



Perancangan **APLIKASI** Visual

NAMA :

NPM :

Sistem Informasi Program Diploma Tiga

Universitas Teknologi Yogyakarta

2022

MODUL PRAKTIKUM PERANCANGAN APLIKASI VISUAL



Disusun Oleh :

Ikrimach, S.Kom., M.Cs.

Sutarman, Ph.D

Murti Retnowo, S.Kom., M.Cs.

**Universitas Teknologi Yogyakarta
2022**

@2022
Diterbitkan Oleh:
Universitas Teknologi Yogyakarta
Jl. Siliwangi, Jombor, Sleman, Yogyakarta
Email : publikasi@uty.ac.id
Website : uty.ac.id

DESAIN WEB

ISBN:

00000000000000

Oleh:

Ikrimach

Sutarman

Murti Retnowo

Edisi ke-1

Cetakan Pertama, 2022

Hak Cipta @2022 pada penulis,

Dilarang memperbanyak atau memindahkan Sebagian atau seluruh buku ini dalam bentuk apapun tanpa ijin dari penulis.

KATA PENGANTAR

Assalamualaikum warahmatullahi wabarokatuh. Alhamdulillah Rabbil Alamin, puji syukur selalu kami panjatkan atas kehadiran Allah SWT. Berkat rahmat dan hidayah-NYA penyusun dapat menyelesaikan modul praktikum desain web ini.

Mudul praktikum ini disusun dari berbagai sumber, yang bertujuan untuk bisa menjadi acuan bagi mahasiswa dalam mengikuti pelaksanaan praktikum mata kuliah perancangan aplikasi visual, Universitas Teknologi Yogyakarta.

Didalam modul ini juga disertakan acuan materi berupa contoh mulai dasar-dasar membangun rangka web, hingga tahap desain *interface*. Modul ini disusun sesuai dengan tahapan belajar desain web dari berbagai sumber, sehingga diharapkan dalam penggunaannya, mengikuti tahapan yang sudah ditentukan.

Penyusun menyadari masih banyaknya kekurangan dalam modul ini, sehingga senantiasa mengharapkan masukan dari pihak-pihak yang terkait dalam penggunaan modul ini, agar penyusun dapat melakukan perbaikan dan menyusun kembali dengan lebih baik. Kritik dan saran bisa dikirimkan melalui email perwakilan penulis: ikrimach@uty.ac.id , Terima kasih kami ucapkan dan semoga bermanfaat.

Wassalamualaikum warahmatullahi Wabarokatuh.

Yogyakarta, Agustus 2022

Tim Penulis

PENDAHULUAN

A. Deskripsi Materi

Perancangan Aplikasi Visual mencakup berbagai keterampilan dan disiplin dalam produksi aplikasi visual. Berbagai bidang meliputi desain grafis web, desain antarmuka, authoring, termasuk kode standar dan perangkat lunak berpemilik, desain pengalaman pengguna, dan optimisasi mesin pencari.

Istilah perancangan aplikasi visual biasanya digunakan untuk menggambarkan proses perancangan yang berkaitan dengan desain front-end (sisi klien) dari situs web termasuk penulisan mark-up. Perancangan Aplikasi Visual sebagian tumpang tindih dengan teknik web dalam lingkup pengembangan aplikasi yang lebih luas. Perancang ini diharapkan memiliki kesadaran akan kegunaan dan jika peran mereka melibatkan pembuatan mark-up maka mereka juga diharapkan untuk selalu mengikuti pedoman aksesibilitas aplikasi perangkat lunak.

Pemahaman pengguna tentang isi sebuah aplikasi sering kali bergantung pada pemahaman pengguna tentang bagaimana aplikasi itu sendiri dapat bekerja. Ini adalah bagian dari *user experience design* (desain pengalaman pengguna). Pengalaman pengguna terkait dengan tata letak, petunjuk dan pelabelan yang jelas pada sebuah aplikasi. Seberapa baik pengguna memahami bagaimana mereka dapat berinteraksi pada aplikasi memungkinkan juga bergantung pada desain interaktif. Jika pengguna merasakan kesenangan pada tampilan yang baik, mereka cenderung terus akan menggunakannya.

Pengguna yang terampil dan berpengalaman menggunakan aplikasi mungkin menemukan antarmuka aplikasi yang lebih khas namun kurang intuitif atau kurang user friendly. Namun, pengguna dengan sedikit pengalaman cenderung tidak melihat kelebihan atau kegunaan antarmuka aplikasi yang kurang intuitif. Hal ini mendorong kecenderungan pengalaman pengguna yang lebih universal dan kemudahan akses untuk menampung pengguna sebanyak mungkin tanpa mepedulikan keahlian pengguna. Sebagian besar desain pengalaman dan desain interaktif pengguna dipertimbangkan dalam desain antarmuka pengguna.

B. Prasyarat

- Mahasiswa mahir dalam mengoperasikan komputer
- Mahasiswa mampu menggunakan teks editor.
- Mahasiswa memiliki pemahaman konsep dasar perancangan aplikasi visual yang baik .

C. Petunjuk Pemakaian

Mudul ini dapat digunakan mahasiswa dengan pertimbangan sebagai berikut :

- Mahasiswa telah memiliki modul dan membaca modul sebelum praktikum dimulai.
- Mahasiswa mempelajari serta mengidentifikasi isi modul yang diuraikan lebih rinci oleh pengampu.
- Mahasiswa dan pengampu mendiskusikan materi untuk mencari penyelesaian pada kasus tertentu.
- Mahasiswa menyimpulkan materi yang telah didiskusikan.
- Mahasiswa menjawab soal latihan yang diberikan.

D. Standar Kompetensi

- Mahasiswa mampu menguasai teknik dan implementasi rancang antarmuka sistem.
- Mahasiswa mengenal dan mampu menggunakan bahasa desain yang meliputi html,xhtml,html5,css,css3.
- Mahasiswa mampu membangun aplikasi visual berbasis web yang lebih interaktif dan memiliki nilai estetika yang tinggi.

DAFTAR ISI

KATA PENGANTAR	iv
PENDAHULUAN	v
DAFTAR ISI	vii
BAB I HTML DAN XHTML	1
1.1 Kompetensi Dasar.....	1
1.2 Indikator	1
1.3 Uraian Materi.....	1
1.3.1 Struktur Dasar HTML	1
1.3.2 XHTML	2
1.3.3 Struktur Dokumen	2
1.3.4 Elemen XHTML.....	3
1.3.5 Atribut XHTML	3
1.3.6 <! DOCTYPE> Wajib.....	3
1.4 Latihan.....	5
1.4.1 Contoh HTML sederhana.....	5
1.4.2 Lists	5
1.4.3 Heading.....	6
1.4.4 Paragraf.....	7
1.4.5 Preformatted Text.....	7
1.4.6 Blockquote	8
1.4.7 Line Break	8
1.4.8 Font	8
1.4.9 Link.....	9
1.5 Tugas.....	11
BAB II HTML DAN XHTML LANJUTAN	12
2.1 Image.....	12
2.1.1 Format Image	12
2.1.2 Insert Image ke Document	12
2.1.3 Attribute Value Description	13
2.1.4 Image Map.....	13
2.2 Tugas	14
BAB III HTML DAN XHTML LANJUTAN	15
3.1 Tabel.....	15
3.2 Latihan.....	15
3.2.1 Tabel sederhana	15
3.2.2 Tabel lanjut.....	17
3.2.3 Iframe.....	17
3.3 Tugas	18
BAB IV HTML DAN XHTML LANJUTAN	19
4.1 Form	19
4.1.1 Textbox.....	19
4.1.2 Checkbox	19
4.1.3 Radio.....	20
4.1.4 Submit dan Reset.....	20
4.2 Latihan.....	20
4.2.1 Membuat Form Input dengan Text box,Submit dan Reset.....	20
4.2.2 Membuat Form Input dengan Check box	21
4.2.3 Membuat Form Input dengan Radio box	21

4.2.4	Membuat Form Input dengan Textarea	21
4.2.5	Membuat Form Input Dengan Select.....	22
4.3	Tugas	22
BAB V	HTML5	23
5.1	Kompetensi Dasar.....	23
5.2	Indikator	23
5.3	Uraian Materi	23
5.3.1	Elemen Baru HTML5.....	23
5.3.2	API Baru HTML5	26
5.4	Latihan.....	26
5.4.1	Contoh kanvas	26
5.4.2	SVG	27
5.5	Tugas	29
BAB VI	HTML5 LANJUTAN.....	30
6.1.1	Maps	30
6.1.2	Video	30
6.1.3	Audio	30
6.1.4	Youtube.....	31
6.1.5	Plug-in	31
6.2	Tugas	32
BAB VII	CSS.....	33
7.1	Kompetensi Dasar.....	33
7.2	Indikator	33
7.3	Uraian Materi	33
7.3.1	Inline Styles	33
7.3.2	Istilah-istilah dalam Style Sheet.....	34
7.3.3	Komentar dalam Style Sheets	35
7.3.4	Pemakaian elemen style	35
7.3.5	Link ke sheet lainnya	35
7.4	Latihan.....	35
7.4.1	Mengatur tepi halaman (page margin).....	35
7.4.2	Mengatur inden kiri dan kanan	36
7.4.3	Mengatur jarak penulisan dari tepi atas dan bawah halaman.....	36
7.4.4	Inden pada baris pertama.....	37
7.5	Tugas	37
BAB VIII	CSS LANJUTAN	38
8.1	Mengatur Format Font.....	38
8.1.1	Model huruf.....	38
8.1.2	Mengatur ukuran huruf.....	38
8.1.3	Mengatur jenis huruf	38
8.1.4	Cara menghindari masalah huruf dan batas tepi halaman web	39
8.1.5	Menambahkan border dan latar belakang	40
8.1.6	Mengatur warna.....	41
8.1.7	Bar navigasi	42
8.2	Tugas	44
BAB IX	CSS3	45
9.1	Kompetensi Dasar.....	45
9.2	Indikator	45
9.3	Uraian Materi	45
9.4	Latihan.....	46
9.4.1	Border-radius CSS3.....	46

1. Empat nilai	46
2. Tiga nilai	46
3. Dua nilai	46
4. Satu nilai	46
9.4.2 Backgrounds CSS3	47
9.5 Tugas	49
BAB X CSS3 LANJUTAN	50
10.1 Warna CSS3	50
10.1.1 Warna RGBA	50
10.1.2 Warna HSL	50
10.1.3 Warna HSLA	51
10.1.4 Warna Kegelapan/opacity	51
10.2 Tugas	52
BAB XI CSS3 LANJUTAN	53
11.1 Transisi Warna	53
11.1.1 Linier gradient	53
11.1.2 Repeat Gradients	54
11.1.3 Radial Gradients	55
11.1.4 Efek bayangan CSS3	56
11.2 Tugas	57
BAB XII CSS3 LANJUTAN	58
12.1 Transformasi CSS3	58
12.1.1 Transformasi 2D	58
12.1.2 Transformasi 3D	60
12.2 Transisi CSS3	61
12.2.1 Ubah Beberapa Nilai Properti	61
12.2.2 Menentukan kecepatan transisi	62
12.2.3 Transisi + Transformasi	63
12.3 Tugas	63
BAB XIII CSS3 LANJUTAN	64
13.1 ANIMASI CSS3	64
13.1.1 Aturan @keyframes	64
13.1.2 Menetapkan Berapa Banyak Animasi yang Harus Dijalankan	66
13.1.3 Menentukan Kurva Kecepatan Animasi	67
13.2 Tugas	69
BAB XIV KOLABORASI HTML,CSS,JAVASCRIPT	70
14.1 POP-UP	70
14.2 SLIDE SHOW	71
14.3 LOGIN FORM	73
14.4 VALIDASI	76
14.5 MULTIPLE STEP FORM	79
14.6 Tugas	83
DAFTAR PUSTAKA	84

BAB I

HTML DAN XHTML

1.1 Kompetensi Dasar

- a. Mahasiswa dapat mengenal tag dan fungsinya dalam membangun rancang awal sistem berbasis web.

1.2 Indikator

- a. Mampu membangun rancang awal sistem berbasis web.

1.3 Uraian Materi

Hypertext Markup Language merupakan standard bahasa yang digunakan untuk menampilkan document web, yang bisa anda lakukan dengan HTML yaitu:

1. Mengontrol tampilan dari web page dan contentnya.
2. Mempublikasikan document secara online sehingga bisa di akses dari seluruh dunia.
3. Membuat online form yang bisa digunakan untuk menangani pendaftaran, transaksi secara online dan lain-lain.
4. Menambahkan object-object seperti image, audio, video dan animasi lainnya.

1.3.1 Struktur Dasar HTML

Hypertext Markup Language (HTML) adalah bahasa yang digunakan untuk menulis halaman web. Ciri utama dokumen HTML adalah adanya tag dan elemen. Elemen dalam dokumen HTML dikategorikan menjadi dua yaitu elemen <HEAD> yang berfungsi memberikan informasi tentang dokumen tersebut dan elemen <BODY> yang menentukan bagaimana isi suatu dokumen ditampilkan oleh browser, seperti paragraf, list(daftar), tabel dan lain-lain. Sedangkan tag dinyatakan dengan tanda lebih kecil “ < “ (tag awal) dan tanda lebih besar “ > “ (tag akhir). Dalam penggunaannya sebagian besar kode HTML tersebut harus terletak di antara tag kontainer. Yaitu diawali dengan <namatag> dan diakhiri dengan </namatag> (terdapat tanda "/").

Dokumen HTML mempunyai tiga buah tag utama yang membentuk struktur dari dokumen HTML yaitu HTML, HEAD dan BODY.

Tag HTML berfungsi untuk menyatakan suatu dokumen HTML, tag HEAD berfungsi untuk memberikan informasi tentang dokumen HTML dan tag BODY berfungsi untuk menyimpan informasi atau data yang akan ditampilkan dalam dokumen HTML.

1.3.2 XHTML

XHTML adalah HTML yang ditulis sebagai XML.

Apa itu XHTML?

- XHTML adalah singkatan dari EXTensible HyperText Markup Language
- XHTML hampir identik dengan HTML
- XHTML lebih ketat dari HTML
- XHTML adalah HTML yang didefinisikan sebagai aplikasi XML
- XHTML didukung oleh semua browser utama

Mengapa XHTML?

Banyak halaman di internet berisi HTML yang "buruk". Kode HTML ini bekerja dengan baik di sebagian besar browser (meskipun tidak mengikuti aturan HTML):

```
<html>
<head>
  <title>HTML</title>
<body>
  <h1>HTML yang buruk
  <p>ini adalah paragraf
</body>
```

Pasar hari ini terdiri dari berbagai teknologi browser. Beberapa browser berjalan dikomputer, dan beberapa browser berjalan diponsel atau perangkat kecil lainnya. Perangkat yang lebih kecil seringkali kekurangan sumber daya atau kekuatan untuk menafsirkan markup yang "buruk".

XML adalah bahasa markup dimana dokumen harus ditandai dengan benar. Dengan menggabungkan kekuatan HTML dan XML, XHTML dikembangkan. XHTML didesain ulang HTML sebagai XML.

Perbedaan Paling Penting dari HTML:

1.3.3 Struktur Dokumen

- XHTML DOCTYPE adalah wajib
- Atribut xmlns di <html> adalah wajib
- <html>, <head>, <title>, dan <body> adalah wajib

1.3.4 Elemen XHTML

- Elemen XHTML harus disisipkan dengan benar
- Elemen XHTML harus selalu ditutup
- Elemen XHTML harus dalam huruf kecil
- Dokumen XHTML harus memiliki satu elemen root

1.3.5 Atribut XHTML

- Nama atribut harus dalam huruf kecil
- Nilai atribut harus dikutip
- Minimalisasi atribut dilarang

1.3.6 <! DOCTYPE> Wajib

Dokumen XHTML harus memiliki deklarasi DOCTYPE XHTML. Elemen <html>, <head>, <title>, dan <body> juga harus ada, dan atribut xmlns di <html> harus menentukan namespace xml untuk dokumen.

Contoh ini menunjukkan dokumen XHTML dengan minimal tag yang diperlukan:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"
    "http://www.w3.org/TR/xhtml-basic/xhtml-basic10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<html>
<head>
<meta charset="utf-8">
<title>judul</title>
</head>
<body>
konten
</body>
</html>
```

Saat membuat dokumen HTML atau XHTML, penting untuk menambahkan deklarasi Doctype. Ini memastikan dokumen akan diuraikan dengan cara yang sama oleh browser yang berbeda.

Jika Anda memerlukan DOCTYPE yang cocok dengan versi spesifik dari (X) HTML, deklarasi DOCTYPE harus tepat (baik dalam ejaan maupun dalam kasus) untuk memiliki efek yang diinginkan, yang terkadang sulit dilakukan.

Untuk memudahkan pekerjaan, anda dapat mengakses <https://www.w3.org/QA/2002/04/valid-dtd-list.html> . dalam website tersebut terdapat daftar deklarasi DOCTYPE yang direkomendasikan yang dapat Anda gunakan dalam dokumen Web Anda.

Elemen XHTML Harus Bersarang dengan Benar

Dalam HTML, beberapa elemen dapat disatukan dengan tidak benar satu sama lain, seperti ini:

```
<b> <i> Teks ini dicetak tebal dan miring </ b> </ i>
```

Dalam XHTML, semua elemen harus saling bersarang satu sama lain, seperti ini:

```
<b> <i> Teks ini dicetak tebal dan miring </ i> </ b>
```

Elemen XHTML Harus Selalu Ditutup

Ini salah:

```
<p> Ini adalah paragraf  
<p> Ini adalah paragraf lain
```

Ini benar:

```
<p> Ini adalah paragraf </ p>  
<p> Ini adalah paragraf lain </ p>
```

Unsur Kosong Juga Harus Ditutup

Ini salah:

```
Break: <br>  
Horizontal rule: <hr>  
Image: <img src = "oke.gif" alt = "setuju">
```

Ini benar:

```
Break: <br />  
Horizontal rule: <hr />  
Image: <img src = "happy.gif" alt = "wajah bahagia" />
```

Unsur XHTML Harus Lower Case

Ini salah:

```
<BODY>  
<P> Ini adalah paragraf </ P>  
</ BODY>
```

Ini benar:

```
<body>  
<p> Ini adalah paragraf </ p>  
</ body>
```

Nama Atribut XHTML Harus Lower case

Ini salah:

```
<table WIDTH = "100%">
```

Ini benar:

```
<table width = "100%">
```

Nilai Atribut Harus Dikutip

Ini salah:

```
<table width = 100%>
```

Ini benar:

```
<table width = "100%">
```

Minimisasi Atribut Dilarang

Salah:

```
<input type = " checkbox " nama = "kendaraan" nilai = "mobil"
dicentang />
```

Benar:

```
<input type = "checkbox" name = "kendaraan" value = "car" checked
= "checked" />
```

Salah:

```
<input type = "text" name = "nama belakang" disabled />
```

Benar:

```
<input type = "text" name = "nama belakang" disabled = "disabled"
/>
```

Cara Mengkonversi dari HTML ke XHTML

- Tambahkan XHTML `<!DOCTYPE>` ke baris pertama setiap halaman
- Tambahkan atribut `xmlns` ke elemen html dari setiap halaman
- Ubah semua nama elemen menjadi huruf kecil
- Tutup semua elemen kosong
- Ubah semua nama atribut menjadi huruf kecil
- Kutip semua nilai atribut

1.4 Latihan

1.4.1 Contoh HTML sederhana

```
<html>
  <head>
    <title>struktur dokumen html </title>
  </head>
  <body>
    hello,... selamat belajar membuat home page dengan html
  </body>
</html>
```

1.4.2 Lists

Terdapat tipe list yang dapat digunakan, yaitu :

1.4.2.1 Unordered Lists `` :

Untuk membuat daftar item dengan tanda bullet. List entries didefinisikan dengan tag ``.

Contoh :

```
<ul>
<li> item nomer 1
<li> item nomer 2
<li> item nomer 3
</ul>
```

1.4.2.2 Ordered Lists :

Juga digunakan untuk membuat daftar item, dengan tiap item dapat menggunakan angka arab atau romawi. List entries juga didefinisikan dengan tag.

Contoh :

```
<ol type=i>
<li> item nomer 1
<li> item nomer 2
<li> item nomer 3
</ol>
```

1.4.2.3 Definition Lists: <DL>

Contoh :

```
<dl>
<dt> item pertama.
<dd> penjelasan tentang item pertama.
<dt> item kedua.
<dd> penjelasan tentang item kedua
</dl>
```

1.4.3 Heading

Heading biasanya digunakan untuk menulis judul dalam sebuah artikel. Heading dibagi menjadi beberapa tingkatan yaitu heading 1 </h1> memiliki tampilan dengan ukuran font paling besar sesuai standart heading. Contoh seperti dibawah ini anda akan melihat perbedaannya.

```
<html>
  <head>
    <title> .:heading dokumen html:. </title>
  </head>
  <body>
    <h1>heading tingkat 1 </h1>
    <h2>heading tingkat 2 </h2>
    <h3>heading tingkat 3 </h3>
    <h4>heading tingkat 4 </h4>
    <h5>heading tingkat 5 </h5>
    <h6>heading tingkat 6 </h6>
  </body>
</html>
```

1.4.4 Paragraf

Paragraf dalam html dideklarasikan menggunakan tag <p>. Biasanya digunakan untuk menulis teks yang memiliki jumlah kalimat banyak dalam dokumen web. Ada empat atribut <p> yang biasa digunakan yaitu :

- Right : Untuk membuat tulisan rata kanan
- Left : Untuk membuat tulisan rata kiri
- Center : Untuk membuat tulisan rata tengah
- Justify : Untuk membuat tulisan sama rata

```
<html>
  <head>
    <title>paragraf</title>
  </head>
  <body>
    <p align="right">
      HTTP adalah suatu protokol yang menentukan aturan yang perlu
      diikuti oleh web browser dalam meminta atau mengambil suatu
      dokumen dan oleh web server dalam menyediakan dokumen yang diminta
      web browser.
    <p align="center">
      Internet berasal dari kata interconnection networking yang
      mempunyai arti hubungan berbagai komputer dan berbagai tipe
      komputer yang membentuk sistem jaringan dan mencakup seluruh dunia
      (jaringan global) dengan melalui jalur telekomunikasi seperti
      telepon, wireless dan lainnya.
    <p align="left">
      Web Browser yang bertindak sebagai client memungkinkan anda untuk
      menginterpretasikan dan melihat informasi pada Web, sedang web
      server yang bertindak sebagai server memungkinkan anda untuk
      menerima informasi yang diminta browser
    </P>
  </body>
</html>
```

1.4.5 Preformatted Text

Preformatted text digunakan untuk membuat tampilan tulisan dalam browser sama seperti letak tulisan aslinya.

```
<html>
  <head>
    <title>preformatted text</title>
  </head>
  <body>
    <pre>
      uniform resource locator (url)
      =====
      url adalah suatu sarana yang dipakai untuk menentukan lokasi
      informasi pada suatu web server.
      url dapat diibaratkan suatu alamat, dimana alamat tersebut
      terdiri dari :
```



```

:) protokol yang digunakan oleh suatu browser untuk mengambil
informasi
:) nama komputer ( server ) dimana informasi tersebut berada
:) jalur/path serta nama file dari suatu informasi
</pre>
</body>
</html>

```

1.4.6 Blockquote

Blockquote digunakan untuk menulis kutipan text

```

<html>
<head>
  <title>blockquote</title>
</head>
<body>
<h3>sesuatu yang tidak perlu dicoba</h3>
<blockquote>
sesuatu yang kelihatan mengasyikkan tapi sebenarnya lebih banyak
menambah masalah pada remaja adalah narkoba, seks, alkohol dan
merokok. jangan sekali-kali mencoba jika tidak ingin membuat
masalah baru</blockquote>
</body>
</html>

```

1.4.7 Line Break

Line break digunakan untuk memindahkan baris ke baris berikutnya dalam dokumen html.

```

<html>
<head>
<title>linebreak</title>
</head>
<body>
<p>banyak cara yang dapat digunakan untuk melakukan instalasi
freebsd. namun ada tiga cara yang paling sering dilakukan , yaitu
:
<br>instalasi melalui ftp<br> instalasi melalui cdrom
<br> instalasi melalui partisi dos
</body>
</html>

```

1.4.8 Font

Font digunakan untuk memodifikasi tulisan teks dalam dokumen html. font memiliki beberapa atribut yang biasa digunakan yaitu :

- Size : Untuk menentukan ukuran font
- Face : Untuk menentukan jenis font
- Color : Untuk membuat warna font

1.4.8.1 Ukuran font dalam HTML :

```
<html>
<head>
  <title>ukuran font</title>
</head>
<body>
<font size=1>ukuran font 1</font><br>
<font size=2>ukuran font 2</font><br>
<font size=3>ukuran font 3</font><br>
<font size=4>ukuran font 4</font><br>
<font size=5>ukuran font 5</font><br>
<font size=6>ukuran font 6</font><br>
<font size=7>ukuran font 7</font><br>
</body>
</html>
```

1.4.8.2 Jenis font dalam HTML :

```
<html>
<head>
  <title>jenis font</title>
</head>
<body>
<font size=5>
<font face = "arial">arial , contoh ac milan <p>
<font face = "verdana">verdana , contoh persebaya <p>
<font face = "tahoma">tahoma , contoh indonesia <p>
</body>
</html>
```

1.4.8.3 Warna font dalam HTML :

```
<html>
<head>
  <title>warna font</title>
</head>
<body>
<font size=5>
<font color= "red">yogyakarta <p>
<font color = "#ff0000">kota budaya <p>
<font color = "#00ff00">kota pariwisata dan pendidikan<p>
</body>
</html>
```

1.4.9 Link

Link biasanya digunakan untuk menghubungkan suatu halaman ke halaman lain atau ke dokumen lain. Perintah <a> digunakan untuk membuat suatu link. Untuk membuat link ke dokumen HTML lain digunakan perintah :

```
<a href="nama_dokumen">teks pada browser</a>
```

Anda juga bisa membuat link dalam sebuah dokumen HTML. Untuk itu perlu disediakan nama anchor lokasi tujuan dari link tersebut. Nama anchor dibuat dengan menambahkan atribut name="" pada atribut <a>. Misalnya:

```
<a name="atas">pemain-pemain ac milan menurut abjad</a>
```

Cara melakukan LINK ke bagian alamat tersebut adalah menggunakan

```
<a href="#atas(nama anchor)">kembali ke atas(teks pada browser)</a>
```

Macam-macam LINK:

- a. 1.Link ke page lain :
- b. 2.Link ke alamat URL :
- c. 3.Link ke e-mail :
- d. 4.Link ke halaman yang sama :
- e. 5.Link ke file yang akan di download :

```
<html>
<head>
<title>link</title>
</head>
<body>
<a name="atas">pemain-pemain timnas RI</a>
<blockquote>
<p>EGI SAPUTRA ,<a href="#info">info lengkap</a>
<p>KURNIA MEGA
<p>IRFAN BACHDIM
<p>BOAS SALOSA
<p>EFAN DIMAS
<p>
<a href="contoh_2.html">kalo mau tahu lagi klik disini</a>
</blockquote>
<p><a name="info">EGI SAPUTRA</a>
<blockquote>
<p>No punggung 10 timnas u-21
<br>dijuluki messi asia
<br>masih jomlo
</blockquote>
<p><a href="#atas">kembali ke atas</a>
</body>
</html>
```

1.4.9.1 Link tujuan dalam HTML :

```
<html>
<head>
<title>link tujuan</title>
</head>
<body>
<p align="center"> maaf, hanya sedikit
<br>chamot
<br>dida
```

```
<br>shevchenko  
<br><a href="praktik10.html">kembali</a>  
</body>  
</html>
```

1.5 Tugas

1. Buatlah halaman web berisi informasi inti dan lengkapi dengan link ke halaman berikutnya untuk penjabarannya.

BAB II

HTML DAN XHTML LANJUTAN

2.1 Image

Digunakan untuk menampilkan image atau animasi gif pada halaman web Anda. Atribut : alt, align=(center, left, right), hspace, vspace, border, width & height

Contoh :

```


```

2.1.1 Format Image

Ada banyak format image, tapi ada tiga jenis format yang paling sering digunakan :

GIF (Graphical Interchange Format) (.GIF)

JPEG (Joint Photographic Expert Image) (.JPG)

PNG (Portable Network Graphics)

2.1.2 Insert Image ke Document

Tag IMG di gunakan untuk menginsertkan image ke document HTML.

Syntax nya:

```
<IMG SRC = "URL">
<html>
<head>
<title>Image</title>
</head>
<body>

</body>
</html>
```

2.1.3 Attribute Value Description

Attribute	Value	Description
Align	Center justify left right Baseline top bottom middle	<ul style="list-style-type: none"> ▪ Top, bottom, middle digunakan untuk menentukan posisi image terhadap text ▪ Left, right, center untuk menentukan posisi image di document

```
<html>
<head>
<title>Image</title>
</head>
<body>
<p>
Default alignment at the bottom</p>
<p>Aligned at Top</p>
<p>Aligned at Middle</p>
</body>
</html>
```

2.1.4 Image Map

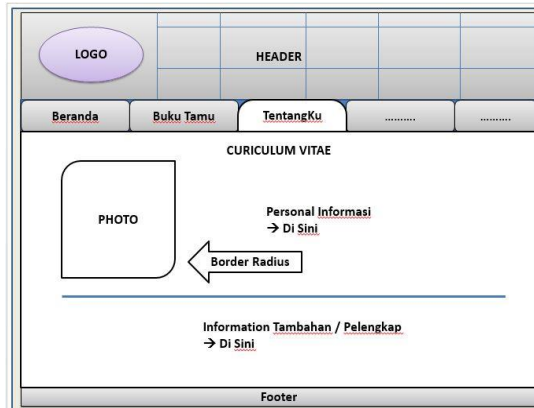
Anda bisa gunakan image yang ada pada website anda untuk membuat image map. Image map yaitu suatu area pada image yang bisa kita beri hyperlink area ini biasanya disebut “hot spots”. Koordinat dari hotspot ditentukan menggunakan bidang geometry seperti rectangle, polygon dan lainnya.

```
<html>
<head>
<title>Creating Hotspot</title>
</head>
<body>

<map name="Map">
<area shape="rect" coords="0,0,200,100" href="file1.htm"
alt="kotak link">
<area shape="circle" coords="354,41,36" href="file2.htm"
alt="lingkaran link">
<area shape="poly"
coords="58,102,97,101,110,134,119,119,177,195,69,196,
47,162,62,143" href="#">
</map>
</body>
</html>
```

2.2 Tugas

1. Buatlah halaman web seperti dibawah ini , dengan memasukkan foto pribadi sesuai dengan layout yang sudah ditentukan.



BAB III HTML DAN XHTML LANJUTAN

3.1 Tabel

Dalam pembuatan homepage, tabel memiliki fungsi yang tidak kalah penting dengan tag-tag HTML lainnya. Pertama tabel berfungsi untuk menampilkan informasi secara terstruktur, ringkas dan mudah dibaca, kedua berfungsi untuk mengatur tampilan homepage agar lebih menarik.

Tag utama untuk tabel yaitu `<TABLE>` `</TABLE>` dan untuk membuat judul tabel tag-nya table header `<TH>``</TH>` lalu untuk membuat data tabel atau isi tabel menggunakan tag table Data `<TD>``</TD>` dan untuk membuat baris adalah memakai tag Table Rows `<TR>` `</TR>`

Table dapat Anda gunakan karena dapat menampilkan informasi dalam bentuk yang ringkas dan mudah dibaca. Tag awal yang digunakan adalah `<TABLE>` dan tag penutupnya `</TABLE>`. Tag `<TABLE>` mewakili beberapa bagian penting :

- `<CAPTION>` ...`</CAPTION>` digunakan untuk membentuk judul table. Judul table akan terletak di luar table di bagian atas atau di bagian bawah.
- `<TH>`...`</TH>` digunakan untuk meletakkan judul table di bagian paling atas atau bagian paling kiri dari table.
- `<TD>` ...`</TD>` digunakan sebagai tempat menulis data atau informasi dalam table.

3.2 Latihan

3.2.1 Tabel sederhana

3.2.1.1 Tabel dasar

Bagaimana membuat tabel dengan jumlah satu baris dan satu kolom, berikut adalah contontonya.

```
<table>
  <tr>
    <td>&nbsp;</td>
  </tr>
</table>
```

Selanjutnya jika anda ingin menambahnya kolom pada baris yang sama ,anda cukup menambahkan tag `<td>` diantara tag `<tr>`, contoh


```
<table>
  <tr>
    <td>&nbsp;</td>
    <td>&nbsp;</td>
  </tr>
</table>
```

Selanjutnya jika anda ingin menambahnya jumlah barisnya, anda bisa menambahnya tag<tr></tr> dengan jumlah kolom<td> sama seperti baris sebelumnya. Contoh

```
<table>
  <tr>
    <td>&nbsp;</td>
  </tr>
  <tr>
    <td>&nbsp;</td>
  </tr>
</table>
```

3.2.1.2 Judul tabel

Untuk membuat judul atau header informasi yang akan ditampilkan dalam table kita dapat menggunakan tag <th>,<th>...</th> digunakan untuk meletakkan judul table di bagian paling atas atau bagian paling kiri dari table. Contoh

```
<table>
  <tr>
    <th>Kampus</th>
    <th>Alamat</th>
  </tr>
  <tr>
    <td>UTY</td>
    <td>Jombor</td>
  </tr>
</table>
```

Contoh diatas untuk menampilkan header secara horizontal, jika anda ingin menampilkan secara vertikal ,maka anda cukup mengganti tag kolom pada baris berikutnya menjadi tag header. Contoh

```
<table>
  <tr>
    <th>Kampus</th>
    <td>UTY</td>
  </tr>
  <tr>
    <th>Alamat</th>
    <td>Jombor</td>
  </tr>
</table>
```

3.2.2 Tabel lanjut

Istilah Merge atau menggabungkan baris atau kolom dalam tabel dapat kita lakukan menggunakan atribut span. Jika anda ingin menggabungkan beberapa kolom anda dapat menggunakan atribut colspan dan tentukan jumlah kolom yang akan digabungkan. Contoh

```
<table>
  <tr>
    <th>Kampus</th>
    <th colspan="2">Alamat</th>
  </tr>
  <tr>
    <td>UTY</td>
    <td>Jombor</td>
    <td>Jogja</td>
  </tr>
</table>
```

Selanjutnya jika anda ingin menggabungkan baris, anda dapat menggunakan atribut rowspan dan tentukan jumlah baris yang akan digabungkan.

Contoh

```
<table>
  <tr>
    <th>Kampus<th>
    <td>UTY </td>
  </tr>
  <tr>
    <th rowspan="2">Alamat</th>
    <td>Jombor</td>
  </tr>
  <tr>
    <td>Jogja</td>
  </tr>
</table>
```

3.2.3 Iframe

Iframe digunakan untuk menampilkan halaman web dalam halaman web.

Sintaks iframe

HTML iframe didefinisikan dengan tag <iframe>:

```
<iframe src = "URL"> </ iframe>
```

Atribut src menentukan URL (alamat web) dari halaman kerangka inline.

```
<html>
<body>
<iframe src="contoh.html" height="200" width="300"></iframe>
</body>
</html>
```

3.3 Tugas

Buatlah rancangan tabel seperti gambar berikut

1.

Universitas Teknologi Yogyakarta						
Fakultas	Teknik		Ekonomi		Bahasa	
Prodi	Managemen Informatika	Teknik Informatika	Managemen Ekonomi	Bisnis	Ingris	Jepang

2.

3.

BAB IV HTML DAN XHTML LANJUTAN

4.1 Form

Form biasanya digunakan untuk mengumpulkan informasi dari user, sehingga memungkinkan suatu web server untuk menerima informasi dari pemakai melalui sejumlah elemen yang disebut kontrol. Kontrol ini bisa berupa suatu textbox, checkbox, radio button, push button, list menu dan lainnya.

Standart penulisan form :

```
<form method="post/get" action=url">  
.....  
.....  
</form>
```

Atribut method memiliki dua nilai post dan get. Metode get mengirimkan data pada server dengan cara meletakkan data pada bagian akhir URL yang ditunjuk. Metode post mengirimkan datanya secara terpisah. Jika data masukan banyak, lebih disarankan untuk menggunakan metode post. Atribut action berisi URL dari program yang dipanggil oleh form tersebut.

4.1.1 Textbox

Textbox merupakan salah satu jenis kontrol untuk memasukan data. HTML menyediakan tag `<input>` dengan atribut `type="text"` untuk membuat kotak input. Atribut lain yaitu name untuk memberi nama input, atribut Value untuk memberi nilai suatu input dan atribut size untuk menentukan batas terpanjang sebuah masukan.

Dalam textbox kita juga dapat menyembunyikan masukan yang ditulis user dengan memberi nilai `type="password"`. Masukan akan menjadi karakter "*".

4.1.2 Checkbox

Checkbox digunakan untuk memberi beberapa pilihan kepada user. Dengan checkbox user dapat memilih salah satu, lebih dari satu pilihan atau tidak sama sekali memilih.

4.1.3 Radio

Pada type Radio, maka user memilih salah satu pilihan yang tersedia, sehingga user tidak bisa memilih kurang atau lebih dari satu pilihan. Atribut Checked memberi tanda bahwa pilihan tersebut sedang diaktifkan.

4.1.4 Submit dan Reset

Setiap form harus memiliki minimal tombol submit dan reset. Untuk membuatnya digunakan atribut type="submit" dan type="reset". Tombol submit digunakan ketika user mengisi formulir dan ingin mengirimkan ke server. Sedangkan reset digunakan ketika user ingin menghapus semua masukan yang ditulis.

4.2 Latihan

4.2.1 Membuat Form Input dengan Text box, Submit dan Reset

```
<table width="248" border="0">
  <tbody>
    <tr>
      <td colspan="3" align="center">Login</td>
    </tr>
    <tr>
      <td width="72">User ID</td>
      <td width="6">:</td>
      <td width="148">
        <input type="text" name="textfield" id="textfield"></td>
      </tr>
    <tr>
      <td>Password</td>
      <td>:</td>
      <td>
        <input type="password" name="password" id="password"></td>
      </tr>
    <tr>
      <td colspan="3" align="center"><input type="submit"
name="submit" id="submit" value="
  log-in">
      <input type="reset" name="reset" id="reset"
value="Reset"></td>
    </tr>
  </tbody>
</table>
```

4.2.2 Membuat Form Input dengan Check box

```
<html>
<head>
<title>form input dengan check box</title>
</head>
<body>
<p>pilih yang anda suka </p>
<p>daftar menu masakan</p>
<p>
  <input type="checkbox" name="c1" checked>ayam goreng<br>
  <input type="checkbox" name="c2">gado-gado<br>
  <input type="checkbox" name="c3">sate kambing<br>
  <input type="checkbox" name="c4">nasi goreng
</p>
</body>
</html>
```

4.2.3 Membuat Form Input dengan Radio box

```
<html>
<head>
<title>form input dengan radio box</title>
</head>
<body>
<p>pilih salah satu </p>
<p>usia anda</p>
<p>
  <input type="radio" value="v1" checked name="r1">< 15 tahun<br>
  <input type="radio" value="v2" name="r1">15-19 tahun<br>
  <input type="radio" value="v3" name="r1">19-25 tahun<br>
  <input type="radio" value="v4" name="r1">25-35 tahun<br>
  <input type="radio" value="v5" name="r1"> > 35 tahun<br>
</p>
</body>
</html>
```

4.2.4 Membuat Form Input dengan Textarea

```
<html>
<head>
<title>Textarea</title>
</head>
<body>
<form>
<textarea name="Alamat" rows="20" cols="30">
</textarea>
</form>
</body>
</html>
```

4.2.5 Membuat Form Input Dengan Select

```

<html>
<head>
<title>Select</title>
</head>
<body>
<form>AGAMA
<select name="AGAMA">
  <option value="ISLAM">ISLAM
  <option value="Katolik">KATOLIK
  <option value="Protestan">PROTESTAN
  <option value="Hindu">HINDU
  <option value="Budha">BUDHA
</option>
</form>
</body>
</html>

```

4.3 Tugas

- a. Buatlah desain form pendaftaran seperti gambar dibawah.

Form Registrasi	
User ID	: Teks
Nama	: Teks
Password	: ●●●●●●●●
Re-pass	: ●●●●●●●●
Alamat	: Teks Area
Jenis Kelamin	: <input type="radio"/> L <input type="radio"/> P
Agama	: <input type="text" value="isi 5 agama"/>
Hobi	: <input type="checkbox"/> makan <input type="checkbox"/> olahraga <input type="checkbox"/> lain-lain
<input type="button" value="Daftar"/> <input type="button" value="Ulangi"/>	

BAB V HTML5

5.1 Kompetensi Dasar

- a. Mahasiswa dapat mengenal tag dan fungsinya dalam membangun rancang awal sistem berbasis web.
- b. Mahasiswa dapat mengetahui perbedaan tag dari generasi HTML sebelumnya, hingga HTML5.

5.2 Indikator

- a. Mampu membangun rancang awal sistem berbasis web.

5.3 Uraian Materi

Deklarasi DOCTYPE untuk HTML5 sangat sederhana:

```
<! DOCTYPE html>
```

Deklarasi karakter pengkodean (charset) juga sangat sederhana:

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

Contoh HTML5:

```
<! DOCTYPE html>
<html>
<head>
<meta charset = "UTF-8">
<title> Judul dokumen </ title>
</ head>
<body>
Isi dokumen .....
</ body>
</ html>
```

Encoding karakter default pada HTML5 adalah UTF-8.

5.3.1 Elemen Baru HTML5

Elemen HTML5 yang paling menarik adalah:

- Elemen semantik baru seperti <header>, <footer>, <article>, dan <section>.
- Atribut baru berupa elemen seperti nomor, tanggal, waktu, kalender, dan jangkauan.
- Elemen grafis baru: <svg> dan <canvas>.
- Elemen multimedia baru: <audio> dan <video>.

Berikut adalah daftar elemen HTML5 yang baru, dan deskripsi tentang apa penggunaannya.

5.3.1.1 Elemen Semantik / Struktural Baru

HTML5 menawarkan elemen baru untuk struktur dokumen yang lebih baik:

<article>	Mendefinisikan sebuah artikel dalam sebuah dokumen
<aside>	Mendefinisikan konten selain konten halaman
<bdi>	Isolat bagian teks yang mungkin diformat dalam arah yang berbeda dari teks lain di luarnya
<details>	Mendefinisikan rincian tambahan yang dapat dilihat atau disembunyikan pengguna
<dialog>	Mendefinisikan sebuah kotak dialog atau jendela
<figcaption>	Mendefinisikan sebuah caption untuk elemen <figure>
<figure>	Mendefinisikan konten mandiri
<footer>	Mendefinisikan footer untuk dokumen atau bagian
<header>	Mendefinisikan sebuah header untuk dokumen atau bagian
<main>	Mendefinisikan konten utama sebuah dokumen
<mark>	Mendefinisikan teks yang ditandai / disorot
<menuitem>	Mendefinisikan sebuah perintah / item menu yang dapat dipanggil pengguna dari menu popup
<meter>	Mendefinisikan pengukuran skalar dalam rentang yang diketahui (alat ukur)
<nav>	Merupakan kemajuan sebuah tugas
<progress>	Mendefinisikan apa yang harus ditunjukkan di browser yang tidak mendukung anotasi ruby
<rp>	Mendefinisikan sebuah penjelasan / pengucapan karakter (untuk tipografi Asia Timur)
<rt>	Mendefinisikan sebuah penjelasan / pengucapan karakter (untuk tipografi Asia Timur)
<ruby>	Mendefinisikan anotasi ruby (untuk tipografi Asia Timur)
<section>	Mendefinisikan sebuah bagian dalam sebuah dokumen
<summary>	Mendefinisikan heading yang terlihat untuk elemen <details>
<time>	Mendefinisikan tanggal / waktu
<wbr>	Mendefinisikan kemungkinan jeda baris

5.3.1.2 Form Element Baru

<datalist>	Menentukan daftar opsi yang telah ditentukan untuk kontrol masukan
<output>	Mendefinisikan hasil perhitungan

5.3.1.3 Jenis Masukan Baru

New Input Types	New Input Attributes
color	autocomplete
date	autofocus
datetime	form
datetime-local	formaction
email	formenctype
month	formmethod
number	formnovalidate
range	formtarget
search	height and width
tel	list
time	min and max
url	multiple
week	pattern (regexp)
	placeholder
	required
	step

5.3.1.4 Sintaks Atribut Baru

HTML5 memungkinkan empat sintaks yang berbeda untuk atribut.

Contoh ini menunjukkan sintaks yang berbeda yang digunakan dalam tag <input>:

Type	Contoh
Empty	<input type="text" value="John" disabled>
Unquoted	<input type="text" value=John>

5.3.1.5 Grafik

Tag	Deskripsi
<canvas>	Menggambar grafis, on the fly, melalui scripting (biasanya JavaScript)
<svg>	Menggambar grafis vektor terukur

5.3.1.6 Elemen Media Baru

Tag	Deskripsi
<audio>	Mendefinisikan konten suara
<embed>	Mendefinisikan sebuah wadah untuk aplikasi eksternal (non-HTML)

5.3.2 API Baru HTML5

API baru yang paling menarik di HTML5 adalah:

- HTML Geolocation
- HTML Drag and Drop
- HTML Local Storage
- HTML Application Cache
- HTML Web Workers
- HTML SSE

Tip: HTML Local Storage adalah pengganti cookie yang hebat.

5.4 Latihan

5.4.1 Contoh kanvas

Kanvas adalah area persegi panjang pada halaman HTML. Secara default, kanvas tidak memiliki batas dan tidak ada konten.

Markup terlihat seperti ini:

```
<kanvas id = "myCanvas" width = "200" height = "100"> </ kanvas>
```

Catatan: Selalu tentukan atribut id (untuk disebut dalam skrip), dan atribut lebar dan tinggi untuk menentukan ukuran kanvas. Untuk menambahkan border, gunakan atribut style.

```
<!DOCTYPE html>
<html>
<body>
<p>Pilih gambar:</p>

<p>Mengisi :</p>
<canvas id="kanvas" width="250" height="300"
style="border:1px solid #d3d3d3;">
Your browser does not support the HTML5 canvas tag.</canvas>

<p><button onclick="kanvas()">Try it</button></p>
<script>
function kanvas() {
```

```

var c = document.getElementById("kanvas");
var ctx = c.getContext("2d");
var img = document.getElementById("gambar");
ctx.drawImage(img,10,10);
}
</script>

</body>
</html>

```

5.4.2 SVG

Apa itu SVG?

- SVG adalah singkatan dari Scalable Vector Graphics
- SVG digunakan untuk mendefinisikan grafis untuk Web
- SVG adalah rekomendasi W3C

Elemen HTML <svg>

Elemen HTML <svg> adalah wadah untuk grafis SVG.

SVG memiliki beberapa metode untuk menggambar jalur, kotak, lingkaran, teks, dan gambar grafis.

5.4.2.1 SVG Circle

```

<!DOCTYPE html>
<html>
<body>
<svg width="100" height="100">
  <circle cx="50" cy="50" r="40"
    stroke="red" stroke-width="4" fill="blue" />
</svg>
</body>
</html>

```

5.4.2.2 SVG Rectangle

```

<!DOCTYPE html>
<html>
<body>
<svg width="400" height="100">
  <rect width="400" height="100"
    style="fill:rgb(255,0,255);stroke-width:10;stroke:rgb(0,255,0)"
  />
</body>
</html>

```

5.4.2.3 SVG Rounded Rectangle

```
<!DOCTYPE html>
<html>
<body>
<svg width="400" height="400">
  <rect x="50" y="20" rx="20" ry="20" width="300" height="300"
    style="fill:red;stroke:black;stroke-width:5;opacity:0.5" />
</svg>
</body>
</html>
```

5.4.2.4 SVG Star

```
<!DOCTYPE html>
<html>
<body>
<svg width="300" height="200">
  <polygon points="100,10 40,198 190,78 10,78 160,198"
    style="
    fill:black;
    stroke:red;
    stroke-width:5;
    fill-rule:evenodd;" />
Sorry, your browser does not support inline SVG.
</svg>
</body>
</html>
```

5.4.2.5 SVG Logo

```
<!DOCTYPE html>
<html>
<body>
<svg height="130" width="500">
  <defs>
    <linearGradient id="grad1" x1="0%" y1="0%" x2="100%" y2="0%">
      <stop offset="0%"
        style="stop-color:rgb(255,255,0);stop-opacity:1" />
      <stop offset="100%"
        style="stop-color:rgb(255,0,255);stop-opacity:1" />
    </linearGradient>
  </defs>
  <ellipse cx="100" cy="70" rx="85" ry="55" fill="url(#grad1)" />
  <text fill="green" font-size="45" font-family="arial"
    x="50" y="86">UTY</text>
</svg>

</body>
</html>
```

5.5 Tugas

1. Buatlah logo seperti gambar dibawah ini menggunakan SVG



BAB VI HTML5 LANJUTAN

6.1.1 Maps

HTML5 mendukung untuk dapat menampilkan peta yang disediakan oleh pihak ketiga yaitu google. peta sangat penting berkaitan dengan navigasi yang dapat memungkinkan kemudahan dalam pencarian lokasi dan pastinya akan memiliki nilai estetika lebih . Contoh

```
Google Maps allows you to display maps on your web page:
<!DOCTYPE html>
<html>
<body>
<h1>Google Map pertama</h1>
<div id="peta" style="width:100%;height:400px;"></div>
<script>
function myMap() {
var mapProp= {
    center:new google.maps.LatLng(51.508742,-0.120850),
    zoom:5,
};
var map=new
google.maps.Map(document.getElementById("peta"),mapProp);
}
</script>
<script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBu-
916DdpKAjTmJNigngS6HL_kDIKU0aU&callback=myMap"></script>
</body>
</html>
```

6.1.2 Video

```
<!DOCTYPE html>
<html>
<body>
<video width="400" controls>
  <source src=".....mp4" type="video/mp4">
</video>
</body>
</html>
```

6.1.3 Audio

```
<!DOCTYPE html>
<html>
<body>
<audio controls>
  <source src=".....ogg" type="audio/ogg">
```

```

<source src="...mp3" type="audio/mpeg">
</audio>
</body>
</html>

```

6.1.4 Youtube

- Memainkan Video YouTube dalam HTML
- Untuk memutar video Anda di halaman web, lakukan hal berikut:
- Upload video ke YouTube
- Ambil catatan id video
- Tentukan elemen <iframe> di halaman web Anda
- Biarkan atribut src mengarah ke URL video
- Gunakan atribut lebar dan tinggi untuk menentukan dimensi pemutar
- Tambahkan parameter lainnya ke URL

```

<!DOCTYPE html>
<html>
<body>
<iframe width="420" height="345"
src="https://www.youtube.com/embed/XGSy3_Czz8k?autoplay=0">
</iframe>
</body>
</html>

```

6.1.5 Plug-in

6.1.5.1 Elemen <object>

Elemen <object> didukung oleh semua browser. Elemen <object> mendefinisikan objek yang disematkan dalam dokumen HTML.

Ini digunakan untuk menanamkan plug-in (seperti applet Java, pembaca PDF, Flash Players) di halaman web.

```

<!DOCTYPE html>
<html>
<body>
<object width="400" height="50"
data="****.swf/.html/.jpeg/dll"></object>
</body>
</html>

```


6.1.5.2 Elemen <embed>

Elemen <embed> didukung di semua browser utama. Elemen <embed> juga mendefinisikan objek yang disematkan dalam dokumen HTML.

Browser web telah mendukung elemen <embed> untuk waktu yang lama. Namun, itu belum menjadi bagian dari spesifikasi HTML sebelum HTML5.

```
<!DOCTYPE html>
<html>
<body>
<embed width="400" height="50" src="*****.swf">
</body>
</html>
```

6.2 Tugas

1. Buatlah Desain web seperti tugas dibab-bab sebelumnya dan lengkapi dengan menu audio,vedio dan maps.

BAB VII

CSS

7.1 Kompetensi Dasar

- a. Mahasiswa dapat mengenal tag dan fungsi CSS dalam membangun rancang awal sistem berbasis web sehingga dapat membuatnya lebih dinamis dari segi desain.

7.2 Indikator

- a. Mahasiswa mampu mendesain web lebih baik dan lebih dinamis.

7.3 Uraian Materi

Style Sheets merupakan feature yang sangat penting dalam membuat Dynamic HTML. Meskipun bukan merupakan suatu keharusan dalam membuat web, akan tetapi penggunaan style sheets merupakan kelebihan tersendiri.

Suatu style sheet merupakan tempat dimana anda mengontrol dan memanager style-style yang ada. Style sheet mendeskripsikan bagaimana tampilan document HTML di layar. Anda juga bias menyebutnya sebagai template dari documents HTML yang menggunakannya. Anda juga bisa membuat efek-efek spesial di web anda dengan menggunakan style sheet. Sebagai contoh anda bisa membuat style sheet yang mendefinisikan style untuk <H1> dengan style bold dan italic dan berwarna biru. Atau pada tag <P> yang akan di tampilkan dengan warna kuning dan menggunakan font verdana dan masih banyak lagi yang bisa anda lakukan dengan style sheet.

Secara teoritis anda bisa menggunakan style sheet technology dengan HTML. Akan tetapi pada prakteknya hanya Cascading Style Sheet (CSS) technology yang support pada hampir semua web Browser. Karena CSS telah disetandardkan oleh World Wide Web Consortium (W3C) untuk di gunakan di web browser.

7.3.1 Inline Styles

Ada dua cara untuk merubah style dari web page anda yaitu dengan :

- Merubah inline style
- Menulis script untuk merubah style anda.

Dengan menggunakan inline style anda dapat membuat dynamic style tanpa harus menambahkan script ke web anda. Inline styles merupakan style yang bisa kita pasang pada element web tertentu saja.

Contoh:

Jika anda ingin menambahkan style pada <H1> dengan warna merah, anda harus mengeset atribut STYLE dari tag <H1>.

```
<H1 STYLE="color:red">
```

Jika anda ingin menggunakan script untuk memodifikasi inline style, anda dapat menggunakan Style Object. Style Object mensupport semua property yang di support CSS untuk style. Untuk menggunakan property pada script :

1. Hilangkan tanda hubung "-" dari property CSS Style
2. Ganti huruf setelah tanda hubung menjadi Capital.

Contoh:

font-weight menjadi font**Weight**
text-align menjadi text**Align**

7.3.2 Istilah-istilah dalam Style Sheet

7.3.2.1 Style rule

Cascading style sheet merupakan kumpulan aturan yang mendefinisikan style dari document. Sebagai contoh kita bisa membuat aturan style yang menentukan bahwa semua <H2> di tampilkan dengan warna orange.

7.3.2.2 Style sheet

Style sheet dapat di embedded ke HTML document. Atau disebut embedded style sheet. Style sheet juga bisa dibuat sebagai external file dan di link ke document HTML. Style role bisa di kenakan pada bagian tertentu dari web page. Sebagai contoh anda bisa menentukan paragraph tertentu di tampilkan dengan style bold dan italic sementara yang lain tetap seperti biasa.

7.3.2.3 Selector

```
selector { property1: value; property2:value, . . . }  
H1 { color:green; background-color:orange }
```

Style sheets terdiri dari dua bagian:

1. Selector

Bagian pertama sebelum tanda "{}" disebut selector

2. Declaration

Terdiri dari property dan nilainya.

7.3.3 Komentar dalam Style Sheets

Comments atau komentar biasanya di gunakan oleh programmer untuk memudahkan mengingat kembali script yang sudah di tulisnya, Comments di CSS hampir sama dengan comments di C atau C++ yaitu dengan menggunakan:

```
/* isi Comments */
```

Contoh:

```
H1 { color:blue;} /* H1 elements akan menjadi biru */
Tags.H1.color = "blue"; /* H1 elements akan menjadi biru */
```

7.3.4 Pemakaian elemen style

Mari kita mulai dengan mengatur warna huruf dan latar belakang. Anda dapat mengerjakan ini dengan menggunakan elemen style untuk mengatur karakter kode tag dokumen anda.

```
<style type="text/css">
body { color: black; background: white; }
</style>
```

Pernyataan yang ditulis antara kode tag <style> dan </style> menunjukkan perintah pengaturan style.

7.3.5 Link ke sheet lainnya

Apabila anda menginginkan style yang sama untuk halaman HTML anda yang lain, disarankan mempergunakan sheet-sheet terpisah namun satu dan lainnya terhubung dengan cara link. Anda dapat mengikuti cara berikut ini :

```
<link rel="stylesheet" href="style.css">
```

Kode tag untuk link ini ditempatkan di bagian "head" dokumen anda. Perintah rel perlu diatur dengan pernyataan "stylesheet" agar supaya browser dapat menemukan perintah href sebagai penunjukan ke alamat Web (URL) sheet anda.

7.4 Latihan

7.4.1 Mengatur tepi halaman (page margin)

Halaman Web akan tampil cantik bila dituliskan dalam margin yang lebih lebar. Anda dapat mengatur sisi kiri dan kanan memakai karakter "margin-left" dan "margin-right". Contoh :

```
<style type="text/css">
body { margin-left: 10%; margin-right: 10%; }
</style>
```

Perintah di atas dituisikan dengan tujuan agar tampilan halaman Web dilayar monitor memiliki batas halaman kiri 10% dari lebar layar monitor.

7.4.2 Mengatur inden kiri dan kanan

Agar halaman Web anda tampil lebih cantik bisa juga diberikan inden(spasi) dari margin kiri beberapa huruf sebelum menuliskan awal kalimat.

Contoh :

```
<style type="text/css">
body { margin-left: 10%; margin-right: 10%; }
h1 { margin-left: -8%;}
h2,h3,h4,h5,h6 { margin-left: -4%; }
</style>
```

7.4.3 Mengatur jarak penulisan dari tepi atas dan bawah halaman

Program Browser biasanya mengerjakan batas atas dan bawah, paragraf dan lain-lain dengan baik. Namun ketika anda ingin membuat ruang disebelah atas atau bawah halaman web anda, atau anda ingin membuat spasi yang khusus, barulah disini anda perlu memiliki cara mengaturnya. Property "margin-top" menentukan ruang sebelah atas dan property "margin-bottom" menentukan ruang sebelah bawah halaman web anda. Bila anda hendak mengatur semuanya dengan heading h2, anda cukup menuliskan dengan perintah HTML sebagai berikut :

```
h2 { margin-top: 8em; margin-bottom: 3em; }
```

Kode em merupakan unit penting dalam mengatur ukuran tinggi font (huruf). Ini lebih mudah bila dibandingkan dengan pengaturan pixel atau titik (point). Unit ini akan sangat berguna pada pembuatan huruf besar.

Satuan titik (Point) umumnya dipergunakan oleh program word processor misalnya dituliskan ukuran huruf 10 pt. Sayangnya untuk ukuran titik yang sama, menghasilkan ukuran huruf yang berbeda pada pemakaian program browser yang berbeda pula. Apa yang anda kerjakan dengan baik dengan menggunakan sebuah program browser, bila dibaca dengan program browser yang lain belum tentu baik !. Pergunakanlah kode em untuk mengatasi hal ini.

Untuk mengatur ruang sebelah atas bagian heading halaman web, anda sebaiknya membuat nama style untuk heading tersebut. Dalam penulisan HTML-nya anda cukup menggunakan atribut class.

Contoh :

```
<h2 class="subsection">Getting started</h2>
```

Kemudian pengaturan ruangnya ditulis dengan perintah berikut :

```
h2.subsection { margin-top: 8em; margin-bottom:3em; }
```

Pengaturan ini dimulai dengan nama tag, sebuah titik dan kemudian nilai dari atribut class. Hati-hati dalam menempatkan ruang sebelum atau sesudah titik tersebut. Bila pengaturan tersebut tidak memberikan hasil. Ada cara lain untuk mengatur style elemen tertentu. Tapi atribut class tetap bersifat fleksibel. Pada saat sebuah "heading" diikuti dengan sebuah paragraf, nilai untuk batas bawah (margin-bottom) untuk heading tersebut tidak ditambahkan dengan nilai batas atas (margin-top) paragraf.

7.4.4 Inden pada baris pertama

Kadang-kadang anda ingin membuat inden pada baris pertama tiap paragraf. Cara berikut dapat anda lakukan.

```
p { text-indent: 2em; margin-top: 0; marginbottom: 0; }
```

Cara tersebut akan membuat inden pada baris pertama paragraf sejauh 2 em dan memberikan jarak antar paragraf.

7.5 Tugas

1. Buatlah paragraf seperti dibawah ini dengan pengaturan jarak penulisan, warna, tepi halaman, menggunakan css

Uraian Materi

Style Sheets merupakan feature yang sangat penting dalam membuat Dynamic **HTML**. Meskipun bukan merupakan suatu keharusan dalam membuat web, akan tetapi penggunaan style sheets merupakan kelebihan tersendiri.

Suatu style sheet merupakan tempat dimana anda mengontrol dan memanager style-style yang ada.

Style sheet mendeskripsikan bagaimana tampilan document **HTML** di layar. Anda juga bias menyebutnya sebagai template dari documents **HTML** yang menggunakannya. Anda juga bisa membuat efek-efek spesial di web anda dengan menggunakan style sheet. [style sheet](#).

BAB VIII CSS LANJUTAN

8.1 Mengatur Format Font.

8.1.1 Model huruf

Model yang umum dipakai adalah teks miring (italik) atau tebal (bold). Umumnya program browser mempergunakan tag `em` untuk huruf italik dan tag `strong` untuk huruf tebal. Misalnya anda ingin menuliskan kode `em` agar huruf tampil berbentuk italik dan tebal dan menuliskan kode `strong` untuk huruf tebal uppercase, perintahnya dituliskan sebagai berikut :

```
em { font-style: italic; font-weight: bold; }
strong { text-transform: uppercase; font-weight: bold; }
```

bila gagal dapat ditambahkan perintah ini :

```
h2 { text-transform: lowercase; }
```

8.1.2 Mengatur ukuran huruf

Kebanyakan program browser mempergunakan huruf yang lebih besar untuk heading yang penting sifatnya. Bila anda menimpa ukuran defaultnya, anda menempuh resiko yaitu huruf menjadi tampak lebih kecil khususnya bila anda mempergunakan ukuran yang anda tambahkan tersebut dengan ukuran titik (point). Karenanya anda disarankan untuk melakukan pengaturan ukuran huruf dengan ukuran yang sama. Contoh berikut mengatur ukuran heading dalam persen relatif terhadap ukuran teks normal.

```
h1 { font-size: 200%; }
h2 { font-size: 150%; }
h3 { font-size: 100%; }
```

8.1.3 Mengatur jenis huruf

Bisa saja jenis huruf favorit anda tidak bisa ditampilkan oleh berbagai jenis browser. Untuk mengatasi hal ini anda dapat menuliskan beberapa jenis huruf yang tidak dapat ditampilkan oleh hampir semua browser. Ada beberapa jenis huruf generik yang dijamin cocok, sehingga anda kami sarankan untuk mengakhiri daftar

perintah HTML anda dengan salah satu jenis huruf berikut : serif, sans-serif, cursive, fantasy, atau monospace, contoh :

```
body { font-family: Verdana, sans-serif; }
h1,h2 { font-family: Garamond, Times New Roman, serif; }
```

Dalam contoh ini heading penting akan ditampilkan dalam bentuk huruf Garamond, bila gagal maka akan ditampilkan dalam bentuk Times New Roman, dan bila juga masih tidak dapat tampil, maka akan tampil sebagai huruf serif. Teks paragraf akan ditampilkan dengan huruf Verdana atau bila masih tidak tampil juga, maka browser masih dapat menampilkannya dengan jenis huruf standar sans-serif.

8.1.4 Cara menghindari masalah huruf dan batas tepi halaman web

Pertama pergunakan elemen p untuk mencegah teks pada body tulisan yang tidak dapat ditampilkan dengan baik. Contoh :

```
<h2>Spring in Wiltshire</h2>
Blossom on the trees, bird song and the sound of
lambs bleating in the fields.
```

Teks yang ditulis mengikuti sebuah heading dapat menimbulkan akibat tampilan jenis huruf yang berbeda pada beberapa jenis program browser. Anda disarankan segera menutup teks pada paragraf tersebut. Contoh :

```
<h2>Spring in Wiltshire</h2>
<p>Blossom on the trees, bird song and the sound of lambs bleating
in the fields.</p>
```

Kedua pergunakan selalu elemen pre ketika anda menuliskan pengaturan jenis huruf yang anda pergunakan.

```
pre { font-family: monospace; }
```

Ketiga pergunakan elemen p dan ul pada waktu mengatur jenis huruf untuk heading. Khususnya ketika anda melakukan pengaturan border atau warna halaman web anda dengan elemen div. Beberapa jenis program browser tidak dapat melakukan pengaturan huruf dengan baik dan cenderung lupa sehingga huruf heading anda tampak menjadi huruf standar saja. Untuk menghindari hal ini anda dapat menuliskan perintah HTML sebagai berikut :

```
h1,h2,h3,h4,h5,p,ul { font-family: sans-serif; }
```


8.1.5 Menambahkan border dan latar belakang.

Anda dapat dengan mudah menambahkan border disekitar heading, daftar (list), atau paragraf atau sekelompok heading, list dan paragraf secara tertutup dengan mempergunakan elemen div. Contoh :

```
div.box{ border: solid; border-width: thin; width:
100% }
```

Catatan : tanpa property "width" beberapa browser akan menempatkan tepi kanan terlalu jauh ke arah kanan. Untuk mencegah hal ini anda dapat menuliskan perintah HTML-nya sebagai berikut :

```
<div class="box"> The content within this DIV
element will be enclosed in a box with a thin line around it.
</div>
```

Ada sedikit jenis border yaitu : dotted, dashed, solid, double, groove, ridge, inset dan outset. Lebar border diatur dengan mempergunakan property "border-width". Nilai dari property ini yaitu thin, medium dan thick yang tampak setipis ukuran 0.1em. Property "border-color" memungkinkan anda mengatur warna. Sebuah efek yang cantik dapat dilakukan dengan memberikan warna latar belakang kotak dengan warna tebal (solid color) atau dengan hamburan gambar ("tile image"). Untuk melakukan ini anda perlu mempergunakan property "background". Anda dapat mengikuti perintah berikut ini.

```
div.color {
background: rgb(204,204,255);
padding: 0.5em;
border: none;
}
```

Tanpa pengaturan property border, biasanya program browser hanya akan menampilkan warna standar saja. Property padding memberikan beberapa ruangan diantara tepi-tepi daerah berrwarna dan teks yang ada didalamnya. Anda dapat mengatur nilai property pading dengan menambahkan padding-left, padding-top, padding-right dan padding-bottom. Pengaturan ini dituliskan misalnya sebagai :

```
padding-left: 1em.
```

Anggaplah anda menginginkan border pada satu sisi halaman web saja. Anda dapat melakukan pengontrolan tiap sisi border dengan memberikan keterangan border-left, border-top, border-right dan border-bottom.

Contoh :

```
p.changed {
padding-left: 0.2em;
```

```
border-left: solid;
border-right: none;
border-top: none;
border-bottom: none;
border-left-width: thin;
border-color: red;
}
```

Susunan perintah diatas memberikan efek pada tampilan web anda dengan border berwarna merah disisi sebelah kiri.

8.1.6 Mengatur warna

Contoh berikut adalah perintah pengaturan warna.

```
body {
color: black;
background: white;
}
strong { color: red }
```

Model pengaturan diatas memberikan warna hitam teks (default) dan latar belakang putih, tetapi memiliki elemen strong pada warna merah. Ada 16 buah nama warna standar yang dijelaskan dalam uraian berikut. Selain itu anda dapat juga mempergunakan nilai desimal untuk warna merah, hijau dan biru, dan masing-masing memiliki interval antara 0 sampai 255. Misalnya rgb(255, 0, 0) akan memberikan warna merah di layar monitor. Anda juga dapat mempergunakan angka hexadesimal yang dimulai karakter # yang diikuti enam angka hexadesimal sebagai pengaturan warna. Sebuah pengubah juga diberikan dibawah ini agar anda dapat melakukan pengubahan nilai dari RGB ke nilai hexadesimal.

8.1.6.1 Mengatur warna link

Anda dapat mempergunakan CSS untuk mengatur warna hypertext link, dengan warna yang berbeda untuk link yang belum pernah anda akses, link yang pernah anda akses dan link yang akan kemudian anda akses serta link yang aktif. Anda bahkan dapat mengatur warnanya pada saat kursor mouse berada diatas daerah yang akan dilink.

Hal ini dapat anda tuliskan dalam bentuk perintah berikut :

```
a:link {color: green;} /* unvisited link */
a:visited {color: red;} /* visited link */
a:hover {color: yellow;} /* mouse over link */
a:active {color: blue;} /* selected link */
```

Kadang-kadang anda ingin memperlihatkan link hypertexts tanpa garis bawah. Anda dapat melakukan ini dengan memberikan property `text-decoration` atau `none`, contoh :

```
a.plain { text-decoration: none }
```

Contoh berikut juga menampilkan link yang tidak bergaris bawah.

```
This is <a class="plain" href="what.html">not underlined</a>
```

Kebanyakan orang ketika mereka melihat garis bawah dibawah sebuah link selalu mengira itu adalah bagian teks yang diberi link. Umumnya orang memberikan warna biru pada teks yang diberi link ke halaman atau alamat internet lain. Anda disarankan untuk meninggalkan warna link ini apabila warna latar belakang menyebabkan teks anda yang diberi link jadi sulit terbaca.

8.1.7 Bar navigasi

Memiliki navigasi yang mudah digunakan penting untuk situs web manapun. Dengan CSS anda bisa mengubah menu HTML yang membosankan menjadi bar navigasi yang bagus.

Navigasi Bar = Daftar Tautan

Sebuah bar navigasi membutuhkan HTML standar sebagai basis.

8.1.7.1 Contoh Bar Navigasi Vertikal

Buat bilah navigasi vertikal dasar dengan warna latar abu-abu dan ubah warna latar belakang tautan saat pengguna mengarahkan mouse ke atasnya:

```
<html>
<head>
<style>
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  width: 200px;
  background-color: #f1f1f1;
}
li a {
  display: block;
  color: #000;
  padding: 8px 16px;
  text-decoration: none;
}
/* rubah warna link -> on hover */
li a:hover {
  background-color: #555;
```

```

        color: white;
    }
</style>
</head>
<body>
<h2>Bar Navigasi Vertical</h2>
<ul>
  <li><a href="#home">Home</a></li>
  <li><a href="#news">News</a></li>
  <li><a href="#contact">Contact</a></li>
  <li><a href="#about">About</a></li>
</ul>
</body>
</html>

```

8.1.7.2 Contoh Bar Navigasi Horizontal

Buat bilah navigasi horizontal dasar dengan warna latar belakang gelap dan ubah warna latar belakang tautan saat pengguna mengarahkan mouse ke atasnya:

```

<html>
<head>
<style>
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #333;
}
li {
  float: left;
}
li a {
  display: block;
  color: white;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}
li a:hover:not(.active) {
  background-color: #111;
}
.active {
  background-color: #4CAF50;
}
</style>
</head>
<body>
<ul>
  <li><a href="#home">Home</a></li>
  <li><a href="#news">Artikel</a></li>
  <li><a href="#contact">Download</a></li>

```

```
<li style="float:right"><a class="active"
href="#about">kontak</a></li>
</ul>
</body>
</html>
```

8.2 Tugas

Buat halaman web dengan menggunakan navigasi bar seperti pada gambar dibawah ini.



BAB IX

CSS3

9.1 Kompetensi Dasar

- a. Mahasiswa dapat mengenal tag dan fungsi CSS dalam membangun rancang awal sistem berbasis web sehingga dapat membuatnya lebih dinamis dari segi desain dan lebih memiliki nilai estetika.

9.2 Indikator

- a. Mahasiswa dapat lebih imajinatif dalam mendesain tatap muka sistem berbasis web.

9.3 Uraian Materi

CSS3 adalah standar terbaru untuk CSS. CSS3 benar-benar kompatibel dengan versi CSS sebelumnya.

Bagian ini mengajarkan Anda tentang fitur baru di CSS3!

Modul CSS3

CSS3 telah dipecah menjadi "beberapa modul". Ini berisi "spesifikasi CSS lama" (yang telah dipecah menjadi potongan yang lebih kecil). Selain itu, modul baru ditambahkan.

Beberapa modul CSS3 yang paling penting adalah:

- Selectors
- Box Model
- Backgrounds and Borders
- Image Values and Replaced Content
- Text Effects
- 2D/3D Transformations
- Animations
- Multiple Column Layout
- User Interface

Sebagian besar properti CSS3 baru diimplementasikan di browser modern.

9.4 Latihan

9.4.1 Border-radius CSS3

Menentukan Setiap Sudut

Jika Anda hanya menentukan satu nilai untuk properti radius perbatasan, radius ini akan diterapkan ke 4 sudut. Namun, Anda bisa menentukan masing-masing sudut secara terpisah jika Anda mau. Berikut aturannya:

1. **Empat nilai:** nilai pertama berlaku untuk nilai kiri atas, nilai kedua berlaku untuk kanan atas, nilai ketiga berlaku untuk kanan bawah, dan nilai keempat berlaku untuk pojok kiri bawah.
2. **Tiga nilai:** nilai pertama berlaku untuk kiri atas, nilai kedua berlaku untuk kanan atas dan kiri bawah, dan nilai ketiga berlaku untuk kanan bawah.
3. **Dua nilai:** nilai pertama berlaku untuk pojok kiri atas dan kanan bawah, dan nilai kedua berlaku untuk pojok kanan atas dan kiri bawah.
4. **Satu nilai:** keempat penjuru dibulatkan sama.

Berikut adalah tiga contohnya:

```
<html>
<head>
<style>
#rcorner4 {
    border-radius: 15px 50px 30px 5px;
    background: #73AD21;
    padding: 20px;
    width: 200px;
    height: 150px;
}

#rcorners3 {
    border-radius: 15px 50px 30px;
    background: #73AD21;
    padding: 20px;
    width: 200px;
    height: 150px;
}

#rcorners2 {
    border-radius: 15px 50px;
    background: #73AD21;
    padding: 20px;
    width: 200px;
    height: 150px;
}
</style>
</head>
<body>
```

```

<p>4 Nilai - border-radius: 15px 50px 30px 5px:</p>
<p id="rcorner4"></p>

<p>3 Nilai - border-radius: 15px 50px 30px:</p>
<p id="rcorners3"></p>

<p>2 Nilai - border-radius: 15px 50px:</p>
<p id="rcorners2"></p>

</body>
</html>

```

9.4.2 Backgrounds CSS3

9.4.2.1 Multiple backgrounds

CSS3 memungkinkan Anda menambahkan beberapa gambar latar belakang untuk sebuah elemen, melalui properti gambar latar belakang.

Gambar latar belakang yang berbeda dipisahkan dengan koma, dan gambar ditumpuk di atas satu sama lain, di mana gambar pertama paling dekat dengan penampil. Contoh berikut memiliki dua gambar latar belakang, gambar pertama adalah bunga (sejajar dengan bagian bawah dan kanan) dan gambar kedua adalah latar belakang kertas (sejajar dengan pojok kiri atas):

```

<html>
<head>
<style>
#contoh1 {
    background-image: url(bunga.gif), url(kertas.gif);
    background-position: right bottom, left top;
    background-repeat: no-repeat, repeat;
    padding: 15px;
}
</style>
</head>
<body>

<div id="contoh1">
<h1>Multiple Background</h1>
<p>Gambar latar belakang yang berbeda dipisahkan dengan koma, dan gambar ditumpuk di atas satu sama lain, di mana gambar pertama paling dekat dengan penampil. Contoh berikut memiliki dua gambar latar belakang, gambar pertama adalah bunga (sejajar dengan bagian bawah dan kanan) dan gambar kedua adalah latar belakang kertas (sejajar dengan pojok kiri atas)</p>
</div>

</body>
</html>

```


9.4.2.2 Background Size

Properti ukuran latar belakang CSS3 memungkinkan Anda menentukan ukuran gambar latar belakang.

Sebelum CSS3, ukuran gambar latar belakang adalah ukuran sebenarnya dari gambar. CSS3 memungkinkan kita untuk menggunakan kembali gambar latar belakang dalam konteks yang berbeda.

Ukuran dapat ditentukan dalam panjang, persentase, atau dengan menggunakan salah satu dari dua kata kunci: berisi atau menutupi.

Contoh berikut mengubah ukuran gambar latar menjadi lebih kecil dari gambar asli (menggunakan piksel):

```
<html>
<head>
<style>
#contoh1 {
  background-image: url(bunga.gif), url(kertas.gif);
  background-position: right bottom, left top;
  background-repeat: no-repeat, repeat;
  background-size: 100px 80px;
  padding: 15px;
}
</style>
</head>
<body>

<div id="contoh1">
<h1>Multiple Background</h1>
<p>Gambar latar belakang yang berbeda dipisahkan dengan koma, dan gambar ditumpuk di atas satu sama lain, di mana gambar pertama paling dekat dengan penampil. Contoh berikut memiliki dua gambar latar belakang, gambar pertama adalah bunga (sejajar dengan bagian bawah dan kanan) dan gambar kedua adalah latar belakang kertas (sejajar dengan pojok kiri atas)</p>
</div>

</body>
</html>
```

9.4.2.3 Tentukan Ukuran Multiple Background Images

Properti ukuran latar belakang juga menerima beberapa nilai untuk ukuran latar belakang (menggunakan daftar yang dipisahkan koma), saat bekerja dengan banyak latar belakang.

Contoh berikut memiliki tiga gambar latar belakang yang ditentukan, dengan nilai latar belakang yang berbeda untuk setiap gambar:

```
<html>
<head>
<style>
```

```

#contoh1 {
  background: url(bunga.gif) left top no-repeat, url(bunga.gif)
right bottom no-repeat, url(kertas.gif) left top repeat;
  padding: 15px;
  background-size: 50px, 130px, auto;
}
</style>
</head>
<body>

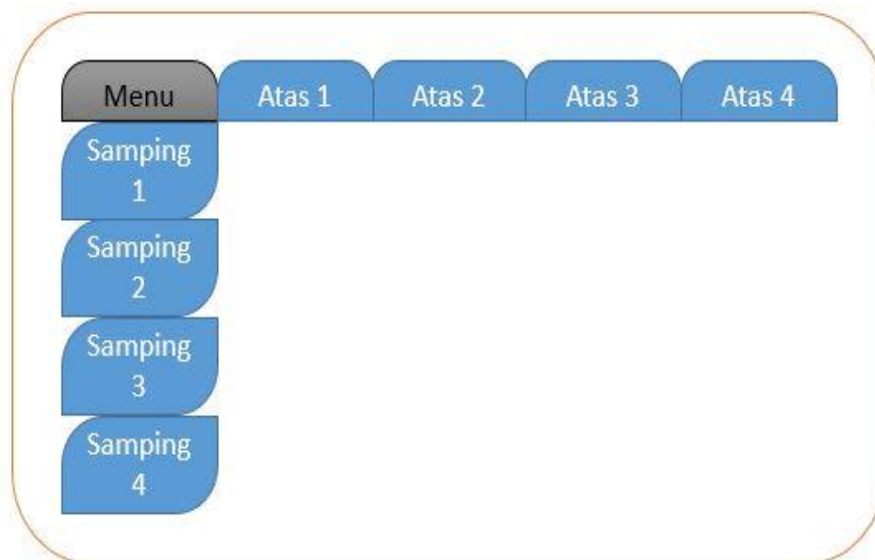
<div id="contoh1">
<h1>Multiple Background</h1>
<p>Gambar latar belakang yang berbeda dipisahkan dengan koma, dan
gambar ditumpuk di atas satu sama lain, di mana gambar pertama
paling dekat dengan penampil. Contoh berikut memiliki dua gambar
latar belakang, gambar pertama adalah bunga (sejajar dengan bagian
bawah dan kanan) dan gambar kedua adalah latar belakang kertas
(sejajar dengan pojok kiri atas)</p>
</div>

</body>
</html>

```

9.5 Tugas

1. Buatlah halaman web dengan susunan menu seperti gambar dibawah, titik lengkungan menggunakan border radius.



BAB X CSS3 LANJUTAN

10.1 Warna CSS3

CSS mendukung nama warna, warna heksadesimal dan RGB.

Selain itu, CSS3 juga memperkenalkan:

- Warna RGBA
- Warna HSL
- Warna HSLA
- Kegelapan/opacity

10.1.1 Warna RGBA

Nilai warna RGBA merupakan perpanjangan dari nilai warna RGB dengan saluran alfa - yang menentukan keburaman untuk warna. Nilai warna RGBA ditentukan dengan: `rgba` (merah, hijau, biru, alfa). Parameter alpha adalah angka antara 0.0 (sepenuhnya transparan) dan 1.0 (buram penuh).

```
rgba(255, 0, 0, 0.2);
```

```
rgba(255, 0, 0, 0.4);
```

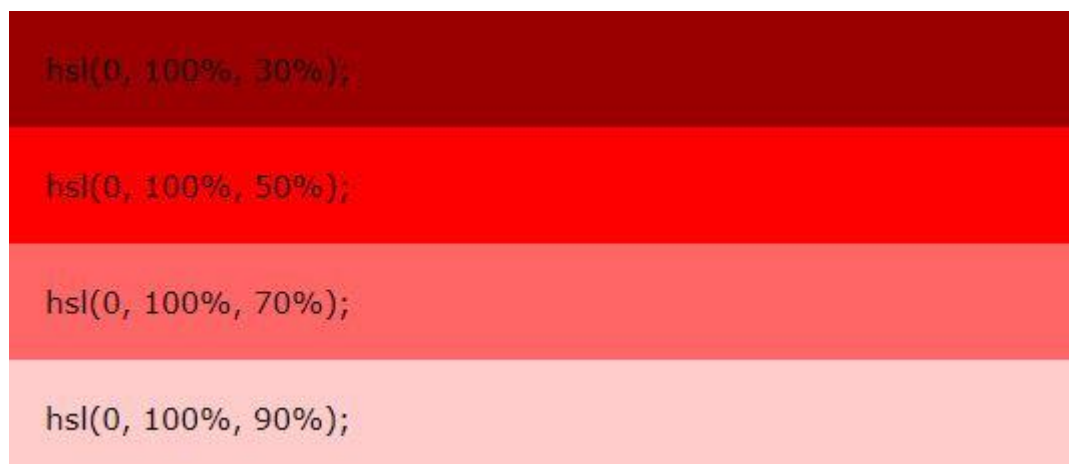
```
rgba(255, 0, 0, 0.6);
```

```
rgba(255, 0, 0, 0.8);
```

10.1.2 Warna HSL

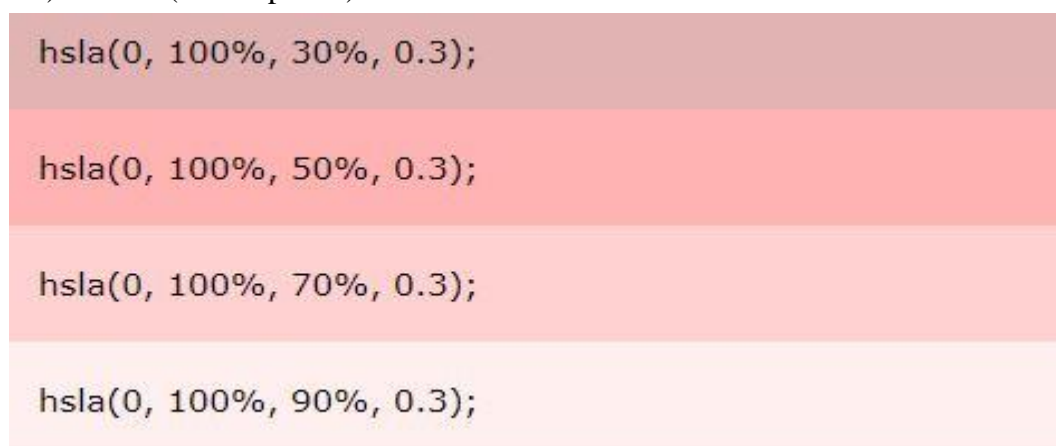
HSL singkatan dari Hue, Saturation and Lightness. Nilai warna HSL ditentukan dengan: `hsl` (hue, saturation, lightness).

- Hue adalah tingkat pada roda warna (dari 0 sampai 360): 0 (atau 360) berwarna merah, 120 berwarna hijau, 240 berwarna biru
- Saturation adalah nilai persentase: 100% adalah warna penuh.
- Lightness juga persentase; 0% gelap (hitam) dan 100% berwarna putih.



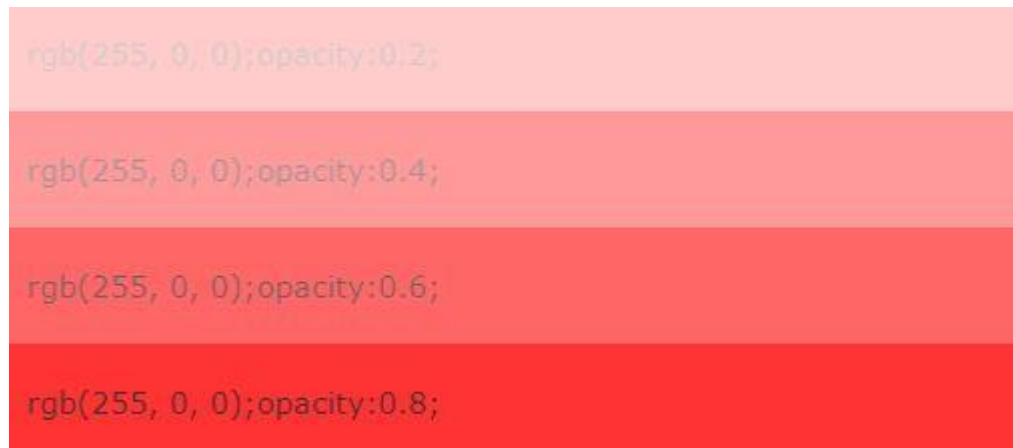
10.1.3 Warna HSLA

Nilai warna HSLA merupakan perpanjangan dari nilai warna HSL dengan alpha channel - yang menentukan opacity untuk sebuah warna. Nilai warna HSLA ditentukan dengan: hsla (hue, saturation, lightness, alpha), di mana parameter alpha mendefinisikan opacity. Parameter alpha adalah angka antara 0.0 (sepenuhnya transparan) dan 1.0 (buram penuh).



10.1.4 Warna Kegelapan/opacity

Properti opacity CSS3 menetapkan opacity untuk keseluruhan elemen (baik warna latar belakang dan teks akan buram / transparan). Nilai properti opacity harus berupa angka antara 0.0 (sepenuhnya transparan) dan 1.0 (sama sekali buram).



Contoh berikut menunjukkan elemen yang berbeda:

```
<html>
<head>
<style>
#p1 {background-color:hsl(120,100%,50%);}
#p2 {background-color: hsla(120, 100%, 50%, 0.3);}
#p3 {background-color:rgb(0,255,0);opacity:0.6;}
#p4 {background-color: rgba(0, 255, 0, 0.3);}
</style>
</head>
<body>

<p>Warna CSS3:</p>
<p id="p1">Green</p>
<p id="p2">Pastel violet</p>
<p id="p3">Pastel violet</p>
<p id="p4">Pastel violet</p>
</body>
</html>
```

10.2 Tugas

1. Buatlah halaman web berisi kolom berwarna , pewarnaan kolom menggunakan metode pewarnaan RGBA,HSL,HSLA dan Opacity.

BAB XI CSS3 LANJUTAN

11.1 Transisi Warna

Gradien CSS3 memungkinkan Anda menampilkan transisi yang mulus antara dua atau lebih warna tertentu. Sebelumnya, Anda harus menggunakan gambar untuk efek ini. Namun, dengan menggunakan gradien CSS3 Anda dapat mengurangi waktu download dan penggunaan bandwidth. Selain itu, elemen dengan gradien terlihat lebih baik saat diperbesar, karena gradiennya dihasilkan oleh browser.

CSS3 mendefinisikan dua jenis gradien:

- Gradien linier (turun / naik / kiri / kanan / diagonal)
- Radial Gradients (didefinisikan oleh pusatnya)

11.1.1 Linier gradient

```
<html>
<head>
<style>
#grad1 {
    height: 200px;
    background: red; /* For browsers that do not support gradients
*/
    background: -webkit-linear-gradient(red, yellow); /* For
Safari 5.1 to 6.0 */
    background: -o-linear-gradient(red, yellow); /* For Opera 11.1
to 12.0 */
    background: -moz-linear-gradient(red, yellow); /* For Firefox
3.6 to 15 */
    background: linear-gradient(red, yellow); /* Standard syntax
(must be last) */
}
#grad2 {
    height: 200px;
    background: red; /* For browsers that do not support gradients
*/
    background: -webkit-linear-gradient(left, red , yellow); /*
For Safari 5.1 to 6.0 */
    background: -o-linear-gradient(right, red, yellow); /* For
Opera 11.1 to 12.0 */
    background: -moz-linear-gradient(right, red, yellow); /* For
Firefox 3.6 to 15 */
    background: linear-gradient(to right, red , yellow); /*
Standard syntax (must be last) */
}
</style>
</head>
```

```

<body>
<h3>Linear Gradient - Atas ke bawah</h3>
<div id="grad1"></div>
<h3>Linear Gradient - samping</h3>
<div id="grad2"></div>
</body>
</html>

```

11.1.2 Repeat Gradients

```

<html>
<head>
<style>
#grad1 {
    height: 200px;
    background: -webkit-repeating-linear-gradient(red, yellow 10%,
green 20%); /* For Safari 5.1 to 6.0 */
    background: -o-repeating-linear-gradient(red, yellow 10%,
green 20%); /* For Opera 11.1 to 12.0 */
    background: -moz-repeating-linear-gradient(red, yellow 10%,
green 20%); /* For Firefox 3.6 to 15 */
    background: repeating-linear-gradient(red, yellow 10%, green
20%); /* Standard syntax (must be last) */
}

#grad2 {
height: 200px;
background: -webkit-repeating-linear-gradient(45deg, red, yellow
7%, green 10%);
background: -o-repeating-linear-gradient(45deg, red, yellow 7%, green
10%);
background: -moz-repeating-linear-gradient(45deg, red, yellow
7%, green 10%);
background: repeating-linear-gradient(45deg, red, yellow 7%, green
10%);
}

#grad3 {
    height: 200px;
background: -webkit-repeating-linear-gradient(190deg, red, yellow
7%, green 10%);
background: -o-repeating-linear-gradient(190deg, red, yellow
7%, green 10%);
background: -moz-repeating-linear-gradient(190deg, red, yellow
7%, green 10%);
background: repeating-linear-gradient(190deg, red, yellow 7%, green
10%);
}

#grad4 {
    height: 200px;
background: -webkit-repeating-linear-gradient(90deg, red, yellow
7%, green 10%);
background: -o-repeating-linear-gradient();
background: -moz-repeating-linear-gradient(90deg, red, yellow
7%, green 10%);
}

```

```
background: repeating-linear-gradient(90deg, red, yellow 7%, green
10%);
}
</style>
</head>
<body>
<h3>Repeating Linear Gradient</h3>
<div id="grad1"></div>
<div id="grad2"></div>
<div id="grad3"></div>
<div id="grad4"></div>
</body>
</html>
```

11.1.3 Radial Gradients

Gradien radial didefinisikan oleh pusatnya. Untuk membuat gradien radial anda juga harus menentukan setidaknya dua warna.

Sintaks

```
background: radial-gradient(shape size at position, start-color,
..., last-color);
```

Secara default, bentuknya elips, ukuran sudut paling jauh, dan posisi tengah.

Radial Gradient - Stop Warna Diam-Diam Diam (ini default)

Contoh berikut menunjukkan gradien radial dengan berhenti warna yang merata:

```
<html>
<head>
<style>
#grad1 {
    height: 150px;
    width: 200px;
    background: red;
    background: -webkit-radial-gradient(red 5%, yellow 15%, green
60%);
background: -o-radial-gradient(red 5%, yellow 15%, green 60%);
background: -moz-radial-gradient(red 5%, yellow 15%, green 60%);
background: radial-gradient(red 5%, yellow 15%, green 60%);
}
#grad2 {
    height: 150px;
    width: 200px;
background: red;
background: -webkit-radial-gradient(red, yellow, green);
background: -o-radial-gradient(red, yellow, green);
background: -moz-radial-gradient(red, yellow, green);
background: radial-gradient(red, yellow, green);
}
</style>
</head>
<body>
<h3>Radial Gradient</h3>
<div id="grad1"></div>
```



```
<div id="grad2"></div>
</body>
</html>
```

11.1.4 Efek bayangan CSS3

Dengan CSS3 Anda bisa menambahkan bayangan ke teks dan elemen. Dalam bab ini Anda akan belajar tentang properti berikut:

- teks-bayangan
- kotak-bayangan

11.1.4.1 Teks bayangan

Properti `text-shadow` CSS3 menerapkan bayangan ke teks. Dalam penggunaannya yang paling sederhana, Anda hanya menentukan bayangan horisontal (2px) dan bayangan vertikal (2px):

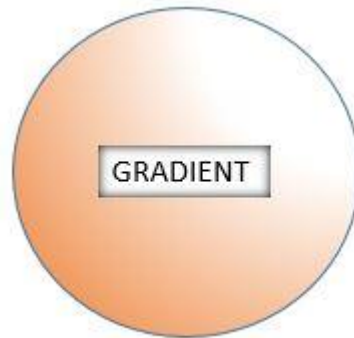
```
<html>
<head>
<style>
h1 {
  text-shadow: 2px 2px;
  text-shadow: 2px 2px red;
  text-shadow: 2px 2px 5px red;
  color: white;
  text-shadow: 2px 2px 4px #000000;
  text-shadow: 0 0 3px #FF0000, 0 0 5px #0000FF;
  text-shadow: 1px 1px 2px black, 0 0 25px blue, 0 0 5px
darkblue;
  text-shadow: -1px 0 black, 0 1px black, 1px 0 black, 0 -1px
black;
}
</style>
</head>
<body>
<h1>Text-shadow effect!</h1>
</body>
</html>
```

11.1.4.2 Kotak bayangan

```
div {
  width: 300px;
  height: 100px;
  padding: 15px;
  background-color: yellow;
  box-shadow: 10px 10px 5px grey;
}
```

11.2 Tugas

1. Buatlah Tampilan Gradian seperti dibawah ini, dengan pewarnaan yang sama.



BAB XII

CSS3 LANJUTAN

12.1 Transformasi CSS3

Transformasi CSS3 memungkinkan Anda menerjemahkan, memutar, skala, dan elemen miring. Transformasi adalah efek yang memungkinkan suatu elemen mengubah bentuk, ukuran dan posisi. CSS3 mendukung transformasi 2D dan 3D.

12.1.1 Transformasi 2D

Dalam bab ini Anda akan belajar tentang metode transformasi 2D berikut ini:

- translate()
- rotate()
- scale()
- skewX()
- skewY()
- matrix()

Berikut contoh pengaplikasiannya

```
<html>
<head>
<style>
div {
    width: 300px;
    height: 100px;
    background-color: yellow;
    border: 1px solid black;
}
div#myDiv {
    -ms-transform: rotate(20deg);
    -webkit-transform: rotate(20deg);
    transform: rotate(20deg);
}
div#myDiv1 {
    margin: 150px;
    width: 200px;
    height: 100px;
    background-color: yellow;
    border: 1px solid black;
    -ms-transform: scale(2,3);
    -webkit-transform: scale(2,3);
    transform: scale(2,3);
}
div#myDiv2 {
    margin: 150px;
```

```

width: 200px;
height: 100px;
background-color: yellow;
border: 1px solid black;
border: 1px solid black;
-ms-transform: scale(0.5,0.5);
-webkit-transform: scale(0.5,0.5);
transform: scale(0.5,0.5);
}
div#myDiv3 {
-ms-transform: skewX(20deg);
-webkit-transform: skewX(20deg);
transform: skewX(20deg);
}
div#myDiv4 {
-ms-transform: skewY(20deg);
-webkit-transform: skewY(20deg);
transform: skewY(20deg);
}
div#myDiv5{
-ms-transform: matrix(1, -0.3, 0, 1, 0, 0);
-webkit-transform: matrix(1, -0.3, 0, 1, 0, 0);
transform: matrix(1, -0.3, 0, 1, 0, 0);
}
div#myDiv6 {
-ms-transform: matrix(1, 0, 0.5, 1, 150, 0);
-webkit-transform: matrix(1, 0, 0.5, 1, 150, 0);
transform: matrix(1, 0, 0.5, 1, 150, 0);
}
</style>
</head>
<body>

<div>
Ini elemen normal div.
</div>
<div id="myDiv">
Metode rotasi clockwise 20 degrees.
</div>
<div id="myDiv1">
Metode scale ()increases
</div>
<div id="myDiv2">
MEtode scale()decreases
</div>
<div id="myDiv3">
Metode skewX()
</div>
<div id="myDiv4">
Metode skewY()
</div>
<div id="myDiv5">
Metode matrix()
</div>
<div id="myDiv6">
Metode matrix() yang lain

```

```

</div>
</body>
</html>

```

12.1.2 Transformasi 3D

Dalam bab ini Anda akan belajar tentang metode transformasi 3D berikut ini:

- rotateX()
- rotateY()
- rotateZ()

Berikut contoh pengaplikasiannya

```

<html>
<head>
<style>
div {
width: 300px;
height: 100px;
background-color: yellow;
border: 1px solid black;
}
div#myDiv {
-webkit-transform: rotateX(150deg);
transform: rotateX(150deg);
}
div#myDiv1 {
-webkit-transform: rotateY(150deg);
transform: rotateY(150deg);
}
div#myDiv2 {
-webkit-transform: rotateZ(90deg);
transform: rotateZ(90deg);
}
</style>
</head>
<body>
<div>
Elemen normal div
</div>
<div id="myDiv">
Metode rotateX() dengan rotated 150 degrees.
</div>
<div id="myDiv1">
Metode rotateY() dengan rotated 150 degrees.
</div>
<div id="myDiv2">
Metode rotateZ() dengan rotated 90 degrees.
</div>
</div>
</body>

```

```
</html>
```

12.2 Transisi CSS3

Transisi CSS3 memungkinkan Anda mengubah nilai properti dengan lancar (dari satu nilai ke nilai lainnya), selama durasi tertentu.

Bagaimana Cara Menggunakan Transisi ?

Untuk membuat efek transisi, Anda harus menentukan dua hal:

- properti CSS yang ingin Anda tambahkan efeknya
- durasi efeknya

Catatan: Jika bagian durasi tidak ditentukan, transisi tidak akan berpengaruh, karena nilai defaultnya adalah 0.

Contoh berikut menunjukkan elemen 100px * 100px red <div>. Elemen <div> juga menentukan efek transisi untuk properti lebar, dengan durasi 2 detik:

```
div {
  width: 100px;
  height: 100px;
  background: red;
  -webkit-transition: width 2s;
  transition: width 2s;
}
```

Efek transisi akan dimulai saat properti CSS (width) yang ditentukan mengubah nilai. Sekarang, mari kita tentukan nilai baru untuk properti lebar saat pengguna menyusui elemen <div>:

```
div {
  width: 100px;
  height: 100px;
  background: red;
  -webkit-transition: width 2s;
  transition: width 2s;
}

div:hover {
  width: 300px;
}
```

Perhatikan bahwa ketika kursor keluar dari elemen, secara bertahap akan berubah kembali ke gaya asalnya.

12.2.1 Ubah Beberapa Nilai Properti

Contoh berikut menambahkan efek transisi baik untuk properti lebar dan tinggi, dengan durasi 2 detik untuk lebar dan 4 detik untuk tinggi:

```
div {
  width: 100px;
```

```

height: 100px;
background: red;
-webkit-transition: width 2s, height 4s;
transition: width 2s, height 4s;
}

div:hover {
width: 300px;
height: 300px;
}

```

12.2.2 Menentukan kecepatan transisi

Properti `transition-timing-function` menentukan kurva kecepatan efek transisi. Properti `transition-timing-function` dapat memiliki nilai berikut:

- kemudahan - menentukan efek transisi dengan awal yang lambat, lalu cepat, lalu akhiri perlahan (ini adalah default)
- linear - menentukan efek transisi dengan kecepatan yang sama dari awal sampai akhir
- kemudahan-in - menentukan efek transisi dengan awal yang lambat
- easy-out - menentukan efek transisi dengan akhir yang lambat
- kemudahan-in-out - menentukan efek transisi dengan awal dan akhir yang lambat
- cubic-bezier (n, n, n, n) - memungkinkan Anda menentukan nilai Anda sendiri dalam fungsi cubic-bezier.

Contoh berikut menunjukkan beberapa kurva kecepatan yang berbeda yang dapat digunakan:

```

<html>
<head>
<style>
div {
width: 100px;
height: 100px;
background: red;
-webkit-transition: width 2s; /* Safari */
transition: width 2s;
}
/* Safari 3.1 to 6.0 */
#div1 {-webkit-transition-timing-function: linear;}
#div2 {-webkit-transition-timing-function: ease;}
#div3 {-webkit-transition-timing-function: ease-in;}
#div4 {-webkit-transition-timing-function: ease-out;}
#div5 {-webkit-transition-timing-function: ease-in-out;}
/* Standard syntax */
#div1 {transition-timing-function: linear;}
#div2 {transition-timing-function: ease;}
#div3 {transition-timing-function: ease-in;}
#div4 {transition-timing-function: ease-out;}

```

```
#div5 {transition-timing-function: ease-in-out;}
div:hover {
    width: 300px;
}
</style>
</head>
<body>
<div id="div1">linear</div><br>
<div id="div2">ease</div><br>
<div id="div3">ease-in</div><br>
<div id="div4">ease-out</div><br>
<div id="div5">ease-in-out</div><br>
</body>
</html>
```

12.2.3 Transisi + Transformasi

Contoh berikut juga menambahkan transformasi ke efek transisi:

```
<html>
<head>
<style>
div {
    width: 100px;
    height: 100px;
    background: red;
    -webkit-transition: width 2s, height 2s, -webkit-transform 2s;
    /* Safari */
    transition: width 2s, height 2s, transform 2s;
}

div:hover {
    width: 300px;
    height: 300px;
    -webkit-transform: rotate(180deg); /* Safari */
    transform: rotate(180deg);
}
</style>
</head>
<body>
<div></div>
</body>
</html>
```

12.3 Tugas

1. Buatlah tampilan Transformasi dan Transisi Logo uty

BAB XIII

CSS3 LANJUTAN

13.1 ANIMASI CSS3

Animasi CSS3 memungkinkan animasi elemen HTML tanpa menggunakan JavaScript atau Flash!

Apa itu Animasi CSS3?

Animasi memungkinkan elemen berubah secara bertahap dari satu gaya ke gaya yang lain. Anda bisa mengubah sebanyak mungkin properti CSS yang Anda inginkan, sebanyak yang Anda inginkan. Untuk menggunakan animasi CSS3, Anda harus terlebih dahulu menentukan beberapa keyframes untuk animasi.

Keyframes menyimpan elemen gaya yang akan dibuuhkan pada waktu-waktu tertentu.

13.1.1 Aturan @keyframes

Bila Anda menentukan gaya CSS di dalam aturan @keyframes, animasi akan berangsur-angsur berubah dari gaya saat ini ke gaya baru pada waktu-waktu tertentu.

Agar animasi bisa bekerja, Anda harus mengikat animasi ke elemen.

Contoh berikut mengikat animasi "contoh" ke elemen <div>. Animasi akan berlangsung selama 4 detik, dan secara bertahap akan mengubah warna latar belakang elemen <div> dari "merah" menjadi "kuning":

```
<html>
<head>
<style>
div {
  width: 100px;
  height: 100px;
  background-color: red;
  -webkit-animation-name: contoh; /* Safari 4.0 - 8.0 */
  -webkit-animation-duration: 4s; /* Safari 4.0 - 8.0 */
  animation-name: contoh;
  animation-duration: 4s;
}
/* Safari 4.0 - 8.0 */
@-webkit-keyframes contoh {
  from {background-color: red;}
  to {background-color: yellow;}
}
/* Standard syntax */
@keyframes contoh {
```

```

    from {background-color: red;}
    to {background-color: yellow;}
}
</style>
</head>
<body>
<div></div>
</body>
</html>

```

Catatan: Jika properti durasi animasi tidak ditentukan, animasi tidak akan berpengaruh, karena nilai defaultnya adalah 0. Pada contoh di atas kita telah menentukan kapan style akan berubah dengan menggunakan kata kunci "from" dan "to" (yang mewakili 0% (start) dan 100% (complete)). Hal ini juga memungkinkan untuk menggunakan persen. Dengan menggunakan persen, Anda bisa menambahkan perubahan gaya sebanyak yang Anda mau.

Contoh berikut akan mengubah warna latar belakang elemen <div> saat animasi selesai 25%, selesai 50%, dan lagi saat animasi 100% selesai:

```

<html>
<head>
<style>
div {
    width: 100px;
    height: 100px;
    background-color: red;
    -webkit-animation-name: contoh; /* Safari 4.0 - 8.0 */
    -webkit-animation-duration: 4s; /* Safari 4.0 - 8.0 */
    animation-name: contoh;
    animation-duration: 4s;
}
/* Safari 4.0 - 8.0 */
@-webkit-keyframes contoh {
    0%   {background-color: red;}
    25%  {background-color: yellow;}
    50%  {background-color: blue;}
    100% {background-color: green;}
}
/* Standard syntax */
@keyframes contoh {
    0%   {background-color: red;}
    25%  {background-color: yellow;}
    50%  {background-color: blue;}
    100% {background-color: green;}
}
</style>
</head>
<body>
<div></div>

```

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

Contoh berikut akan mengubah warna latar belakang dan posisi elemen <div> saat animasi selesai 25%, selesai 50%, dan lagi saat animasi 100% selesai:

```
<html>
<head>
<style>
div {
  width: 100px;
  height: 100px;
  background-color: red;
  position: relative;
  -webkit-animation-name: contoh; /* Safari 4.0 - 8.0 */
  -webkit-animation-duration: 4s; /* Safari 4.0 - 8.0 */
  animation-name: contoh;
  animation-duration: 4s;
}
/* Safari 4.0 - 8.0 */
@-webkit-keyframes contoh {
  0%   {background-color:red; left:0px; top:0px;}
  25%  {background-color:yellow; left:200px; top:0px;}
  50%  {background-color:blue; left:200px; top:200px;}
  75%  {background-color:green; left:0px; top:200px;}
  100% {background-color:red; left:0px; top:0px;}
}
/* Standard syntax */
@keyframes contoh {
  0%   {background-color:red; left:0px; top:0px;}
  25%  {background-color:yellow; left:200px; top:0px;}
  50%  {background-color:blue; left:200px; top:200px;}
  75%  {background-color:green; left:0px; top:200px;}
  100% {background-color:red; left:0px; top:0px;}
}
</style>
</head>
<body>
<div></div>
</body>
</html>
```

13.1.2 Menetapkan Berapa Banyak Animasi yang Harus Dijalankan.

Properti `animation-iteration-count` menentukan berapa kali animasi dijalankan.

Contoh berikut akan menjalankan animasi 5 kali sebelum berhenti dalam 1 detik pertama:

```

<html>
<head>
<style>
div {
    width: 100px;
    height: 100px;
    background-color: red;
    position: relative;
    -webkit-animation-name: contoh; /* Safari 4.0 - 8.0 */
    -webkit-animation-duration: 4s; /* Safari 4.0 - 8.0 */
    -webkit-animation-iteration-count: 3; /* Safari 4.0 - 8.0 */
    animation-name: contoh;
    animation-duration: 1s;
    animation-iteration-count: 5;
    /*animation-iteration-count: infinite;*//*untuk berjalan terus*/
}
/* Safari 4.0 - 8.0 */
@-webkit-keyframes contoh {
    0% {background-color:red; left:0px; top:0px;}
    25% {background-color:yellow; left:200px; top:0px;}
    50% {background-color:blue; left:200px; top:200px;}
    75% {background-color:green; left:0px; top:200px;}
    100% {background-color:red; left:0px; top:0px;}
}
/* Standard syntax */
@keyframes contoh {
    0% {background-color:red; left:0px; top:0px;}
    25% {background-color:yellow; left:200px; top:0px;}
    50% {background-color:blue; left:200px; top:200px;}
    75% {background-color:green; left:0px; top:200px;}
    100% {background-color:red; left:0px; top:0px;}
}
</style>
</head>
<body>
<div></div>
</body>
</html>

```

13.1.3 Menentukan Kurva Kecepatan Animasi

Properti fungsi timing-animasi menentukan kurva kecepatan animasi. Properti fungsi timing-animasi dapat memiliki nilai berikut:

- Ease - menentukan animasi dengan awal yang lambat, lalu cepat, lalu linggah perlahan (ini adalah default)
- linear - menentukan animasi dengan kecepatan yang sama dari awal sampai akhir
- Ease-in - menentukan animasi dengan awal yang lambat
- Ease-out - menentukan animasi dengan akhir yang lambat

- Ease-in-out - menentukan animasi dengan awal dan akhir yang lambat
- cubic-bezier (n, n, n, n) - memungkinkan Anda menentukan nilai Anda sendiri dalam fungsi cubic-bezier.
-

Contoh berikut menunjukkan beberapa kurva kecepatan yang berbeda yang dapat digunakan:

```

<html>
<head>
<style>
div {
    width: 100px;
    height: 50px;
    background-color: red;
    font-weight: bold;
    position: relative;
    -webkit-animation: mymove 15s infinite; /* Safari 4.0 - 8.0 */
    animation: mymove 15s infinite;
}
/* Safari 4.0 - 8.0 */
#div1 {-webkit-animation-timing-function: linear;}
#div2 {-webkit-animation-timing-function: ease;}
#div3 {-webkit-animation-timing-function: ease-in;}
#div4 {-webkit-animation-timing-function: ease-out;}
#div5 {-webkit-animation-timing-function: ease-in-out;}
/* Standard syntax */
#div1 {animation-timing-function: linear;}
#div2 {animation-timing-function: ease;}
#div3 {animation-timing-function: ease-in;}
#div4 {animation-timing-function: ease-out;}
#div5 {animation-timing-function: ease-in-out;}
/* Safari 4.0 - 8.0 */
@-webkit-keyframes mymove {
    from {left: 0px;}
    to {left: 300px;}
}
/* Standard syntax */
@keyframes mymove {
    from {left: 0px;}
    to {left: 300px;}
}
</style>
</head>
<body>
<div id="div1">linear</div>
<div id="div2">ease</div>
<div id="div3">ease-in</div>
<div id="div4">ease-out</div>
<div id="div5">ease-in-out</div>
</body>
</html>

```

13.2 Tugas

1. Buatlah Tampilan Seperti tugas Bab XII, dan kolaborasikan dengan pengaturan kecepatan dan banyaknya yang dijalankan.

BAB XIV KOLABORASI HTML,CSS,JAVASCRIPT

14.1 POP-UP

```
<!DOCTYPE html>
<html>
<head>
<style>
.popup {
    position: relative;
    display: inline-block;
    cursor: pointer;
    -webkit-user-select: none;
    -moz-user-select: none;
    -ms-user-select: none;
    user-select: none;
}
.popup .popuptext {
    visibility: hidden;
    width: 160px;
    background-color: #555;
    color: #fff;
    text-align: center;
    border-radius: 6px;
    padding: 8px 0;
    position: absolute;
    z-index: 1;
    bottom: 125%;
    left: 50%;
    margin-left: -80px;
}
.popup .popuptext::after {
    content: "";
    position: absolute;
    top: 100%;
    left: 50%;
    margin-left: -5px;
    border-width: 5px;
    border-style: solid;
    border-color: #555 transparent transparent transparent;
}
.popup .show {
    visibility: visible;
    -webkit-animation: fadeIn 1s;
    animation: fadeIn 1s;
}
@-webkit-keyframes fadeIn {
    from {opacity: 0;}
    to {opacity: 1;}
}
@keyframes fadeIn {
    from {opacity: 0;}

```

```

        to {opacity:1 ;}
    }
</style>
</head>
<body style="text-align:center">
<h2>Popup</h2>
<div class="popup" onclick="myFunction()">Klik disini
    <span class="popuptext" id="myPopup">inilah popup</span>
</div>
<script>
function myFunction() {
    var popup = document.getElementById("myPopup");
    popup.classList.toggle("show");
}
</script>
</body>
</html>

```

14.2 SLIDE SHOW

```

<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-
scale=1">
<style>
* {box-sizing:border-box}
body {font-family: Verdana,sans-serif;}
.mySlides {display:none}
.slideshow-container {
    max-width: 1000px;
    position: relative;
    margin: auto;
}
.text {
    color: #f2f2f2;
    font-size: 15px;
    padding: 8px 12px;
    position: absolute;
    bottom: 8px;
    width: 100%;
    text-align: center;
}
.numbertext {
    color: #f2f2f2;
    font-size: 12px;
    padding: 8px 12px;
    position: absolute;
    top: 0;
}
.dot {
    height: 15px;
    width: 15px;
    margin: 0 2px;
}

```



```

background-color: #bbb;
border-radius: 50%;
display: inline-block;
transition: background-color 0.6s ease;
}
.active {
background-color: #717171;
}
.fade {
-webkit-animation-name: fade;
-webkit-animation-duration: 1.5s;
animation-name: fade;
animation-duration: 1.5s;
}

@-webkit-keyframes fade {
from {opacity: .4}
to {opacity: 1}
}
@keyframes fade {
from {opacity: .4}
to {opacity: 1}
}
@media only screen and (max-width: 300px) {
.text {font-size: 11px}
}
</style>
</head>
<body>
<h2>Automatic Slideshow</h2>
<p>Change image every 2 seconds:</p>
<div class="slideshow-container">
<div class="mySlides fade">
<div class="numbertext">1 / 3</div>

<div class="text">Caption Text</div>
</div>
<div class="mySlides fade">
<div class="numbertext">2 / 3</div>

<div class="text">Caption Two</div>
</div>
<div class="mySlides fade">
<div class="numbertext">3 / 3</div>

<div class="text">Caption Three</div>
</div>
</div>
<br>
<div style="text-align:center">
<span class="dot"></span>
<span class="dot"></span>
<span class="dot"></span>
</div>
<script>
var slideIndex = 0;

```

```

showSlides();

function showSlides() {
    var i;
    var slides = document.getElementsByClassName("mySlides");
    var dots = document.getElementsByClassName("dot");
    for (i = 0; i < slides.length; i++) {
        slides[i].style.display = "none";
    }
    slideIndex++;
    if (slideIndex > slides.length) {slideIndex = 1}
    for (i = 0; i < dots.length; i++) {
        dots[i].className = dots[i].className.replace(" active",
        "");
    }
    slides[slideIndex-1].style.display = "block";
    dots[slideIndex-1].className += " active";
    setTimeout(showSlides, 2000); // ganti gambar setiap 2 detik
}
</script>
</body>
</html>

```

14.3 LOGIN FORM

```

<!DOCTYPE html>
<html>
<style>
/* input fields */
input[type=text], input[type=password] {
    width: 100%;
    padding: 12px 20px;
    margin: 8px 0;
    display: inline-block;
    border: 1px solid #ccc;
    box-sizing: border-box;
}
/* gaya untuk semua tombol */
button {
    background-color: #4CAF50;
    color: white;
    padding: 14px 20px;
    margin: 8px 0;
    border: none;
    cursor: pointer;
    width: 100%;
}

button:hover {
    opacity: 0.8;
}

/* gaya tambahan untuk tombol cancel */
.cancelbtn {

```

```

        width: auto;
        padding: 10px 18px;
        background-color: #f44336;
    }

    /* Pusatkan gambar dan posisikan tombol cancel */
    .imgcontainer {
        text-align: center;
        margin: 24px 0 12px 0;
        position: relative;
    }

    img.avatar {
        width: 40%;
        border-radius: 50%;
    }

    .container {
        padding: 16px;
    }

    span.psw {
        float: right;
        padding-top: 16px;
    }

    /* The Modal (background) */
    .modal {
        display: none; /* sembunyi */
        position: fixed; /* posisi */
        z-index: 1; /* posisi diatas */
        left: 0;
        top: 0;
        width: 100%; /* Full lebar */
        height: 100%; /* Full tinggi */
        overflow: auto; /* Enable scroll jika dibutuhkan */
        background-color: rgb(0,0,0); /* warna Fallback */
        background-color: rgba(0,0,0,0.4); /* Black w/ opacity */
        padding-top: 60px;
    }

    /* Modal Content/Box */
    .modal-content {
        background-color: #fefefe;
        margin: 5% auto 15% auto; /* 5% dari atas, 15% dari bawah and
        terpusat */
        border: 1px solid #888;
        width: 80%; /* Bisa lebih atau kurang, tergantung ukuran layar
    */
    }

    /* Close Button (x) */
    .close {
        position: absolute;
        right: 25px;
        top: 0;
    }

```

```

        color: #000;
        font-size: 35px;
        font-weight: bold;
    }

    .close:hover,
    .close:focus {
        color: red;
        cursor: pointer;
    }

    /* zoom animasi */
    .animate {
        -webkit-animation: animatezoom 0.6s;
        animation: animatezoom 0.6s
    }

    @-webkit-keyframes animatezoom {
        from {-webkit-transform: scale(0)}
        to {-webkit-transform: scale(1)}
    }

    @keyframes animatezoom {
        from {transform: scale(0)}
        to {transform: scale(1)}
    }

    /* Ubah gaya untuk bentang dan batalkan tombol pada layar ekstra
    kecil */
    @media screen and (max-width: 300px) {
        span.psw {
            display: block;
            float: none;
        }
        .cancelbtn {
            width: 100%;
        }
    }
</style>
<body>

<h2>Login Form</h2>

<button
onclick="document.getElementById('id01').style.display='block'"
style="width:auto;">Login</button>

<div id="id01" class="modal">

    <form class="modal-content animate" action="/action_page.php">
        <div class="imgcontainer">
            <span
onclick="document.getElementById('id01').style.display='none'"
class="close" title="Close Modal">&times;</span>
            
        </div>

```

```

<div class="container">
  <label><b>Username</b></label>
  <input type="text" placeholder="Enter Username" name="uname"
required>

  <label><b>Password</b></label>
  <input type="password" placeholder="Enter Password"
name="psw" required>

  <button type="submit">Login</button>
  <input type="checkbox" checked="checked"> Remember me
</div>

<div class="container" style="background-color:#f1f1f1">
  <button type="button"
onclick="document.getElementById('id01').style.display='none'"
class="cancelbtn">Cancel</button>
  <span class="psw">Forgot <a href="#">password?</a></span>
</div>
</form>
</div>

<script>
var modal = document.getElementById('id01');
window.onclick = function(event) {
  if (event.target == modal) {
    modal.style.display = "none";
  }
}
</script>
</body>
</html>

```

14.4 VALIDASI

```

<!DOCTYPE html>
<html>
<head>
<style>
input {
  width: 100%;
  padding: 12px;
  border: 1px solid #ccc;
  border-radius: 4px;
  box-sizing: border-box;
  margin-top: 6px;
  margin-bottom: 16px;
}
/* gaya submit button */
input[type=submit] {
  background-color: #4CAF50;
  color: white;
}
/* gaya container untuk inputs */
.container {

```

```

        background-color: #f1f1f1;
        padding: 20px;
    }
    /* Kotak pesan ditampilkan saat pengguna mengklik kolom kata sandi
    */
    #message {
        display:none;
        background: #f1f1f1;
        color: #000;
        position: relative;
        padding: 20px;
        margin-top: 10px;
    }
    #message p {
        padding: 10px 35px;
        font-size: 18px;
    }
    /* Tambahkan warna teks hijau dan tanda centang bila
    persyaratannya benar */
    .valid {
        color: green;
    }
    .valid:before {
        position: relative;
        left: -35px;
        content: "✓";
    }
    /* ambahkan warna teks merah dan "x" bila persyaratannya salah */
    .invalid {
        color: red;
    }
    .invalid:before {
        position: relative;
        left: -35px;
        content: "✗";
    }
}
</style>
</head>
<body>

<h3>VALIDASI</h3>
<p>MASUKKAN :</p>
<div class="container">
    <form action="/action_page.php">
        <label for="usrname">ID</label>
        <input type="text" id="usrname" name="usrname" required>

        <label for="psw">Sandi</label>
        <input type="password" id="psw" name="psw"
pattern="(?!.*\d) (?!.*[a-z]) (?!.*[A-Z]).{8,}" title="Harus berisi
setidaknya satu nomor dan satu huruf besar dan huruf kecil, dan
paling sedikit 8 karakter atau lebih" required>
        <input type="submit" value="login">
    </form>
</div>
<div id="message">

```

```

<h3>Sandi harus berisi yang berikut ini:</h3>
  <p id="letter" class="invalid">A <b>lowercase</b> </p>
  <p id="capital" class="invalid">A <b>capital </b> </p>
  <p id="number" class="invalid">A <b>nomor</b></p>
  <p id="length" class="invalid">Minimal <b>8 karakter</b></p>
</div>
<script>
var myInput = document.getElementById("psw");
var letter = document.getElementById("letter");
var capital = document.getElementById("capital");
var number = document.getElementById("number");
var length = document.getElementById("length");
// Saat pengguna mengklik kolom kata sandi, tampilkan kotak
pesannya
myInput.onfocus = function() {
  document.getElementById("message").style.display = "block";
}
// Saat pengguna mengeklik di luar bidang kata sandi, sembunyikan
kotak pesan
myInput.onblur = function() {
  document.getElementById("message").style.display = "none";
}
// Saat pengguna mulai mengetikkan sesuatu di dalam kolom kata
sandi
myInput.onkeyup = function() {
  // Validasi huruf kecil
  var lowerCaseLetters = /[a-z]/g;
  if(myInput.value.match(lowerCaseLetters)) {
    letter.classList.remove("invalid");
    letter.classList.add("valid");
  } else {
    letter.classList.remove("valid");
    letter.classList.add("invalid");
  }
  // Validasi huruf besar
  var upperCaseLetters = /[A-Z]/g;
  if(myInput.value.match(upperCaseLetters)) {
    capital.classList.remove("invalid");
    capital.classList.add("valid");
  } else {
    capital.classList.remove("valid");
    capital.classList.add("invalid");
  }
  // Validasi nomor
  var numbers = /[0-9]/g;
  if(myInput.value.match(numbers)) {
    number.classList.remove("invalid");
    number.classList.add("valid");
  } else {
    number.classList.remove("valid");
    number.classList.add("invalid");
  }
  // Validasi panjang
  if(myInput.value.length >= 8) {
    length.classList.remove("invalid");
    length.classList.add("valid");
  }
}

```

```

    } else {
      length.classList.remove("valid");
      length.classList.add("invalid");
    }
  }
</script>
</body>
</html>

```

14.5 MULTIPLE STEP FORM

```

<!DOCTYPE html>
<html>
<meta name="viewport" content="width=device-width, initial-
scale=1.0">
<style>
* {
  box-sizing: border-box;
}

body {
  background-color: #f1f1f1;
}

#regForm {
  background-color: #ffffff;
  margin: 100px auto;
  font-family: Raleway;
  padding: 40px;
  width: 70%;
  min-width: 300px;
}

h1 {
  text-align: center;
}

input {
  padding: 10px;
  width: 100%;
  font-size: 17px;
  font-family: Raleway;
  border: 1px solid #aaaaaa;
}

/* Tandai kotak masukan yang mendapat kesalahan saat validasi: */
input.invalid {
  background-color: #ffdddd;
}

/* Sembunyikan semua langkah secara default: */
.tab {
  display: none;
}

```



```

button {
  background-color: #4CAF50;
  color: #ffffff;
  border: none;
  padding: 10px 20px;
  font-size: 17px;
  font-family: Raleway;
  cursor: pointer;
}

button:hover {
  opacity: 0.8;
}

#prevBtn {
  background-color: #bbbbbb;
}

/* Buat lingkaran yang menunjukkan langkah-langkah dari formulir:
*/
.step {
  height: 15px;
  width: 15px;
  margin: 0 2px;
  background-color: #bbbbbb;
  border: none;
  border-radius: 50%;
  display: inline-block;
  opacity: 0.5;
}

.step.active {
  opacity: 1;
}

/* Tandai langkah-langkah yang telah selesai dan valid: */
.step.finish {
  background-color: #4CAF50;
}
</style>
<body>

<form id="regForm" action="/action_page.php">
  <h1>Register:</h1>
  <!-- Satu "tab" untuk setiap langkah dalam bentuk: -->
  <div class="tab">Nama:
    <p><input placeholder="Nama Depan..." oninput="this.className
= ''" name="fname"></p>
    <p><input placeholder="Mama Lanjutan..."
oninput="this.className = ''" name="lname"></p>
  </div>
  <div class="tab">Info Kontak:
    <p><input placeholder="E-mail..." oninput="this.className =
''" name="email"></p>
    <p><input placeholder="Phone..." oninput="this.className = ''"
name="phone"></p>

```

```

</div>
<div class="tab">Kelahiran:
  <p><input placeholder="dd" oninput="this.className = ''"
name="dd"></p>
  <p><input placeholder="mm" oninput="this.className = ''"
name="nn"></p>
  <p><input placeholder="yyyy" oninput="this.className = ''"
name="yyyy"></p>
</div>
<div class="tab">Info Login:
  <p><input placeholder="Username..." oninput="this.className =
''" name="uname"></p>
  <p><input placeholder="Password..." oninput="this.className =
''" name="pword" type="password"></p>
</div>
<div style="overflow:auto;">
  <div style="float:right;">
    <button type="button" id="prevBtn" onclick="nextPrev(-
1)">Kembali</button>
    <button type="button" id="nextBtn"
onclick="nextPrev(1)">Lanjut</button>
  </div>
</div>
<!-- Lingkaran yang menunjukkan langkah-langkah dari formulir: -
->
<div style="text-align:center;margin-top:40px;">
  <span class="step"></span>
  <span class="step"></span>
  <span class="step"></span>
  <span class="step"></span>
</div>
</form>

<script>
var currentTab = 0; // Tab saat ini diatur menjadi tab pertama (0)
showTab(currentTab); // Tampilkan tab saat ini

function showTab(n) {
  // Fungsi ini akan menampilkan tab yang ditentukan dari form ...
  var x = document.getElementsByClassName("tab");
  x[n].style.display = "block";
  //... dan perbaiki tombol Sebelumnya / Berikutnya:
  if (n == 0) {
    document.getElementById("prevBtn").style.display = "none";
  } else {
    document.getElementById("prevBtn").style.display = "inline";
  }
  if (n == (x.length - 1)) {
    document.getElementById("nextBtn").innerHTML = "Submit";
  } else {
    document.getElementById("nextBtn").innerHTML = "Next";
  }
  //... dan menjalankan fungsi yang akan menampilkan indikator
langkah yang benar:
  fixStepIndicator(n)
}

```

```

function nextPrev(n) {
    // Fungsi ini akan mencari tahu tab mana yang akan ditampilkan
    var x = document.getElementsByClassName("tab");
    // Keluar dari fungsi jika bidang apa pun di tab aktif tidak
    valid:
    if (n == 1 && !validateForm()) return false;
    //Sembunyikan tab saat ini:
    x[currentTab].style.display = "none";
    // Meningkatkan atau menurunkan tab saat ini dengan 1:
    currentTab = currentTab + n;
    // jika Anda telah mencapai akhir formulir ...
    if (currentTab >= x.length) {
        // ... formulir akan dikirim:
        document.getElementById("regForm").submit();
        return false;
    }
    // Jika tidak, tampilkan tab yang benar:
    showTab(currentTab);
}

function validateForm() {
    // Fungsi ini berhubungan dengan validasi kolom formulir
    var x, y, i, valid = true;
    x = document.getElementsByClassName("tab");
    y = x[currentTab].getElementsByTagName("input");
    // Lingkaran yang memeriksa setiap bidang masukan pada tab saat
    ini:
    for (i = 0; i < y.length; i++) {
        //Jika bidang kosong ...
        if (y[i].value == "") {
            // tambahkan kelas "tidak valid":
            y[i].className += " invalid";
            // dan setel status valid saat ini menjadi false
            valid = false;
        }
    }
    // Jika statusnya benar, tandai langkahnya sebagai selesai dan
    valid:
    if (valid) {
        document.getElementsByClassName("step")[currentTab].className
        += " finish";
    }
    return valid; // kembalikan status yang valid
}

function fixStepIndicator(n) {
    // Fungsi ini menghapus kelas "aktif" dari semua langkah ...
    var i, x = document.getElementsByClassName("step");
    for (i = 0; i < x.length; i++) {
        x[i].className = x[i].className.replace(" active", "");
    }
    //... dan menambahkan kelas "aktif" pada langkah saat ini:
    x[n].className += " active";
}
</script>

```

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

14.6 Tugas

Buatlah Web sistem informasi pariwisata komplit menggunakan semua unsur yang ada dalam modul ini.

DAFTAR PUSTAKA

Chapman, Cameron, *The Evolution of Web Design*, Six Revisions, archived from the original on 30 October 2013.

Christensen, Mathias Biilmann (2015-11-16). "Static Website Generators Reviewed: Jekyll, Middleman, Roots, Hugo". *Smashing Magazine*. Retrieved 2016-10-26.

Niederst, Jennifer (2006). *Web Design In a Nutshell*. United States of America: O'Reilly Media. pp. 12–17. ISBN 0-596-00987-9.

Nielsen, Jakob; Tahir, Marie (October 2001), *Homepage Usability: 50 Websites Deconstructed*, New Riders Publishing, ISBN 978-0735711020

Tutorial web script : <https://www.w3schools.com>, Diakses Juli 2022

World Wide Web Consortium: Doctype Declarations to use in your Web document , <https://www.w3.org/QA/2002/04/valid-dtd-list.html>. Diakses Juli 2022

World Wide Web Consortium: Understanding Web Content Accessibility Guidelines 2.2.2: <https://www.w3.org/TR/UNDERSTANDING-WCAG20/time-limits-pause.html>. Diakses Juli 2022.



Diterbitkan Oleh :

Universitas Teknologi Yogyakarta

Jl. Siliwangi, Jombor, Seman, D.I.Yogyakarta.
E-mail : publikasi@uty.ac.id
www.uty.ac.id

@2022