
<label for="cars">Choose a car:</label>
<select id="cars" name="cars">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="fiat">Fiat</option>
  <option value="audi">Audi</option>
</select>
```

The <option> elements defines an option that can be selected. By default, the first item in the drop-down list is selected. To define a pre-selected option, add the *selected* attribute to the option:

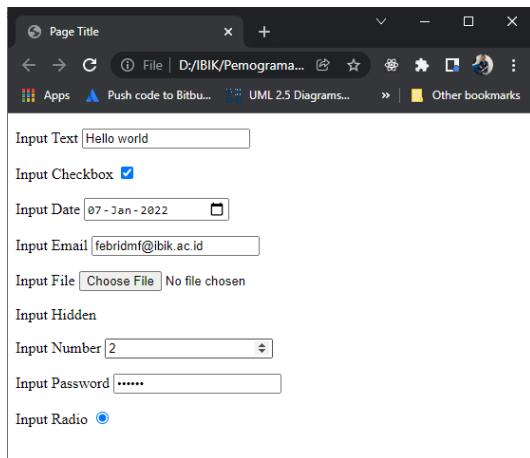
```
<option value="fiat" selected>Fiat</option>
```

### 3. Input Types

Berikut adalah berbagai jenis input yang dapat digunakan dalam HTML:

<pre>&lt;input type="text"&gt; &lt;input type="checkbox"&gt; &lt;input type="date"&gt; &lt;input type="email"&gt; &lt;input type="file"&gt;</pre>	<pre>&lt;input type="hidden"&gt; &lt;input type="number"&gt; &lt;input type="password"&gt; &lt;input type="radio"&gt; &lt;input type="range"&gt;</pre>
---	--

Display:



#### 4. Input Atributes

Bagian ini menjelaskan attribut yang berbeda untuk elemen <input> HTML.

Attribute	Description
value	Menentukan nilai awal untuk bidang input
readonly	Menentukan bahwa bidang input hanya-baca
disabled	Menentukan bahwa bidang input harus dinonaktifkan
size	Menentukan lebar yang terlihat, dalam karakter, dari bidang input
maxlength	Menentukan jumlah karakter maksimum yang diizinkan dalam bidang input
min & max	Menentukan nilai minimum dan maksimum untuk bidang input
pattern	Menentukan ekspresi reguler yang memeriksa nilai bidang input, saat formulir dikirimkan.
placeholder	Menentukan petunjuk singkat yang menjelaskan nilai yang diharapkan dari bidang input (nilai sampel atau deskripsi singkat tentang format yang diharapkan)
required	Menentukan bahwa bidang input harus diisi sebelum mengirimkan formulir

### 1.3. Tugas dan Latihan

Kerjakan dan unggahlah jawaban ke repository GitHub Anda!

#### 1. Latihan *Typography*

Buatlah dokumen HTML bernama *Latihan-1*.

Buatlah sebuah artikel yang berisi *Heading*, *Paragraph* dan *Formatting*. Minimum penggunaan paragraf sebanyak 4 buah.

#### 2. Latihan *Image*

Buatlah dokumen HTML bernama *Latihan-2*.

Dari dokumen HTML Latihan-1, tambahkanlah elemen Image. Penggunaan image minimum 2 buah. Dan tambahkan atribut gambar untuk mengatur panjang dan lebar dari gambar tersebut.

### 3. Latihan *Table*

Buatlah dokumen HTML bernama Latihan-3.

Silakan implementasikan bentuk dibawah ini kedalam scripting HTML:

Oksigen mempunyai beberapa SIFAT FISIKA, diantaranya yang terdapat dalam table berikut:

Sifat Fisika	Oksigen
Massa atom <i>relative</i>	15,994
Nomor Atom	8
Konfigurasi <i>electron</i>	$2s^22p^4$
Jari-jari atom (nm)	0,074
Jari-jari $X^{2-}$ (nm)	0.140
Keelectronegatifan	3,5
Energy ionisasi I (kJ/mol)	1316
Energy ionisasi II (kJ/mol)	3396
Kerapatan ( $g/cm^3$ )	1.27 (padatan)
Titik leleh ( $^{\circ}C$ )	+183
Titik beku ( $^{\circ}C$ )	-219
Potensial elektroda (V)	+0.401
$X_{(g)} + 2e^{+}_{(aq)} \rightarrow 2X^{-}_{(aq)}$	-

### 4. Latihan List

Buatlah dokumen bernama Latihan-4.

Ubahlah bentuk dibawah ini kedalam scripting HTML:

# The 10 Most Popular Programming Languages to Learn in 2022

There's no question that software programming is a hot career right now. The [U.S. Bureau of Labor Statistics](#) projects 21 percent growth for programming jobs from 2018 to 2028, which is more than four times the average for all occupations. What's more, the median annual pay for a software programmer is about \$106,000, which nearly three times the median pay for all U.S. workers.

Not all programming jobs are the same, however. Different roles, companies, and types of software require knowing and understanding different programming languages—and it's often beneficial to know multiple languages. Trying to break into the field of software programming can be a daunting experience, especially for professionals with no prior programming experience.

The [Master of Science in Computer Science Align program](#) at Northeastern University is specifically designed for students who want to transition into computer science from another field of study.

*“Our aim is to transport students from a variety of different backgrounds and have them come out as software engineers,”* says Ian Gorton, PhD and director of the graduate computer science programs at Northeastern University—Seattle. *“We focus on math, programming, and a variety of computer science and engineering concepts.”*

Whether you're new to programming or looking to brush up on your skills, it helps to know which languages are in high demand. Here are 10 of the most popular programming languages of 2020 based on the number of job postings listed on job search site Indeed, the average annual salary for those jobs, and factors such as ease of use and potential for growth.

## Top 10 Most Popular Programming Languages

1. Python  
**Average annual salary: \$120,000**
2. JavaScript  
**Average annual salary: \$118,000**
3. Java  
**Average annual salary: \$104,000**
4. C#  
**Average annual salary: \$97,000**
5. C  
**Average annual salary: \$97,000**
6. C++  
**Average annual salary: \$97,000**
7. Go  
**Average annual salary: \$93,000**
8. R  
**Average annual salary: \$93,000**
9. Swift  
**Average annual salary: \$93,000**
10. PHP  
**Average annual salary: \$81,000**

## 7 Other Programming Languages to Consider

- Web-based startups are more likely to be programming in **Python** and **JavaScript**.
- Larger companies tend to develop their internal software applications using **C#** or **Java** and their **Web applications using PHP**.
- Programs for data *analytics* typically use the **R** and **MATLAB** programming languages.
- Embedded devices, such as those in the automotive and healthcare industries, run software written in **C**, **C++**, or **Rust**.
- Applications that run on the cloud are increasingly written in **Go** or **Scala**.
- **Mobile applications** are increasingly written in Swift or Kotlin.

### 5. Latihan Iframe, *Block* dan *Inline*

Buatlah dokumen HTML bernama *Latihan-5*.

- a. Ubahlah bentuk dibawah ini menjadi script HTML:

#### Programming Language Base on Career



#### Top 10 Programming Language in 2022

...Iframe to *Latihan-4*...

#### List of programming languages from Wikipedia

The page displays a list of programming languages, each with its name and a small icon. The categories listed in the sidebar are:

- Programming language lists
  - Alphabetical
  - Categorical
  - Chronological
  - Generational

- b. Buatlah 1 buah iframe yang memanggil file dokumen Latihan-4
  - c. Buatlah 1 buah iframe yang memanggil sebuah url
6. Latihan Link
- Buatlah dokumen HTML bernama Latihan-6.*
- a. Implementasikan bentuk dibawah ini kedalam script HTML:

## Top 10 Daftar Bahasa Pemograman

### Daftar isi:

- [Python](#)
- [JavaScript](#)
- [Java](#)
- [C#](#)
- [C](#)
- [C++](#)
- [Go](#)
- [R](#)
- [Swift](#)
- [PHP](#)

### Python

Python adalah ...

### JavaScript

JavaScript adalah ...

*...dan seterusnya...*

Buatlah *hyperlink* pada elemen **Daftar Isi**, dimana link tersebut merujuk ke *diverent area*, ketika mengklik salah satu *link* tersebut maka akan merujuk ke area yang telah dituju. Contoh: jika mengklik *link Java*, maka akan dirujuk ke area *Java*.

- b. Buatlah tiga buah link baru dimana:
  - 1) Link “*Go to Latihan-4*” menuju ke dokumen HTML Latihan-4 dengan menggunakan target [self](#)
  - 2) Link “*Go to Latihan-5*” menuju ke dokumen HTML Latihan-5 dengan menggunakan target [blank](#)

3) Link “*Go to URL*” menuju URL dengan menggunakan target *\_parent*

## 7. Latihan Form

Buatlah dokumen HTML bernama *Latihan-7*.

Buatlah form yang berisi isian field sebagai berikut:

- NPM (input text)
- Fullname (input text)
- Email (input email)
- Password (input password)
- Place Birth (input text)
- Birthdate (input date)
- Gender (radio button, *Male and Female*)
- Address (textarea)

Dari form diatas memiliki dua buah button. Button submit dengan nama Save dan button reset dengan nama Clear.

## 8. Latihan Layout

Buatlah folder project dengan nama *my-portfolio*. Didalam folder tersebut terdapat dua buah folder yaitu folder *css* dan *images*. Dan terdapat satu buah file bernama *index.html*. Dimana file tersebut berisi script HTML yang akan menampilkan bentuk portfolio atau curiculum vitae pada *browser*. Silakan membuat bentuk portfolio dengan bentuk bebas namun mengimplementasikan bentuk elemen HTML yang telah diberikan minimal 5 buah elemen.

## MODUL 2

### PENGENALAN CASCADING STYLE SHEETS (CSS)

#### 2.1. Tujuan Praktikum

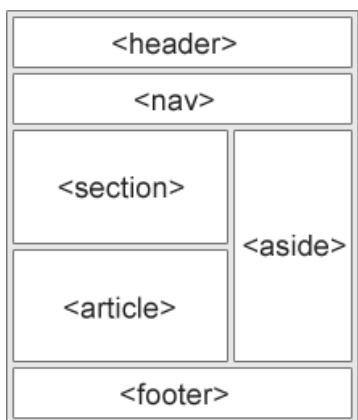
Tujuan dari praktikum pada chapter ini ialah memperkenalkan template yang berupa style untuk mempercantik tampilan website.

#### 2.2. Dasar Teori

##### 2.2.1 Pengenalan CSS

CSS atau Cascading Style Sheets adalah sebuah dokumen yang berisi aturan yang digunakan untuk memisahkan isi dengan layout dalam halaman-halaman web yang dibuat. CSS memperkenalkan “template” yang berupa style untuk dibuat dan mengijinkan penulisan kode yang lebih mudah dari halaman-halaman web yang dirancang. CSS mampu menciptakan halaman yang tampak sama pada resolusi layar yang berbeda-beda tanpa memerlukan penggunaan tabel seperti pada html klasik.

Umumnya HTML memiliki beberapa elemen semantik yang mendefinisikan berbagai bagian halaman web seperti gambar dibawah ini:



- <header> □ Mendefinisikan header untuk dokumen atau bagian
- <nav> □ Mendefinisikan sekumpulan link navigasi
- <section> □ Mendefinisikan bagian dalam dokumen
- <article> □ Mendefinisikan konten mandiri yang mandiri
- <aside> □ Mendefinisikan konten selain dari konten (seperti sidebar)
- <footer> □ Mendefinisikan footer untuk dokumen atau bagian

Gambar diatas menggambarkan sebuah Layouting dari sebuah template website. Untuk mempercantik tampilan dari layout tersebut dapat menggunakan CSS sebagai template.

CSS dapat ditambahkan ke dokumen HTML dengan 3 cara:

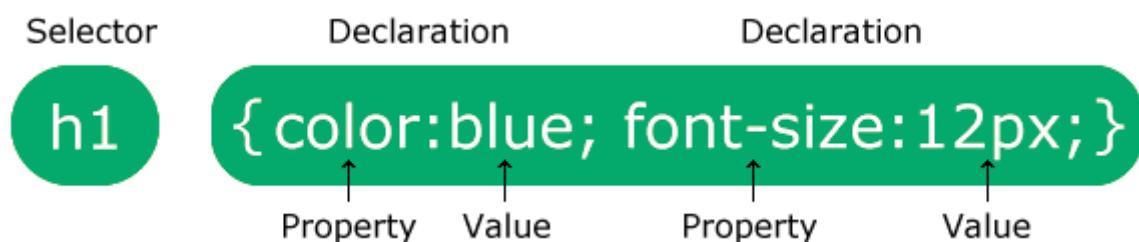
- Inline - dengan menggunakan atribut style di dalam elemen HTML
- Internal - dengan menggunakan elemen <style> di bagian <head>
- Eksternal - dengan menggunakan elemen <link> untuk menautkan ke file CSS eksternal

Cara paling umum untuk menambahkan CSS, adalah dengan menyimpan style di file CSS eksternal.

### 2.2.2. Cara penggunaan CSS

Aturan CSS terdiri dari selector dan blok deklarasi:

- Selector menunjuk ke elemen HTML yang ingin diberikan style. Blok deklarasi berisi satu atau lebih deklarasi yang dipisahkan oleh titik koma (;).
- Setiap deklarasi menyertakan nama properti CSS dan nilai, dipisahkan oleh titik dua(:).
- Beberapa deklarasi CSS dipisahkan dengan titik koma, dan blok deklarasi dikelilingi oleh kurung kurawal.



Contoh pemberian style pada element HTML paragraph <p> :

```
p {  
    color: red;  
    text-align: center;  
}
```

Penjelasan:

- `p` adalah pemilih di CSS (ini menunjuk ke elemen HTML yang ingin diberikan style: <p>).
- `color` adalah *properti*, dan `red` adalah *nilai properti*
- `text-align` adalah *properti*, dan `center` adalah *nilai properti*

### 2.2.3. Selector

Selektor CSS digunakan untuk "menemukan" (atau memilih) elemen HTML yang ingin diberikan style. Pemilih selector CSS dibagi menjadi lima kategori, yaitu:

- a. Selektor sederhana (pilih elemen berdasarkan nama, id, kelas)

- b. Selektor kombinator (memilih elemen berdasarkan hubungan spesifik di antara mereka)
- c. Selektor Pseudo-class (memilih elemen berdasarkan status tertentu)
- d. Selektor Pseudo-element (memilih dan menata bagian elemen)
- e. Selektor atribut (memilih elemen berdasarkan atribut atau nilai atribut)

Contoh syntax selector:

- Selektor Sederhana

<pre>p {     text-align: right;     color: red; }</pre>	<pre>#para1 {     text-align: center;     color: blue; }</pre>	<pre>p.description {     text-align: left;     color: green; }</pre>
<pre>&lt;p&gt;Every paragraph will be affected by the style.&lt;/p&gt; &lt;p id="para1"&gt;Me too!&lt;/p&gt; &lt;p class="description"&gt;And me!&lt;/p&gt;</pre>		

- Selektor kombinator

<pre>div p {     background-color: yellow; }</pre>
<pre>&lt;h1&gt;Descendant Selector&lt;/h1&gt; &lt;div&gt;     &lt;p&gt;Every paragraph will be affected by the style.&lt;/p&gt; &lt;/div&gt;</pre>

- Selektor *Pseudo-class*

*Pseudo-class* digunakan untuk mendefinisikan keadaan khusus suatu elemen.

Misalnya, dapat digunakan untuk:

- style elemen saat pengguna mengarahkan mouse ke atasnya
- style tautan yang dikunjungi dan yang belum dikunjungi secara berbeda
- style elemen saat mendapat fokus

### Syntax *pseudo-class*

```
selector:pseudo-class {
    property: value;
}
```

Contoh:

```
.para1:hover {  
    background-color: tomato;  
    font-weight: bold;  
}  
  
<h1>Descendant Selector</h1>  
<div id="content">  
    <p class="para1">Every paragraph will be affected by the style.</p>  
    <p>Every paragraph will be affected by the style.</p>  
</div>
```

Berikut ini adalah beberapa pseudo-class yang dapat digunakan:

Selector	Example	Example description
<a href="#">:active</a>	a:active	Selects the active link
<a href="#">:checked</a>	input:checked	Selects every checked <input> element
<a href="#">:disabled</a>	input:disabled	Selects every disabled <input> element
<a href="#">:empty</a>	p:empty	Selects every <p> element that has no children
<a href="#">:enabled</a>	input:enabled	Selects every enabled <input> element
<a href="#">:first-child</a>	p:first-child	Selects every <p> elements that is the first child of its parent
<a href="#">:first-of-type</a>	p:first-of-type	Selects every <p> element that is the first <p> element of its parent
<a href="#">:focus</a>	input:focus	Selects the <input> element that has focus
<a href="#">:hover</a>	a:hover	Selects links on mouse over
<a href="#">:in-range</a>	input:in-range	Selects <input> elements with a value within a specified range
<a href="#">:invalid</a>	input:invalid	Selects all <input> elements with an invalid value
<a href="#">:lang(language)</a>	p:lang(it)	Selects every <p> element with a lang attribute value starting with "it"
<a href="#">:last-child</a>	p:last-child	Selects every <p> elements that is the last child of its parent
<a href="#">:last-of-type</a>	p:last-of-type	Selects every <p> element that is the last <p> element of its parent
<a href="#">:link</a>	a:link	Selects all unvisited links
<a href="#">:not(selector)</a>	:not(p)	Selects every element that is not a <p> element
<a href="#">:nth-child(n)</a>	p:nth-child(2)	Selects every <p> element that is the second child of its parent
<a href="#">:nth-last-child(n)</a>	p:nth-last-child(2)	Selects every <p> element that is the second child of its parent, counting from the last child
<a href="#">:nth-last-of-type(n)</a>	p:nth-last-of-type(2)	Selects every <p> element that is the second <p> element of its parent, counting from the last child
<a href="#">:nth-of-type(n)</a>	p:nth-of-type(2)	Selects every <p> element that is the second <p> element of its parent
<a href="#">:only-of-type</a>	p:only-of-type	Selects every <p> element that is the only <p> element of its parent

<a href="#">:only-child</a>	p:only-child	Selects every <p> element that is the only child of its parent
<a href="#">:optional</a>	input:optional	Selects <input> elements with no "required" attribute
<a href="#">:out-of-range</a>	input:out-of-range	Selects <input> elements with a value outside a specified range
<a href="#">:read-only</a>	input:read-only	Selects <input> elements with a "readonly" attribute specified
<a href="#">:read-write</a>	input:read-write	Selects <input> elements with no "readonly" attribute
<a href="#">:required</a>	input:required	Selects <input> elements with a "required" attribute specified
<a href="#">:root</a>	root	Selects the document's root element
<a href="#">:target</a>	#news:target	Selects the current active #news element (clicked on a URL containing that anchor name)
<a href="#">:valid</a>	input:valid	Selects all <input> elements with a valid value
<a href="#">:visited</a>	a:visited	Selects all visited links

#### - Selektor Pseudo-element

Elemen pseudo CSS digunakan untuk menata bagian tertentu dari suatu elemen.

Misalnya, dapat digunakan untuk:

- Gaya huruf pertama, atau baris, dari suatu elemen
  - Sisipkan konten sebelum, atau setelah, konten elemen

## Syntax pseudo-elements

```
selector::pseudo-element {  
    property: value;  
}
```

## Contoh:

```
p::first-line {  
    color: #ff0000;  
    font-variant: small-caps;  
}
```

Berikut ini adalah beberapa pseudo-elements yang dapat digunakan:

Selector	Example	Example description
<a href="#">::after</a>	p::after	Insert something after the content of each <p> element
<a href="#">::before</a>	p::before	Insert something before the content of each <p> element

<code>::first-letter</code>	<code>p::first-letter</code>	Selects the first letter of each <code>&lt;p&gt;</code> element
<code>::first-line</code>	<code>p::first-line</code>	Selects the first line of each <code>&lt;p&gt;</code> element
<code>::marker</code>	<code>::marker</code>	Selects the markers of list items
<code>::selection</code>	<code>p::selection</code>	Selects the portion of an element that is selected by a user

- Selektor atribut

Selector [attribute] digunakan untuk memilih elemen dengan atribut tertentu. Dimungkinkan untuk menata elemen HTML yang memiliki atribut atau nilai atribut tertentu. Contoh berikut memilih semua `<a>` elemen dengan atribut target:

Syntax atribut

<code>a[target] {     background-color: yellow; }</code>	<code>a[target="_blank"] {     background-color: purple; }</code>
<code>&lt;h2&gt;CSS [attribute="value"] Selector&lt;/h2&gt; &lt;a href="https://www.cnnindonesia.com/"&gt;CNN Indonesia&lt;/a&gt; &lt;a target="_self" href="https://www.detik.com/"&gt;Detik.com&lt;/a&gt; &lt;a target="_blank" href="https://www.kompas.com/"&gt;Kompas.com&lt;/a&gt;</code>	

#### 2.2.4. CSS Inline

Atribut style digunakan untuk menambahkan gaya ke elemen, seperti warna, font, ukuran, dan lainnya. Pengaturan style pada suatu elemen, dapat dilakukan dengan atribut style. Atribut style HTML memiliki sintaks berikut:

<code>&lt;tagname style="property:value;"&gt;</code>
--

Properti adalah properti CSS. Value adalah nilai CSS. Contoh penggunaan *css inline*:

<code>&lt;h1 style="text-align:center;"&gt;Centered Heading&lt;/h1&gt; &lt;p style="text-align:center;"&gt;Centered paragraph.&lt;/p&gt;</code>
<code>&lt;h1 style="font-size:300%;"&gt;This is a heading&lt;/h1&gt; &lt;p style="font-size:160%;"&gt;This is a paragraph.&lt;/p&gt;</code>
<code>&lt;h1 style="font-family:verdana;"&gt;This is a heading&lt;/h1&gt; &lt;p style="font-family:courier;"&gt;This is a paragraph.&lt;/p&gt;</code>
<code>&lt;h1 style="color:blue;"&gt;This is a heading&lt;/h1&gt; &lt;p style="color:red;"&gt;This is a paragraph.&lt;/p&gt;</code>
<code>&lt;h1 style="background-color:blue;"&gt;This is a heading&lt;/h1&gt; &lt;p style="background-color:tomato;"&gt;This is a paragraph.&lt;/p&gt;</code>

## 2.2.5. CSS Internal

Kode CSS internal diletakkan di dalam bagian `<head>` pada halaman. *Class* dan *ID* bisa digunakan untuk merujuk pada kode CSS, namun hanya akan aktif pada halaman tersebut. Style CSS yang dipasang dengan metode ini akan di-download setiap kali halaman dipanggil, jadi ini akan meningkatkan kecepatan akses.

Namun, ada beberapa kasus dimana penggunaan internal *stylesheet* justru berguna. Salah satu contohnya adalah untuk mengirimkan template halaman ke seseorang – karena semuanya bisa terlihat dalam 1 halaman, maka akan lebih mudah untuk melihat previewnya. CSS internal diletakkan di dalam tag `<style></style>`. Contohnya:

```
<head>
<style>
  body {background-color: blue;}
  h1   {color: blue; font-family: verdana; font-family: courier; }
  p    {color: red;}
  .center {text-align: "center";}
  #card {border: 2px solid blue; margin: 50px;}
  #card > h1 {color: tomato}
  #card > p {background-color: pink}
</style>
</head>
<body>
<h1>Centered Heading</h1>
<p>Centered paragraph.</p>

<h1 class="center">This is a heading</h1>
<p class="center">This is a paragraph.</p>

<div id="card">
  <h1>This is a heading</h1>
  <p>This is a paragraph.</p>
</div>
</body>
```

## 2.2.6. CSS Eksternal

Salah satu cara yang paling nyaman untuk menambahkan CSS ke website adalah dengan menghubungkannya ke file .CSS eksternal. Dengan cara tersebut, perubahan apapun yang Anda buat pada file CSS akan tampil pada website Anda secara keseluruhan. File CSS eksternal biasanya diletakkan setelah bagian `<head>` pada halaman:

```
<head>
  <link rel="stylesheet" href="styles.css">
</head>
```

## File styles.css

```
body {background-color: blue;}  
h1 {color: blue; font-family: verdana; font-family: courier; }  
p {color: red;}  
.center {text-align: "center";}  
#card {border: 2px solid blue; margin: 50px; }  
#card > h1 {color: tomato}  
#card > p {background-color: pink}
```

### 2.2.7. Component Properties

#### 1. Comments

Komentar digunakan untuk menjelaskan kode, dan dapat membantu saat Anda mengedit kode sumber di kemudian hari. Komentar diabaikan oleh browser. Komentar CSS ditempatkan di dalam elemen <style>, dan dimulai dengan /\* dan diakhiri dengan \*/:

```
/* Set text paragraph color to red */  
p {  
    color: red;  
}
```

#### 2. Colors

Color ditentukan menggunakan nama warna yang telah ditentukan sebelumnya, atau nilai RGB, HEX, HSL, RGBA, HSLA. Di CSS, warna dapat ditentukan dengan menggunakan nama warna yang telah ditentukan:



Mengatur warna latar belakang untuk elemen HTML:

```
<h1 style="background-color:DodgerBlue;">Hello World</h1>  
<p style="background-color: #4CAF50;">Lorem ipsum...</p>
```

Mengatur set color pada text:

```
<h1 style="color:Tomato;">Hello World</h1>  
<p style="color: #1e90ff;">Lorem ipsum...</p>  
<p style="color: rgb(30 255 203);">Ut wisi enim...</p>
```

Mengatur set color pada border:

```
<h1 style="border:1px dashed Tomato;">Hello World</h1>
<p style="border:2px solid #1e90ff;">Lorem ipsum...</p>
<p style="border:3px dotted rgb(30 255 203);">Ut wisi enim...</p>
```

### 3. Backgrounds

Properti *background* CSS digunakan untuk menambahkan efek latar belakang untuk elemen. Dalam bagian ini, berikut ini beberapa properti background CSS yang dapat digunakan:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position
- background

### 4. Borders

Properti *border* CSS memungkinkan untuk menentukan style, lebar, dan warna pada setiap elemen. Properti border-style menentukan jenis border apa yang akan ditampilkan.

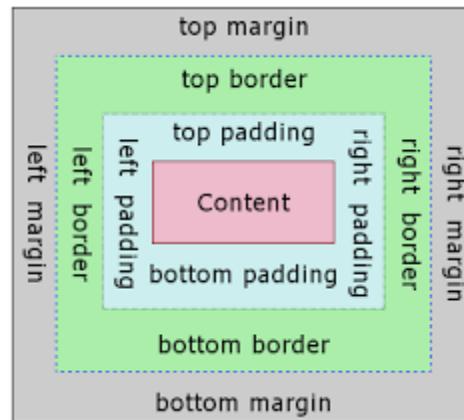
```
p{border-style: value;}
```

Berikut ini beberapa jenis **value** border yang dapat digunakan:

- dotted
- dashed
- solid
- double
- groove
- ridge
- inset
- outset
- none
- hidden

## 5. Margins dan Paddings

Margin dan padding merupakan properties CSS yang diperlukan untuk menentukan batas setiap element. Margin diperlukan untuk menentukan batas luar sedangkan padding diperlukan untuk batas dalam pada suatu elemen.



### - Margin

Margin digunakan untuk membuat ruang di sekitar elemen, di luar batas yang ditentukan. Properti margin CSS digunakan untuk membuat ruang di sekitar elemen, batas diluar elemen. Dengan CSS, memiliki kontrol penuh atas margin. Ada properti untuk mengatur margin untuk setiap sisi elemen (top, right, bottom, dan left).

CSS memiliki properti untuk menentukan margin untuk setiap sisi elemen:

- margin-top
- margin-right
- margin-bottom
- margin-left
- margin

Semua properti margin dapat memiliki nilai sebagai berikut:

- auto □ browser menghitung margin
- length □ menentukan margin dalam px, pt, cm, dll.
- % □ menentukan margin dalam % dari lebar elemen yang mengandung
- inherit □ menentukan bahwa margin harus diwarisi dari elemen induk

Contoh

```

p {
  margin-top: 100px;
  margin-bottom: 100px;
  margin-right: 150px;
  margin-left: 80px;
}
h1{ margin: 25px 50px 75px 100px; }

```

### - **Paddings**

Jika margin diperuntukan untuk menentukan batas luar pada element, berbeda dengan padding. Padding diperuntukan untuk menentukan batas dalam pada element.

Properti padding CSS digunakan untuk menghasilkan ruang di sekitar konten elemen, di dalam batas yang ditentukan. Dengan CSS, memiliki kontrol penuh atas padding. Ada properti untuk mengatur padding untuk setiap sisi elemen (atas, kanan, bawah, dan kiri):

- padding-top
- padding-right
- padding-bottom
- padding-left
- padding

Semua properti margin dapat memiliki nilai sebagai berikut:

- length □ menentukan margin dalam px, pt, cm, dll.
- % □ menentukan margin dalam % dari lebar elemen yang mengandung
- inherit □ menentukan bahwa margin harus diwarisi dari elemen induk

Contoh:

```

div {
  padding-top: 50px;
  padding-right: 30px;
  padding-bottom: 50px;
  padding-left: 80px;
}
.card {
  padding: 25px 50px 75px 100px;
}

```

## 6. Height dan Width

Properti Height dan Width digunakan untuk mengatur tinggi dan lebar suatu elemen. Properti Height dan Width tidak termasuk padding, border, atau margin. Ini mengatur tinggi/lebar area di dalam padding, border, dan margin elemen.

Properti height dan width mungkin memiliki nilai berikut:

- auto □ Ini adalah default. Browser menghitung tinggi dan lebar
- length □ Mendefinisikan tinggi/lebar dalam px, cm dll.
- % □ Mendefinisikan tinggi/lebar dalam persen dari blok yang berisi
- initial □ Mengatur tinggi/lebar ke nilai default
- inherit □ Tinggi/lebar akan diwarisi dari nilai induknya

Contoh:

```
div {  
    height: 200px;  
    width: 50%;  
    background-color: powderblue;  
}
```

## 7. Text

- Text alignment

Properti text-align digunakan untuk mengatur perataan *horizontal* teks. Sebuah teks dapat diratakan kiri atau kanan, di tengah, atau diratakan. Contoh berikut menunjukkan rata tengah, dan teks rata kiri dan kanan (perataan kiri adalah default jika arah teks dari kiri ke kanan, dan perataan kanan adalah default jika arah teks dari kanan ke kiri):

```
h1 {text-align: center; }  
h2 {text-align: left; }  
h3 {text-align: right; }  
p {text-align: justify; }
```

- Text Decoration

Properti text-decoration digunakan untuk mengatur atau menghapus dekorasi dari teks. Nilai text-decoration: none; sering digunakan untuk menghapus garis bawah dari tautan:

```
a {text-decoration: none; }
```

Contoh:

```
h2 {text-decoration: overline; }  
h3 {text-decoration: line-through; }  
h4 {text-decoration: underline; }
```

- Text Transformation

Properti text-transform digunakan untuk menentukan huruf besar dan huruf kecil dalam sebuah teks. Ini dapat digunakan untuk mengubah semuanya menjadi huruf besar atau kecil, atau menggunakan huruf besar untuk huruf pertama setiap kata:

```
.title {text-transform: uppercase;}  
span {text-transform: lowercase;}  
p {text-transform: capitalize;}
```

- Text Spacing

Text spacing merupakan properti yang diperuntukan untuk memberikan jarak tertentu pada suatu text word. Berikut ini ada beberapa properti yang dapat digunakan untuk menentukan text spacing:

```
p {text-indent: 50px;}  
h1 {letter-spacing: 5px;}  
h2 {letter-spacing: -2px;}  
p.big {line-height: 1.8;}  
.two {word-spacing: -2px;}  
p {white-space: nowrap;}
```

- Text Shadow

Properti text-shadow menambahkan bayangan ke teks. Dalam penggunaannya yang paling sederhana, Anda hanya menentukan bayangan horizontal (2px), bayangan vertikal (2px), efek blur (5px) dan warna red untuk bayangannya:

```
h1 {text-shadow: 2px 2px 5px red; }
```

## 8. Font

Memilih font yang tepat memiliki dampak besar pada bagaimana pembaca mengalami situs web. Font yang tepat dapat menciptakan identitas yang kuat untuk merek Anda. Menggunakan font yang mudah dibaca adalah penting. Font menambah nilai pada teks Anda. Penting juga untuk memilih warna dan ukuran teks yang benar untuk font.

Property	Description
<a href="#">font</a>	Sets all the font properties in one declaration
<a href="#">font-family</a>	Specifies the font family for text
<a href="#">font-size</a>	Specifies the font size of text
<a href="#">font-style</a>	Specifies the font style for text
<a href="#">font-variant</a>	Specifies whether or not a text should be displayed in a small-caps font
<a href="#">font-weight</a>	Specifies the weight of a font

Contoh

```
.p1 {  
    font-family: "Times New Roman", Times, serif;  
    font-size: 40px;  
}  
.normal {  
    font-style: normal;  
    font-variant: normal;  
    font-weight: normal;  
    font-style: normal;  
}  
.italic {font-style: italic;}  
.oblique {font-style: oblique;}  
.thick {font-weight: bold;}  
.small {font-variant: small-caps;}
```

## 9. List

Dalam HTML, ada dua jenis daftar utama:

- List unordered (<ul>) □ item daftar ditandai dengan peluru atau bulat
- List ordered (<ol>) □ item daftar ditandai dengan angka atau huruf

Properti daftar CSS memungkinkan Anda untuk:

- Tetapkan penanda item daftar yang berbeda untuk daftar yang dipesan
- Tetapkan penanda item daftar yang berbeda untuk daftar yang tidak berurutan
- Tetapkan gambar sebagai penanda item daftar
- Tambahkan warna latar belakang ke daftar dan daftar item

Properti list-style-type menentukan tipe penanda item daftar. Contoh berikut menunjukkan beberapa penanda item daftar yang tersedia:

```
ul.a {list-style-type: circle;}  
ul.b {list-style-type: square;}  
ol.c {list-style-type: upper-roman;}  
ol.d {list-style-type: lower-alpha;}
```

## 10. Display

Properti display menentukan perilaku tampilan (tipe kotak rendering) dari suatu elemen. Dalam HTML, nilai properti display default value diambil dari spesifikasi HTML atau dari style css default browser/pengguna. Nilai default dalam XML adalah inline, termasuk elemen SVG.

```
p.ex1 {display: none;}  
p.ex2 {display: inline;}  
p.ex3 {display: block;}  
p.ex4 {display: inline-block;}
```

## 11. Float

Properti float digunakan untuk memposisikan dan memformat konten, misalkan biarkan gambar melayang ke kiri ke teks dalam wadah. Properti float dapat memiliki salah satu dari nilai berikut:

- left ┃ Elemen mengapung di sebelah kiri wadahnya
- right ┃ Elemen mengapung di sebelah kanan wadahnya
- none ┃ Elemen tidak mengapung (akan ditampilkan tepat di tempat kemunculannya dalam teks). Ini adalah default
- inherit ┃ Elemen mewarisi nilai float dari induknya

Dalam penggunaannya yang paling sederhana, properti float dapat digunakan untuk membungkus teks di sekitar gambar.

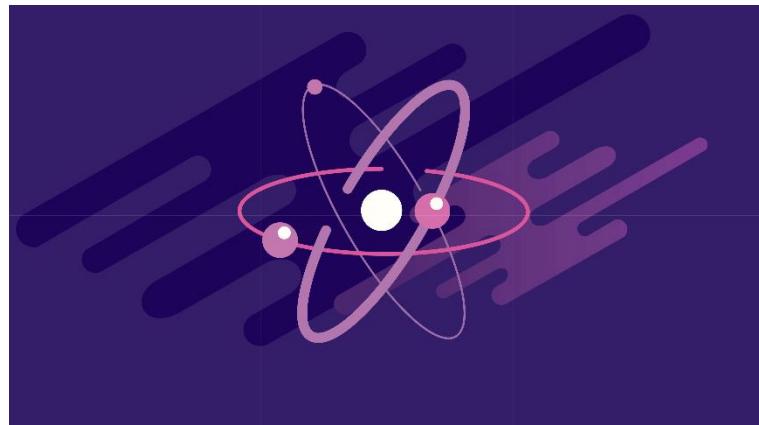
```
img {float: right;}
```

### 2.3. Tugas dan Latihan

1. Buatlah file dokumen HTML bernama Latihan-1, ubahlah contoh dibawah ini kedalam scripting HTML dan menggunakan CSS sebagai variasi style pada elemen HTML:

#### Mengenal Partikel dan Notasi Atom

Nama anda, 20 Feb 2022 – 08:55 am



Perhatikan sekeliling kalian, matahari terbit dari timur di pagi hari, bulan muncul pada malam hari, bumi mengelilingi matahari dalam dua belas bulan, dan banyak lagi keteraturan di alam semesta ini. Hebat ya *Sang Pencipta* kita mengatur alam semesta ini dengan rapi. Bahkan, sampai tingkat paling kecil pun, elektron-elektron di alam semesta ini telah diatur dengan rapi menurut bilangan kuantumnya! **Wow apa tuh bilangan kuantum?**

Elektron-elektron tersebar di sekeliling atom dengan teratur berdasarkan tingkat energinya. Nah, tingkat energi inilah yang digambarkan dengan bilangan kuantum. Artinya, dari bilangan kuantum, lokasi-lokasi penyebaran elektron dapat digambarkan. Sedetail itu loh Sang Pencipta kita mengurnya. Bayangkan kalau elektron, penyusun segala sesuatu di alam semesta ini, tidak teratur. Alam semesta ini tidak stabil dong. Mana bisa kita hidup di dunia seperti itu. Keren kan?

Salah satu contoh atom di alam semesta ini adalah atom karbon. Atom kabon adalah penyusun dari berbagai benda yang sangat berguna. Mulai dari bensin, plastik, berlian, bahkan tubuh kita pun tersusun dari karbon! Nah, karbon (*biasa dilambangkan dengan huruf C*) punya 6 elektron. Bagaimana bilangan kuantum dari elektron terakhirnya? Tinggal ikuti deh langkah-langkahnya.

#### Partikel Dasar Penyusun Atom dan Lambang Atom

Partikel dasar penyusun atom ada tiga yaitu proton (*p*), neutron (*n*) dan elektron (*e*). Jadi, massa atom = (massa *p* + massa *n*) + massa *e*. Massa elektron jauh lebih kecil dari pada massa proton dan massa neutron, maka massa elektron dapat diabaikan.

Dengan demikian: massa atom = massa *p* + massa *n*.

Partikel	Lambang	Massa(g)	Muatan	
			Satuan	Coulomb
proton	p	$1.673 \times 10^{-24}$	+1	$1.6 \times 10^{-9}$
neutron	n	$1.673 \times 10^{-24}$	0	0
elektron	e	$9.109 \times 10^{-28}$	-1	$1.6 \times 10^{-19}$

### Lambang Atom



X Simbol dari unsur.

- a nomor atom merupakan jumlah proton. Saat netral (tidak bermuatan) akan sama dengan jumlah elektron.
- b nomor massa melambangkan jumlah proton ditambah jumlah neutron atau disebut juga jumlah nukleon.
- c Muatan/bilangan oksidasi (biloks) terdiri dari melepas elektron (positif) dan menangkap elektron atau bertambah (negatif).

Bagikan artikel ini



2. Buatlah file dokumen HTML bernama Latihan-2, ubahlah contoh dibawah ini kedalam scripting HTML dan menggunakan CSS sebagai variasi style pada elemen HTML:

### Daftar Hewan Dilindungi



#### Orang Utan

Orang utan dicirikan oleh rambut di seluruh badannya yang berwarna kemerahan. Satwa ini merupakan mamalia arboreal terbesar yang menghabiskan hampir seluruh waktunya di pepohonan. Lengannya yang panjang dan kuat serta tangan dan kakinya yang dapat mencengkeram erat.



#### Kera Emas

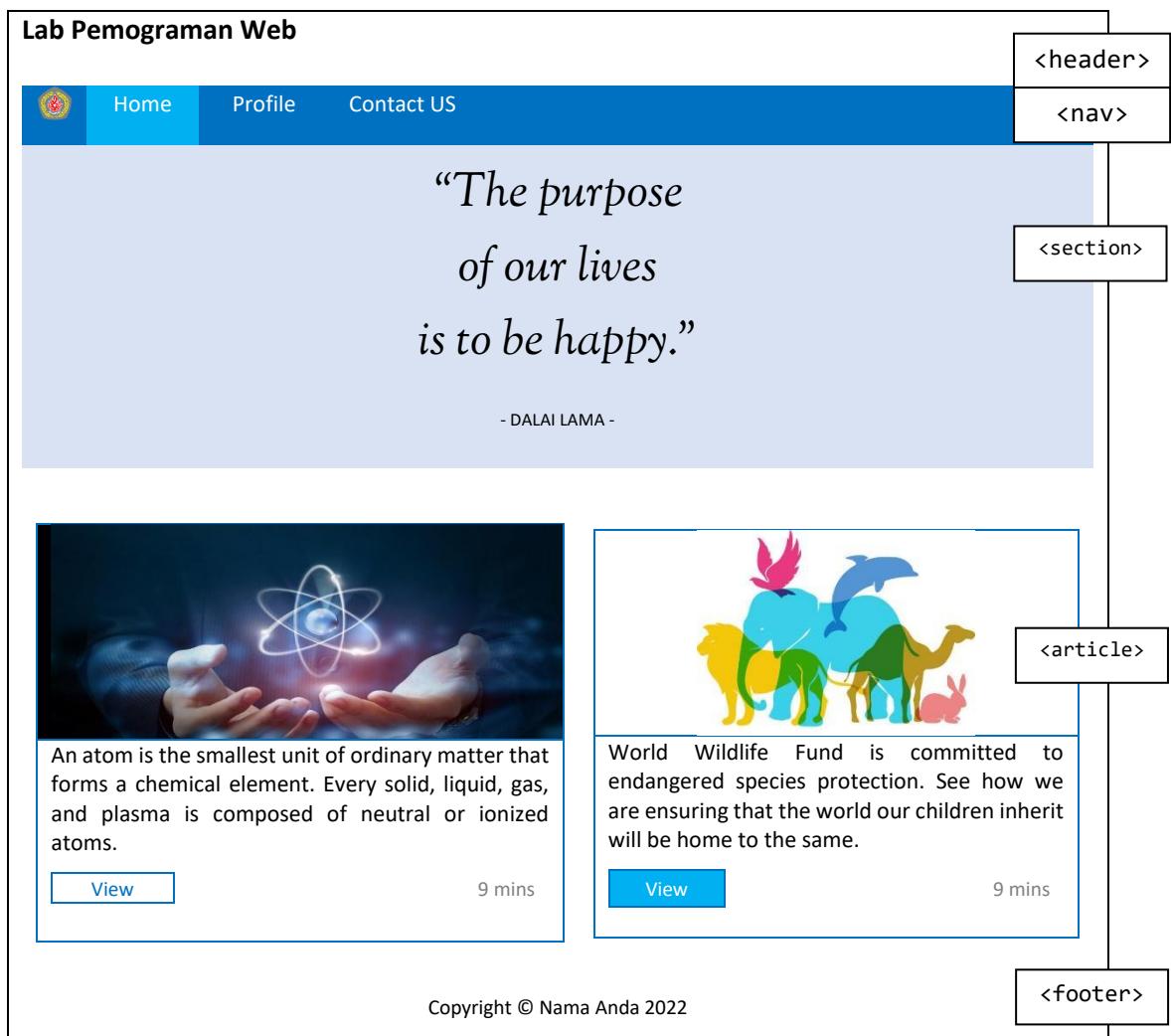
Monyet yang unik ini juga dikenal sebagai hewan yang memiliki adaptasi yang sangat baik dengan lingkungan yang bersuhu sangat dingin. Umumnya, monyet hidung pesek rambut emas tinggal di hutan pegunungan.



#### Burung Cenderawasih

Burung cenderawasih termasuk ke dalam hewan langka dan hanya bisa ditemukan di Papua. Burung cenderawasih punya tampilan yang cantik yang membuat burung ini sering menjadi perburuan liar untuk didagangkan.

3. Buatlah file dokumen HTML bernama Latihan-3, ubahlah contoh dibawah ini kedalam scripting HTML dan menggunakan CSS sebagai variasi style pada elemen HTML:



Pada storyboard diatas untuk elemen <article> terdapat dua buah artikel dimana masing-masing artikel tersebut akan me-redirect ke halaman *Latihan-1.html* dan *Latihan-2.html* ketika mengklik tombol *view*. Jika tombol *view* di sorot maka menampilkan background color.

4. Buatlah file dokumen HTML bernama Latihan-4, buatlah satu halaman dokument HTML berisi data portfolio anda dan gunakan CSS sebagai style untuk membentuk Layout. Sambungkan halaman tersebut dengan Latihan-3 ketika mengklik tombol Profile pada elemen <nav>

## **MODUL 3**

### **PENGENALAN BAHASA PEMROGRAMAN JAVASCRIPT**

#### **3.1. Tujuan Praktikum**

Mahasiswa memahami simbol atau karakter yang biasa dilibatkan dalam program untuk melakukan sesuatu operasi atau manipulasi.

#### **3.2. Dasar Teori**

##### *3.2.1. Pengantar*

Javascript diperkenalkan pertama kali oleh Netscape pada tahun 1995. Pada awalnya bahasa ini dinamakan “LiveScript” yang berfungsi sebagai bahasa sederhana untuk browser Netscape Navigator 2. Pada masa itu bahasa ini banyak di kritik karena kurang aman, pengembangannya yang terkesan buru buru dan tidak ada pesan kesalahan yang di tampilkan setiap kali kita membuat kesalahan pada saat menyusun suatu program. Kemudian sejalan dengan sedang giatnya kerjasama antara Netscape dan Sun (pengembang bahasa pemrograman “Java” ) pada masa itu, maka Netscape memberikan nama “JavaScript” kepada bahasa tersebut pada tanggal 4 desember 1995. Pada saat yang bersamaan Microsoft sendiri mencoba untuk mengadaptasikan teknologi ini yang mereka sebut sebagai “Jscript” di browser Internet Explorer 3.

Javascript adalah bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML, sepanjang sejarah internet bahasa ini adalah bahasa skrip pertama untuk web. Bahasa ini adalah bahasa pemrograman untuk memberikan kemampuan tambahan terhadap bahasa HTML dengan mengijinkan pengeksekusian perintah perintah di sisi user, yang artinya di sisi browser bukan di sisi server web.

Javascript bergantung kepada browser(navigator) yang memanggil halaman web yang berisi skrip skrip dari Javascript dan tentu saja terselip di dalam dokumen HTML. Javascript juga tidak memerlukan kompilator atau penterjemah khusus untuk menjalankannya (pada kenyataannya kompilator Javascript sendiri sudah termasuk di dalam browser tersebut). Lain halnya dengan bahasa “Java” (dengan mana JavaScript selalu di banding bandingkan) yang memerlukan kompilator khusus untuk menterjemahkannya di sisi user/klien. Untuk mempelajari pemrograman Java Script, ada dua piranti yang diperlukan, yaitu :

- Teks Editor

Digunakan untuk menuliskan kode-kode Java Script, teks editor yang dapat digunakan antara lain notepad dan ultra edit.

- Web Browser

Digunakan untuk menampilkan halaman web yang mengandung kode-kode Java Script. Web browser yang digunakan harus mendukung Java Script. Browser yang dapat digunakan adalah internet explorer dan Netscape Navigator.

### *3.2.2. Penulisan Javascript*

Kode Java Script dituliskan pada file HTML. Terdapat dua cara untuk menuliskan kode-kode Java Script agar dapat ditampilkan pada halaman HTML, yaitu :

- a) Java script ditulis pada file yang sama

Untuk penulisan dengan cara ini, perintah yang digunakan adalah

<SCRIPT LANGUAGE = "JavaScript" > program java script disini </SCRIPT>. Perintah tersebut biasanya diletakkan diantara Tag <BODY> ... </BODY> Contoh Penulisan :

```

<HTML>
<HEAD>
<TITLE>.....</TITLE>
<SCRIPT LANGUAGE="Javascript">
    kode javascript disini
</SCRIPT>
</HEAD>
<BODY>
    kode HTML disini
</BODY>
</HTML>
```

- b) Javascript ditulis pada file terpisah

Kode Javascript bisa juga kita buat dalam file terpisah dengan tujuan agar dokumen HTML isinya tidak terlalu panjang. Atribut yang digunakan adalah

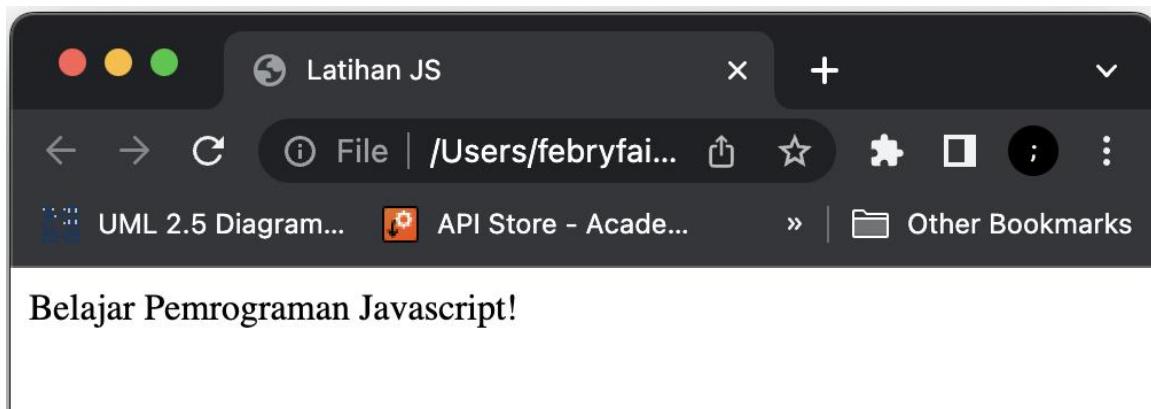
<SCRIPT SRC= "namafile.js" > ... </SCRIPT>

Diantara tag <SCRIPT.....> dan <SCRIPT> tidak diperlukan lagi kode Javascriptnya karena sudah dibuat dalam file erpisah. File yang mengandung kode Javascript berekstensi .js Pada bagian ini kita akan membuat program dengan menggunakan Javascript. Program ini akan menampilkan teks “ Belajar Pemrograman Javascript”

```

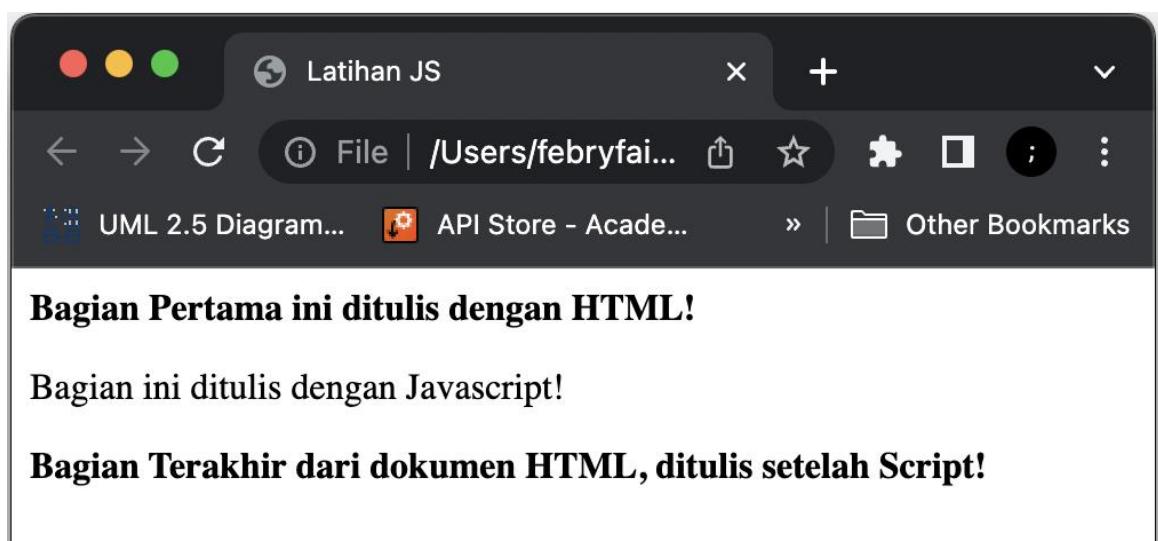
<HTML>
<HEAD>
<SCRIPT language="JavaScript">
<!--
document.write("Belajar Pemrograman Javascript!");
//-->
</SCRIPT>
</HEAD>
```

```
<BODY></BODY>  
</HTML>
```



Berikut diberikan beberapa contoh program sederhana dengan menggunakan Javascript

```
<HTML>  
<HEAD>  
</HEAD>  
<BODY>  
<B>Bagian Pertama ini ditulis dengan HTML!</B>  
<P>  
<SCRIPT language="JavaScript">  
<!--  
document.write("Bagian ini ditulis dengan Javascript!");  
//-->  
</SCRIPT>  
<P>  
<B>Bagian Terakhir dari dokumen HTML, ditulis setelah Script!</B>  
</BODY>  
</HTML>
```



```
<HTML>  
<HEAD><TITLE>BELajar Javascript Yuuuuuu</TITLE>
```

```

</HEAD>
<BODY BGCOLOR="gray">
<SCRIPT language="JavaScript">
<!--
document.writeln("<PRE>"); document.write("<B><FONT SIZE=5>");
document.writeln("SELAMAT DATANG DI JAVASCRIPT");
document.write("</B></FONT>"); document.write("<i>");
document.writeln("Program ini merupakan contoh sederhana menampilkan Teks!");
document.write("</i>");
document.writeln("Dengan Javascript!!!!!!!");
//-->
</SCRIPT>
</BODY>
</HTML>

```



Objek document mempunyai dua metode untuk menuliskan teks, yaitu **write** dan **writeln**. Metode **write** digunakan untuk menuliskan teks tanpa ganti baris, sedangkan metode **writeln** digunakan untuk menuliskan teks dengan ganti baris.

Berikut diberikan contoh Program Javascript yang diletakkan di file lain dan dipanggil melalui suatu file HTML

#### File external.js

```

document.writeln("<CENTER>");
document.writeln("<HR WIDTH=600 COLOR=Black>");
document.writeln("<H1>INSTITUTE BISNIS DAN INFORMATIK KESATUAN BOGOR</H1>");
document.writeln("<H2>FAKULTAS INFORMATIKA DAN PARIWISATA<H2>");
document.writeln("<H3>JURUSAN TEKNOLOGI INFORMASI<H3>");
document.writeln("<HR WIDTH=600 COLOR=Black>");
document.writeln("</CENTER>");

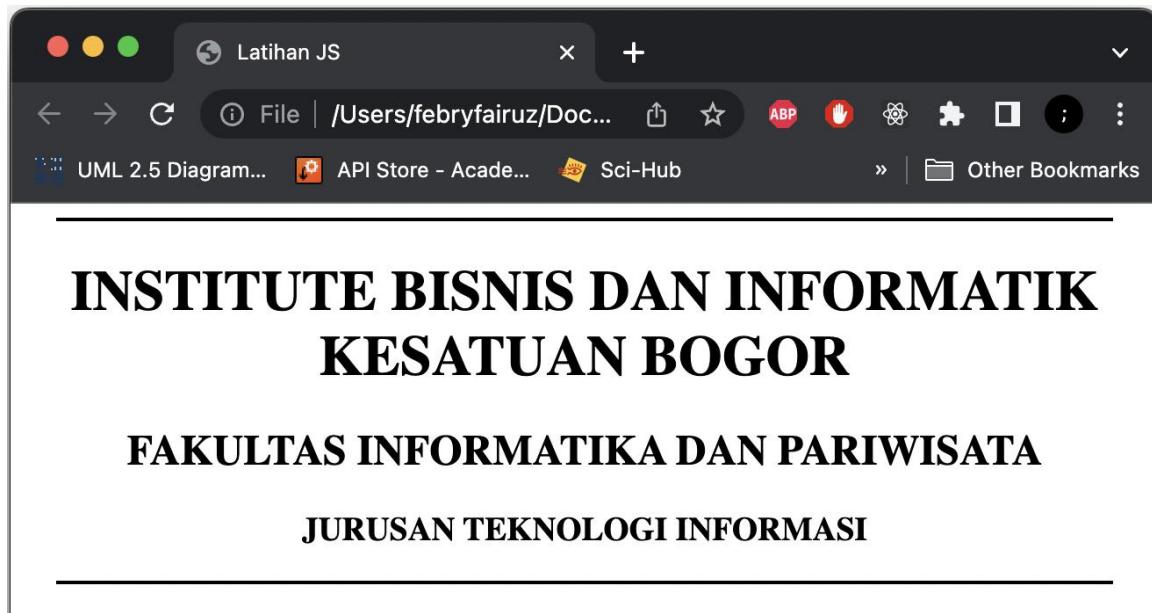
```

#### FileHTML

```

<HTML>
<HEAD>
<SCRIPT language="JavaScript" SRC="./external.js"></SCRIPT>
</HEAD>
<BODY>
<B>Contoh ini menggunakan Javascript yang diambil dari File lain..</B>
</BODY>
</HTML>

```



Sama seperti bahasa pemrograman lain. Javascript juga menyediakan fasilitas untuk menuliskan komentar, komentar ini berguna bila nantinya anda atau orang lain membaca program. Pemberian komentar dalam Javascript dapat dilakukan dengan dua cara yaitu dengan menuliskan komentar setelah tanda garis miring dua kali, contoh :

//ini komentar

atau

/\*ini juga komentar \*/

### 3.2.3. Variables

Variabel adalah empat dimana kita menyimpan nilai-nilai atau informasi-informasi pada JavaScript. Variabel yang dideklarasikan dapat diisi dengan nilai apa saja. Dalam JavaScript pendeklarasian sebuah variabel sifatnya opsional, artinya anda boleh mendeklarasikan atau tidak hal tersebut tidak menjadi masalah. Jika anda memberi nilai pada variabel, maka dalam JavaScript dianggap bahwa anda telah mendeklarasikan variabel tersebut.

Aturan penamaan variabel :

- Harus diawali dengan karakter (huruf atau baris bawah)
- Tidak boleh menggunakan spasi

- Huruf Kapital dan kecil memiliki arti yang berbeda
- Tidak boleh menggunakan kata-kata yang merupakan perintah dalam JavaScript.

### Deklarasi Variabel

**Var nama\_variabel = nilai**

Atau

**Nama\_variabel = nilai**

### Contoh :

var nama;	Nama = "Bunga Lestari"
var nama = " Zaskia Mecca"	X = 1990;
var X = 1998; Y = 08170223513 var Y;	

### *3.2.4. Tipe Data*

Tidak seperti bahasa pemrograman lainnya, JavaScript tidak memiliki tipe data secara explisit. Hal ini dapat dilihat dari beberapa contoh variabel diatas. Anda mendeklarasikan variabel tapi tidak menentukan tipenya. Meskipun JavaScript tidak memiliki tipe data secara explisit. JavaScript mempunyai tipe data implisit. Terdapat empat macam tipe data implisit yang dimiliki oleh JavaScript yaitu:

- Numerik, seperti : 0222532531, 1000, 45, 3.146789 dsb
- String, seperti : "Hallo", "April", "Jl. Setiabudi No 17A", "Cece Kirani" dsb
- Boolean, bernilai true atau false

Null, variabel yang tidak diinisialisasi

### *3.2.5. Tipe Numerik*

Pada dasarnya JavaScript hanya mengenal dua macam tipe numerik, yaitu bilangan bulat (integer) dan bilangan pecahan(real/float).

Untuk bilangan bulat, kita dapat merepresentasikan dengan basis desimal, oktal atau heksadesimal.

### Contoh :

var A = 100;

var B = 0x2F;

untuk pendeklarasian tipe bilangan real, dapat menggunakan tanda titik atau notasi ilmiah (notasi E).

Contoh :

```
var a = 3.14533567;  
var b = 1.23456E+3;
```

### *3.2.6 Tipe String*

Untuk mendeklarasikan tipe string dapat dilakukan dengan cara menuliskan string diantara tanda petik tunggal ('') atau tanda petik ganda ("") Contoh : var str ='Contoh deklarasi string';

```
var str1 = "cara ini juga bisa untuk menulis string";
```

### *3.2.7 Tipe Boolean*

Tipe boolean hanya mempunyai nilai True atau False. Tipe ini biasanya digunakan untuk mengecek suatu kondisi atau keadaan.

Contoh :

```
var X = (Y > 90); contoh diatas menunjukkan bahwa jika Y lebih besar dari 90 maka X akan bernilai True.
```

### *3.2.8 Tipe Null*

Tipe Null digunakan untuk merepresentasikan variabel yang tidak diberi nilai awal (inisialisasi).

### *3.2.9. Operator*

Operator pada JavaScript terbagi menjadi enam, yaitu :

- Aritmatika
- Pemberian nilai (Assign)
- Pemanipulasi bit (bitwise)
- Pembanding
- Logika
- String

### a. Operator Aritmatika

Digunakan untuk operan beripe numerik. Ada dua macam operator aritmatik, yaitu operator numerik tunggal dan operator aritmatik biner. Perbedaan kedua operator terletak pada jumlah operan yang harus dioperasikan.

Operator	Tunggal/Biner	Keterangan
+	Biner	Penjumlahan
-	Biner	Pengurangan
*	Biner	Perkalian
/	Biner	Pembagian
%	Biner	Modulus
-	Tunggal	Negasi
++	Tunggal	Penambahan dengan satu
--	Tunggal	Pengurangan dengan satu

### b. Operator Pemberian Nilai

Digunakan untuk memberikan nilai ke suatu operan atau mengubah nilai suatu operan.

Operator	keterangan	Contoh	Ekuivalen
=	Sama dengan	X=Y	
+=	Ditambah dengan	X+=Y	X=X+Y
-=	Dikurangi dengan	X-=Y	X=X-Y
*=	Dikali dengan	X*=Y	X=X*Y
/=	Dibagi dengan	X/=Y	X=X/Y
%=	Modulus dengan	X%=Y	X=X% Y
&=	Bit AND dengan	X&Y	X=X& Y
=	Bit OR	X =Y	X=X Y

### c. Operator Manipulasi Bit

Operasi ini berhubungan dengan pemanipulasi bit pada operan bertipe bilangan bulat.

Operator	Keterangan
&	Bit AND
	Bit OR
^	Bit XOR
~	Bit NOT
<<	Geser ke kiri
>>	Geser ke Kanan
>>>	Geser ke kanan dengan diisi nol

Contoh : var A = 12; // A = 1100b var B = 10; // B = 1010b var C = A & B  
maka akan dihasilkan bilangan seperti berikut :

1100b

1010b AND

1000b

var A = 12; var C = A<< 2

var D = A >> 1

maka variabel C akan bernilai 48(0011 0000b) variabel D  
akan bernilai 6 (0110b)

#### d. Operator Pembanding

Digunakan untuk membandingkan dua buah operan. Operan yang dikenal operator ini dapat bertipe string, numerik, maupun ekspresi lain.

Operator	Keterangan
==	Sama dengan
!=	Tidak sama dengan
>	Lebih besar
<	Lebih kecil
>=	Lebih besar atau sama dengan
<=	Lebih kecil atau sama dengan

#### e. Operator Logika

Digunakan untuk mengoperasikan operan yang bertipe boolean.

Operator	Keterangan
&&	Operator logika AND
	Operator Logika OR
!	Operator logika NOT

Contoh :

```
var A = true; var B = false;  
var C = A && B; //menghasilkan false  
var D = A || B ; // false  
var E = !A; //false
```

#### f. Operator String

Selain operator pembanding, operator string pada JavaScript juga mengenal satu operator lagi yang bernama PENGGABUNGAN. Operator ini digunakan untuk menggabungkan beberapa string menjadi sebuah string yang lebih panjang.

Contoh : nama = "Java" + "Script";

akan menghasilkan "JavaScript" pada variabel nama

#### Contoh Program JavaScript

```
<HTML>
<HEAD><TITLE>Operasi Aritmatika</TITLE></HEAD>
<BODY>
<P><SCRIPT language="JavaScript">
<!--
document.writeln("<PRE>");
document.writeln("<H1>Operasi Aritmatik</H1>"); var A = "100"; var B = "200"; var C = 300;
var D = 400; var E = A + B;
document.writeln("100" + "200" = ' + E); E = B + C;
document.writeln("200" + 300 = ' + E); E = C + D;
document.writeln('300 + 400 = ' + E); document.writeln("<PRE>");
//-->
</SCRIPT></P>
</BODY>
</HTML>
```

The screenshot shows a Mac OS X style browser window titled "Latihan JS". The address bar shows the path "/Users/febryfai...". Below the address bar, there are several tabs: "UML 2.5 Diagram...", "API Store - Acade...", and "Other Bookmarks". The main content area displays the following text:

```
"100" + "200" = 100200
"200" + 300 = 200300
300 + 400 = 700
```

### 3.2.10 Masukan Data

Untuk memasukkan data dari keyboard dapat dilakukan dengan menggunakan perintah **input**.

Contoh Program JavaScript

```
<HTML>
<HEAD><TITLE>Memasukkan Bilangan</TITLE></HEAD>
<BODY>
<P><SCRIPT language="JavaScript">
<!--
function jumlah()
{
    var bil1 = parseFloat(document.form.bilangan1.value); if (isNaN (bil1))
bil1=0.0;
    var bil2 = parseFloat(document.form.bilangan2.value); if (isNaN (bil2))
bil2=0.0;  var hasil = bil1 + bil2;
    alert ("Hasil Penjumlahan = " + hasil);
}
//--></SCRIPT></P>
<FORM NAME ="fform">
<H1><BR>Memasukkan Data Lewat Keyboard</H1>
```

```
<PRE>
```

```
Bilangan Pertama :<input type="text" size="11" name="bilangan1">
```

```
Bilangan Kedua :<input type="text" size="11" name="bilangan2">
```

```
</PRE>
```

```
<P>
```

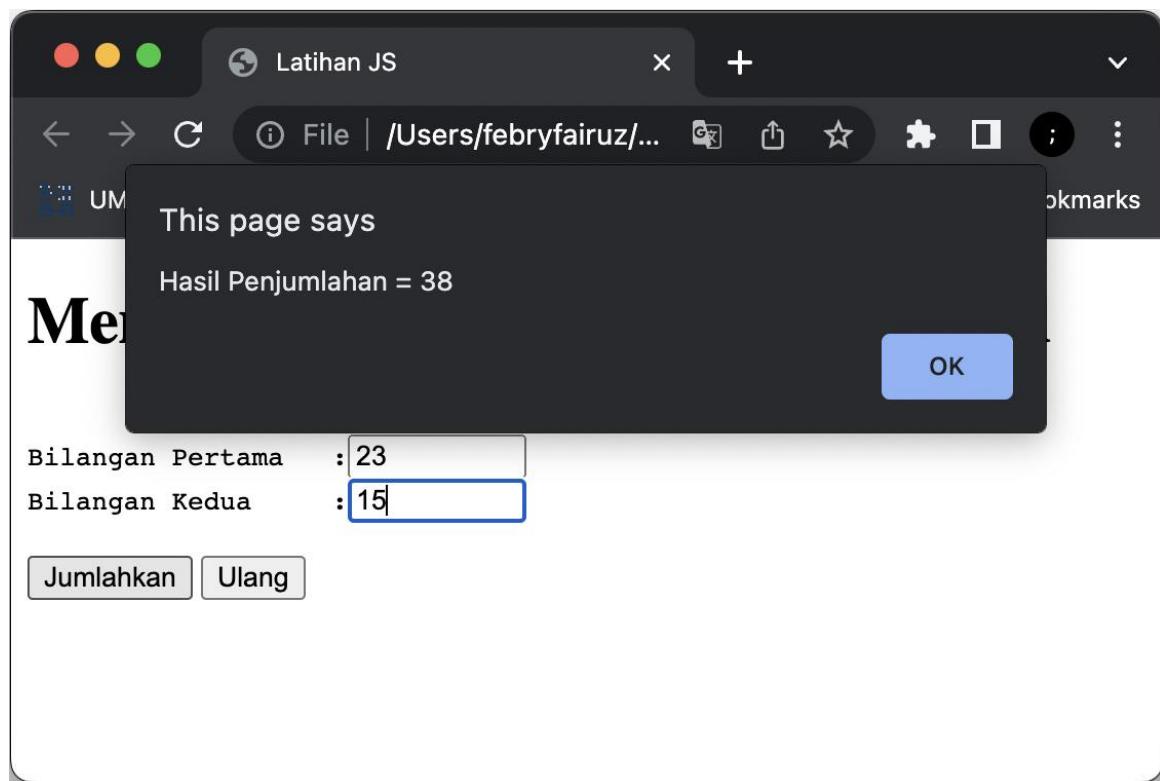
```
<INPUT TYPE="button" value="Jumlahkan" onclick="jumlah()">
```

```
<INPUT TYPE="reset" value="Ulang">
```

```
</FORM>
```

```
</BODY>
```

```
</HTML>
```



### 3.2.11. Object Javascript

#### - Objek Untuk memasukkan Data

Terdapat beberapa objek yang dapat digunakan untuk memasukkan data. Objek-objek tersebut biasanya terdapat dalam suatu form. Adapun objek-objek tersebut meliputi Objek Text, Objek Radio, Objek Checkbox, Objek Textarea, dan Objek Select.

## - Objek Text

Untuk menginputkan data kita dapat menggunakan komponen/objek text. Contoh penggunaannya dapat kita lihat pada contoh berikut :

```
<html>

<head><title>Latihan Dengan Objek Text</title></head>

<body>

<script languange ="JavaScript">

<!--

function tekan()

{

var namastr = (document.fform.nama.value); var alamatstr =
(document.fform.alamat.value); document.fform.onama.value =
namastr; document.fform.oalamat.value = alamatstr;

}

//-->

</script>

<form name ="fform">

<H1> Memasukkan Data Dengan Objek Teks</H1><hr>

<PRE>

Nama Anda :<input type="text" size="11" name="nama">

Alamat :<input type="text" size="25" name="alamat">

</PRE>

<BR>

<input type="button" value="kirim" onclick="tekan()">

<input type="reset" value="ulang">

<H3>Output</H3>

<PRE>

Jadi Nama Anda adalah :<input type="text" size="11" name="onama">

Alamat Anda di :<input type="text" size="25" name="oalamat">

</form>

</body>

</html>
```

The screenshot shows a web browser window titled "Latihan Dengan Objek Text". The address bar indicates the file is located at "/Users/febryfairuz/Docu...". The page content is titled "Memasukkan Data Dengan Objek Teks". It contains a form with two text input fields:

Nama Anda :	Febri D Fairuz
Alamat :	Jl. Kutilang Berbunyi

Below the form are two buttons: "kirim" (blue background) and "ulang" (gray background).

**Output**

Jadi Nama Anda adalah :	Febri D Fairuz
Alamat Anda di :	Jl. Kutilang Berbunyi

### - Objek Radio

Objek radio adalah komponen yang digunakan untuk melakukan suatu pemilihan data. Karena selalu berupa Array , untuk mengakses satu tombol radio digunakan radio[indeks]. Disamping itu objek radio juga mempunyai nilai True jika dipilih dan False jika tidak. Untuk memilih suatu objek radio menggunakan properti Checked.

```
<html>
<head><title>Latihan Dengan Objek Radio</title></head>
<body>
<script languange ="JavaScript">
<!--
function radio_box(form)
{
var ket = "";
if (form.wanita.checked == true)
{
```

```

ket = "Wanita";
} else {
    ket = "Pria";
}
alert('Anda adalah seorang ' +ket);
}

//-->

</script>

<form>

<H1> Memasukkan Data Dengan Objek Radio</H1><hr>

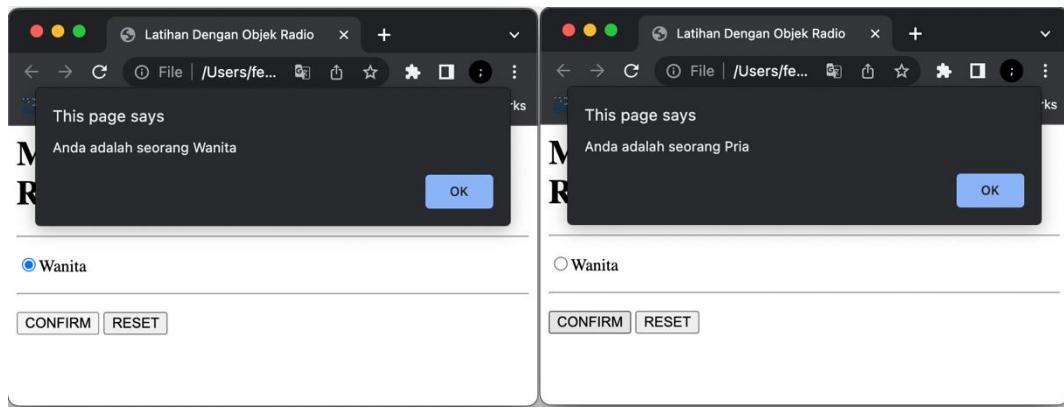
<p><input type="radio" value="wanita" name="wanita">Wanita</p>
<hr>

<p><input type="button" value="CONFIRM"
onclick="radio_box(this.form)"> <input type="reset"
value="RESET"></p>

</form>

</body>
</html>

```



### - Objek Checkbox

Objek checkbox menyimpan informasi tentang elemen form yang berupa kotak cek. Penggunaannya hampir sama seperti objek radio.

```

<html>
<head><title>Latihan Dengan Objek Checkbox</title></head>
<body>

```

```

<script languange ="JavaScript">
<!--
function radio_box(form)
{
var ket = ""; var
ket1 = "";
if (form.bola.checked == true)
{
    ket = "Nonton Bola";
}
if (form.tv.checked == true)
{
    ket1 = " Nonton Tv";
}
alert('Hobby anda ' +ket+''+ket1);
}

//-->
</script>

<form>

<H1> Memasukkan Data Dengan Objek Checkbox</H1><hr>

<p>Hobby anda :

<input type="checkbox" value="ON" name="bola">Nonton Sepak Bola
<input type="checkbox" value="ON" name="tv">Nonton televisi</p>

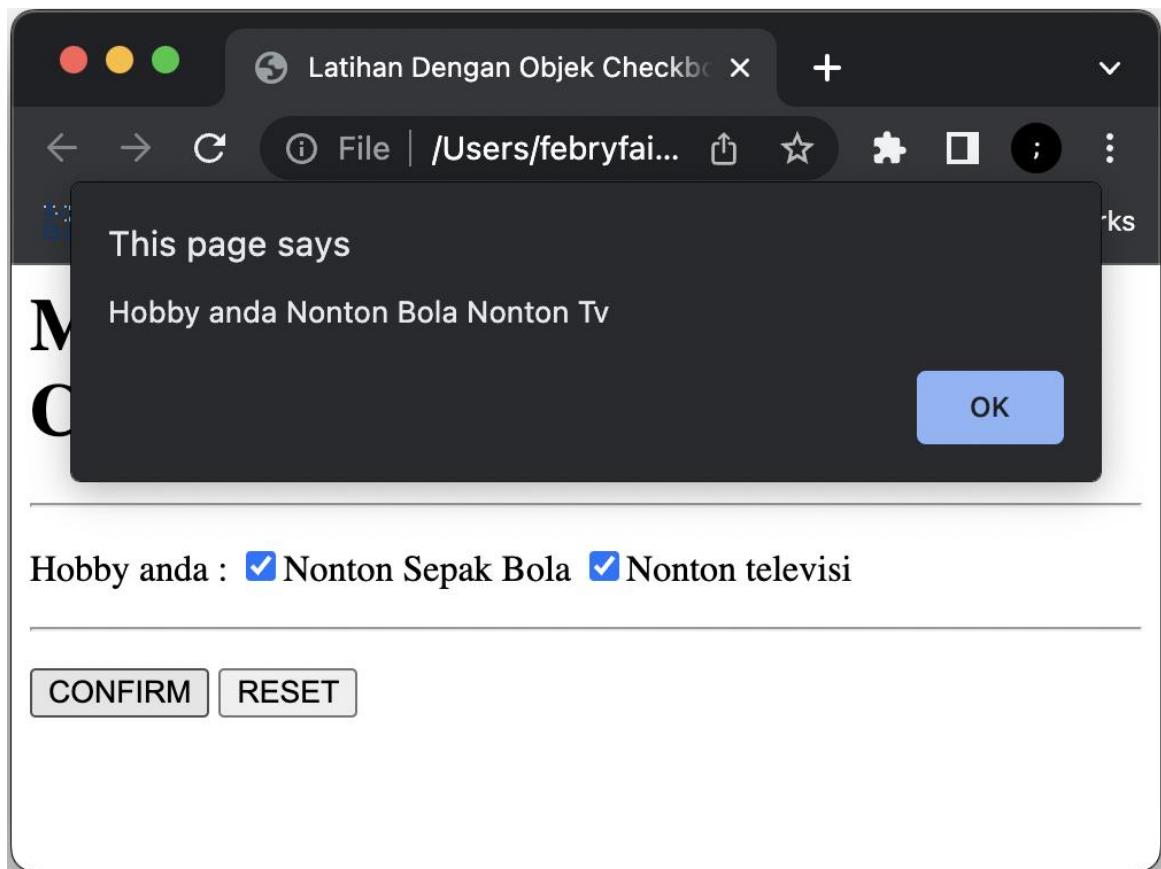
<hr>

<p><input type="button" value="CONFIRM"
onclick="radio_box(this.form)"> <input type="reset"
value="RESET"></p>

</form>

</body>
</html>

```

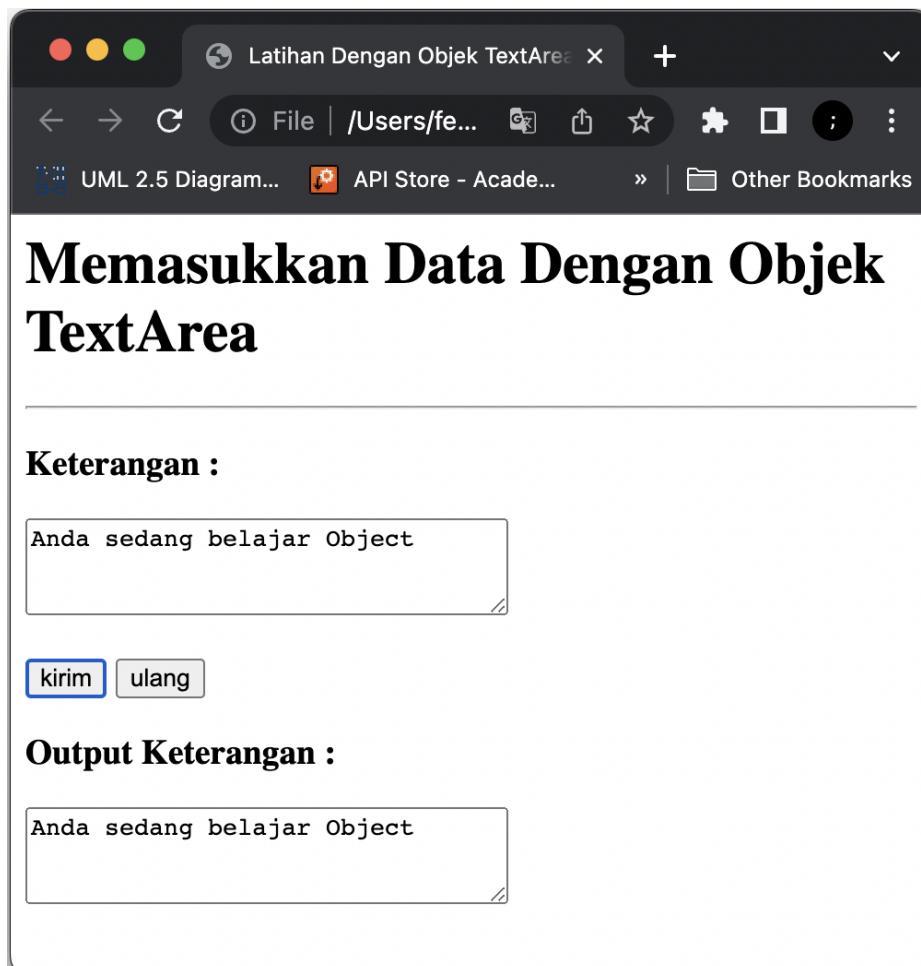


### - Objek TextArea

Objek textarea menyimpan informasi tentang elemen form yang berupa kotak teks dengan banyak baris.

```
<html>
<head><title>Latihan Dengan Objek TextArea</title></head>
<body>
<script languange ="JavaScript">
<!--
function tekan()
{
var ketstr = (document.fform.Ket.value); document.fform.Oket.value = ketstr;
}
//-->
</script>
<form name ="fform">
```

```
<H1> Memasukkan Data Dengan Objek TextArea</H1><hr>
<h3>Keterangan :<h3><br>
<textarea name="Ket" rows="3" cols="30"></textarea>
<BR><BR>
<input type="button" value="kirim" onclick="tekan()">
<input type="reset" value="ulang">
<H3>Output Keterangan :</H3>
<textarea name="Oket" rows="3" cols="30"></textarea>
</form>
</body>
</html>
```



## - Objek Select

Objek Select menyimpan informasi tentang elemen form yang berupa kotak daftar. Objek select berguna apabila di dalam form terdapat banyak pilihan yang telas mempunyai nilai tertentu.

```
<html>
<head><title>Latihan Dengan Objek Select</title></head>
<body>
<script languange ="JavaScript">
<!--
function tekan()
{
var jurusanstr = (document.form.Jurusan.value); document.form.Ojurusan.value = jurusanstr;
}
//-->
</script>
<form name ="fform">
<H1> Penggunaan Objek Select</H1><hr>
<h3>Masukan :<h3>
Jurusan Di IBIK :
<select name="Jurusan" Size="1">
<option value ="Teknologi Informasi"> S1 Teknologi Informasi </option>
<option value ="Pariwisata"> S1 Pariwisata</option>
<option value ="Teknik Industri"> Teknik Industri </option>
</select>
<p><input type="button" value="kirim" onclick="tekan()">
<input type="reset" value="ulang">
<H3>Output jurusan :</H3>
<input type="text" name="Ojurusan" size="30">
</form>
</body>
</html>
```

The screenshot shows a web browser window titled "Latihan Dengan Objek Select". The address bar shows the path "/Users/fe...". Below the address bar are various icons for file operations like copy, paste, and search. A toolbar below the address bar includes "UML 2.5 Diagram...", "API Store - Acade...", and "Other Bookmarks". The main content area features a large title "Penggunaan Objek Select" in bold black font. Below it is a section labeled "Masukan :" followed by a dropdown menu set to "S1 Teknologi Informasi". There are two buttons: "kirim" (highlighted in blue) and "ulang". Underneath is a section labeled "Output jurusan :" with a text input field containing "Teknologi Informasi".

### 3.3. Kontrol Program

#### 3.3.1. Percabangan

Untuk membuat suatu halaman yang dinamis dan interaktif, perancang halaman Web membutuhkan perintah-perintah yang dapat mengatur aliran dari informasi. Berdasarkan hasil komputasi yang telah dilakukan, JavaScript akan membuat keputusan jalur mana yang akan dieksekusi. Pada dasarnya dalam JavaScript terdapat dua macam pernyataan percabangan yaitu if..else dan switch

- **IF..ELSE**

Pernyataan ini digunakan untuk menguji sebuah kondisi dan kemudian mengeksekusi pernyataan tertentu bila kondisi tersebut terpenuhi, dan mengeksekusi pernyataan lain bila kondisi tersebut tidak terpenuhi.

```
if (kondisi) {  
    //pernyataan1 dieksekusi
```

```

    //bila kondisi terpenuhi
} else {
    //pernyataan2 dieksekusi
    //bila kodisi tidak terpenuhi
}

```

**kondisi** adalah ekspresi JavaScript yang mana hasil evaluasinya memiliki nilai Boolean **true** atau **false**

Untuk kasus yang melibatkan lebih banyak kondisi, maka kita dapat meletakkan pernyataan if lain setelah else

```

if (kondisi1) {
    //pernyataan1 dieksekusi
    //bila kondisi1 terpenuhi
} else if (kondisi2) {
    //pernyataan2 dieksekusi
    //bila kodisi1 tidak terpenuhi
} else {
    //pernyataan3 dieksekusi
    //bila kodisi2 tidak terpenuhi
}

```

Contoh program

```

<HTML>
<HEAD><TITLE>Percabangan IF-ELSE</TITLE></HEAD>
<BODY>
<SCRIPT language="JavaScript">
<!--
function tanyabilangan(){
    var bil = parseFloat(document.form.bilangan.value);
    var jenis = " ";
    if(isNaN(bil)) {
        alert("Anda Belum memasukkan Bilangan");
    } else {

```

```

if (bil > 0)  {
    jenis = " Adalah bilangan Positif";
}else if (bil < 0) {
    jenis = " Adalah bilangan Negatif";
}else{
    jenis = " Adalah Nol";
}
alert (bil+" "+jenis);
}

}

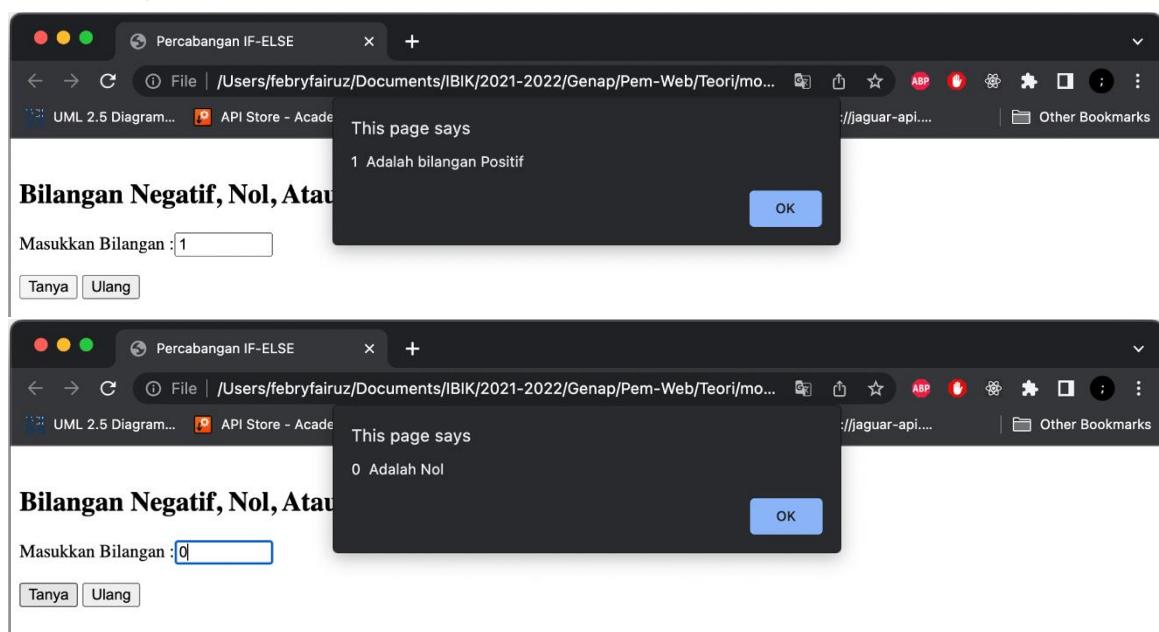
//--></SCRIPT>

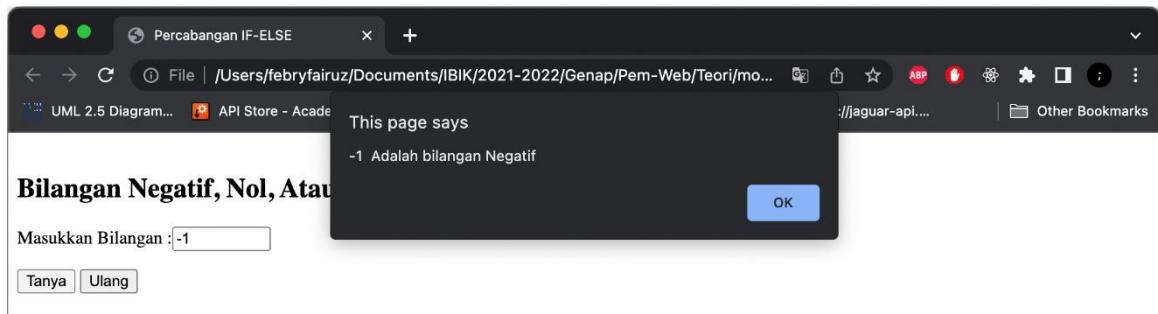
<FORM NAME ="fform">

<H2><BR>Bilangan Negatif, Nol, Atau Positif ???</H2>

Masukkan Bilangan :<input type="text" size="11" name="bilangan"> <P>
<INPUT TYPE="button" value="Tanya" onclick="tanyabilangan()">
<INPUT TYPE="reset" value="Ulang"> </p>
</FORM>
</BODY>
</HTML>

```





- **IF.. ELSE IF.. ELSE**

Percabangan majemuk adalah suatu percabangan yang dapat melibatkan lebih dari 1 kondisi di dalam percabangannya. Biasanya percabangan seperti ini menggunakan operator tambahan seperti AND, OR dan sebagainya.

<HTML>

```
<HEAD><TITLE>Percabangan IF-ELSE 3</TITLE></HEAD>
```

```
<BODY>
```

```
<SCRIPT language="JavaScript">
```

```
<!--
```

```
function hitungip(){
```

```
    var quis = parseFloat(document.fform.iquis.value);
```

```
    var tugas = parseFloat(document.fform.itugas.value);
```

```
    var uts = parseFloat(document.fform.iuts.value);
```

```
    var uas = parseFloat(document.fform.iuas.value);
```

```
    var ip = " "; var ket = " ";
```

```
    var na = (0.10 * quis) + (0.20 * tugas) + (0.30 * uts) + (0.40 * uas);
```

```
    if ((na > 80) && (na <= 100)) {
```

```
        ip = "A";
```

```
        ket = "Lulus dengan Sangat Baik";
```

```
    } else if ((na > 68) && (na <= 80)) {
```

```
        ip = "B";
```

```
        ket = "Lulus dengan Baik";
```

```
    } else if ((na > 55) && (na <= 68)) {
```

```
        ip = "C";
```

```
        ket = "Lulus dengan Cukup";
```

```

} else if ((na > 38) && (na <=55)) {
    ip ="D";
    ket="Lulus dengan Kurang";
} else {
    ip ="E";  ket="Tidak Lulus";
}
document.fform.oip.value=ip; document.fform.oket.value=ket;
// gunakan untuk mengecek alert (ip+"<"+na);
}

//-->

</SCRIPT>

<FORM NAME ="fform">

<table border="1" width="100%" ALIGN="center" > <tr>

<td width="100%" colspan="4"><H2 ALIGN="center">Menghitung Indeks
Prestasi</H2></td>

</tr>
<tr>

<td width="25%">Quis (10%) :<input type="text" size="10" name="iquis"> </td>
<td width="25%">Tugas (20%):<input type="text" size="10" name="itugas">
</td>

<td width="25%">UTS (30%):<input type="text" size="10" name="iuts"> </td>
<td width="25%">UAS (40%) :<input type="text" size="10" name="iuas"> </td>
</tr>
<tr>

<td width="100%" colspan="4"><P Align="center">
<INPUT TYPE="button" value="Hitung" onclick="hitungip()">
<INPUT TYPE="reset" value="Ulang"> </p></td>
</tr>
<tr>

<td width="100%" colspan="4" align="center">
Indeks Prestasi :<input type="text" size="5" name="oip">

```

```

Keterangan :<input type="text" size="25" name="oket"></td>
</tr>
</table>
</FORM>
</BODY>
</HTML>

```

Menghitung Indeks Prestasi

Quis (10%) :	Tugas (20%):	UTS (30%):	UAS (40%) :
70	80	75	88
<b>Hitung</b>		<b>Ulang</b>	
Indeks Prestasi : A	Keterangan : Lulus dengan Sangat Baik		

- **SWITCH**

Selain menggunakan if..else, percabangan juga dapat ditangani dengan perintah switch. Dengan kata lain pernyataan switch digunakan untuk menyederhanakan pernyataan if..else yang terlalu banyak.

```

<HTML>
<HEAD><TITLE>Percabangan Switch</TITLE></HEAD>
<BODY>
<SCRIPT language="JavaScript">
<!--
function tanyabulan(){
    var bulan = parseFloat(document.fform.ibulan.value);
    var namabulan= " ";
    switch (bulan) {
        case 1 : namabulan="Bulan ke 1 adalah = Januari";break;
        case 2 : namabulan="Bulan ke 2 adalah = Februari";break;
        case 3 : namabulan="Bulan ke 3 adalah = Maret";break;
    }
}
-->

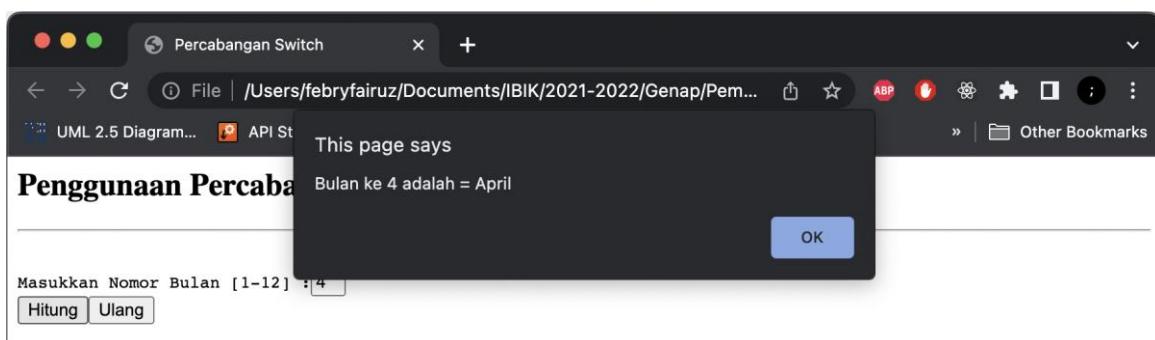
```

```

        case 4 : namabulan="Bulan ke 4 adalah = April";break;
        case 5 : namabulan="Bulan ke 5 adalah = Mei";break;
        case 6 : namabulan="Bulan ke 6 adalah = Juni";break;
        case 7 : namabulan="Bulan ke 7 adalah = Juli";break;
        case 8 : namabulan="Bulan ke 8 adalah = Agustus";break;
        case 9 : namabulan="Bulan ke 9 adalah = September";break;
        case 10 : namabulan="Bulan ke 10 adalah = Oktober";break;
        case 11 : namabulan="Bulan ke 11 adalah = November";break;
        case 12 : namabulan="Bulan ke 12 adalah = Desember";break;
    default : namabulan="Anda salah mengisi";
}
alert(namabulan);
}
//--></SCRIPT>

<FORM NAME ="fform">
<H2>Penggunaan Percabangan Switch</H2><HR>
<PRE>
Masukkan Nomor Bulan [1-12] :<input type="text" size="2" name="ibulan">
<INPUT TYPE="button" value="Hitung" onclick="tanyabulan()"><INPUT
TYPE="reset" value="Ulang">
</PRE>
</FORM>
</BODY>
</HTML>

```



### 3.3.2. Perulangan

Untuk mengulang kejadian beberapa kali maka kita membutuhkan proses perulangan. Pada JavaScript dikenal beberapa metode/cara perulangan.

- **Perulangan For**

Digunakan untuk mengeksekusi pernyataan-pernyataan beberapa kali. Perulangan For paling sering dipakai, jika anda sudah tahu akhir dari perulangan tersebut. . Perintah for

mengulang suatu loop sampai kondisi menghasilkan evaluasi true atau loop keluar dengan perintah break .

Contoh :

```
for (nilai awal;kondisi;penambahan) {  
    ulang pernyataan ini;  
}
```

Contoh dalam program :

```
for(x=1;x<=10;x++) {  
    document.writeln("Belajar JavaScript Yuuuuu..");  
}
```

### Contoh Program

```
<HTML>  
<HEAD><TITLE>Latihan Perulangan I</TITLE></HEAD>  
<SCRIPT LANGUAGE="JavaScript">  
document.writeln("<H2>Latihan Perulangan For</H2>");  
document.writeln("-----");  
for (i=1; i<=10; i++) {  
    sq=i*i;  
    document.writeln("<PRE>");  
    document.write("Angka:" + i + " ----> Kuadrat: " + sq + "<BR>");  
    document.writeln("</PRE>");  
}  
document.writeln("-----");  
</SCRIPT>  
</HEAD>  
<BODY>  
</BODY>  
</HTML>
```

```
-----  
Angka:1 -----> Kuadrat: 1  
Angka:2 -----> Kuadrat: 4  
Angka:3 -----> Kuadrat: 9  
Angka:4 -----> Kuadrat: 16  
Angka:5 -----> Kuadrat: 25  
Angka:6 -----> Kuadrat: 36  
Angka:7 -----> Kuadrat: 49  
Angka:8 -----> Kuadrat: 64  
Angka:9 -----> Kuadrat: 81  
Angka:10 -----> Kuadrat: 100  
-----
```

- **Perulangan While**

Perulangan lain yang dapat digunakan adalah dengan menggunakan perintah While. Perintah while digunakan untuk perulangan yang tidak diketahui berapa kali proses perulangannya. Perintah while terus mengulangi loop selama kondisi memiliki nilai true. Syntax untuk perintah while adalah sebagai berikut :

```
while (kondisi){ ulang pernyataan ini; }
```

#### Contoh Program

```
<HTML>  
<HEAD><TITLE>Latihan Perulangan II</TITLE></HEAD>  
<SCRIPT LANGUAGE="JavaScript">  
  
var deret = prompt('Masukkan Jumlah Deret : ' );  
  
document.writeln("<H2>Latihan Perulangan While</H2>");  
  
document.writeln("-----");  
  
document.writeln("<BR>");  
  
var jml = 0.0; var angka = 1;  
  
while (angka <= deret) {  
  
    jml= jml+angka;  
  
    angka++
```

```
}

document.writeln("<BR>");

document.writeln("Jumlah Deret dari 1 sampai "+deret+" adalah = "+jml);

document.writeln("<BR>");

document.writeln("-----");

</SCRIPT>

</HEAD>

<BODY>

</BODY>

</HTML>
```

The screenshot shows a web browser window titled "Latihan Perulangan II". A modal dialog box is displayed in the foreground, asking for input. The dialog title is "This page says" and the question is "Masukkan Jumlah Deret :". Inside the input field, the number "4" is typed. Below the input field are two buttons: "Cancel" and "OK". In the background, the main page content is visible, featuring a large heading "Latihan Perulangan While" and a text block stating "Jumlah Deret dari 1 sampai 4 adalah = 10".

This page says

Masukkan Jumlah Deret :

4

Cancel OK

Latihan Perulangan While

Jumlah Deret dari 1 sampai 4 adalah = 10

- **Perulangan Do While**

Perulangan ini hampir sama seperti while, digunakan apabila kita belum tahu berapa kali perulangan harus dilakukan. Bedanya pernyataan do..while pengujiannya dilakukan di akhir pernyataan.

```
do{
    //pernyataan1 dieksekusi
}while (kondisi);
```

Contoh Program

```
<HTML>

<HEAD><TITLE>Latihan Perulangan III(Do-While)</TITLE></HEAD>

<BODY>

<SCRIPT LANGUAGE="JavaScript">

function pass(){

    var coba = 1;

    do {
        p = prompt("Tuliskan password dengan benar","");
        if (p=="IBIKJAYA"){

            alert("Selamat Datang Friends");
            window.open("Latiahn-1.html");
            break;
        } else {
            alert("Password Salah !!! Ulangi lagi.");
        } if (coba==3)  {

            alert("maaf, kesempatan anda hanya 3 kali"); history.go(-1);
        } coba=coba+1;
    } while (coba<=3);

}

</SCRIPT>

<FORM METHOD="post">

<INPUT TYPE="button" VALUE="Password" ONCLICK="pass()">

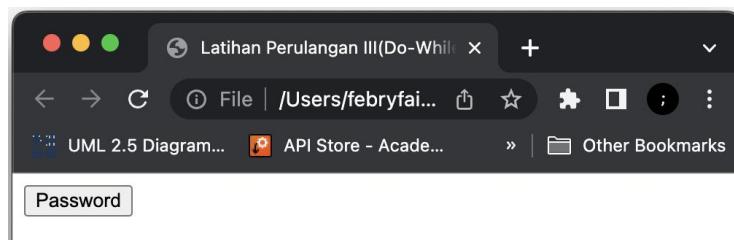
</FORM>

</BODY>
```

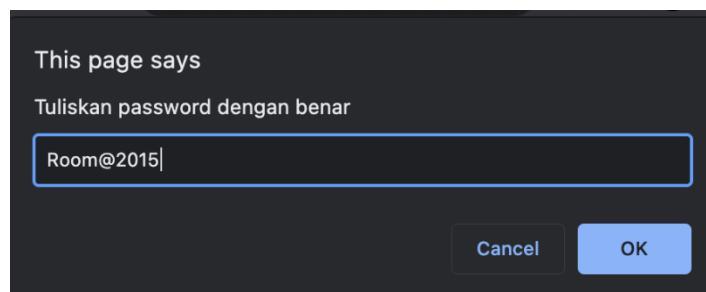
</HTML>

Adapun hasil yang diperoleh adalah sebagai berikut :

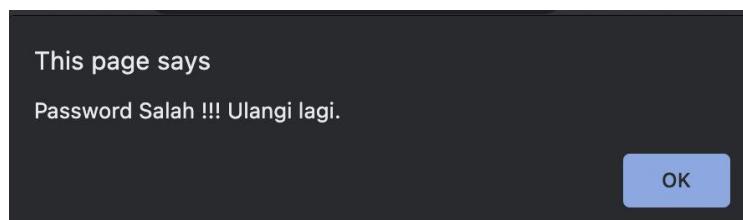
1. Tampilan awal



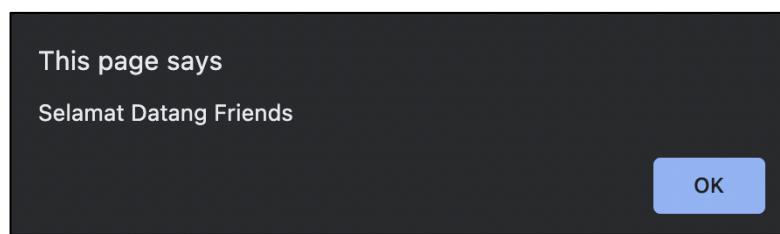
2. Prompt password



3. Alert apabila terjadi kesalahan password



4. Apabila password benar



### 3.4. Tugas dan Latihan

1. Buatlah halaman seperti berikut, yang mencakup semua materi yang terdapat pada modul ini.
  - Input berupa : NIM, NAMA, JENIS KELAMIN, AGAMA, STATUS, JURUSAN, KOMENTAR. (SESUAIKAN OBJEK YANG DIPAKAI)
  - Proses terjadi ditombol KIRIM
  - Output berupa isian yang sama dengan soal nomor 1.
  - Tugas individu, dikumpulkan minggu ke 5, dalam bentuk print out KODE HTMLnya serta Tampilannya
2. Buatlah sebuah halaman form inputan, yang mencakup semua materi yang terdapat pada modul ini.

Perusahaan Travel IBIK Abadi memiliki armada dengan tujuan Jakarta, Solo dan Surabaya. Setiap tujuan memiliki kelas Eksekutif, Bisnis dan Ekonomi.

	<b>Eksekutif</b>	<b>Bisnis</b>	<b>Ekonomi</b>
<b>Jakarta</b>	70000	40000	10000
<b>Solo</b>	80000	50000	20000
<b>Surabaya</b>	90000	60000	30000

Diskon 10% diberikan apabila pemesan tiket merupakan anggota Travel Bintang Abadi.

**Input** : Nama Pemesan, Tujuan, Kelas, Banyak Tiket dan Status Member/Bukan

**Output** : Harga Tiket, Subtotal, Diskon dan Total Bayar

3. Buat program untuk menentukan faktorial dan jumlah deret sampai ke N.

<b>PERHITUNGAN</b>		
<b>FAKTORIAL dan DERET</b>		
<b>N</b>	<input type="text"/>	
<b>Hasil</b>		
<b>Faktorial</b>	<input type="text"/>	<b>Submit</b> <b>Reset</b>
<b>Jumlah</b>	<input type="text"/>	

4. Buat program untuk menghitung saldo akhir dari suatu tabungan dengan bunga dan jangka waktu tertentu :

Input :

Saldo : 100000

Bunga : 10

Waktu : 3

Ketika user menekan tombol Hitung maka akan muncul hasil seperti berikut :

Saldo Bulan 1 = Rp. 110000

Saldo Bulan 2 = Rp. 121000

Saldo Bulan 3 = Rp. 133100

## MODUL 4

### JAVASCRIPT 2

#### 4.1. Tujuan Praktikum

Mahasiswa memahami bentuk-bentuk dari berbagai macam object yang disediakan oleh Javascript untuk membantu membuat tampilan web lebih menarik dengan berbagai animasi.

#### 4.2. Dasar Teori

##### 4.2.1. Object Array

Array adalah suatu variable yang dapat memuat beberapa nilai secara berurutan atau seri. Artinya variable yang dideklarasikan sebagai array isinya tidak satu. Berikut adalah pendeklarasian untuk array.

```
nama = new Array(3)
```

Pernyataan diatas menunjukan bahwa variabel nama memiliki 3 elemen. Ketiga elemen tersebut akan memiliki nilai masing-masing 0, 1, 2. Nilai pertama = 0. Untuk mengisikan ketiga elemen tersebut kita dapat melakukan dengan cara :

```
nama[0] = "Febri", nama[1] = "Damatraseta", nama[2] = "Fairuz"
```

Selain cara diatas kita juga bisa mendeklarasikan array sekaligus dengan mengisikan elemen-elemennya.

```
nama = new Array("Febri", "Damatraseta", "Fairuz")
```

#### Contoh Program

<pre>&lt;HTML&gt; &lt;HEAD&gt; &lt;TITLE&gt; LATihan Objek Array&lt;/TITLE&gt; &lt;BODY&gt; &lt;h3&gt;Latihan Objek Array I&lt;/H3&gt; Nama pada data ke 3 adalah : &lt;SCRIPT LANGUAGE="JavaScript"&gt; function cobaarray(){     nama = new Array("Febri",     "Damatraseta", "Fairuz");     document.write(nama[2]); } &lt;/SCRIPT&gt; &lt;SCRIPT LANGUAGE="JavaScript"&gt; cbaarray(); &lt;/SCRIPT&gt; &lt;/BODY&gt; &lt;/HTML&gt;</pre>	<pre>&lt;HTML&gt; &lt;HEAD&gt; &lt;TITLE&gt; LATihan Objek Array&lt;/TITLE&gt; &lt;BODY&gt; &lt;h3&gt;Latihan Objek Array II&lt;/H3&gt; Pemanggilan data Array dengan Perintah Perulangan For&lt;BR&gt; &lt;SCRIPT LANGUAGE="JavaScript"&gt; var nilai = new Array(3); nilai[0]="A"; nilai[1]="B"; nilai[2]="C"; for (a=0;a&lt;3;++a) {     document.writeln("&lt;B&gt;Nilai ke "+ [a+1] +     " : "+nilai[a]+"&lt;BR&gt;"); } &lt;/SCRIPT&gt; &lt;SCRIPT LANGUAGE="JavaScript"&gt; &lt;/SCRIPT&gt; &lt;/BODY&gt; &lt;/HTML&gt;</pre>
--	---

#### 4.2.2. Object Date

Objek ini digunakan untuk memanipulasi tanggal dan waktu pada JavaScript. Untuk pendeklarasiannya adalah sebagai berikut :

```
tanggal = new Date()
```

Pernyataan diatas menyatakan bahwa variabel tanggal mengandung unsur tanggal dan waktu.

Metode-metode untuk Objek Date

Metode	Kegunaan
getDate()	Menghasilkan tanggal (integer) mulai 1 – 31.
getDay()	Menghasilkan hari(integer) mulai 0-6. Minggu = 0, Senin = 1,.....
getMonth()	Menghasilkan bulan(integer) mulai 0-11. Januari=0, Feb=1,.....
getFullYear()	Menampilkan tahun menjadi 4 digit
getHours()	Menghasilkan jam mulai 0-23
getMinutes()	Menghasilkan menit mulai 0-59
getSeconds()	Menghasilkan detik mulai 0-59

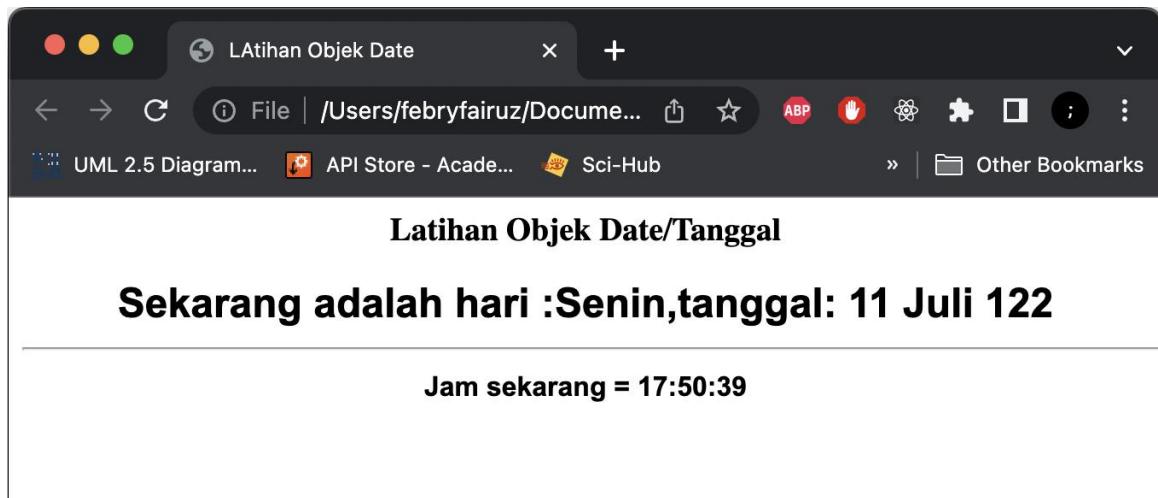
#### Contoh Program

```
<HTML>
<HEAD>
<TITLE> LATihan Objek Date</TITLE>
<BODY>
<h3 align="center">Latihan Objek Date/Tanggal</H3>
<SCRIPT LANGUAGE="JavaScript">
var hari = new Array("Senin","Selasa","Rabu","Kamis","Jumat","Sabtu","Minggu");
var bulan = new Array("Januari","Februari","Maret","April",
                     "Mei","Juni","Juli","Agustus",
                     "September","Oktober","November","Desember");
var t = new Date();
var hari_ini=hari[t.getDay()-1];
var tanggal=t.getDate();
var bulan_ini=bulan[t.getMonth()];
var tahun=t.getYear();
var jam =t.getHours();
var menit =t.getMinutes();
var detik =t.getSeconds();
document.write("<font size=5 face=arial>");
document.write("<b><center>Sekarang adalah hari :" +hari_ini+",tanggal: "+ tanggal +" "+ bulan_ini +" "+tahun);
document.write("<hr width=700>"); document.write("</font>");
```

```

document.write("<font size=3 face=arial>");
document.write("<b><center>Jam sekarang = "+ jam + ":"+menit+ ":"+detik);
document.write("</font>");
</SCRIPT>
<SCRIPT LANGUAGE="JavaScript">
</SCRIPT>
</BODY>
</HTML>

```



#### 4.2.3. Object Math

Math digunakan untuk menangani perhitungan matematis yang rumit. Bentuk penulisan:

*Math.metode(nilai)*

Metode Untuk Objek Math

Metode	Keterangan
abs(a)	Nilai absolut dari a
acos(a)	Nilai arc-kosinus dari a
asin(a)	Nilai arc-sinus dari a
atan(a)	Nilai arc-tan dari a
ceil(a)	Membulatkan nilai ke integer diatasnya
cos(a)	Nilai kosinus dari a
exp(a)	Nilai E pangkat a
log(a)	Nilai logaritma dari a
max(a,d)	Nilai terbesar dari a dan d
min(a,d)	Nilai terkecil dari a dan d
pow(a,d)	Nilai dari a pangkat d
random(a)	Nilai acak antara 0 dan 1
round(a)	Membulatkan nilai a ke integer terdekat

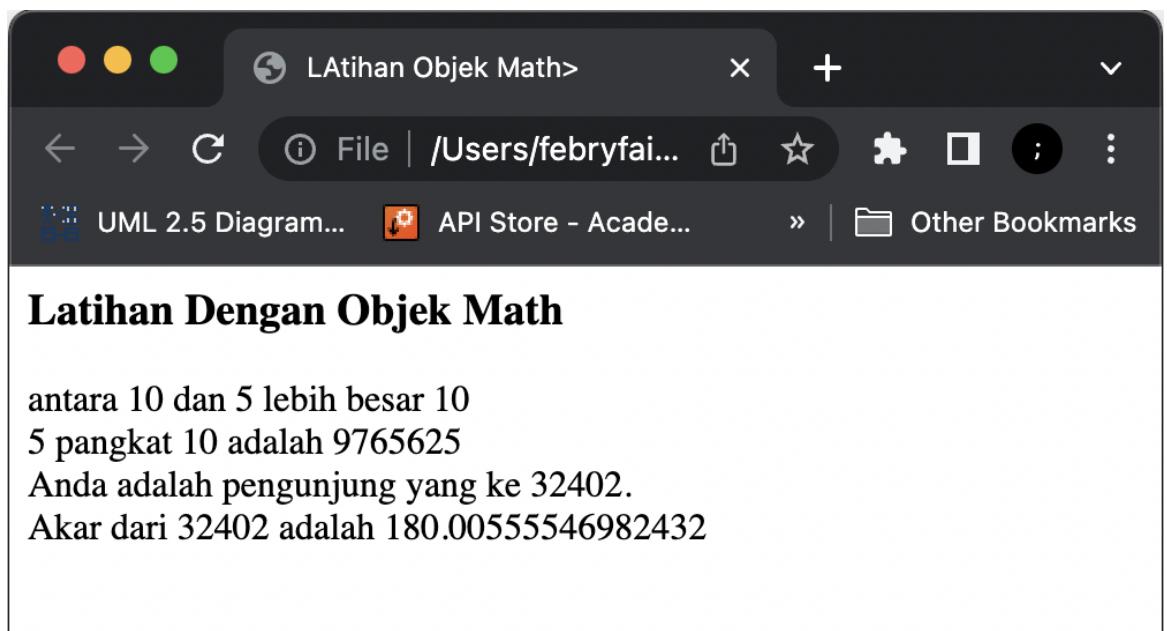
sqrt(a)	Nilai akar dari kuadrat a
sin(a)	Nilai sinus dari a
tan(a)	Nilai tangen dari a

## Contoh Program

```

<HTML>
<HEAD>
<TITLE> LATihan Objek Math</TITLE>
<BODY>
<h3>Latihan Dengan Objek Math</h3>
<SCRIPT LANGUAGE="JavaScript">
var a=10; var b=5; besar=Math.max(a,b);
document.write("antara " + a + " dan " + b + " lebih besar "+besar+"<br>");
pangkat=Math.pow(b,a);
document.write(b+ " pangkat " + a + " adalah "+pangkat+"<br>");
var ran;
ran = Math.round(Math.random()*50000);
document.write("Anda adalah pengunjung yang ke " + ran + ".<br>");
var akar; akar =Math.sqrt(ran);
document.write ("Akar dari " + ran + " adalah "+akar);
</SCRIPT>
</BODY>
</HTML>

```



#### 4.2.4. Object String

String adalah suatu objek yang merupakan kumpulan dari elemen karakter-karakter.

Dalam Javascript string atau karakter harus diapit dengan tanda petik ganda(“) atau tanda petik tunggal(').

Contoh pendeklarasian Objek String :

```
Nama = "Shafana Vevica"
```

```
Panjang = Nama.length; // Panjang akan berisi 14
```

Length adalah property yang sering digunakan dalam objek string yang digunakan

Untuk mengetahui banyaknya karakter dalam suatu string.

Objek String juga memiliki method yang dapat digunakan untuk memanipulasi string tersebut. Adapun Method yang dapat digunakan meliputi :

Method	Fungsi
big()	Tercetak lebih besar
blink()	Efek berkedip aktif pada browserNetscape
bold()	Tercetak tebal
charAt(n)	Mengambil karakter ke -n dari string. Index string dimulai dari 0
fixed()	Tercetak fixed-pitch
fontcolor('warna')	Tercetak sesuai warna yang didefinisikan
indexOf('char')	Mengambil nilai indeks dari suatu karakter
italic()	Tercetak miring
link('url')	Menjadikan string hyperlink
small()	Tercetak lebih kecil
strike()	Tercetak dengan coretan
sub()	Tercetak subscript
substring(a,b)	Mengambil karakter dari posisi a sampai b-1
sup()	Tercetak superscript
toLowerCase()	Tercetak huruf kecil
toUpperCase()	Tercetak huruf besar
split('')	Menjadikan string diuraikan/dipisahkan berdasarkan tanda (''). Hasil dari split akan dihasilkan sebuah array dengan indeks 0 untuk string ke 1 dan seterusnya.

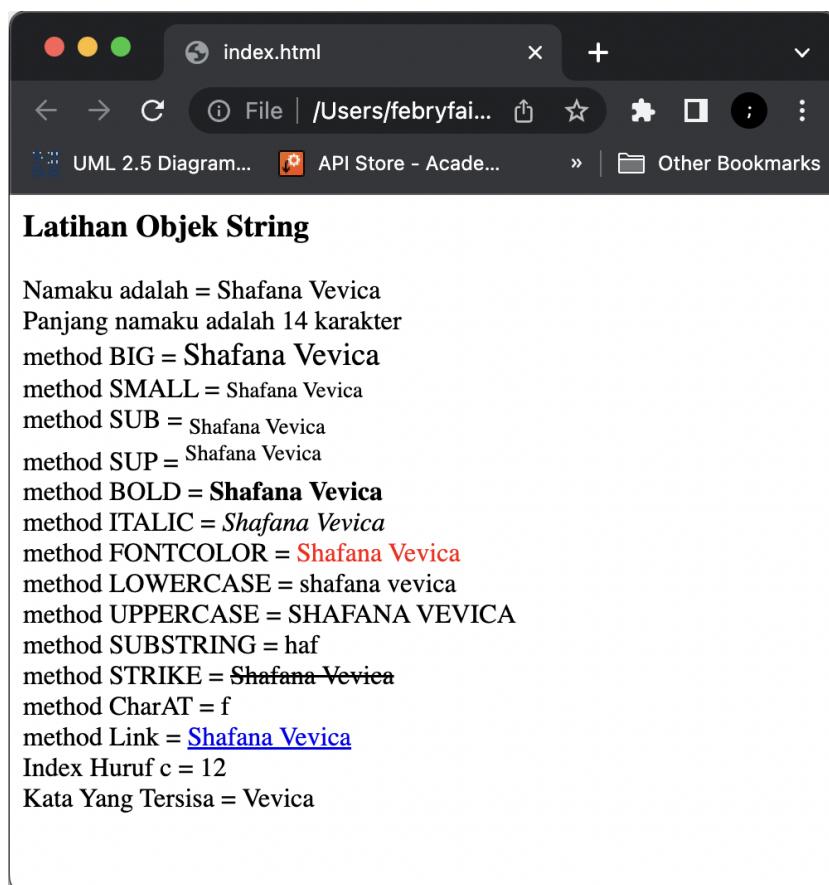
#### Contoh Program

```
<HTML>
<BODY>
<H3>Latihan Objek String</H3>
<SCRIPT LANGUAGE="Javascript"> nama ="Shafana Vevica";
panjang=nama.length; n=nama.substring(1,4);
```

```

besar=nama.toUpperCase();
namakulink=nama.link('shafa.html');
document.writeln('Namaku adalah = '+ nama +'  
');
document.writeln('Panjang namaku adalah '+ panjang + ' karakter <BR>');
document.writeln('method BIG = '+ nama.big() +'<BR>');
document.writeln('method SMALL = '+ nama.small() +'<BR>');
document.writeln('method SUB = '+ nama.sub() +'<BR>');
document.writeln('method SUP = '+ nama.sup() +'<BR>');
document.writeln('method BOLD = '+ nama.bold() +'<BR>');
document.writeln('method ITALIC = '+ nama.italics() +'<BR>');
document.writeln('method FONTCOLOR = '+ nama.fontcolor('red') +'<BR>');
document.writeln('method LOWERCASE = '+nama.toLowerCase() +'<BR>');
document.writeln('method UPPERCASE = '+ besar +'<BR>');
document.writeln('method SUBSTRING = '+ n +'<BR>');
document.writeln('method STRIKE = '+ nama.strike() +'<BR>');
document.writeln('method CharAT = '+ nama.charAt(3) +'<BR>');
document.writeln('method Link = '+ namakulink +'<BR>');
document.writeln('Index Huruf c = '+ nama.indexOf("c") +'<BR>');
awal=nama.indexOf('V'); akhir=nama.length;
document.writeln('Kata Yang Tersisa = '+ nama.substring(awal,akhir) +'<BR>');
</SCRIPT>
</BODY>
</HTML>

```



#### 4.2.5. Object Document

Objek ini digunakan untuk mengakses informasi mengenai dokumen HTML, tampilan output dan memanipulasinya.

Property dari objek document meliputi :

Property	Fungsi
bgColor	Memberikan warna latar belakang
fgColor	Memberikan warna foreground atau warna huruf
link[]	Mengakses objek anchor/link(dapat digunakan nama objek anchor/link)
linkColor=warna	Memberikan warna link
alinkColor=warna	Memberikan warna pada active link
vlinkColor=warna	Memberikan warna pada visited link
title=judul window	Memberikan judul/title window
image[]	Mengakses objek image(dapat digunakan nama objek anchor/link)
forms[]	Mengakses objek form(dapat digunakan nama objek form)

Method dari objek document meliputi :

Method	Fungsi
open()	Menciptakan/membuka document HTML
close()	Mengakhiri document HTML
write(output)	Memberikan output ke browser
writeln(output)	Memberikan output ke browser dengan menyertakan perpindahan baris

Khusus untuk output ke browser ada beberapa hal yang perlu diperhatikan :

- Diisi dengan string(“) atau (‘)
- Dapat diberikan tag HTML
- Dapat digunakan untuk menampilkan isi dari variable
- Terdapat karakter spesial :

\b = untuk backspace

\f = untuk form feed \n

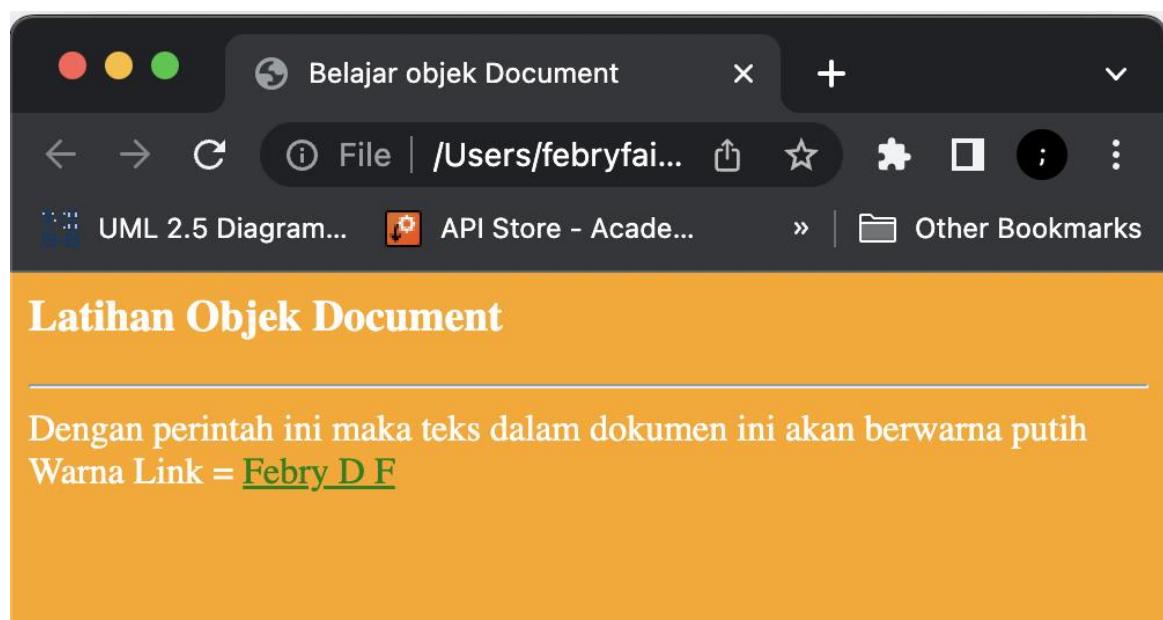
= untuk baris baru

\r = untuk carriage return

\t = untuk tab

## Contoh Program

```
<HTML>
<BODY>
<H3>Latihan Objek Document</H3><hr>
Dengan perintah ini maka teks dalam dokumen ini akan berwarna putih<BR>
<SCRIPT LANGUAGE="Javascript">
nama="Febry D F";
document.bgColor="orange";
document.fgColor="white";
document.title="Belajar objek Document";
document.linkColor="red";
document.vlinkColor="green";
document.alinkColor="white";
namakulink=nama.link('https://www.linkedin.com/in/febri-d-2b9537b1/');
document.writeln('Warna Link = '+ namakulink +'  
');
</SCRIPT>
</BODY>
</HTML>
```



### 4.2.5. Object Window

Objek window merupakan objek tertinggi dalam objek Javascript. Objek ini digunakan untuk memanipulasi tampilan jendela dari document HTML.

### Property pada Objek window

<b>Property</b>	<b>Fungsi</b>
length	Mengetahui jumlah frame pada window
location.href	Mengakses objek location untuk melakukan redirect atau berpindah ke alamat tertentu.
Status=nilai_status	Memberikan nilai status window

### Metode-metode untuk Objek window

<b>Method</b>	<b>Fungsi</b>
alert(pesan)	Memunculkan messagebox sebuah pesan kesalahan
confirm(pesan)	Memunculkan pesan konfirmasi. Method ini akan menghasilkan dua nilai kembalian yaitu true untuk Ok dan false untuk Cancel
prompt(pesan,nilai default)	Memunculkan pesan yang menunggu sebuah input
close()	Menutup jendela aktif
open(url file,windowname ,feature)	Membuka jendela baru dengan feature meliputi : toolbar=yes no mengaktifkan toolbar status=yes no mengaktifkan window status menubar=yes no mengaktifkan menubar scrollbars=yes no mengaktifkan scrollbar resizable=yes no jendela resizable
	width = ukuran lebar jendela height = ukuran tinggi jendela
print()	Membuka jendela dialog print

### **Contoh Program**

```

<HTML>
<BODY>
<H3>Latihan Objek Window</H3><hr>
<SCRIPT LANGUAGE="Javascript">
window.status="Welcome"; window.alert("Selamat
Datang"); angka=window.prompt("Inputkan Angka ?",0);
document.write("Angka vaforit anda adalah =<strong>" +angka+
"</strong><br>");
tampung=window.confirm("jenis kelamin anda Pria ?"); if(tampung) {
document.write("Boleh Kenalan donk");
} else {
document.write("Ok dech");
}
window.close();
</SCRIPT>
</BODY>

```

```
</HTML>
```

Contoh penggunaan perintah window.open dan window.location untuk membuka halaman web lain.

### **Contoh Program**

```
<HTML>
<BODY>
<CENTER><H3>Latihan Objek Document</H3><hr>
Membuka Web Page dengan Perintah Window.Open dan Window.Location
</CENTER>
<SCRIPT LANGUAGE="Javascript">
function koneksi1()
{
    window.open("utsb.HTML");
}
function koneksi2()
{
    window.location="kunci_jawaban UTS.HTML";
}
</SCRIPT>
<FORM METHOD="post">
<P><CENTER>
<INPUT TYPE="button" VALUE="Kunci Jawaban UTS A"
ONCLICK="koneksi1()">
<INPUT TYPE="button" VALUE="Kunci Jawaban UTS B"
ONCLICK="koneksi2()">
</FORM></CENTER>
</BODY>
</HTML>
```

Contoh penggunaan objek window.location.href untuk membuka halaman web yang lain.

### **Contoh Program**

```
<HTML>
<BODY>
<CENTER><H3>Latihan Objek Document</H3><hr>
Membuka Web Page dengan Perintah Window.Location.href
</CENTER>
<SCRIPT LANGUAGE="Javascript">
function koneksi1()
{
    if(document.pilihan.pilih.options[0].selected)
    {
        window.location.href="latobjekradio.HTML";
    }
}
```

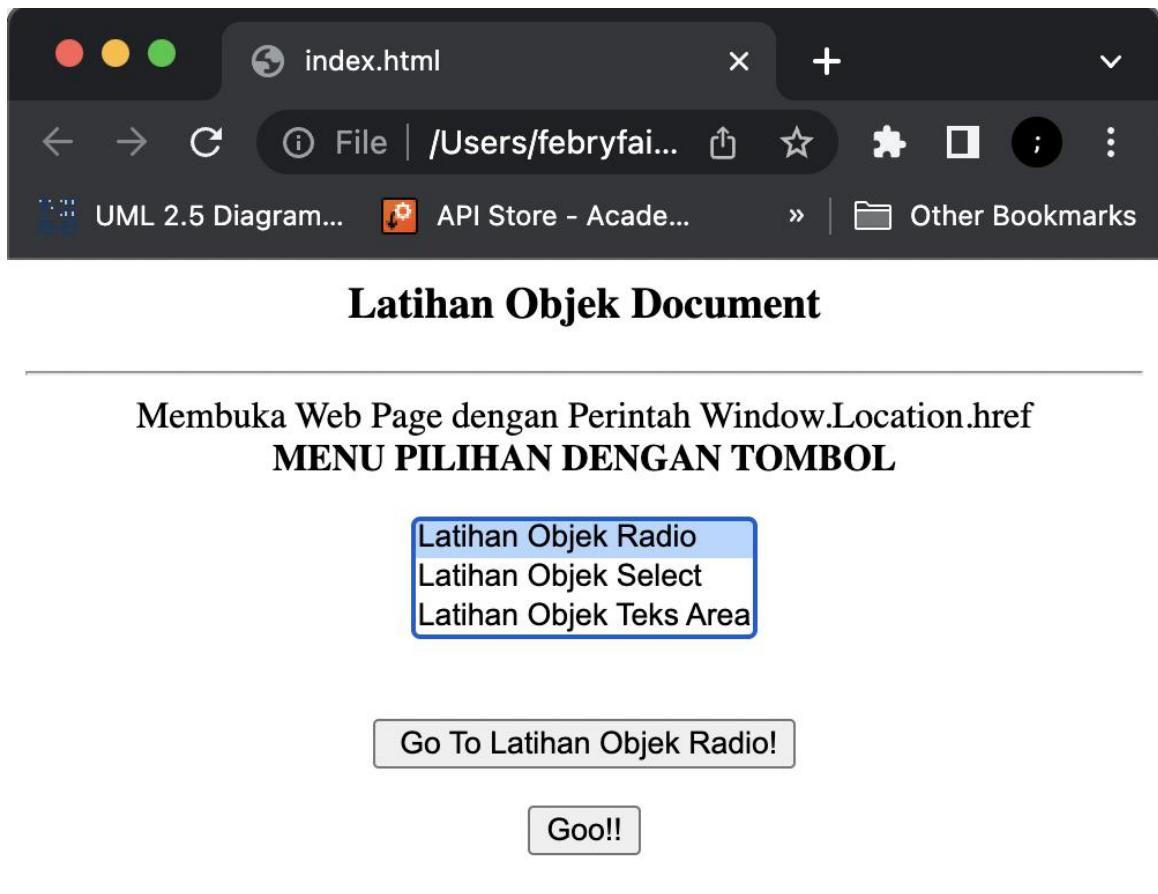
```

}
else if (document.pilihan.pilih.options[1].selected)
{
    window.location.href="latobjekselect.HTML";
}
else if (document.pilihan.pilih.options[2].selected)
{
    window.location.href="latobjekteksarea.HTML";
} return true;
}

function koneksi2()
{ var pilihint; var pilihstr;
pilihint=document.pilihan.pilih.selectedIndex;
pilihstr=document.pilihan.pilih.options[pilihint].text;
document.pilihan.pilihteks.value=" Go To " + pilihstr + "!" ;
}

</SCRIPT>
<CENTER>
<FORM NAME="pilihan">
<B>MENU PILIHAN DENGAN TOMBOL</B>
<P><SELECT NAME="pilih" ONCHANGE="koneksi2()" MULTIPLE SIZE="3">
    <OPTION>Latihan Objek Radio</OPTION>
    <OPTION>Latihan Objek Select</OPTION>
    <OPTION>Latihan Objek Teks Area</OPTION>
</SELECT>
</P>
<P><BR>
<INPUT TYPE="button" name="pilihteks" value="" size="40" maxlength="40">
</P>
<P>
<INPUT TYPE="button" NAME="Gobutton" VALUE="Goo!!"
ONCLICK="koneksi1()">
</P>
</FORM></CENTER>
</BODY>
</HTML>

```



### 4.3. Event Handler

Even adalah sesuatu yang terjadi pada halaman HTML. Berikut ini terdapat beberapa bentuk kejadian yaitu jika pengguna memuat dokumen, pengguna memasukkan data, pengguna mengklik tombol dan sebagainya. Hal-hal tersebut diatur oleh even. Semua kejadian pada Javascript dapat anda tangani dengan menentukan kejadiannya. Biasanya kejadian(even) adalah sebuah fungsi, tetapi pada beberapa kasus, kita dapat menuliskan pernyataan-pernyataannya secara langsung. Berikut ini adalah daftar kejadian(even) pada JavaScript :

Event	Keterangan
onClick	Kejadian yang dibangkitkan bila pengguna mengklik sebuah elemen form atau link.
onChange	Dibangkitkan bila informasi masukan pada sebuah elemen form (text, textarea, select) diubah oleh pengguna.
onBlur	Dibangkitkan ketika suatu elemen kehilangan focus masukan, yaitu ketika pengguna menekan tombol <tab> atau mengklik elemen lain form lainnya.
onFocus	Dibangkitkan bila sebuah elemen form menerima focus masukan; yaitu bila pengguna mengklik

	elemen form tersebut atau menekan tombol <tab> sehingga focus masukan berpindah ke elemen ini.
onAbort	Dibangkitkan bila pengguna menghentikan pemuatan citra (tag<img>) yaitu bila pengguna menekan tombol stop atau mengklik link.
onError	Dibangkitkan bila terjadi kesalahan saat browser memuat dokumen atau citra.
onLoad	Dibangkitkan bila browser selesai memuat document
onUnload	Dibangkitkan bila pengguna keluar dari dokumen
onMouseOver	Dibangkitkan bila kursor mouse berada di atas sebuah link.
onMouseOut	Dibangkitkan bila kursor mouse keluar dari daerah link atau peta citra.
onReset	Dibangkitkan bila pengguna menekan tombol reset
onSelect	Kejadian yang dibangkitkan bila pengguna memilih sebagian atau seluruh teks pada elemen form yang berupa kotak teks.
onSubmit	Dibangkitkan ketika pengguna menekan tombol submit.

#### 4.3.1. Penanganan Kejadian (Event)

Berikut ini akan diberikan beberapa contoh program-program yang menggunakan kejadian-kejadian dalam aplikasinya.

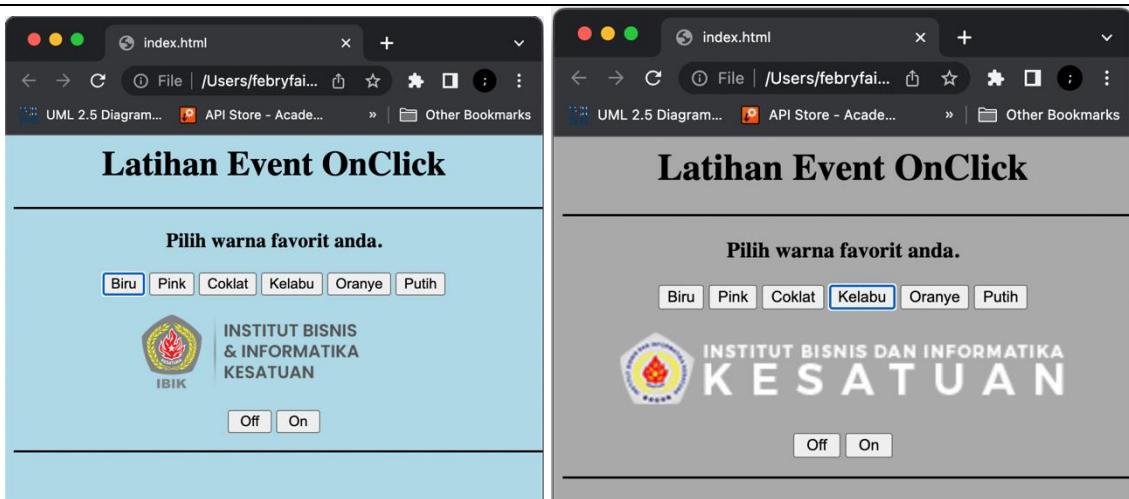
Contoh program yang menggunakan even **OnClick** :

```
<HTML>
<BODY>
<SCRIPT LANGUAGE="JavaScript">
function warna(pilihan){
    alert("Anda Memilih Warna " + pilihan); document.bgColor=pilihan;
}
</SCRIPT>
<h1 align="center">Latihan Event OnClick</h1>
<hr width="500" color="black" noshade>
<h3 align="center">Pilih warna favorit anda.</h3>
<CENTER>
<FORM>
<INPUT TYPE="button" VALUE="Biru" onClick="warna('lightblue')">
<INPUT TYPE="button" VALUE="Pink" onClick="warna('pink')">
<INPUT TYPE="button" VALUE="Coklat" onClick="warna('burlywood')">
<INPUT TYPE="button" VALUE="Kelabu" onClick="warna('darkgray')">
<INPUT TYPE="button" VALUE="Oranye" onClick="warna('peachpuff')">
```

```

<INPUT TYPE="button" VALUE="Putih" onClick="warna('white')">
</FORM>
<FORM>
    <IMG NAME="coolfan" SRC="https://www.ibik.ac.id/wp-content/uploads/2020/08/logo-ibik-
web.png" height=72><BR><BR>
    <INPUT TYPE=BUTTON VALUE=" Off " onClick="coolfan.src =
'https://www.ibik.ac.id/wp-content/uploads/2019/10/logo-ibik-web.png'">
    <INPUT TYPE=BUTTON VALUE=" On " onClick="coolfan.src =
'https://www.ibik.ac.id/wp-content/uploads/2020/08/logo-ibik-web.png'">
</FORM>
</CENTER>
<hr width="500" color="black" noshade>
</BODY>
</HTML>

```



Contoh program yang menggunakan even **OnBlur** dan **onFocus** :

```

<HTML>
<BODY>
<SCRIPT LANGUAGE="JavaScript">
function masukannim() {
if (document.f.inim.value=="")
{
alert("anda belum memasukkan nim");
}}
function masukannama()
{
if (document.f.inama.value=="")
{
alert("anda belum memasukkan nama");
}}
function masukanalamat()
{
if (document.f.ialamat.value=="")

```

```

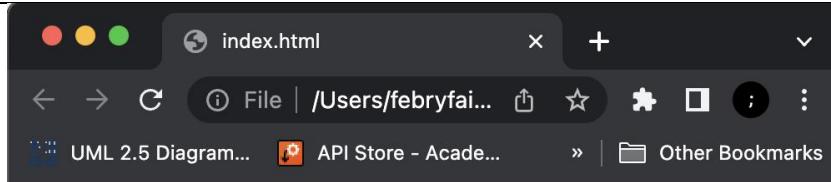
{
  alert("anda belum memasukkan alamat");
}
}

function terimakasih()
{
if ((document.f.inim.value!="")&&(document.f.inama.value!="")&&
(document.f.ialamat.value!=""))
{
  alert("Terima Kasih Telah mengisi Data");
} else
  alert("Mohon Data Dilengkapi");
}

</SCRIPT>
<h1 align="center">Latihan Event OnFocus dan OnBlur</h1>
<hr width="600" color="black" noshade size="10">
<font face="arial">
<h3 align="center">R E G I S T R A S I</h3>
<hr width="600" color="black" noshade size="2">
<CENTER>
<form name="f" method="get">
<TABLE>
<tr>
  <td width="31%">NIM</td>
  <td width="7%">:</td>
  <td width="62%"><input type="text" name="inim" size="9"
onFocus="window.status='Silahkan Mengisi NIM Anda';"    onBlur="masukannim()"></td>
</tr>
<tr>
  <td width="31%">NAMA</td>
  <td width="7%">:</td>
  <td width="62%"><input type="text" name="inama" size="23"
onFocus="window.status='Silahkan Mengisi Nama Anda';"    onBlur="masukannama()"></td>
</tr>
<tr>
  <td width="31%">ALAMAT</td>
  <td width="7%">:</td>
  <td width="62%"><input type="text" name="ialamat" size="34"
onFocus="window.status='Silahkan Mengisi Alamat Anda';"
onBlur="masukanalamat()"></td>
</tr>
</table>
<hr width="600" color="black" noshade size="2">
<p><input type="BUTTON" value="Kirim" Onclick="terimakasih()">
  <input type="reset" value="Reset"></p>
</form>

```

```
</center>
</BODY>
</HTML>
```



## Latihan Event OnFocus dan OnBlur

### REGISTRASI

NIM :   
NAMA :   
ALAMAT :

Contoh program yang menggunakan even **onLoad** dan **onUnload**:

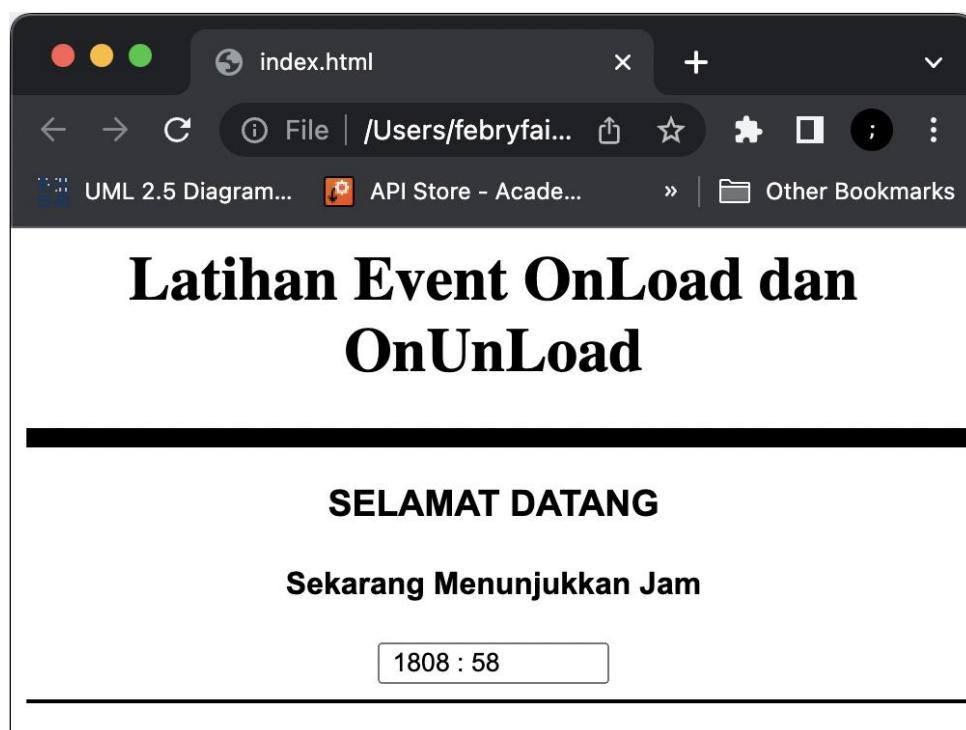
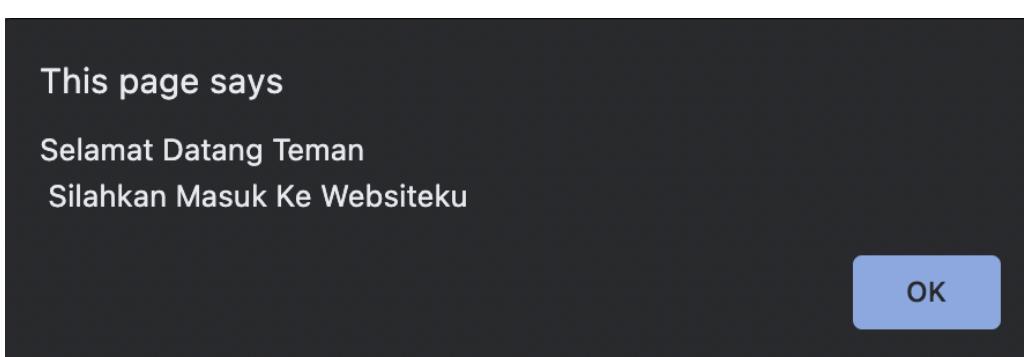
```
<HTML>
<BODY Onload="tanggal()" OnUnload="tutup()">
<SCRIPT LANGUAGE="JavaScript">
function tanggal() {
var d = new Date();
var y = d.getFullYear();
var m = d.getMonth() + 1;
var d = d.getDate();
var t = d + '/' + m + '/' + y + ' ';
defaultStatus = "Anda datang pada tanggal " + t + ".";
alert("Selamat Datang Teman \n Silahkan Masuk Ke Websiteku");
}

function timer() {
var d = new Date();
var jam = d.getHours();
var menit = d.getMinutes();
var detik = d.getSeconds();
var strwaktu = (jam<10)?"0"+jam:jam;
strwaktu +=(menit<10)?"0"+menit:" "+menit;
strwaktu +=(detik<10)?"0"+detik:" "+detik;
document.f.txtwaktu.value=" "+strwaktu;
setTimeout("timer()",200);
}
function tutup(){
```

```

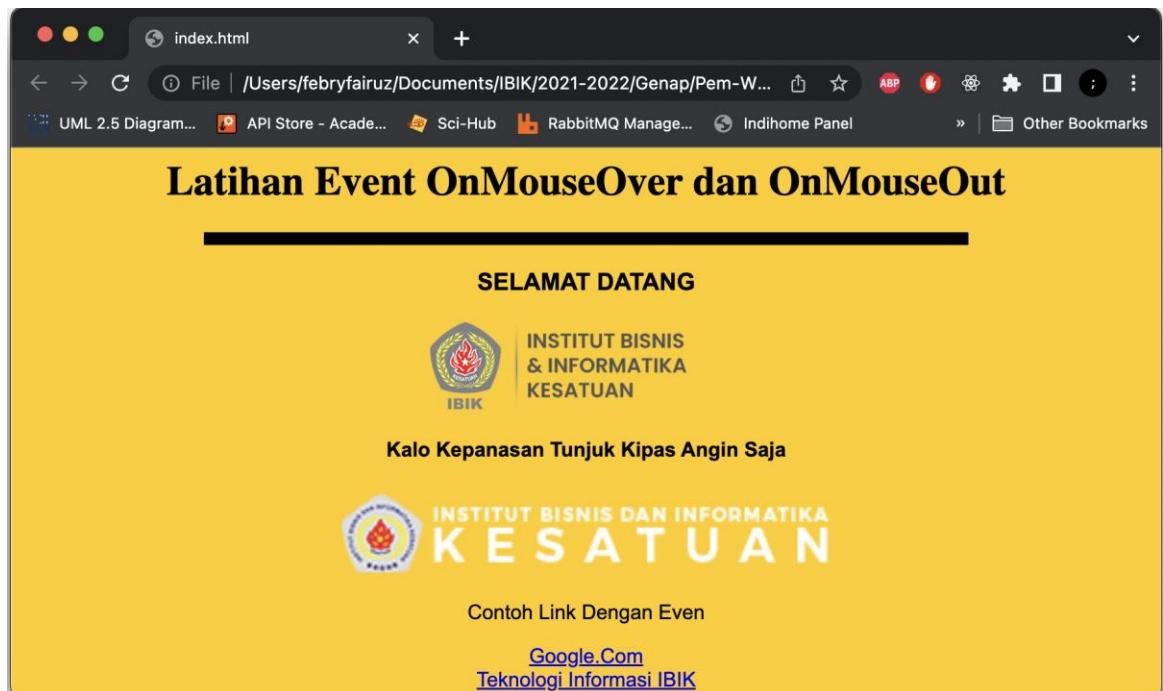
window.alert("Terimakasih Telah Berkunjung\nJangan lupa Datang kembali Ya..");
}
// end hiding -->
</SCRIPT>
<h1 align="center">Latihan Event OnLoad dan OnUnLoad</h1>
<hr width="600" color="black" noshade size="10">
<font face="arial">
<h3 align="center">SELAMAT DATANG</h3>
<CENTER>
<form name="f">
<h4 align="center">Sekarang Menunjukkan Jam</h4>
<input type="Text" size="16" name="txtwaktu">
<hr width="600" color="black" noshade size="2">
</form>
<SCRIPT LANGUAGE="JavaScript">timer()</SCRIPT>
</center>
</BODY>
</HTML>

```



Contoh program yang menggunakan **even onMouseOut** dan **onMouseOver**:

```
<HTML>
<BODY bgcolor="lightblue">
<SCRIPT LANGUAGE="JavaScript">
function g1(){
    document.f.imgLogo.src="https://www.ibik.ac.id/wp-content/uploads/2020/08/logo-ibik-
web.png"
}
function g2(){
    document.f.imgLogo.src="https://www.ibik.ac.id/wp-content/uploads/2019/10/logo-ibik-
web.png"
}
</SCRIPT>
<h1 align="center">Latihan Event OnMouseOver dan OnMouseOut</h1>
<hr width="600" color="black" noshade size="10">
<font face="arial">
<h3 align="center">SELAMAT DATANG</h3>
<CENTER>
<form name="f">
<Img Name="logo" height="72" src="https://www.ibik.ac.id/wp-content/uploads/2020/08/logo-
ibik-web.png"
onmouseover="document.logo.src='https://www.ibik.ac.id/wp-content/uploads/2019/10/logo-
ibik-web.png';window.status='Ibik 1'"
onmouseout="document.logo.src='https://www.ibik.ac.id/wp-content/uploads/2020/08/logo-
ibik-web.png';window.status='Ibik 2'"
<h4 align="center">Kalo Kepanasan Tunjuk Kipas Angin Saja</h4>
<A HREF="latevenonclick.html" onMouseOver="g1()" onMouseOut="g2()">
<IMAGE NAME="imgLogo" SRC="https://www.ibik.ac.id/wp-content/uploads/2020/08/logo-ibik-
web.png" HEIGHT=72 BORDER=0>
</A>
<p>
Contoh Link Dengan Even
</p>
<A HREF="http://www.google.com" onMouseOver="document.bgColor='#ffcc00';
    onMouseOver=window.status='Ke Website Google.Com'; return true">Google.Com</A><BR>
<A HREF="https://www.ibik.ac.id/s1-teknologi-informasi/" onMouseOver="window.status='Oh,
jangan deket-deket..';
    return true" onMouseOut="alert('Nah gitu dong'); return true">Teknologi Informasi IBIK </A>
<hr width="600" color="black" noshade size="2">
</form>
</center>
</BODY>
</HTML>
```



#### 4.3.2. Event on Submit

Event on submit akan dibangkitkan apabila seorang user menekan tombol submit. Dengan event ini data yang diinputkan akan dikirimkan ke tempat lain (email, file teks atau ke dalam suatu tabel).

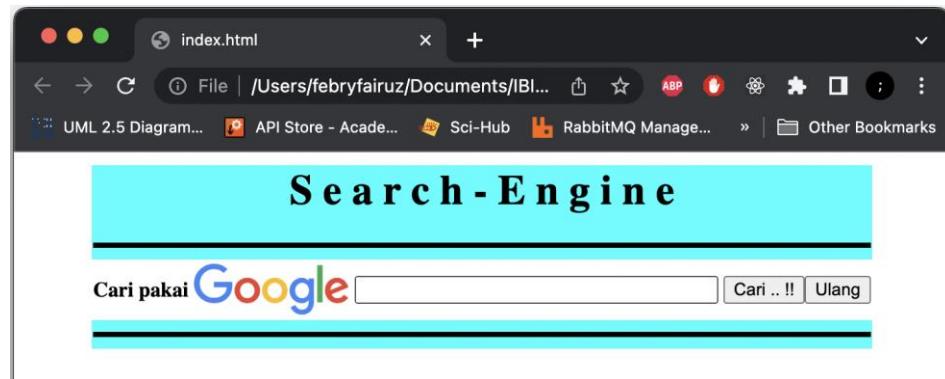
#### Contoh program

```
<Html>
<Body>
<SCRIPT LANGUAGE="JavaScript">
function cari() {
var kata = document.formcari.keyword.value; var hasil =
"http://www.google.com/search?q=" + kata ; window.open(hasil, 'google',
config='height=500,width=750 scrollbars=yes location=yes')
}
</SCRIPT>
<FORM NAME="formcari" onSubmit="cari()">
<center>
<table>
<tr>
<td colspan="4" bgcolor="orange"><h1 align="center">S e a r c h -
E n g i n e</h1><hr color="black" size="4" ></td>
</tr>
<tr>
<td><b>Cari pakai</b></td>
<td></td>
<td><INPUT NAME="keyword" SIZE="40" TYPE="text"></td>
<td><INPUT TYPE="submit" VALUE="Cari .. !!"><input type="reset"
```

```

Value="Ulang"></td>
</tr>
<tr>
  <td colspan="4" bgcolor="orange"><hr color="black" size="4"
></td>
</tr>
</table>
</center>
</FORM>
</body>
</html>

```



Berikut diberikan contoh event submit yang terjadi pada form pengisian data guestbook yang hasilnya akan dikirim ke suatu email.

```

<Html>
<Body>
<SCRIPT LANGUAGE="JavaScript">
function isiform(form) {
isinama(form);
isiemail(form);
isikomentar(form);
kosongkan(form);
}
function kosongkan(form){
if((isinama(form) && isiemail(form) && isikomentar(form)))
{
  form.submit;
}
if((isinama(form)== false || isiemail(form)== false || isikomentar(form)== false))
{
  salahisi(form);
}
function salahisi(form)

```

```

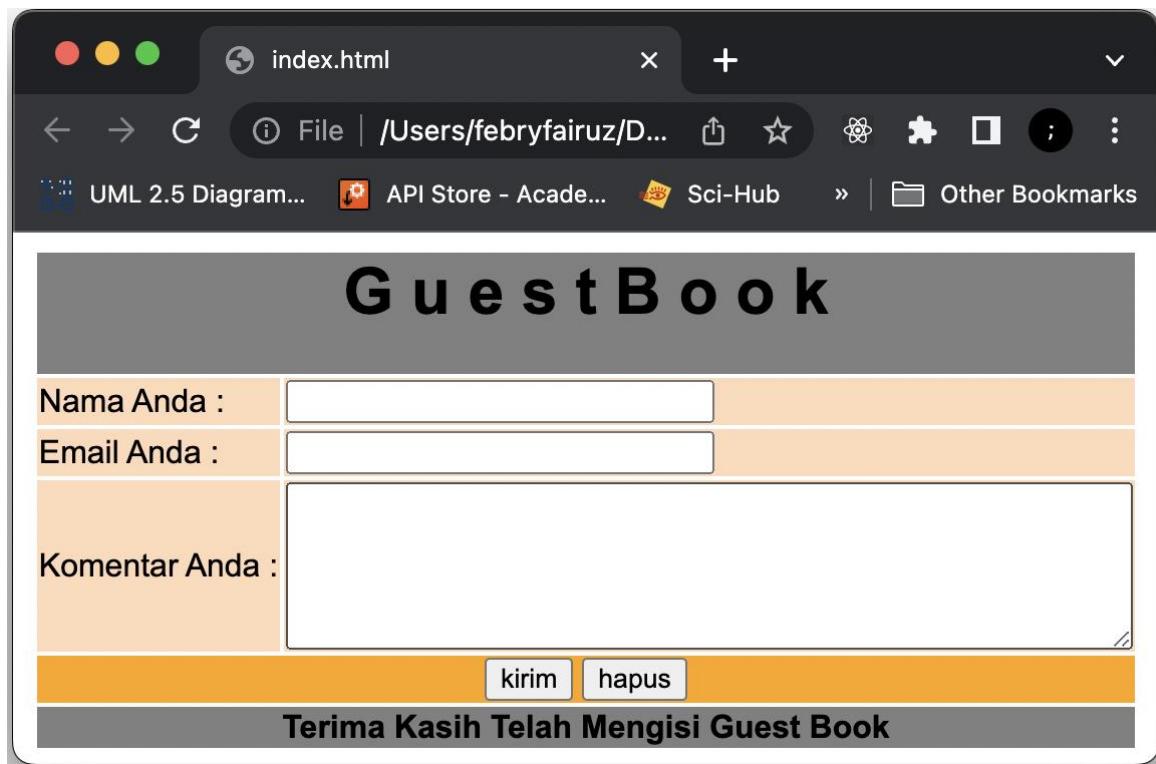
{
var teks ="Ada Kesalahan Isian :"; if (isinama(form)== false) {teks +="\nNama Anda";} if
(isiemail(form)==false) {teks +="\nEmail Anda";} if (isikomentar(form)==false) {teks
+="\nKomentar Anda";} alert(teks);
}
function isinama(form)
{
if (form.nama.value=="") {return false;} else {return true;}
}
function isiemail(form)
{
if((form.email.value=="" || form.email.value.indexOf('@',0)==-1) ||
form.email.value.indexOf('.')==-1)
{return false;} else {return true;}
}
function isikomentar(form)
{
if(form.cs.value=="") {return false;} else
{return true;}
}
</script>
<form name="form-guest" method="post" action="mailto:febrid@ibik.ac.id?subject=FormJS">
<font face="Arial">
<table align="center">
<tr bgcolor="gray">
<td colspan="2" align="center"><h1>G u e s t B o o k</h1></td>
</tr>
<tr bgcolor="peachpuff">
<td>Nama Anda :</td>
<td><input type="text" value="" name="nama" size="30"></td>
</tr>
<tr bgcolor="peachpuff">
<td>Email Anda :</td>
<td><input type="text" value="" name="email" size="30"></td>
</tr>
<tr bgcolor="peachpuff">
<td>Komentar Anda :</td>
<td><textarea name="cs" rows="5" cols="50"></textarea></td>
</tr>
<tr bgcolor="orange">
<td colspan="2" align="center">
<input type="button" name="thesubmit" value="kirim" onClick="isiform(this.form)">
<input type="reset" value="hapus">
</td>
</tr>

```

```

<tr bgcolor="gray">
<td colspan="2" align="center"><b>Terima Kasih Telah Mengisi
Guest Book</b> </td>
</tr>
</table>
</font>
</form>
</body>
</html>

```



Untuk memuat suatu image, pada Javascript terdapat objek Image. Untuk membuat objek tersebut pendeklarasiannya adalah sebagai berikut :

```
img1 = new Image()
```

```
img1.src = "pic1.gif"
```

Artinya membuat objek image dengan isinya adalah image pic1.gif berikut akan diberikan contoh mengenai objek image :

### Contoh Program

```

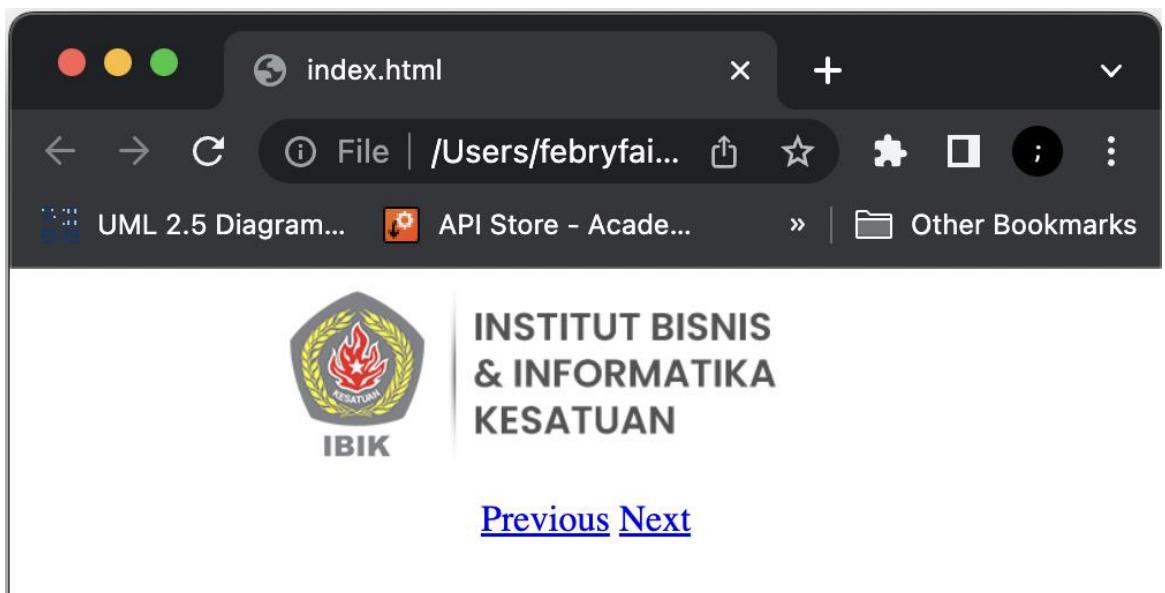
<HTML>
<BODY>
<SCRIPT LANGUAGE="JavaScript">

```

```

var num=1 ;
img1 = new Image();
img1.src = "pic1.gif";
img2 = new Image();
img2.src = "pic2.gif";
img3 = new Image();
img3.src = "pic3.gif";
img4 = new Image();
img4.src = "pic4.gif";
img5 = new Image();
img5.src = "pic5.gif";
img6 = new Image();
img6.src = "pic6.gif";
function slideshow(x){
num=num+x;
if (num==7) {
    num=1
}
if (num==0) {
    num=6
}
document.mypic.src=eval("img"+num+".src")
}
</SCRIPT>
<CENTER>
<IMG SRC="pic1.gif" NAME="mypic" BORDER=0 height="200" width="150"><p>
<A HREF="JavaScript:slideshow(-1)">Previous</A>
<A HREF="JavaScript:slideshow(1)">Next</A>
</CENTER>
</BODY>
</HTML>

```



### Contoh Program selanjutnya

```
<html>
<head>
<script language="javascript">
var image1=new Image();
image1.src="1.gif";
var image2=new Image();
image2.src="2.gif";
var image3=new Image();
image3.src="3.gif"
</script>
</head>
<body>
<center>
<h2>Penggunaan Objek Image Untuk Membuat SlideShow</h2><hr size=5 color="black">

<script>
function slideit(){
    document.images.slide.src=eval("image"+step+".src");
    if (step<3) step++ else step=1 ;
}
slideit();
</script>
<hr size=5 color="black">
</body>
</html>
```

**Penggunaan Objek Image Untuk Membuat SlideShow**

---

 INSTITUT BISNIS  
& INFORMATIKA  
KESATUAN

---

#### 4.4. Tugas dan Latihan

1. Buatlah program dengan javascript untuk memunculkan alert Selamat Pagi, Selamat Siang dan Selamat Malam. Sesuai dengan waktu yang tertera di komputer !!!!!
2. Buat program untuk menghitung/mencari akar-akar dari suatu persamaan  
$$F(x) = ax^2 + bx + c$$
Rumus mencari akar  $x_1$  dan  $x_2$  adalah :

## **DAFTAR PUSTAKA**

- Dorucher, David, 2021. HTML & CSS QuickStart Guide. ClydeBank Media LLC.
- Robbins, Jennifer. 2018. Learning Web Design. O'Reilly Media, Inc.
- Meloni, Julie, 2018. HTML, CSS, and JavaScript All in One, Sams Teach Yourself. Pearson Education.
- Firdaus, Thoriq dan Frain, Ben. 2016. HTML5 and CSS3: Building Responsive Websites. Packt Publishing Ltd.
- Haverbeke, Marjin. 2014. Eloquent JavaScript, 2nd Ed. No Starch Press.

### **Website :**

<https://www.w3schools.com/html/>.

<https://www.w3schools.com/css/default.asp>.

<https://www.w3schools.com/js/default.asp>.



INSTITUT BISNIS  
& INFORMATIKA  
KESATUAN

# PEMROGRAMAN

---

## WEB

---