



PROGRAM STUDI DIPLOMA 3

Manajemen Informatika

Modul Praktek Pemrograman Web

PROGRAM STUDI DIPLOMA 3 MANAJEMEN INFORMATIKA

Modul Praktek Pemrograman Web

Sunaryo Winardi, S.Kom.

© STMIK MIKROSKIL

Kampus A : Jl Thamrin No 124 Medan

Kampus B : Jl Thamrin No 140 Medan

Kampus C : Jl Thamrin No 112 Medan

Telp : 061 4573767, 4533708, 4533705 Fax : 061 4567789

Daftar Isi

JAVA SCRIPT	4
Memulai Java Script.....	4
Penulisan Java Script.....	4
Komentar	6
Variable.....	6
Operator Aritmatika	7
Operator Assignment	7
Operator Perbandingan.....	8
Operator Logika	8
Operator String	8
Operator Kondisional	8
Statement Percabangan	9
Statement Perulangan	12
Fungsi	14
Kegiatan Mandiri.	16
EVENT DAN OBJEK-OBJEK JAVA SCRIPT	18
Event.....	18
Objek-objek JavaScript	19
Kegiatan Mandiri.	27
HTML DOM.....	32
Properties dan Method HTML DOM.....	33
Mengakses HTML DOM.....	34
Memodifikasi HTML DOM	36
Mengubah konten HTML.....	36
Mengubah Gaya CSS	36
Mengubah Atribut HTML	37
Membuat Elemen HTML Baru.....	38
Menghapus Elemen HTML	39
Mengubah Elemen HTML	39
Event HTML DOM.....	40
Kegiatan Mandiri.	40
DHTML	44
DHTML WINDOWS.....	44

Cookie.....	45
document.cookie.....	46
setTimeout & clearTimeout	49
setInterval & clearInterval.....	50
Kegiatan Mandiri.	51
JQUERY	53
Memulai JQuery	53
Penulisan JQuery.....	53
jQuery Selectors	55
jQuery Events	56
Efek-Efek dengan jQuery	57
Kegiatan Mandiri.	71
JQUERY LANJUTAN	75
Manipulasi HTML dengan jQuery.....	75
Manipulasi CSS dengan jQuery	78
Kegiatan Mandiri.	82
PHP	85
Proses Instalasi XAMPP	85
Memulai PHP	90
Penulisan Script PHP.....	90
Komentar	91
Variable.....	91
Tipe Data	91
Operator Aritmatika	91
Operator Assignment	92
Operator Perbandingan.....	92
Operator Logika	93
Operator String	93
Operator Kondisional	93
Statement Percabangan	94
Statement Perulangan	97
Fungsi	98
Kegiatan Mandiri.	100
OBJEK-OBJEK PHP	101
Objek Untuk Memasukkan Data	101
Kegiatan Mandiri.	106
My SQL.....	109
Create Database.....	109
Create Tabel	110
Insert, Select Tabel	112
Update dan Delete Tabel	114
Kegiatan Mandiri.	116

Mengakses MySQL Dengan PHP	118
Koneksi ke Database	118
Menutup Koneksi ke Database	119
Memilih Database	119
Menjalankan Perintah SQL	119
Mengambil Nilai Dari Record	120
Kegiatan Mandiri.	122
AJAX	124
XMLHttpRequest.....	124
AJAX XML FILE	128
Kegiatan Mandiri.	131
AJAX DATABASE.....	134
Contoh AJAX Database Select	134
Contoh AJAX Database Insert	135
Contoh AJAX Database Delete.....	137
Kegiatan Mandiri.	139

JAVA SCRIPT

JavaScript merupakan bahasa yang berbentuk kumpulan skrip yang pada fungsinya berjalan pada suatu dokumen HTML. Bahasa ini adalah bahasa pemrograman untuk bahasa HTML dengan mengijinkan pengekseskusi perintah perintah di sisi user, yang artinya pengekseskusiannya dilakukan pada sisi browser bukan di sisi server web.

Memulai Java Script

Untuk mempelajari pemrograman Java Script, ada dua piranti yang diperlukan, yaitu :

- Teks Editor
Teks Editor digunakan untuk menuliskan kode-kode Java Script. Teks editor yang dapat digunakan antara lain NotePad, TextPad, dan Macromedia Dreamweaver.
- Web Browser
Web Browser digunakan untuk menampilkan halaman web yang mengandung kode-kode Java Script. Web Browser yang dapat digunakan seperti Internet Explorer, FireFox, dan Google Chrome.

Penulisan Java Script

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

a) Java script ditulis pada file yang sama

Untuk penulisan dengan cara ini, perintah yang digunakan adalah

```
<SCRIPT Type="Text/JavaScript">  
    program java script disini  
</SCRIPT>.
```

Perintah tersebut biasanya diletakkan diantara Tag *<HEAD>* atau Tag *<BODY>*

JAVA SCRIPT

Contoh Penulisan :

```
<HTML>
<HEAD>
<TITLE>.....</TITLE>

<SCRIPT Type="Text/JavaScript">
    kode javascript disini
</SCRIPT>

</HEAD>
<BODY>

    kode HTML disini

<SCRIPT Type="Text/JavaScript">
    kode javascript disini
</SCRIPT>

    kode HTML disini

</BODY>
</HTML>
```

Contoh 1.1:

ketikan file HTML berikut dan jalankan dengan menggunakan web browser:

```
<html>
<head>
<title>JavaScript in the Head</title>
<script type="text/javascript">
function alertTest()
{
    document.write("Belajar Pemrograman Javascript!");
}
</script>
</head>
<body>
<h2 align="center">Script in the Head</h2>
<hr />
<script type="text/javascript">
    alertTest();
</script>
<hr />
</body>
</html>
```

Fungsi dari “document.write” adalah digunakan untuk menuliskan output dengan menggunakan javascript.

b) Java script ditulis pada file yang berbeda

Kode Javascript bisa dibuat dalam file terpisah / file eksternal dengan tujuan agar isi dari dokumen HTML tidak terlalu panjang. Atribut yang digunakan adalah :

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

Contoh 1.2:

Langkah no.1: Buat file dengan nama MyScript.js dengan Kode JavaScript Berikut:

```
document.write("<hr />");  
document.write(" My First External JavaScript");  
document.write(" <hr />");
```

Langkah no.2: ketikkan file HTML berikut dan jalankan dengan menggunakan web browser:

```
<html>  
<body>  
  <script src="myscript.js"></script>  
</body>  
</html>
```

Komentar

Sama seperti bahasa pemrograman lain. Javascript juga menyediakan fasilitas untuk menuliskan komentar. Pemberian komentar dalam Javascript dapat dilakukan dengan dua cara yaitu dengan menggunakan tanda // atau dituliskan antara tanda /* ...*/

```
//ini komentar
```

atau

```
/*ini juga komentar */
```

Variable

Aturan penamaan variabel adalah:

- Harus diawali dengan karakter (huruf)
- Tidak boleh menggunakan spasi
- Huruf Kapital dan kecil memiliki arti yang berbeda
- Tidak boleh menggunakan kata-kata yang merupakan perintah dalam JavaScript.

Cara mendeklarasi Variabel :

```
Var nama_variabel = nilai
```

Atau

J A V A S C R I P T

```
nama_variabel = nilai
```

Contoh:

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

Operator Aritmatika

Operator aritmatika digunakan untuk melakukan operasi aritmatika antara variabel. Misalkan bahwa $y = 5$, tabel di bawah menjelaskan operator aritmatika:

Operator	Deskripsi	Contoh	Hasil x	Hasil y
+	Addition	$x=y+2$	7	5
-	Subtraction	$x=y-2$	3	5
*	Multiplication	$x=y*2$	10	5
/	Division	$x=y/2$	2.5	5
%	Modulus (division remainder)	$x=y\%2$	1	5
++	Increment	$x=++y$	6	6
		$x=y++$	5	6
--	Decrement	$x=--y$	4	4
		$x=y--$	5	4

Operator Assignment

Assignment operator digunakan untuk menempatkan nilai ke variabel dalam JavaScript. Misalkan $x = 10$ dan $y = 5$, tabel di bawah menjelaskan assignment operator:

Operator	Contoh	Sama Dengan	Hasil
=	$x=y$		$x=5$
+=	$x+=y$	$x=x+y$	$x=15$
-=	$x-=y$	$x=x-y$	$x=5$
=	$x=y$	$x=x*y$	$x=50$
/=	$x/=y$	$x=x/y$	$x=2$
%=	$x\%=y$	$x=x\%y$	$x=0$

Operator Perbandingan

Operator perbandingan digunakan untuk menentukan kesetaraan atau perbedaan antara variabel. Misalkan $x = 5$, tabel di bawah menjelaskan operator perbandingan:

Operator	Deskripsi	Perbandingan	Hasil
<code>==</code>	Sama dengan	$x == 8$ $x == 5$	<i>Salah</i> <i>Benar</i>
<code>===</code>	Tepat sama dengan (nilai dan tipe)	$x === "5"$ $x === 5$	<i>Salah</i> <i>Benar</i>
<code>!=</code>	Tidak sama dengan	$x != 8$	<i>Salah</i>
<code>!==</code>	Tidak sama dengan (nilai dan tipe)	$x !== "5"$ $x !== 5$	<i>Benar</i> <i>Salah</i>
<code>></code>	Lebih besar dari	$x > 8$	<i>Salah</i>
<code><</code>	Lebih kecil dari	$x < 8$	<i>Benar</i>
<code>>=</code>	Lebih besar atau sama dengan	$x >= 8$	<i>Salah</i>
<code><=</code>	Lebih kecil atau sama dengan	$x <= 8$	<i>Benar</i>

Operator Logika

Operator logika digunakan untuk menentukan logika antara variabel. Misalkan $x = 5$ dan $y = 3$, tabel di bawah menjelaskan operator logis:

Operator	Deskripsi	Contoh	Hasil
<code>&&</code>	and	$(x < 10 \ \&\& \ y > 1)$	<i>Benar</i>
<code> </code>	or	$(x == 5 \ \ y == 5)$	<i>Salah</i>
<code>!</code>	not	$!(x == y)$	<i>Benar</i>

Operator String

Operator string pada JavaScript digunakan untuk menggabungkan beberapa string menjadi sebuah string yang lebih panjang.

Contoh :

```
nama = "Java" + "Script";
```

akan menghasilkan "JavaScript" pada variabel nama

Operator Kondisional

operator kondisional yang memberikan nilai ke variabel didasarkan pada beberapa kondisi.

Syntax:

```
variableName = (kondisi) nilai1 : ? nilai2
```

contoh

```
status = (usia <18) "Terlalu muda": "Cukup tua";?
```

Keterangan:

Jika variabel usia lebih kecil dari 18, status dari variabel nilai akan bernilai "Terlalu muda" , jika nilai usia lebih besar dari 18, status akan bernilai "Cukup tua ".

Statement Percabangan

JavaScript memiliki statement percabangan sebagai berikut:

if Statement : digunakan untuk mengeksekusi baris kode hanya jika kondisi yang ditentukan benar.

Syntax:

```
if (kondisi)
{
    baris kode yang akan dieksekusi jika kondisi benar
}
```

Contoh :

```
<html>
<head>
</head>
<body>
<script type="text/javascript">

var x="Good";
var y=12;
if (y<12)
{
    x=x+" day";
}
document.write(x);

</script>

</body>
</html>
```

if-else Statement : digunakan untuk mengeksekusi baris kode jika kondisi benar dan kode lain jika kondisi salah.

Syntax:

```
if (kondisi)
{
    baris kode yang akan dieksekusi jika kondisi benar
}
else
{
    baris kode yang akan dieksekusi jika kondisi salah
}
```

Contoh:

```
<html>
<head>
</head>
<body>
<script type="text/javascript">

var x="Good";
var y=13;
if (y<12)
{
    x=x+" day";
}
else
{
    x=x+" evening";
}

document.write(x);

</script>

</body>
</html>
```

- **If...else if...else Statement** : digunakan untuk untuk memilih salah satu dari banyak baris kode yang akan dieksekusi.

Syntax:

```
if (kondisi 1)
{
    baris kode yang akan dieksekusi jika kondisi 1 benar
}
else if ((kondisi 2)
{
    baris kode yang akan dieksekusi jika kondisi 2 benar
}
else
{
    baris kode yang akan dieksekusi jika kondisi 1 dan
    kondisi 2 salah
}
```

Contoh:

```
<html>
<head>
</head>
<body>
<script type="text/javascript">

var x="Good";
var y=0;
if (y<12)
{
    x=x+" day";
}
else if (y<24)
{
    x=x+" evening";
}
else
{
    x=x+" morning";
}

document.write(x);

</script>

</body>
</html>
```

- **Switch Statement** : digunakan untuk memilih salah satu dari banyak baris kode yang akan dieksekusi.

Syntax :

```
switch(n)
{
case 1:
    eksekusi baris kode 1
    break;
case 2:
    eksekusi baris kode 2
    break;
default:
    baris kode yang dieksekusi jika n tidak termasuk case 1 dan case 2.
}
```

Contoh :

```
<html>
<body>
<script type="text/javascript">

var x = "Hari ini Hari";
var d = 1;
switch (d)
{
  case 1:
    x= x+" Senin";
    break;
  case 2:
    x= x+" Selasa";
    break;
  case 3:
    x= x+" Rabu";
    break;
  case 4:
    x= x+" Kamis";
    break;
  case 5:
    x= x+" Jumat";
    break;
  case 6:
    x= x+" Sabtu";
    break;
  default:
    x= x+" Minggu";
  }
document.write(x);

</script>

</body>
</html>
```

Statement Perulangan

JavaScript memiliki statement perulangan sebagai berikut:

- **While Statement** : digunakan untuk menjalankan baris kode selama kondisi tertentu adalah benar.

Syntax:

```
while (Kondisi)
{
  baris kode yang akan dieksekusi
}
```

Contoh:

```
<html>
<body>

<script>
var x="",i=0;
while (i<5)
{
    document.write(i+"<br>");
    i++;
}
</script>

</body>
</html>
```

- **Do ... While Statement** : merupakan varian dari while loop. Loop ini akan mengeksekusi baris kode sekali, sebelum memeriksa kondisi while. Jika kondisi benar, maka loop akan diulangi selama kondisi benar.

Sintax :

```
do
{
    baris kode yang akan dieksekusi
}
while (Kondisi);
```

Contoh:

```
<html>
<body>

<script>
var x="",i=0;
do
{
    document.write(i+"<br>");
    i++;
}
while (i<5);
</script>

</body>
</html>
```

- **For Statement** : digunakan untuk loop baris kode beberapa kali.

Syntax :

```
for (pernyataan 1; pernyataan 2, pernyataan 3)
{
    blok kode yang akan dieksekusi
}
```

Dimana :

Pernyataan 1 : dijalankan sebelum loop (baris kode) dimulai.

Pernyataan 2 : mendefinisikan kondisi untuk menjalankan loop (baris kode).

Pernyataan 3 : dijalankan setiap kali setelah loop (baris kode) telah dieksekusi.

Contoh :

```
<html>
<body>

<script>

for (var i=0;i<5;i++)
{
    document.write("Angka " + i + "<br>");
}

</script>

</body>
</html>
```

Funksi

Selain memanfaatkan function yang telah disediakan JavaScript, function juga dapat dibuat sendiri (*User-defined functions*). *User-defined functions* dalam dokumen HTML di didefinisikan di bagian tag <head> dan pemanggilan function untuk menjalankan function tersebut dapat dilakukan di tag <body>. Function ini dibagi menjadi 2:

- **Function Tanpa Parameter (Argumen)**

Syntax:

```
function namafunction()
{
    blok kode yang akan dieksekusi
}
```

Contoh :

JAVA SCRIPT

```
<html>
<head>

<script>
function myFunction()
{
document.write("Hello World!");
}
</script>

</head>

<body>

<script>
myFunction();
myFunction();

</script>

</body>
</html>
```

- **Function Dengan Parameter (Argumen)**

Syntax:

```
function namafunction(var1, Var2, ... )
{
    blok kode yang akan dieksekusi
}
```

Contoh :

```
<html>
<head>

<script>
function myFunction(a,b)
{
    return a*b;
}
</script>

</head>

<body>

<script>
document.write(myFunction(2,3)+"<br>");
document.write(myFunction(3,1)+"<br>");
</script>

</body>
</html>
```

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya.

1. Buat kode HTML dan JavaScript Untuk menampilkan Tampilan Berikut dimana Text ditampilkan menggunakan JavaScript:

```

SELAMAT DATANG DI JAVASCRIPT
Program ini merupakan contoh sederhana menampilkan Teks!
Nama :
Nomor Absensi :
NIM :
Nilai yang saya harapkan untuk pelajaran JavaScript :
Cara saya mendapatkan Nilai tersebut Adalah :

```

2. Buat kode HTML dan JavaScript Untuk menampilkan Tabel Faktorial berikut:

Tabel Faktorial

Nilai	Hasil Faktorial
1	1
2	2
3	6
4	24
5	120
6	720
7	5040
8	40320
9	362880
10	3628800

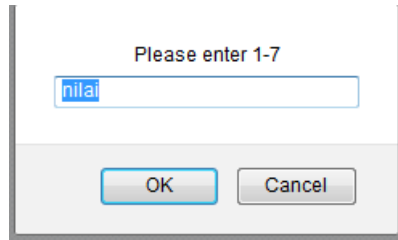
3. Buat kode HTML dan JavaScript Untuk menampilkan angka-angka berikut:

```

1
-2
3
-4
5
-6
7
-8
9
-10

```

4. Buat kode HTML dan JavaScript Untuk menghasilkan tampilan Berikut:

A screenshot of a web browser dialog box. The dialog box has a light gray background and a thin border. At the top, it says "Please enter 1-7". Below this text is a text input field with a blue border. Inside the input field, the word "nilai" is written in blue. At the bottom of the dialog box, there are two buttons: "OK" and "Cancel". The "OK" button is on the left and the "Cancel" button is on the right. Both buttons have a light gray background and a thin border.

Gambar 1.1

Hari ini Hari Minggu

Gambar 1.2

Nilai yang anda masukkan salah

Gambar 1.3

Keterangan:

- Pada saat web browser ditampilkan, akan muncul sebuah input box seperti tampilan gambar 1.1 untuk meminta user memasukan nilai dari 1-7
- Input 1 menyatakan hari senin, input 2 menyatakan hari selasa dan seterusnya sehingga muncul output seperti Gambar 1.2. Jika input yang dimasukkan diluar angka 1 sampai 7, Output yang ditampilkan adalah gambar 1.3.

EVENT DAN OBJEK-OBJEK JAVA SCRIPT

Event

Event merupakan suatu kejadian ketika user melakukan aksi pada halaman web seperti mengklik tombol mouse pada sebuah elemen tertentu yang dapat memicu fungsi javascript. Event umumnya dikombinasikan dengan *function*, dimana *function* tersebut tidak akan dijalankan sebelum event terjadi.

Tabel List Event

Kejadian	Keterangan
onAbort	Dibangkitkan bila pengguna menghentikan pemuatan citra (tag) yaitu bila pengguna menekan tombol stop atau mengklik link.
onChange	Dibangkitkan bila informasi masukan pada sebuah elemen form (text, textarea, select) diubah oleh pengguna.
onBlur	Dibangkitkan ketika suatu elemen kehilangan focus masukan, yaitu ketika pengguna menekan tombol <tab> atau mengklik elemen lain dari form.
onFocus	Dibangkitkan bila sebuah elemen form menerima focus masukan; yaitu bila pengguna mengklik elemen form tersebut atau menekan tombol <tab> sehingga focus masukan berpindah ke elemen ini.
onClick	Kejadian yang dibangkitkan bila pengguna mengklik sebuah elemen form atau link.
onError	Dibangkitkan bila terjadi kesalahan saat browser memuat dokumen atau citra.
onLoad	Dibangkitkan bila browser selesai memuat document
onUnload	Dibangkitkan bila pengguna keluar dari dokumen
onMouseOver	Dibangkitkan bila kursor mouse berada di atas sebuah link.
onMouseOut	Dibangkitkan bila kursor mouse keluar dari daerah link atau peta citra.

onReset	Dibangkitkan bila pengguna menekan tombol reset
onSelect	Kejadian yang dibangkitkan bila pengguna memilih sebagian atau seluruh teks pada elemen form yang berupa kotak teks.
onSubmit	Dibangkitkan ketika pengguna menekan tombol submit.

Objek-objek JavaScript

1. Objek Untuk memasukkan Data

Terdapat beberapa objek yang dapat digunakan untuk memasukkan data. Objek-objek tersebut biasanya terdapat dalam suatu form. Adapun objek-objek tersebut meliputi :

- **Objek Text**

Objek Text dapat digunakan untuk menginputkan data. Contoh penggunaannya dapat dilihat pada contoh berikut :

Contoh Program JavaScript

```
<html>
<head><title>Latihan Dengan Objek Text</title>
<script type ="Text/JavaScript">

function tekan()
{
var namastr = document.frm.nama.value;
var nimstr = document.frm.alamat.value;
document.frm.outputnama.value = namastr;
document.frm.outputalamat.value = nimstr;
}
</script>
</head>
<body>

<form name ="frm">
<H1> Objek Text</H1><hr>
Nama : <input type="text" size="11" name="nama">
Nim : <input type="text" size="25" name="alamat">
<input type="button" value="kirim" onclick="tekan()">
<H3>Output</H3>
Nama Anda adalah :<input type="text" size="11"
name="outputnama">
Nim Anda di :<input type="text" size="25"
name="outputalamat">
</form>
</body>
</html>
```

- **Objek Radio**

Objek radio adalah komponen yang digunakan untuk melakukan suatu pemilihan data. Karena Objek Radio merupakan Array , maka untuk mengakses satu tombol radio digunakan `radio[indeks]`. Objek radio juga mempunyai nilai `True` jika dipilih dan `False` jika tidak.

Contoh Program JavaScript

```
<html>
<head><title>Latihan Dengan Objek Radio</title>
<script type ="Text/JavaScript">

function radio_box()
{
var ket = "Anda adalah seorang";
if (document.frm.Lulusan[0].checked == true)
{
ket = ket +" D1";
}
else if (document.frm.Lulusan[1].checked == true)
{
ket = ket +" D3";
}
else if (document.frm.Lulusan[2].checked == true){
ket = ket +" S1";
}
else{
ket ="Pilih Salah Satu";
}
alert(ket);

}

</script>

</head>
<body>
<form name="frm">
<H1> Objek Radio</H1><hr>
<input type="radio" value="D1" name="Lulusan">D1<br />
<input type="radio" value="D3" name="Lulusan">D3<br />
<input type="radio" value="S1" name="Lulusan">S1<br />

<hr>
<input type="button" value="CONFIRM" onclick="radio_box()" ">
</form>
</body>
</html>
```

- **Objek Checkbox**

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

Contoh Program JavaScript

```

<html>
<head><title>Objek Checkbox</title></head>
<script type ="Text/JavaScript">

function radio_box()
{
var ket = "Buah Pilihan Anda :";
if (document.frm.buah[0].checked == true)
{
ket = ket+"\nAnggur";
}
if (document.frm.buah[1].checked == true)
{
ket = ket+"\nJeruk";
}
if (document.frm.buah[2].checked == true)
{
ket = ket+"\nMangga";
}
alert(ket);
}

</script>

<body>
<form name="frm">
<H1>Objek Checkbox</H1><hr>
Nama Buah-buahan : <br />
<input type="checkbox" value="Anggur" name="buah">Anggur
<br/>
<input type="checkbox" value="Jeruk" name="buah">Jeruk
<br/>
<input type="checkbox" value="Mangga" name="buah">Mangga
<hr>
<input type="button" value="CONFIRM" onclick="radio_box()">

</form>
</body>
</html>

```

- **Objek TextArea**

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

Contoh Program JavaScript

```
<html>
<head><title>Latihan Dengan Objek TextArea</title>
<script type="Text/JavaScript">
function tekan()
{
var ketstr = document.frm.Ket.value;
document.frm.Output.value = ketstr;
}
</script>
</head>
<body>
<form name ="frm">
<H1>Objek TextArea</H1><hr>
<h3>Keterangan :</h3><br>
<textarea name="Ket" rows="5" cols="30"></textarea>
<BR><BR>
<input type="button" value="kirim" onclick="tekan()"><hr/>
<H3>Output Keterangan :</H3>
<textarea name="Output" rows="5" cols="30"></textarea>
</form>
</body>
</html>
```

- **Objek Select**

Objek Select menyimpan informasi tentang elemen form yang berupa daftar kotak.

Contoh Program JavaScript

```
<html>
<head><title>Latihan Dengan Objek Select</title>
<script type="Text/JavaScript">
function tekan()
{
ket=document.frm.Jurusan.value;
alert(ket);
document.frm.Output.value =ket ;
}
</script>
</head>
<body>
<form name ="frm">
<H1>Objek Select</H1><hr>
<select name="Jurusan" Size="1">
<option value ="satu"> 1</option>
<option value ="dua"> 2</option>
<option value ="tiga"> 3</option>
</select>
<input type="button" value="kirim" onclick="tekan()">
<hr>
<H3>Output jurusan :</H3>
<input type="text" name="Output" size="30">
</form>
</body>
</html>
```


2. Objek Array (Array Object)

Array adalah suatu variable yang dapat memuat beberapa nilai secara berurutan. pendeklarasian untuk array adalah:

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

Untuk mengisi ketiga elemen tersebut dapat dilakukan dengan cara :

```
nama[0] ="Dian"
nama[1] ="Andri"
nama[2]="Irawan"
```

Pendeklarasikan array juga dapat dilakukan sekaligus dengan mengisi elemen-elemennya.

```
nama = new Array("Dian","Andri","Irawan")
```

Tabel Metode-metode untuk Objek Array

Method	Keterangan
concat ()	Menggabungkan dua array
join ()	Bergabung semua elemen array ke dalam string
pop ()	Hapus elemen terakhir dari array
push ()	Menambahkan elemen baru ke akhir array
reverse ()	Membalik urutan elemen dalam array
shift ()	Menghapus elemen pertama dari array
slice ()	Memilih elemen dari array
sort ()	Mengurutkan array
splice ()	Menambahkan elemen ke posisi 2 dalam array
toString ()	Mengkonversi sebuah array ke string

3. Objek Tanggal (Date Object)

Objek Tanggal digunakan untuk memanipulasi tanggal dan waktu pada JavaScript. Untuk pendeklarasian Objek Tanggal adalah sebagai berikut :

```
Var tgl = new Date()
```

Pernyataan diatas menyatakan bahwa variable tgl mengandung unsur tanggal dan waktu.

Tabel Metode-metode untuk Objek Date

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

4. Objek Math

Objek Math digunakan untuk menangani perhitungan matematis yang rumit. Bentuk penulisan untuk Objek Math adalah sebagai berikut :

```
Math.metode(nilai)
```

Tabel Metode Untuk Objek Math

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

sqrt(a)	Mendapatkan nilai akar dari kuadrat a
sin(a)	Mendapatkan nilai sinus dari a
tan(a)	Mendapatkan nilai tangen dari a

5. Objek String

String adalah suatu objek yang merupakan kumpulan dari elemen karakter-karakter. Dalam Javascript string atau karakter harus diapit dengan tanda petik ganda("") atau tanda petik tunggal(').

Untuk mendeklarasikan Objek String adalah sebagai berikut:

```
Nama = "Shafana Vevica"
Panjang = Nama.length; // Panjang akan berisi 14
```

Length adalah property yang sering digunakan dalam objek string yang digunakan untuk mengetahui banyaknya karakter dalam suatu string.

Tabel Metode Untuk Objek String

Method	Keterangan
big()	Tercetak lebih besar
blink()	Efek berkedip (aktif pada browser Netscape)
bold()	Tercetak tebal
charAt(n)	Mengambil karakter ke -n dari string. Index string dimulai dari 0
fixed()	Tercetak fixed-pitch
fontcolor('warna')	Tercetak sesuai warna yang didefinisikan
indexOf('char')	Mengambil nilai indeks dari suatu karakter
italic()	Tercetak miring
link('url')	Menjadikan string hyperlink
small()	Tercetak lebih kecil
strike()	Tercetak dengan coretan
sub()	Tercetak subscript
substring(a,b)	Mengambil karakter dari posisi a sampai b-1
sup()	Tercetak superscript
toLowerCase()	Tercetak huruf kecil
toUpperCase()	Tercetak huruf besar

split("")	Menjadikan string diuraikan/dipisahkan berdasarkan tanda ("). Hasil dari split akan dihasilkan sebuah array dengan indeks 0 untuk string ke 1 dan seterusnya.
-----------	---

6. Objek Boolean

Objek Boolean mewakili dua nilai: "benar" atau "salah". Untuk mendeklarasikan Object Boolean adalah sebagai berikut:

```
var myBoolean = new Boolean ();
```

7. Objek RegExp

Sebuah regular expression adalah obyek yang menggambarkan pola karakter. Regular expressions digunakan untuk melakukan pencocokan pola dan "pencarian kata dan menggantikan pola kata" pada sebuah teks.

Untuk mendeklarasikan Object RegExp adalah sebagai berikut:

```
var patt=new RegExp(pattern,modifiers);
```

dimana :

- pattern menentukan pola ekspresi
- modifiers menentukan apakah pencarian harus global, case-sensitive, dll

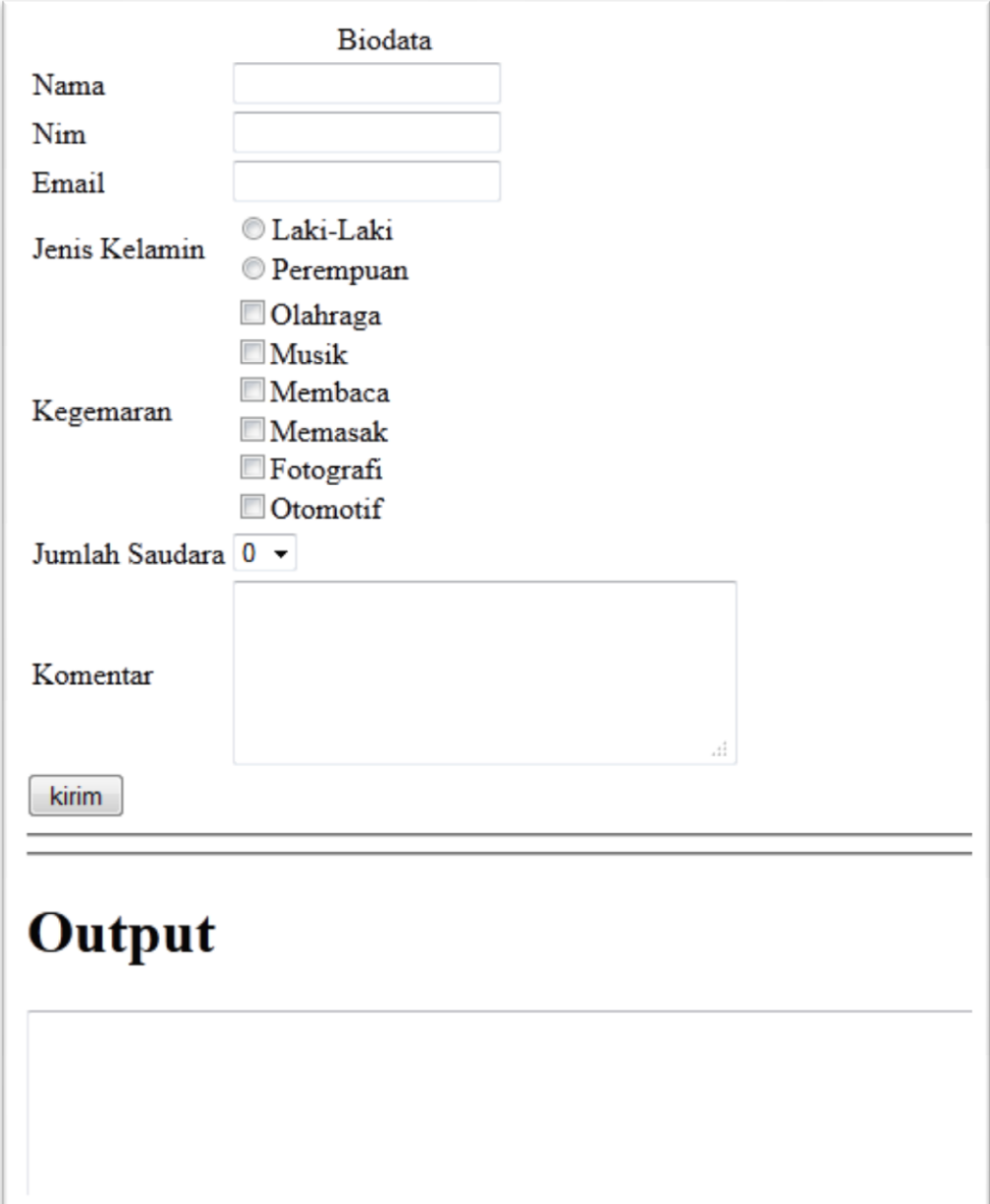
Tabel Metode Untuk Objek RegExp

Method	Keterangan
compile()	Untuk menggantikan suatu pola regex dengan pola regex yang baru
exec()	Melakukan test dan mengembalikan nilainya dalam bentuk sebuah array berisi substring yang cocok dengan pola regex
test()	Memeriksa string sesuai dengan pola regex

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya.

1. Buat kode HTML dan JavaScript Untuk menampilkan Tampilan Berikut:



The image shows a web form titled "Biodata" with the following fields and controls:

- Nama**: Text input field.
- Nim**: Text input field.
- Email**: Text input field.
- Jenis Kelamin**: Radio buttons for ☐ Laki-Laki and ☐ Perempuan.
- Kegemaran**: Checkboxes for ☐ Olahraga, ☐ Musik, ☐ Membaca, ☐ Memasak, ☐ Fotografi, and ☐ Otomotif.
- Jumlah Saudara**: A dropdown menu currently showing "0".
- Komentar**: A large text area.
- Submit**: A button labeled "kirim".

Below the form is a section titled "Output" with a large empty box for displaying results.

Gambar 2.1

Output

```
Nama :Nama
Nim :Nim
Email :re
Jenis Kelamin :Laki-Laki
Kegemaran :
Musik
Memasak
Jumlah Saudara satu
Komentar:
-non-
```

Gambar 2.2

Keterangan:

Setelah data pada form diisi, maka tampilkanlah data form tersebut untuk ditampilkan pada TextArea pada Bagian Output setelah tombol kirim di tekan. Sehingga hasilnya akan tampak seperti Gambar 2.2

2. Buat kode HTML dan JavaScript Untuk menampilkan Tampilan Berikut:

Java Scrip String Object

Input String :

Panjang String :

Ubah Bentuk Tulisan

☐ To Upper Case

☐ to Lower Case

Cari

String Pencarian:

Hasil Kemunculan :

Ganti

Ganti kata dengan:

Ganti 2

Masukkan Kata :

Ganti kata dengan:

Keterangan :

- a. Panjang string akan Otomatis di hitung di Textbox Panjang String saat User mengetikkan karakter di Kotak input String.
- b. Saat fokus keluar dari textbox input string, maka textbox input string bersifat disabled.
- c. Pada Saat pilihan Upper Case maupun Lower Case dipilih, ubah tulisan yang ada di input string menjadi huruf besar atau huruf kecil.

- d. Tombol Cari berfungsi untuk menghitung jumlah kemunculan kata pada input string
 - e. Tombol Ganti pada Grupbox Ganti berfungsi menggantikan kata di input string.
 - f. Tombol Ganti pada Grupbox Ganti2 berfungsi menggantikan sebuah huruf di input string sesuai dengan huruf yang digantikan.
3. Buat kode HTML dan JavaScript Untuk menampilkan Tampilan Berikut:

Bioskop 22
Nama Bioskop :

Nama Film :

Jam Tayang :

Jumlah Penonton :

Gambar 3.1

Bioskop 22

Nama Bioskop : Harmes ▼

Nama Film : Expensive ▼

Jam Tayang : 13.30 ▼

Jumlah Penonton : 3 ▼

Pesan

harga Tiket = 25000
Jumlah Yang dibayar = 75000

Gambar 3.2

Keterangan :

- Daftar pilihan Nama Bioskop adalah Harmes, Sun, dan Paladium
- Jika Nama Bioskop yang dipilih adalah Harmes, maka Nama film yang dapat dipilih adalah Life(dengan jam Tayang :13.00, 15.30, 18.00) ,Expensife (dengan jam Tayang :13.30, 16.00, 18.30) dan Hitman (dengan jam Tayang :13.15, 15.45, 18.15)
- Jika Nama Bioskop yang dipilih adalah Sun, maka Nama film yang dapat dipilih adalah Money(dengan jam Tayang :13.00, 15.30, 18.00) ,Bank (dengan jam Tayang :13.30, 16.00, 18.30) dan Sorow (dengan jam Tayang :13.15, 15.45, 18.15)
- Jika Nama Bioskop yang dipilih adalah Paladium, maka Nama film yang dapat dipilih adalah Kita(dengan jam Tayang :13.00, 15.30, 18.00) ,Dia (dengan jam Tayang :13.30, 16.00, 18.30) dan Saya (dengan jam Tayang :13.15, 15.45, 18.15)
- Pilihan Jumlah Penonton adalah 1-5;

- f. Tombol Pesan akan menampilkan biaya tiket yang dibayar dengan harga tiket Harmes Rp25.000/orang, Sun Rp30.000/orang dan Paladium Rp20.000/orang sehingga hasilnya tampak seperti gambar 3.2.

HTML DOM

DOM (Document Object Model) merupakan standar dari W3C (world Wide Web Consortium) yang memungkinkan untuk mengakses dan mengupdate dokumen HTML secara Dinamis. HTML DOM mendefinisikan objek dan properti untuk semua elemen HTML, serta metoda untuk mengaksesnya. HTML DOM juga merupakan standarisasi bagaimana mengambil, dan memanipulasi element HTML.

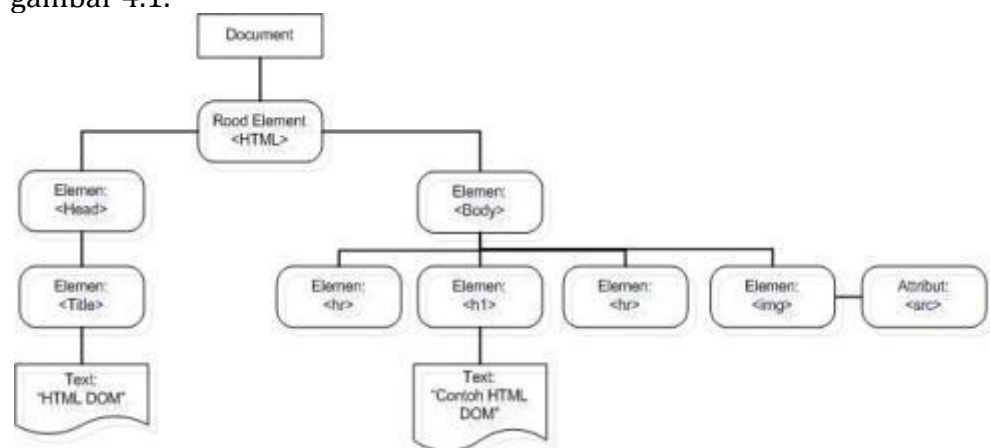
Dom membuat sebuah dokumen HTML sebagai “tree-structur” dengan elemen, atribut dan text.

Contoh Dokumen HTML.

```
<html>
<head><title>HTML DOM</title>
</head>
<body>
<hr />
<H1>Contoh HTML DOM</H1>
<hr />

</body>
</html>
```

Jika diilustrasikan dalam gambar, maka dapat dilihat pada gambar 4.1.



Gambar 4.1

Properties dan Method HTML DOM

Properti adalah nilai yang Anda bisa mendapatkan atau menetapkan (seperti nama atau isi dari node).

Tabel Properties

DOM Properties	Node	Description
nodeName		Berisi nama node
nodeValue		Berisi nilai dalam node, umumnya hanya berlaku untuk node teks
nodeType		berisikan nomor sesuai dengan jenis node, (lihat tabel Node Type)
parentNode		Sebuah referensi ke node induk dari objek, jika ada
childNodes		Akses ke node anak
firstChild		Referensi ke node anak pertama dari elemen, jika ada
lastChild		Referensi ke node anak terakhir dari elemen, jika ada
previousSibling		Referensi ke saudara node sebelum dari node
nextSibling		Referensi ke saudara node selanjutnya dari node
attributes		Daftar atribut untuk elemen dalam Tag
ownerDocument		Referensi ke objek Dokumen HTML di mana unsur yang terkandung dalam Dokumen HTML

Tabel Node Type

Node Type Number	Type	Description	Example
1	Element	Elemen dalam (X)HTML or XML	<p>...</p>
2	Attribute	Attribute untuk elemen HTML or XML	align="center"
3	Text	Sebuah teks yang akan ditutup oleh elemen HTML atau XML	This is a text
8	Comment	Komentar pada HTML	<!-- This is a comment -->

9	Document	Root Dokumen yaitu elemen teratas dari pohon	<html>
10	Document Type	definisi jenis dokumen	<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

Sedangkan metode adalah tindakan yang dapat Anda lakukan (seperti menambahkan atau memodifikasi elemen).

Tabel Method

DOM Node Method	Description	Example
createAttribute(<i>name</i>)	Menciptakan atribut untuk sebuah elemen.	myAlign = document.createAttribute("align");
createComment(<i>string</i>)	Menciptakan sebuah komentar	myComment = document.createComment("Just a comment");
createDocumentFragment()	Menciptakan sebuah fragmen dokumen	myFragment = document.createDocumentFragment(); myFragment.appendChild(temp);
createElement(<i>tagName</i>)	Menciptakan suatu elemen	myHeading = document.createElement("h1");
createTextNode(<i>string</i>)	Menciptakan text node	newText = document.createTextNode("Some new text");

Mengakses HTML DOM

Untuk mengakses elemen pada document HTML, terdapat beberapa cara untuk melakukannya, antara lain:

- **GetElementById**

Syntax:

```
Document.getElementById("Nama ID")
```

Jika nama Id ditemukan, maka metoda ini akan mengembalikan elemen sebagai objek.

Contoh:

```
<html>
<head><title>GetElementById</title></head>
<body>
<p>Hallo</p>
<p id="into"></p>
<hr>

<script>
x=document.getElementById("into");
x.innerHTML="Mari Belajar DOM HTML";
</script>

</body>
</html>
```

- **GetElementByTagName**

Syntax:

```
Document.getElementsByTagName ("Tag Name")
```

Contoh :

```
<html>
<body>
<p>0</p>
<p>1</p>
<p>2</p>

<script>
var y=document.getElementsByTagName('p');
y[0].innerHTML="Hello World";
y[1].innerHTML="Selamat Datang";
y[2].innerHTML="DOM HTML";
</script>

</body>
</html>
```

- **getElementsByClassName()**

Syntax:

```
Document.getElementsByClassName("Class Name")
```

Penggunaan `getElementsByClassName` digunakan jika ada elemen HTML dengan nama kelas yang sama untuk mendapatkan semua daftar elemen dengan class yang sama.

Contoh :

```
<html>
<body>

<p class="para">0</p>
<p>1</p>
<p class="para">2</p>

<script>
x=document.getElementsByClassName("para");
x[0].innerHTML="DOM HTML";
x[1].innerHTML="DOM HTML";
</script>

</body>
</html>
```

Memodifikasi HTML DOM

Terdapat banyak hal yang berbeda dalam memodifikasi HTML DOM, seperti mengubah konten HTML, mengubah gaya CSS, mengubah atribut HTML, membuat elemen HTML baru, menghapus elemen HTML dan mengubah event (handler)

Mengubah konten HTML

Menggunakan properti `innerHTML` merupakan cara termudah untuk mengubah isi dari suatu elemen HTML.

Contoh:

```
<html>
<body>

<p id="p1">paragraph 1</p>

<script>
document.getElementById("p1").innerHTML="Hello World";
</script>

</body>
</html>
```

Mengubah Gaya CSS

Untuk mengakses objek style dalam elemen HTML dengan DOM HTML, dapat digunakan sintaks Berikut:

Syntax:

```
document.getElementById(id).style.property=new style
```

Contoh:

```
<html>
<body>

<p id="p1">Hello world!</p>
<p id="p2">Hello world!</p>

<script>
document.getElementById("p2").style.color="blue";
document.getElementById("p2").style.textTransform="uppercase";
</script>

</body>
</html>
```

Mengubah Atribut HTML

DOM memasok 3 metode berikut untuk mengatasi atribut HTML:

- getAttribute ()
- setAttribute ()
- removeAttribute ()

Metode-metode ini dapat digunakan pada setiap elemen yang mendukung atribut.

- getAttribute () digunakan untuk mengambil nilai dari atribut. Jika atribut tidak ada, string kosong dikembalikan. Sebagai contoh:

```
<html>
<body>


<p id="para">Dapat</p>
<script>
var
getvalue=document.getElementById("myimage").getAttribute("src");
document.getElementById("para").innerHTML=getvalue;
</script>

</body>
</html>
```

- setAttribute () secara dinamis memodifikasi nilai dari sebuah atribut elemen. Metode ini membutuhkan dua parameter yaitu nama atribut dan nilai baru atribut tersebut. Sebagai contoh:

```
<html>
<body>


<script>
document.getElementById("myimage").setAttribute("alt","gambar smiley");
</script>

</body>
</html>
```

- `removeAttribute ()` digunakan untuk menghapus atribut HTML seluruh dari elemen. Sebagai contoh:

```
<html>
<body>


<script>
document.getElementById("myimage").removeAttribute("alt");
</script>

</body>
</html>
```

Membuat Elemen HTML Baru

Untuk menambahkan elemen baru ke DOM HTML, Node elemen harus dibuat lebih dulu dengan `createElement("Tag Name")`, dan kemudian menambahkan ke elemen yang ada. Metode `appendChild(node)` digunakan Untuk menyisipkan node yang ditetapkan pada akhir objek node. Contoh:

```
<html>
<body>

<div id="div1">
<p id="p1">paragraph 1.</p>
<p id="p2">paragraph 2.</p>
</div>

<script>
var para=document.createElement("p");
var node=document.createTextNode("Paragraf Baru.");
para.appendChild(node);

var element=document.getElementById("div1");
element.appendChild(para);
</script>

</body>
</html>
```

Metode `appendChild ()` dalam contoh sebelumnya, menambahkan unsur baru sebagai anak terakhir dari parent. Jika menggunakan metode `insertBefore ("newElement",`

`targetElement`) : Menyisipkan node "newElement" baru sebagai node anak dari simpul dan "targetElement" menentukan di mana "newElement" dimasukkan dalam daftar node anak. Contoh:

```
<html>
<body>

<div id="dvparent">
<div id="dv1">a</div>
<div id="dv2">c</div>
</div>

<script type="text/javascript">

var newd=document.createElement("div");
var oldd=document.getElementById("dv2");
newd.innerHTML="B";
document.getElementById("dvparent").insertBefore(newd,
oldd);

</script>

</body>
</html>
```

Menghapus Elemen HTML

Method *removeChild()* digunakan untuk menghapus elemen HTML dari induk elemen. Elemen HTML yang ingin dihapus, terlebih dahulu induk elemennya harus diketahui. Contoh:

```
<html>
<body>

<div id="dvparent">
<div id="dv1">a</div>
<div id="dv2">c</div>
</div>

<script type="text/javascript">

var hapus=document.getElementById("dv2");
var parentd=document.getElementById("dvparent");

parentd.removeChild(hapus);

</script>

</body>
</html>
```

Mengubah Elemen HTML

Method *replaceChild()* digunakan untuk menggantikan satu node anak dari node dengan node lain. Contoh:

```
<html>
<body>

<div id="adiv">
<span id="innerspan">Text</span>
</div>

<script type="text/javascript">
var oldel=document.getElementById("innerspan")
var newel=document.createElement("p")
var node=document.createTextNode("Paragraf Baru.");
newel.appendChild(node);

document.getElementById("adiv").replaceChild(newel, oldel)
</script>

</body>
</html>
```

Event HTML DOM

HTML DOM memungkinkan JavaScript untuk bereaksi terhadap event HTML. Sebuah JavaScript dapat dieksekusi ketika sebuah event terjadi, seperti ketika mengklik elemen HTML. Untuk mengeksekusi ketika pengguna mengklik elemen, tambahkan kode JavaScript untuk atribut event HTML seperti onclick = JavaScript.

Contoh:

```
<html>
<body>

<h1 onmouseover="this.innerHTML='Mouse Over!'"
onmouseout="this.innerHTML='Mouse Out!'"
onclick="this.innerHTML='Click on this text!'">Click on
this text!</h1>

</body>
</html>
```

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya.

1. Gambarkan Dom Tree dari Dokumen HTML berikut.

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

```

<title>Tugas 1</title>
</head>
<body background=gambar.jpg>
  <font face=arial size=6 color=green>
    <b>Pemrograman Web</b><br>
  </font>
  <font face=arial size=4 color=blue>
    <i>Terus Gapai Impian Walau Hidup Susah</i>
  </font>
  <font face=arial>
    <hr>
    |<a href=satu.html>H O M E</a>
    |<a href=kedua.html>H O B I</a>
    |<a href=ketiga.html>PENDIDIKAN</a>
    |<a href=keempat.html>social networking</a>|
    <hr>
  </font>
  <font face=verdana size=5 color=purple>
    Hobi saya adalah :<br>
    <ul type=circle>
      <li>Basket</li>
      <li>futsal</li>
      <li>Biliard</li>
      <li>komputer dan pemrograman</li>
      <li>Art</li>
    </ul>
  </font>
</body>
</html>

```

2. Buat kode HTML dan JavaScript Untuk menampilkan Tampilan Berikut:

1. Java
2. C++
3. C#

Gambar 3.1

Keterangan:

1. Tombol Append digunakan untuk menambahkan item pada list dan ditambahkan pada akhir list.
2. Isi List baru dimasukkan dari Text Box

3. Tombol remove digunakan untuk menghapus list dan list yang di hapus adalah list no 1
3. Buat kode HTML dan JavaScript Untuk menampilkan Tampilan Berikut:

K-Pop Shop	
List Album	Keterangan
<ul style="list-style-type: none"> • SNSD - I Got A Boy (4th Album) • Super Junior - Sexy, Free and Single (Vol.6) • CN Blue - Re:Blue (4th Mini Album) Special Limited Edition • GD - One Of A Kind 	no image

Gambar 3.2

Keterangan :

1. Jika list “SNSD – I Got A Boy (4th Album)” diklik, maka image Album tersebut akan ditampilkan beserta keterangannya(seperti Gambar 3.3.)
 Harga: Rp. 205,000
 Details :
 1. 10 Cover (9 Members + 1 Group)
 2. Poster + free paralon / tube
 3. READY STOCK all cover
 4. Bonus
2. Jika list “Super Junior - Sexy, Free and Single (Vol.6)” diklik maka image Album tersebut akan ditampilkan beserta keterangannya :
 Harga: Rp. 255,000
 Details :
 1. Photos 84 pages
 2. Free Poster + Paralon
 3. Berat 850 gram
 4. Free gift

3. Jika list “CN Blue - Re:Blue (4th Mini Album) Special Limited Edition” diklik maka image Album tersebut akan ditampilkan beserta keterangannya :

Harga: Rp. 245,000

Details :

1. 4 Cover
2. MV making DVD
3. 104p photobook
4. Free gift

4. Jika list “GD - One Of A Kind” diklik maka image Album tersebut akan ditampilkan beserta keterangannya :

Harga: Rp. 210,000

Details :

1. Random cover
2. Free Poster + Paralon

K-Pop Shop	
List Album	Keterangan
<ul style="list-style-type: none"> ▪ SNSD - I Got A Boy (4th Album) ▪ Super Junior - Sexy, Free and Single (Vol.6) ▪ CN Blue - Re:Blue (4th Mini Album) Special Limited Edition ▪ GD - One Of A Kind 	 <p>Harga: Rp. 205,000</p> <p>Details :</p> <ol style="list-style-type: none"> 1. 10 Cover (9 Members + 1 Group) 2. Poster + free paralon / tube 3. READY STOCK all cover 4. Bonus

Gambar 3.3

DHTML

Dynamic HTML, atau DHTML, merupakan istilah umum untuk sekumpulan teknologi yang digunakan bersama-sama untuk membuat situs web interaktif dan memiliki animasi dengan menggunakan kombinasi bahasa markup statis (seperti HTML), bahasa scripting (seperti JavaScript) , bahasa definisi presentasi (seperti CSS), dan Document Object Model. DHTML mendukung isi dan visualisasi dokumen yang dinamis.

DHTML WINDOWS

Window adalah suatu object yang berfungsi untuk menghasilkan window baru pada web page.

Syntax untuk membuka window baru:

```
window.open(url, name, features, replace)
```

Dimana:

- *url* → dokumen yang akan ditampilkan pada window baru
- *name* → nama dari window
- *features* → string yang terdiri atas fitur-fitur dari window yang dipisahkan dengan tanda titik koma (;)
- *replace* → optional, jika true maka dokumen baru akan menimpa isi dari window yang dibuka, dan sebaliknya jika false. (berlaku pada window yang telah diciptakan sebelumnya)

Contoh:

-Langkah 1: Ketikkan dokumen HTML Berikut dan simpan dengan nama DHTML1.html

```
<html>
<head>
<title>Second Window</title>
</head>
<body>
<h1 align='center'>Another window!
</h1>
<hr>
</body>
</html>
```

DHTML

- Langkah 2: Untuk menampilkan DHTML1.html pada window baru ketikkan Ketikkan dokumen HTML Berikut dan simpan dengan nama DHTML2.html

```
<html>
<head>
<title>Window Tester</title>
<meta http-equiv="content-type" content="text/html;
charset=ISO-8859-1" />
<script type="text/javascript">
function createWindow()
{
    secondwindow = window.open(' DHTML2.html',
    'example', 'height=300,width=200,scrollbars=yes');
    secondwindow.focus();
}
</script>
</head>
<body>
<form>
<input type="button" value="new window"
onclick="createWindow();" />
</form>
</body>
</html>
```

Cookie

Cookie adalah suatu informasi kecil berupa merupakan serangkaian teks yang diberikan oleh web server untuk di simpan di browser client untuk mengumpulkan informasi pengguna. Cookie dapat digunakan untuk mengetahui beberapa hal, misalnya:

- Menyimpan tanggal terakhir kunjungan ke web site
- Berapa kali kunjungan ke web site telah dilakukan
- Sudah pernah atau tidak melakukan jajak pendapat yg hanya boleh di lakukan sekali

Syntax Cookie :

```
cookieName=cookieValue; expires=expirationDateGMT;
path=urlPath; domain=siteDomain; secure
```

Dimana:

- *cookieName* → Mengatur nama cookie dalam value string.
- *expires* → Mengatur tanggal berakhirnya cookie.
- *path* → menunjukkan subset dari jalur di domain yang cookie akan dikembalikan.
- *domain* → Menetapkan domain untuk cookie
- *secure* → Menunjukkan bahwa cookie hanya harus dikembalikan melalui koneksi HTTPS

document.cookie

Cookie dapat dibuat, dibaca dan dihapus dengan JavaScript. Cookie dapat diakses melalui properti *document.cookie*.

Yang diperlukan untuk membuat cookie adalah mengisi/memberikan informasi yang dibutuhkan di *document.cookie*.

Contoh Write Cookie:

```
<html>
<head>
<title>Cookie</title>
<script type="text/javascript">

function setcookie()
{
    var user = document.getElementById("nama").value;
    var expire = new Date();
    expire.setMonth(expire.getMonth() + 6);
    document.cookie = "userName=" + user
    +";path=/;expires=" + expire.toGMTString();
    alert("Nama Anda Selesai Diregister");
}

</script>
</head>
<body>
<form action=#>
<h1> Masukkan Nama: <input type="text" id="nama"/></h1>
<input type="button" value="Register"
onclick="javascript:setcookie();" />
</form>
</body>
</html>
```

Untuk membaca cookie dapat dilakukan dengan menampilkan/memanggil *document.cookie*.

Contoh Read Cookie:

```
<html>
<head>
<title>Cookie</title>
<script type="text/javascript">
function ReadCookie()
{
    var allcookies = document.cookie;
    alert("All Cookies : " + allcookies );

    // Get all the cookies pairs in an array
    cookiearray = allcookies.split(';');

    // Now take key value pair out of this array
    for(var i=0; i<cookiearray.length; i++){
        name = cookiearray[i].split('=')[0];
        value = cookiearray[i].split('=')[1];
    }
}
```



```
        alert("Key is : " + name + " and Value is : " +
value);
    }
}

</script>
</head>
<body>
<form name="myform" action="">
<input type="button" value="Get Cookie"
onclick="ReadCookie()" />
</form>
</body>
</html>
```

Delete cookie dapat dilakukan dengan membuat expiredate dari cookie tersebut melewati tanggal kadaluarsanya. Browser akan melihat bahwa tanggal kadaluwarsa yang telah berlalu dan segera menghapus cookie tersebut.

Contoh Delete Cookie:

```
<html>
<head>
<title>Cookie</title>
<script type="text/javascript">

function setcookie()
{
    var user = document.getElementById("nama").value;
    var expire = new Date();
    expire.setMonth(expire.getMonth() - 6);
    document.cookie = "userName=" + user
    +";path=/;expires=" + expire.toGMTString();
    alert("Cookie Anda Selesai Dihapus");
}

</script>
</head>
<body>
<form action=#>
<h1> Masukkan Nama: <input type="text" id="nama"/></h1>
<input type="button" value="Delete"
onclick="javascript:setcookie();" />
</form>
</body>
</html>
```

Cookie dapat digunakan untuk menghitung berapa banyak jumlah kunjungan seseorang ke web side, yaitu dengan menggunakan Syntax Count Cookie Berikut:

Contoh Count Cookie:

```

<html>
<head>
<title>Cookie</title>
<script type="text/javascript">
function tampilsambutan()
{
    if (document.cookie == "")
    {
        document.write("Anda Harus Register Sebelum
        Menggunakan Web Site Ini");
        return;
    }
    else
    {
        var msg="", nama, count;
        nama = ambilnama();
        if (nama == "")
        {
            document.write("Anda Harus Register Sebelum
            Menggunakan Web Site Ini");
            return;
        }

        count = ambilcount();
        var expire = new Date();
        expire.setMonth(expire.getMonth() + 6);
        if (count == 0)
        {
            document.write("Selamat Datang, " + nama
            + "!!! Ini pertama kali anda mengunjungi
            website ini.");
            document.cookie = "count=" + 1 +
            ";path=/;expires=" +
            expire.toGMTString();
        }
        else
        {
            count++;
            document.write("Selamat Datang, " + nama
            + "!!! Anda telah mengunjungi website ini
            sebanyak " + count + " kali.");
            document.cookie = "count=" + count +
            ";path=/;expires=" +
            expire.toGMTString();
        }
    }
}

function ambilnama()
{
    var y=document.cookie.split("; ");
    for (i=0;i<y.length;i++)
    {
        var z = y[i].split("=");
        if (z[0] == "userName")
            return z[1];
    }
}

```

```

        return "";
    }

    function ambilcount()
    {
        var y=document.cookie.split("; ");
        for (i=0;i<y.length;i++)
        {
            var z = y[i].split("=");
            if (z[0] == "count")
                return z[1];
        }
        return 0;
    }
</script>
</head>
<body onload="javascript:tampilsambutan()">
</body>
</html>

```

setTimeout & clearTimeout

setTimeout & *clearTimeout* digunakan untuk memproses sebuah statement atau function secara berkala. *setTimeout* adalah method dari window yang dapat memproses suatu function secara berkala. *clearTimeout* berfungsi untuk menghentikan *setTimeout*.

Syntax:

```

TimerID = setTimeout(function/exp, milisecond)
clearTimeout (TimerID)

```

Contoh *setTimeout* & *clearTimeout*:

```

<html>
<head>
<script language="Javascript">
var timeout;

function timeout_trigger() {
    document.getElementById('timeout_text').innerHTML =
    "Time Out Selesai";
}

function timeout_init() {
    timeout = setTimeout('timeout_trigger()', 3000);
    document.getElementById('timeout_text').innerHTML =
    "Time Out Dijalankan";
}

function timeout_clear() {
    clearTimeout(timeout);
    document.getElementById('timeout_text').innerHTML =
    "Time Out Berhenti";
}

```

```

</script>
</head>
<body>
  <input type="button" value="test timeout"
    onclick="timeout_init()" />
  <input type="button" value="clear timeout"
    onclick="timeout_clear()" />
  <p id="timeout_text"></p>
</body>
</html>

```

setInterval & clearInterval

setInterval dan *clearInterval* memiliki fungsi yang sama dengan *Timeout*. Perbedaan *Interval* dan *Timeout* adalah *Timeout* akan menjalankan function/ekspresi hanya sekali, kemudian proses *Timeout* akan berhenti. Sedangkan *Interval* akan menjalankan function/ekspresi secara berkala tanpa berhenti

Syntax:

```

TimerID = setInterval(function/exp, milisecond)
clearInterval (TimerID)

```

Contoh *setInterval* dan *clearInterval*:

```

<html>
<head>
<script>
var myVar;

function myFunction()
{
myVar=setInterval(function() {myTimer() },1000);
}

function myTimer()
{
var d=new Date();
var t=d.toLocaleTimeString();
document.getElementById("demo").innerHTML=t;
}

function myStopFunction()
{
clearInterval(myVar);
}
</script>

</head>
<body>
<button onclick="myFunction()">Try it</button>
<button onclick="myStopFunction()">Stop time</button>
<p id="demo"></p>
</body>
</html>

```

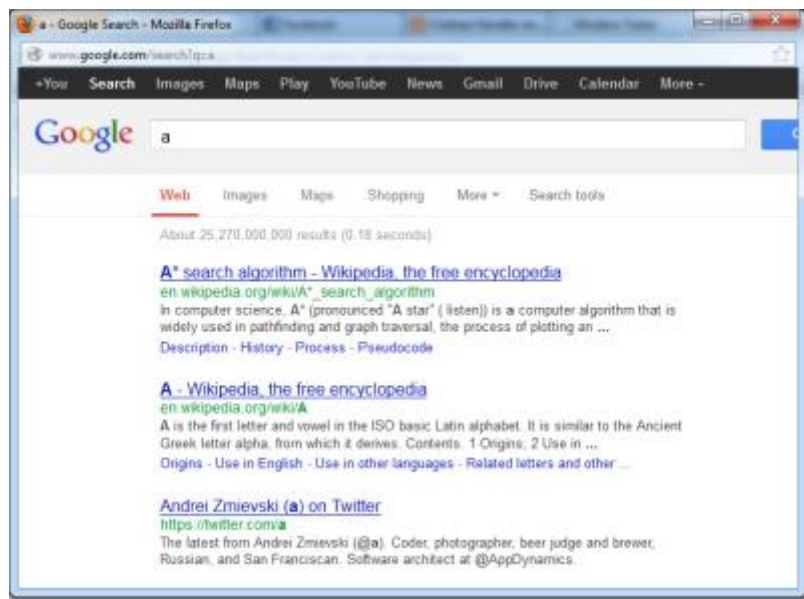
Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya.

1. Buat kode JavaScript dan Html Untuk Membuat tampilan seperti Gambar 4.1



Gambar 4.1



Gambar 4.2

Keterangan:

- a) Setiap kata yang dimasukkan, akan dihubungkan ke Google.com untuk dilakukan pencarian terhadap kata kunci yang dimasukkan.
- b) Hasil pencarian akan ditampilkan pada window baru seperti pada Gambar 4.2.

2. Buat kode JavaScript dan Html Untuk Membuat tampilan seperti Gambar 4.3

-
- Kampus
 - Profil

Gambar 4.3

Keterangan:

- a) Tanda “disc” pada list menandakan ada menu lain.
- b) Tanda “circle” pada list menandakan tidak ada Menu lain

- c) Jika Tanda “disc” diklik maka menu yang bersangkutan akan ditampilkan sehingga jika semua tanda “disc” diklik maka hasilnya akan menjadi seperti Gambar 4.4.
- d) Jika tanda “circle” diklik maka menu akan disembunyikan.

- Kampus
 - Lokasi
 - Sarana dan Prasarana
- Profil
 - Sejarah
 - Sikap Budaya, Visi dan Misi, dan Tujuan
 - Sikap Budaya
 - Visi dan Misi
 - Tujuan

Gambar 4.4

3. Buat kode JavaScript dan Html Untuk Membuat Menu Drop-down seperti Gambar 4.5

Search Sites	E-commerce Sites	Book Related Sites
		JavaScriptRef
		W3C
		PINT

Gambar 4.5

Keterangan:

- a) Pada Search List, terdapat menu Drop Down berupa Link, ke Google(www.google.co.id), Yahoo (www.Yahoo.co.id) , Teoma (www.teoma.com), MSN (<http://plasa.msn.com>), dan dmoz (www.dmoz.org)
- b) Pada Menu E-commerce terdapat menu Drop Down berupa Link, ke Amazon(www.amazon.com), Ebuy(www.ebay.com), dan Buy.Com (www.buy.com)
- c) Pada Menu Book Related Site terdapat menu Drop Down berupa Link, ke JavaScriptRef (www.javascript.com), W3C(www.w3c.org) , dan PINT(www.pint.com)

JQUERY

JQuery adalah javascript library yang cepat dan ringan untuk menangani dokumen HTML, menangani event, membuat animasi dan interaksi ajax. jQuery dirancang untuk memperingkas kode-kode javascript (“write less, do more”) dengan mengubah cara penulisan javascript. Tujuannya adalah untuk memberikan kemudahan dalam menggunakan JavaScript. jQuery juga menyederhanakan banyak hal-hal rumit dari JavaScript, seperti panggilan AJAX dan manipulasi DOM.

Memulai JQuery

Untuk memulai menjalankan jQuery, file JQuery harus dimasukkan ke file HTML. File JQuery “jQuery.js” dapat diunduh melalui situs <http://www.jquery.com> dan tempatkan file yang didownload dalam direktori yang sama dengan halaman di mana htmlnya berada. Setiap menulis kode javascript yang menggunakan jquery, file jquery.js tersebut harus dimasukkan kedalam kode javascript dengan syntax:

```
<head>
<script type="text/javascript" src="jquery.js"></script>
</head>
```

File JQuery juga dapat digunakan dengan menyertakannya dari CDN (Content Delivery Network) baik dari Google dan Microsoft. Untuk menggunakan jQuery dari Google atau Microsoft, maka File html harus terhubung dengan internet dan ditambahkan syntax:

```
<head>
<script
src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.
min.js">
</script>
</head>
```

Penulisan JQuery

Sintaks jquery dibuat untuk memilih elemen-elemen dalam HTML dan melakukan aksi terhadap elemen yang dipilih.

Syntax :

```
$(selector).action()
```

Dimana:

- Tanda dollar (\$), untuk mendefinisikan jQuery
- (selector), untuk menunjukkan elemen yang dipilih atau dituju
- action(), adalah jQuery action yang akan dilakukan terhadap elemen yang dipilih.

Contoh:

```
$(this).hide()      :menyembunyikan elemen saat ini
$("p").hide()       :menyembunyikan semua paragraf atau konten
                    :dari tag <p>
$(".test").hide()   :menyembunyikan elemen yang mempunya
                    :class="test"
$("#test").show()   :menampilkan elemen yang mempunyai id="test"
```

Karena hampir segala sesuatu yang menggunakan jQuery merupakan document object model (DOM), perlu diperlukan sebuah event ready setelah DOM siap. Untuk melakukan hal ini, tambahkan kode ready untuk dokumen.

```
$(document).ready(function() {
//kode anda di sini
});
```

Kode di atas berarti kode akan dijalankan apabila halaman HTML telah di-load semuanya.

cth:

```
$(document).ready(function() {
    $(".tombol1").click(function() {
        $("p").hide(1000);
    });
});
```

Kode \$(".tombol1") adalah jQuery selector untuk memilih elemen dengan class="tombol1". Fungsi click() berguna untuk memberikan event onclick untuk elemen yang dipilih (elemen yang mempunya class='tombol1'). apabila event onclick terjadi maka jquery akan melaksanakan fungsi yang diberikan. Jadi kode diatas dapat diartikan apabila elemen dengan class="tombol1" diklik maka lakukan fungsi \$("p").hide(1000). Fungsi hide() dan show() adalah fungsi built in dari jQuery.

jQuery Selectors

Selectors digunakan untuk memanipulasi elemen HTML sebagai kelompok atau sebagai elemen tunggal. jQuery untuk element selectors dan attribute selectors digunakan untuk memilih elemen HTML (atau kelompok elemen) dengan nama tag, nama atribut atau konten sehingga elemen HTML tersebut dapat dimanipulasi.

jQuery Element Selectors

jQuery mirip CSS dalam hal memilih elemen HTML. Contoh:

`$("p")` memilih semua elemen `<p>`

`$("p.intro")` memilih semua elemen `<p>` yang mempunyai `class="intro"`.

`$("p#demo")` memilih elemen `<p>` yang mempunyai `id="demo"`.

jQuery Attribute Selectors

Contoh:

`$("[href]")` memilih semua elemen dengan atribut `href`.

`$("[href='#']")` memilih semua elemen dengan atribut `href` bernilai `"#"`.

`$("[href!='#']")` memilih semua elemen dengan atribut `href` dengan nilai tidak sama dengan `"#"`.

`$("[href$='.jpg']")` memilih semua elemen dengan atribut `href` yang mengandung `".jpg"`.

Berikut tabel daftar jQuery Selector.

Tabel daftar jQuery Selector.

Selector	Contoh	Yang dipilih
*	<code>\$("*")</code>	Semua elemen
#id	<code>\$("#lastname")</code>	Elemen yang mempunyai <code>id=lastname</code>
.class	<code>\$(".intro")</code>	Semua elemen yang mempunyai <code>class="intro"</code>
element	<code>\$("p")</code>	Semua elemen <code><p></code>
.class.class	<code>\$(".intro.demo")</code>	Semua elemen yang mempunyai <code>class=intro</code> dan <code>class=demo</code>
:first	<code>\$("p:first")</code>	Elemen <code><p></code> yang pertama
:last	<code>\$("p:last")</code>	Elemen <code><p></code> yang terakhir
:even	<code>\$("tr:even")</code>	Semua elemen <code><tr></code> yang genap
:odd	<code>\$("tr:odd")</code>	Semua elemen <code><tr></code> yang ganjil
:eq(index)	<code>\$("ul li:eq(3)")</code>	Elemen ke-empat dari suatu list (index starts at 0)
:gt(no)	<code>\$("ul li:gt(3)")</code>	Elemen-elemen dari suatu list dengan index lebih besar dari 3
:lt(no)	<code>\$("ul li:lt(3)")</code>	Elemen-elemen dari suatu list dengan index kecil dari 3
:not(selector)	<code>\$("input:not(:empty)")</code>	Semua input elemen yang tidak kosong
:header	<code>\$(":header")</code>	Semua elemen header <code><h1><h2>...</code>
:animated		Semua elemen animasi

:contains(text)	<code>\$(":contains('W3Schools'))"</code>	Semua elemen yang mengandung teks 'W3Schools'
:empty	<code>\$(":empty")</code>	Semua elemen yang tidak mempunyai child (elements) nodes
:hidden	<code>\$("p:hidden")</code>	Semua elemen <code><p></code> yang tersembunyi
:visible	<code>\$("table:visible")</code>	Semua tabel yang visible
:s1,s2,s3	<code>\$("th,td,intro")</code>	Semua elemen yang cocok dengan th, td, intro
[attribute]	<code>\$("[href]")</code>	Semua elemen yang mempunyai atribut href
[attribute=value]	<code>\$("[href='#']")</code>	Semua elemen yang mempunyai atribut href bernilai="#"
[attribute!=value]	<code>\$("[href!='#']")</code>	Semua elemen yang mempunyai atribut href bernilai <"#"
[attribute\$=value]	<code>\$("[href\$='.jpg']")</code>	Semua elemen yang mempunyai atribut href yang nilainya mengandung ".jpg"
:input	<code>\$(":input")</code>	Semua elemen <code><input></code>
:text	<code>\$(":text")</code>	Semua elemen <code><input></code> dengan <code>type="text"</code>
:password	<code>\$(":password")</code>	Semua elemen <code><input></code> dengan <code>type="password"</code>
:radio	<code>\$(":radio")</code>	Semua elemen <code><input></code> dengan <code>type="radio"</code>
:checkbox	<code>\$(":checkbox")</code>	Semua elemen <code><input></code> dengan <code>type="checkbox"</code>
:submit	<code>\$(":submit")</code>	Semua elemen <code><input></code> dengan <code>type="submit"</code>
:reset	<code>\$(":reset")</code>	Semua elemen <code><input></code> dengan <code>type="reset"</code>
:button	<code>\$(":button")</code>	Semua elemen <code><input></code> dengan <code>type="button"</code>
:image	<code>\$(":image")</code>	Semua elemen <code><input></code> dengan <code>type="image"</code>
:file	<code>\$(":file")</code>	Semua elemen <code><input></code> dengan <code>type="file"</code>
:enabled	<code>\$(":enabled")</code>	Semua elemen <code><input></code> yang enabled
:disabled	<code>\$(":disabled")</code>	Semua elemen <code><input></code> yang disabled
:selected	<code>\$(":selected")</code>	Semua elemen <code><input></code> yang terseleksi
:checked	<code>\$(":checked")</code>	Semua elemen <code><input></code> yang ter-check

jQuery Events

Salah satu kemampuan utama jquery adalah menangani event. Dalam pemrograman jquery, jika selector digunakan untuk mencari elemen-elemen apa saja yang akan dilakukan operasi terhadapnya, maka event memutuskan kapan dilakukan operasi tersebut, misalnya elemen yang telah dipilih (Selectors) akan dilakukan operasi

terhadapnya setelah diklik (click) atau setelah klik 2x (dblclick), setelah mouse berada diatas objek (hover), dan sebagainya. pada contoh

Cth:

```
$("document").ready(function() {
    $('a').click(function() {
        alert("Hello World");
    });
});
```

Berikut daftar event-event yang dapat terjadi dari elemen HTML.

Event function

<div> <div>\$(document).ready(function)</div> <div>\$(selector).blur(function)</div> <div>\$(selector).change(function)</div> <div>\$(selector).dblclick(function)</div> <div>\$(selector).error(function)</div> <div>\$(selector).focus(function)</div> <div>\$(selector).keydown(function)</div> <div>\$(selector).keypress(function)</div> <div>\$(selector).keyup(function)</div> <div>\$(selector).load(function)</div> <div>\$(selector).mousedown(function)</div> </div>		<div> <div>\$(selector).mouseenter(function)</div> <div>\$(selector).mouseleave(function)</div> <div>\$(selector).mousemove(function)</div> <div>\$(selector).mouseout(function)</div> <div>\$(selector).mouseover(function)</div> <div>\$(selector).mouseup(function)</div> <div>\$(selector).resize(function)</div> <div>\$(selector).scroll(function)</div> <div>\$(selector).select(function)</div> <div>\$(selector).submit(function)</div> <div>\$(selector).unload(function)</div> </div>	
Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

Efek-Efek dengan jQuery

Salah satu kemampuan jquery adalah, adanya fungsi efek yang siap pakai seperti menampilkan dan menyembunyikan suatu elemen, menggerakkan elemen, dan menganimasikan elemen tersebut.

Berikut adalah efek-efek siap pakai yang disediakan oleh jquery.

jQuery show() Effect

Berguna untuk menampilkan elemen yang tersembunyi. Untuk mengatur elemen yang tersembunyi melalui CSS adalah `display:none` (bukan `visibility:hidden`).

Syntax :

JQUERY

```
$(selector).show(speed, callback)
```

Parameter	Deskripsi
speed	Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0. Nilainya bisa berupa: <ul style="list-style-type: none">• milliseconds (contoh : 1500)• "slow"• "normal"• "fast"
callback	Opsional. Suatu fungsi yang akan dijalankan apabila efek show selesai dijalankan.

Contoh :

```
<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script type="text/javascript">
$(document).ready(function() {
    $(".tombol1").click(function() {
        $("p").show(1000, tampil);
    });
});
function tampil() {
    document.getElementById("ket").innerHTML="Paragrafnya
    muncul";
}
</script>
</head>
<body>
<p style="display:none">Ini adalah paragraph
tersembunyi.</p>
<button class="tombol1">Show</button>
<p id="ket"></p>
</body>
</html>
```

jQuery hide() Effect

Berguna untuk menyembunyikan elemen yang dipilih.

Syntax :

```
$(selector).hide(speed, callback)
```

Parameter	Deskripsi
speed	<p>Opsional. Menentukan kecepatan elemen muncul dari visible ke hidden. Defaultnya adalah 0.</p> <p>Nilainya bisa berupa:</p> <ul style="list-style-type: none"> • milliseconds (contoh : 1500) • "slow" • "normal" • "fast"
callback	Opsional. Suatu fungsi yang akan dijalankan apabila efek show selesai dijalankan.

Contoh :

```
<html>
<head>
<script src=jquery-1.9.1.min.js">
</script>
<script type="text/javascript">
$(document).ready(function() {
    $(".tombol1").click(function() {
        $(".p").hide(1000,tampilkan);
    });
});
function tampilkan() {
    document.getElementById("ket").innerHTML="Paragrafnya
    diatas telah disembunyikan";
}
</script>
</head>
<body>
<p>paragraf akan di sembunyi.</p>
<button class="tombol1">Show</button>
<div id="ket"></div>
</body>
</html>
```

jQuery toggle() Effect

jQuery toggle() Adalah gabungan fungsi hide dan show. Jadi toggle() berfungsi menampilkan yang tersembunyi, menyembunyikan yang tampak.

Syntax :

```
$(selector).toggle(speed,callback,switch)
```

Parameter	Deskripsi
speed	Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0. Nilainya bisa berupa: <ul style="list-style-type: none"> • milliseconds (contoh : 1500) • "slow" • "normal" • "fast"
callback	Opsional. Suatu fungsi yang akan dijalankan apabila efek show selesai dijalankan.
switch	Opsional. Bernilai Boolean <ul style="list-style-type: none"> • True, hanya untuk menampilkan semua elemen • False, hanya menyembunyikan semua elemen Jika parameter ini diset, parameter speed dan callback parameters tidak bisa digunakan.

Contoh :

```

<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script type="text/javascript">
$(document).ready(function() {
$(".tombol1").click(function() {
$("p").toggle();
});
});
</script>
</head>
<body>
<p>Ini adalah paragraf</p>
<button class="tombol1">Tampilkan paragraf</button>
</body>
</html>

```

jQuery slideDown() Effect

Berguna untuk menampilkan element yang tersembunyi dengan efek sliding.

Syntax :

```
$(selector).slideDown(speed,callback)
```

Parameter	Deskripsi
speed	<p>Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0.</p> <p>Nilainya bisa berupa:</p> <ul style="list-style-type: none"> • milliseconds (contoh : 1500) • "slow" • "normal" • "fast"
callback	Opsional. Suatu fungsi yang akan dijalankan apabila efek show selesai dijalankan.

Contoh :

```

<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script>
$(document).ready(function() {
    $("#flip").click(function() {
        $("#panel").slideDown("slow");
    });
});
</script>

<style type="text/css">
#panel, #flip
{
padding: 5px;
text-align: center;
border: solid 1px #c3c3c3;
}
#panel
{
padding: 50px;
display: none;
}
</style>
</head>
<body>
<div id="flip">Click</div>
<div id="panel">Hello world!</div>
</body>
</html>

```

jQuery slideUp() Effect

Berguna untuk menyembunyikan element dengan efek sliding.

Syntax :

```
$(selector).slideUp(speed, callback)
```

Parameter	Deskripsi
speed	<p>Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0.</p> <p>Nilainya bisa berupa:</p> <ul style="list-style-type: none"> • milliseconds (contoh : 1500) • "slow" • "normal" • "fast"
callback	Opsional. Suatu fungsi yang akan dijalankan apabila efek show selesai dijalankan.

Contoh :

```

<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script>
$(document).ready(function() {
    $("#flip").click(function() {
        $("#panel").slideUp("slow");
    });
});
</script>

<style type="text/css">
#panel, #flip
{
padding: 5px;
text-align: center;
border: solid 1px #c3c3c3;
}
#panel
{
padding: 50px;
}
</style>
</head>
<body>
<div id="flip">Click</div>
<div id="panel">Hello world!</div>
</body>
</html>

```

jQuery slideToggle() Effect

jQuery slideToggle() merupakan gabungan antara slideDown() dan slideUp(). Menyembunyikan elemen jika dalam keadaan visible, menampilkan elemen jika dalam keadaan hidden dengan efek sliding.

Syntax :

JQUERY

```
$(selector).slideToggle(speed,callback)
```

Parameter	Deskripsi
speed	Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0. Nilainya bisa berupa: <ul style="list-style-type: none">• milliseconds (contoh : 1500)• "slow"• "normal"• "fast"
callback	Opsional. Suatu fungsi yang akan dijalankan apabila efek show selesai dijalankan.

Contoh :

```
<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script>
$(document).ready(function() {
    $("#flip").click(function() {
        $("#panel").slideToggle("slow");
    });
});
</script>

<style type="text/css">
#panel,#flip
{
padding:5px;
text-align:center;
border:solid 1px #c3c3c3;
}
#panel
{
padding:50px;
display:none;
}
</style>
</head>
<body>
<div id="flip">Click</div>
<div id="panel">Hello world!</div>
</body>
</html>
```

jQuery fadeIn() Effect

Berguna untuk menampilkan elemen yang dipilih jika tersembunyi, dengan efek memudar.

Syntax :

```
$(selector).fadeIn(speed,callback)
```

Parameter	Deskripsi
speed	<p>Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0.</p> <p>Nilainya bisa berupa:</p> <ul style="list-style-type: none"> • milliseconds (contoh : 1500) • "slow" • "normal" • "fast"
callback	Opsional. Suatu fungsi yang akan dijalankan apabila efek show selesai dijalankan.

Contoh :

```

<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script>
$(document).ready(function() {
    $("button").click(function() {
        $("#div1").fadeIn();
    });
});
</script>
</head>

<body>
<button>Click to fade</button>
<br><br>
<div id="div1" style="width:180px;height:180px;
background-color:blue; display:none;"></div>
</body>
</html>

```

jQuery fadeOut() Effect

Berguna untuk menampilkan element yang dipilih dengan efek memudar.

Syntax :

```
$(selector).fadeOut(speed,callback)
```

Parameter	Deskripsi
speed	<p>Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0.</p> <p>Nilainya bisa berupa:</p> <ul style="list-style-type: none"> • milliseconds (contoh : 1500) • "slow" • "normal" • "fast"
callback	Opsional. Suatu fungsi yang akan dijalankan apabila efek show selesai dijalankan.

Contoh :

```
<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script>
$(document).ready(function() {
    $("button").click(function() {
        $("#div1").fadeOut();
    });
});
</script>
</head>

<body>
<button>Click to fade</button>
<br><br>
<div id="div1" style="width:180px;height:180px;
background-color:blue;"></div>
</body>
</html>
```

jQuery fadeToggle() Effect

jQuery fadeToggle() merupakan gabungan antara fadeIn() dan fadeOut(). Menyembunyikan elemen jika dalam keadaan visible, menampilkan elemen jika dalam keadaan hidden dengan efek memudar.

Syntax :

```
$(selector).fadeToggle(speed,callback)
```

Parameter	Deskripsi
speed	<p>Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0.</p> <p>Nilainya bisa berupa:</p> <ul style="list-style-type: none"> • milliseconds (contoh : 1500) • "slow"

JQUERY

	<ul style="list-style-type: none">• "normal"• "fast"
callback	Opsional. Suatu fungsi yang akan dijalankann apabila efek show selesai dijalankan.

Contoh :

```
<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script>
$(document).ready(function() {
    $("button").click(function() {
        $("#div1").fadeToggle("slow");
    });
});
</script>
</head>

<body>
<button>Click to fade</button>
<br><br>
<div id="div1" style="width:180px;height:180px;
background-color:blue;"></div>
</body>
</html>
```

jQuery fadeTo() Effect

jQuery fadeTo() berguna untuk mengatur tingkat keputaran elemen terpilih menuju tingkat opacity yang ditentukan.

Syntax :

```
$(selector).fadeTo(speed,opacity,callback)
```

Parameter	Deskripsi
speed	Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0. Nilainya bisa berupa: <ul style="list-style-type: none">• milliseconds (contoh : 1500)• "slow"• "normal"• "fast"
callback	Opsional. Suatu fungsi yang akan dijalankann apabila efek show selesai dijalankan.
opacity	menentukan opacity dari elemen dengan nilai 0 sampai 1.

Contoh :

```

<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script>
$(document).ready(function() {
    $("button").click(function() {
        $("#div1").fadeTo("slow",0.15);
    });
});
</script>
</head>

<body>
<button>Click to fade</button>
<br><br>
<div id="div1" style="width:180px;height:180px;
background-color:blue;"></div>
</body>
</html>

```

jQuery animate() Effect

jQuery animate() berguna mengubah suatu elemen dari satu keadaan ke keadaan yang lainnya berdasarkan CSS. Nilai properti CSS yang berubah secara berangsur-angsur akan menciptakan efek animasi. Nilai properti CSS yang bisa diubah adalah nilai bertipe angka, baik satuannya pixel atau persen(%). Untuk tipe string tidak bisa dianimasikan.

Syntax :

```
$(selector).animate(styles,speed,easing,callback)
```

Parameter	Deskripsi
styles	<p>Wajib. Menentukan properti CSS dan nilainya yang akan di-animasi.</p> <p>Properti CSS yang bisa di-animasi :</p> <ul style="list-style-type: none"> • backgroundPosition • borderWidth • borderBottomWidth • borderLeftWidth • borderRightWidth • borderTopWidth • borderSpacing • margin • marginBottom • marginLeft • marginRight

	<ul style="list-style-type: none"> • marginTop • outlineWidth • padding • paddingBottom • paddingLeft • paddingRight • paddingTop • height • width • maxHeight • maxWidth • minHeight • maxwidth • font • fontSize • bottom • left • right • top • letterSpacing • wordSpacing • lineHeight • textIndent
speed	<p>Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0.</p> <p>Nilainya bisa berupa:</p> <ul style="list-style-type: none"> • milliseconds (contoh : 1500) • "slow" • "normal" • "fast"
easing	<p>Opsional. Menentukan fungsi easing yang diset pada speed animasi.</p> <p>Built in fungsi easing adalah:</p> <ul style="list-style-type: none"> • swing • linear
callback	<p>Opsional. Suatu fungsi yang akan dijalankan apabila efek show selesai dijalankan.</p>

Contoh :

```

<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script type="text/javascript">
$(document).ready(function()
{
$(".tombol1").click(function(){
$("#box").animate({height:"300px",opacity:"0",width:"150px"},3000);
});
$(".tombol2").click(function(){
$("#box").animate({height:"100px",opacity:"1",width:"100px"});
});
});
</script>
</head>
<body>
<div id="box"
style="background:blue;height:100px;width:100px;margin:6px;">
</div>
<button class="tombol1">Animasi</button>
<button class="tombol2">Reset</button>
</body>
</html>

```

jQuery stop() Effect

jQuery stop() berguna untuk menghentikan animasi yang sedang berjalan.

Syntax :

```
$(selector).stop(stopAll,goToEnd)
```

Parameter	Deskripsi
stopAll	Opsional. Bernilai boolean, menentukan apakah menghentikan semua animasi, termasuk yang ngantri untuk dijalankan pada elemen yang dipilih.
goToEnd	Opsional. Bernilai Boolean, menentukan apakah animasi yang sedang jalan diselesaikan atau tidak. Parameter ini hanya bisa digunakan jika parameter stopAll di-set.

Contoh :

```

<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script type="text/javascript">
$(document).ready(function() {
    $("#start").click(function() {
        $("#box").animate({height:300},"slow");
        $("#box").animate({width:300},"slow");
        $("#box").animate({height:100},"slow");
        $("#box").animate({width:100},"slow");
    });
    $("#stop").click(function() {
        $("#box").stop(true,false);
    });
});
</script>
</head>
<body>
<p><button id="start">Start Animasi</button>
<button id="stop">Stop Animasi</button></p>
<div id="box"
style="background:#ababab;height:100px;width:100px;positi
on:relative">
</div>
</body>
</html>

```

jQuery delay() Effect

jQuery delay() berguna untuk menunda fungsi yang akan dijalankan untuk elemen yang dipilih.

Syntax :

```
$(selector).delay(speed)
```

Parameter	Deskripsi
speed	<p>Opsional. Menentukan kecepatan elemen muncul dari hidden ke visible. Defaultnya adalah 0.</p> <p>Nilainya bisa berupa:</p> <ul style="list-style-type: none"> • milliseconds (contoh : 1500) • "slow" • "normal" • "fast"

Contoh :

```

<html>
<head>
<script src="jquery-1.9.1.min.js">
</script>
<script>
$(document).ready(function() {
    $("button").click(function() {
        $("#div1").delay(3000).fadeIn();
        $("#div2").fadeIn();
        $("#div3").fadeIn();
    });
});
</script>
</head>

<body>

<button>Click to fade in with a delay</button>
<br><br>
<div id="div1"
style="width:180px;height:180px;display:none;background-
color:black;"></div><br>
<div id="div2"
style="width:180px;height:180px;display:none;background-
color:red;"></div><br>
<div id="div3"
style="width:180px;height:180px;display:none;background-
color:blue;"></div><br>
</body>
</html>

```

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya.

1. Buat kode JavaScript, Html dan JQuery untuk Membuat tampilan seperti Gambar 5.1

jQuery

Sebelum anda memulai mempelajari jQuery, anda harus mempunyai pengetahuan dasar mengenai HTML, CSS dan Javascript.

Gambar 5.1

Keterangan:

- a. Jika kata JQuery diklik, maka akan muncul keterangan mengenai JQuery seperti tampilan gambar 5.2

jQuery

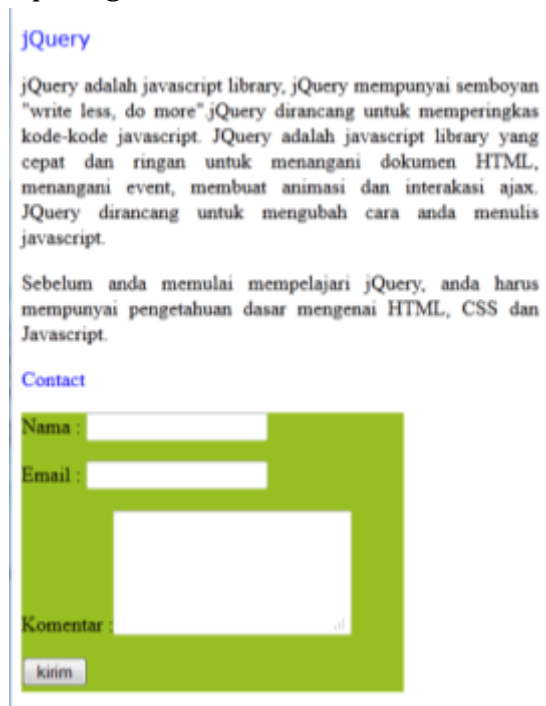
jQuery adalah javascript library, jQuery mempunyai semboyan "write less, do more". jQuery dirancang untuk memperringkas kode-kode javascript. JQuery adalah javascript library yang cepat dan ringan untuk menangani dokumen HTML, menangani event, membuat animasi dan interaksi ajax. JQuery dirancang untuk mengubah cara anda menulis javascript.

Sebelum anda memulai mempelajari jQuery, anda harus mempunyai pengetahuan dasar mengenai HTML, CSS dan Javascript.

Contact

Gambar 5.2

- b. Jika kata contact diklik dari gambar 5.2, maka akan muncul form isian seperti gambar 5.3



The screenshot shows a web page with a blue border. At the top, the word "jQuery" is written in blue. Below it, there is a paragraph of text explaining that jQuery is a JavaScript library designed to simplify writing JavaScript code, and it is fast and lightweight for handling HTML documents, events, animations, and AJAX interactions. Another paragraph follows, stating that before learning jQuery, one must have basic knowledge of HTML, CSS, and JavaScript. Below the text, the word "Contact" is written in blue. Underneath "Contact", there is a contact form with a green background. The form contains three input fields: "Nama :", "Email :", and "Komentar :". The "Komentar :" field is a larger text area. At the bottom left of the form, there is a button labeled "kirim".

Gambar 5.3

2. Buat kode JavaScript, Html dan JQuery untuk Membuat tampilan seperti Gambar 5.4

Demo 1: Animate text

Demo 2: Animate movement

Gambar 5.4

Keterangan:

- Pada saat Run Demo pada Demo 1 diklik, maka, akan muncul animasi text dari ukuran 12px sampai 30px
- Tombol restat Demo 1, digunakan untuk mengembalikan text ke ukuran 12px
- Pada saat Run Demo pada Demo 2, akan muncul animasi mobil berjalan
- Tombol restat Demo 2, digunakan untuk mengembalikan mobil ke posisi semula

Demo 1: Animate text

Animate text

Demo 2: Animate movement



- Buat kode JavaScript, Html dan JQuery untuk Membuat Menu Drop-down seperti Gambar 5.6

Search Sites	E-commerce Sites	Book Releated Sites
		JavaScriptRef
		W3C
		PINT

Gambar 5.6

Keterangan:

- a) Pada Search List, terdapat menu Drop Down berupa Link, ke Google(www.google.co.id), Yahoo (www.Yahoo.co.id) , Teoma (www.teoma.com), MSN (<http://plasa.msn.com>), dan dmoz (www.dmoz.org)
- b) Pada Menu E-commerce terdapat menu Drop Down berupa Link, ke Amazon(www.amazon.com), Ebuy(www.ebay.com), dan Buy.Com (www.buy.com)
- c) Pada Menu Book Releated Site terdapat menu Drop Down berupa Link, ke JavaScriptRef (www.javascript.com), W3C(www.w3c.org) , dan PINT(www.pint.com)

JQUERY LANJUTAN

Manipulasi HTML dengan jQuery

jQuery mempunyai kemampuan yang powerfull dalam hal memanipulasi, mengambil atau menambah konten, dan sebagainya terhadap HTML.

html()

Untuk me-set konten (biasanya disebut innerHTML) dari elemen HTML yang terpilih, dapat digunakan syntax jquery *\$(selector).html(content)*. Apabila parameter content tidak ditentukan maka fungsi html() berguna untuk mendapatkan konten dari HTML.

Contoh:

```
<html>
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(function(){
    $("#lihat").click(function(){
        var isinya = $("#paragraf").html();
        alert(isinya);
    });
});
</script>
<body>
<button id="lihat">Lihat</button>
<p id="paragraf">
jQuery mempunyai kemampuan yang powerfull dalam hal
manipulasi, mengambil atau menambah konten, dan
sebagainya terhadap HTML
</p>
</body>
</html>
```

val()

Metode val() digunakan untuk mengambil/mengubah/menyisipkan nilai dari elemen-elemen form. Sama dengan fungsi html(), apabila parameter nilai dari val() dimasukkan, maka val() akan me-set nilai dari elemen yang dipilih.

Contoh:

```

<html>
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(function() {
    $("#isikan").click(function() {
        var isinya = $("#isi").val();
        alert(isinya);
        $("#paragraf").val(isinya);
    });
});
</script>
<body>
Teks :
<input type="text" id="isi">
<br/>
<textarea id="paragraf"></textarea>
<br/>
<button id="isikan">Isikan</button>
</body>
</html>

```

attr()

Metode attr() digunakan untuk mendapatkan nilai dari suatu properti elemen HTML yang dipilih.

Syntax :

```
$(selector).attr(properties,value);
```

Parameter	Deskripsi
properties	nama properti yang ingin diambil atau ubah
value	digunakan untuk me-set nilai properti

Contoh:

```

<html>
<head>
<script type="text/javascript" src="jquery-1.9.1.min.js"></script>
<script type="text/javascript">
$(function() {
    $(".text").click(function() {
        var id=$(this).attr("type");
        alert(id);
        $(this).attr("type","button");
    });
});
</script>
</head>
<body>
<input type="text" value="isi">
</body>
</html>

```

addClass()

Berguna untuk menambahkan atau mengubah class untuk elemen yang dipilih.

Syntax :

```
$(selector).addClass(namakelas);
```

Contoh :

```
<html>
<head>
<style>
.besar
{
font-size:106px;
}
.kecil
{
font-size:12px;
text-transform:lowercase
}
</style>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(function() {
    $('.ubahclass').click(function() {
        $(".besar").addClass("kecil");
    });
});
</script>
</head>
<body>
<button class="ubahclass" >Klik </button>
<div class="besar">HELLO WORLD</div>
</body>
</html>
```

Ada banyak fungsi-fungsi lainnya untuk memanipulasi HTML :

Manipulasi	Description
<code>\$(selector).html(content)</code>	Set konten (inner HTML) dari elemen yang dipilih
<code>\$(selector).text(text)</code>	Sama seperti <code>html()</code> , tapi tag akan di-escape
<code>\$(selector).attr(attr,value)</code>	Set atribut dan nilainya dari elemen yang dipilih
<code>\$(selector).val(value)</code>	Set nilai dari elemen yang dipilih
Getting Contents	
<code>\$(selector).html()</code>	Mengambil konten (inner HTML) dari elemen yang dipilih
<code>\$(selector).text()</code>	Mengambil konten teks dari elemen yang dipilih
<code>\$(selector).attr(attr)</code>	Mengambil nilai dari suatu atribut elemen yang dipilih.
<code>\$(selector).val()</code>	Mengambil nilai/value dari elemen yang dipilih.

Adding Content	
<code>\$(selector).after(content)</code>	Menambah konten setelah elemen dipilih.
<code>\$(selector).before(content)</code>	Menambah konten sebelum elemen dipilih.
<code>\$(selector).insertAfter(selector)</code>	Menambah elemen yang ditentukan setelah elemen yang dipilih.
<code>\$(selector).insertBefore(selector)</code>	Menambah elemen yang ditentukan sebelum elemen yang dipilih.
Manipulate CSS Description	
<code>\$(selector).addClass(content)</code>	Menambah atau mengubah class elemen yang dipilih.
<code>\$(selector).removeClass(content)</code>	Menghilangkan class dari elemen yang dipilih
<code>\$(selector).toggleClass(content)</code>	Toggle antara penambahan dan penghilangan class dari elemen yang dipilih.
<code>\$(selector).hasClass(content)</code>	Cek apakah elemen memiliki class.
Adding Inner Content	
<code>\$(selector).append(content)</code>	Menambahkan konten ke dalam konten dari elemen yang dipilih
<code>\$(selector).prepend(content)</code>	"Prepend" konten ke dalam konten dari elemen yang dipilih
<code>\$(content).appendTo(selector)</code>	Menambahkan elemen ke dalam konten dari elemen yang dipilih.
<code>\$(content).prependTo(selector)</code>	"Prepend" elemen ke dalam konten dari elemen yang dipilih.
Wrapping	
<code>\$(selector).wrap(content)</code>	Wrap elemen yang dipilih dalam sebuah konten
<code>\$(selector).wrapAll(content)</code>	Wrap semua elemen yang dipilih menjadi satu konten
<code>\$(selector).wrapInner(content)</code>	Wrap konten inner child yang dipilih
<code>\$(selector).unwrap()</code>	Remove dan replace parents dari elemen yang dipilih
Copy, Replace, Remove	
<code>\$(content).replaceAll(selector)</code>	Menukar elemen yang dipilih dengan elemen yang ditentukan
<code>\$(selector).replaceWith(content)</code>	Menukar elemen yang dipilih dengan konten yang baru
<code>\$(selector).empty()</code>	Remove semua konten dan child elements dari elemen yang dipilih
<code>\$(selector).remove()</code>	Remove elemen yang dipilih
<code>\$(selector).removeAttr(attr)</code>	Remove atribut yang ditentukan dari elemen yang dipilih
<code>\$(selector).clone()</code>	Clone elemen yang dipilih
<code>\$(selector).detach()</code>	Remove elemen yang dipilih dari DOM

Manipulasi CSS dengan jQuery

Manipulasi properti CSS

jQuery juga mempunyai kemampuan dalam hal memanipulasi CSS. Ada 3 fungsi utama dalam jQuery untuk melakukan manipulasi :

- `$(selector).css(name,value)`
- `$(selector).css({properties})`
- `$(selector).css(name)`

Dimana:

- `css()` : berguna untuk mendapatkan atau memberikan sebuah properti CSS atau lebih untuk elemen yang dipilih.
- `{properties}` : Untuk me-set nilai-nilai untuk properti CSS satu atau lebih
- `name` : untuk mendapat nilai dari properti CSS yang diinginkan dari elemen yang dipilih
- `value` : untuk memberikan nilai dari properti CSS yang diinginkan dari elemen yang dipilih

Contoh:

```
<html>
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(document).ready(function() {
    $("#tombol1").click(function() {
        $(".p1").css("color", "red");
        $(".p2").css({"color": "white", "font-
        family": "Arial", "fontsize": "20px",
        "padding": "5px"});
    });
    $("#tombol2").click(function() {
        var nilai1 = $(".p1").css("color");
        var nilai2 = $(".p2").css("color");
        alert(nilai1);
        alert(nilai2);
    });
});
</script>
</head>
<body>
<p class="p1">Paragraf ini cuma berubah satu properti
CSS</p>
<p class="p2">Paragraf ini cuma berubah banyak properti
CSS</p>
<button id="tombol1">ubah CSS paragraf</button>
<button id="tombol2">Ambil nilai CSS paragraf 1</button>
</body>
</html>
```

Manipulasi CSS Size

Untuk memanipulasi ukuran dari elemen yang dipilih (tinggi dan lebar), maka ada dua fungsi, yaitu :

- `$(selector).height(value)`
- `$(selector).width(value)`

Parameter value digunakan untuk memberikan nilai height atau width dari elemen yang anda pilih. Untuk mengambil nilai height atau width dari elemen yang dipilih maka parameter value dikosongkan.

Contoh:

```
<html>
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(document).ready(function() {
    $("#tombol1").click(function() {
        $("#kotak").width("300px");
        $("#kotak").height("300px");
    });
    $("#tombol2").click(function() {
        var nilai = $("#kotak").width();
        alert("width:"+nilai);
        var nilai = $("#kotak").height();
        alert("height:"+nilai);
    });
});
</script>
</head>
<body>
<div id="kotak" style="background-
color:pink;width:100px;height:100px;">
Hello
</div>
<p>
<button id="tombol1">ubah ukuran kotak</button>
<button id="tombol2">Ambil nilai ukuran kotak</button>
</body>
</html>
```

Manipulasi Posisi

Untuk melakukan manipulasi posisi dari suatu elemen HTML, berikut beberapa fungsi jQuery untuk melakukannya.

offset()

Untuk mendapatkan atau me-set offset dari elemen yang dipilih. Relatif terhadap dokumen.

Syntax :

```
$(selector).offset({top:value,left:value});
```

Fungsi offset() akan menghasilkan objek yang mempunyai 2 properti yaitu top dan left.

Contoh :

```

<html>
<head>
<style>
#div1{
    background-color : yellow;
    width : 100px;
    height : 100px;
    position : relative;
    left : 300;
}
</style>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(document).ready(function() {
    $("#tombol1").click(function() {
        var pos=new Object();
        pos.left=$("#x").val();
        pos.top=$("#y").val();
        $("#div1").offset(pos);
    });
    $("#tombol2").click(function() {
        var pos=$("#div1").offset();
        alert("left: "+pos.left + "\ntop: "+pos.top);
    });
});
</script>
</head>
<body>
<div id="div1">
Hello World!
</div>
x = <input type="text" size="4" id="x"><br>
y = <input type="text" size="4" id="y"><br>
<button id="tombol1">Set Offset</button>
<button id="tombol2">Get Offset</button>
</body>
</html>

```

offsetParent()

Fungsi `offsetParent()` adalah untuk mendapatkan element parent terdekat dengan posisi yang telah ditentukan.

Syntax :

```
$(selector).offsetParent()
```

Contoh :

```

<html>
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(document).ready(function() {
    $("button").click(function() {

        $("div#child").offsetParent().css("background-
color", "lightblue");
    });
});

```

```

</script>
</head>
<body>
<div
style="width:70%;position:absolute;left:100px;top:100px"
id="parent">
<div style="margin:50px;background-color:yellow"
id="child">
<div>
</div>
<button>ubah warna latar</button>
</body>
</html>

```

position()

Fungsi `position()` adalah untuk mendapatkan posisi dari elemen, tetapi relatif terhadap parent. Fungsi `position()` akan menghasilkan objek yang mempunyai 2 properti yaitu `top` dan `left`.

Syntax :

```
$(selector).position()
```

Contoh :

```

<html>
<head>
<script src="jquery.js">
</script>
<script>
$(document).ready(function() {
    $("button").click(function() {
        x=$("#p").position();
        alert("Top position: " + x.top + " Left position: " +
x.left);
    });
});
</script>
</head>
<body>
<div style="width:250px;padding:10px;border:5px solid
gray;margin:10px; position:fixed;top:30px;left:5px;">
<p>This is a paragraph.</p>
</div>
<button>Get position</button>

</body>
</html>

```

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya.

JQUERY LANJUTAN

1. Buat kode JavaScript, Html dan JQuery untuk Membuat tampilan seperti Gambar 6.1

Click on any square:



Gambar 6.1

Keterangan:

Jika saat terjadi mouse over pada kotak diatas, maka letak posisi dari kotak tersebut akan ditampilkan dengan warna tulisan mengikuti focus dari mouse over seperti gambar 3.2.

Click on any square:



Gambar 6.2

2. Buat kode JavaScript, Html dan JQuery untuk Membuat tampilan seperti Gambar 6.1

JQuery

jQuery is a lightweight, "write less, do more", JavaScript library. The purpose of jQuery is to make it much easier to use JavaScript on your website. jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code. jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

What You Should Already Know

- HTML
- CSS
- JavaScript

jQuery Syntax

The jQuery syntax is tailor made for selecting HTML elements and performing some action on the element(s). Basic syntax is: `$(selector).action()`

jQuery Selectors

jQuery selectors allow you to select and manipulate HTML element(s). With jQuery selectors you can find elements based on their id, classes, types, attributes, values of attributes and much more. It's based on the existing CSS Selectors, and in addition, it has some own custom selectors. All type of selectors in jQuery, start with the dollar sign and parentheses: `$()`.

jQuery Event

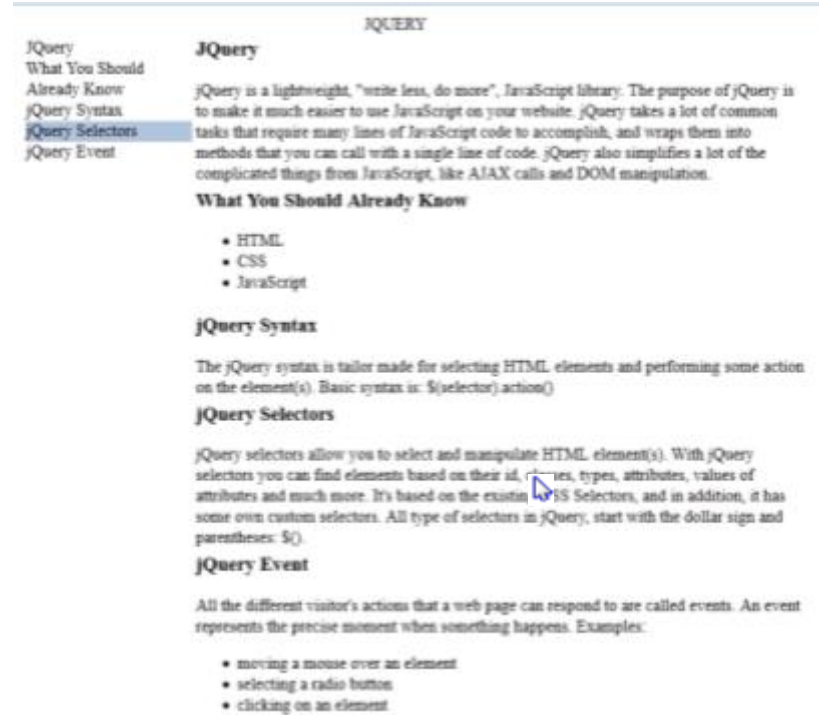
All the different visitor's actions that a web page can respond to are called events. An event represents the precise moment when something happens. Examples:

- moving a mouse over an element
- selecting a radio button
- clicking on an element

Gambar 6.3

Keterangan:

- Saat paragraph difokuskan, maka akan muncul menu tampilan di samping paragraf. (Seperti tampilan gambar 6.4)
- Menu disamping paragram akan berubah warna sesuai judul paragraph yang sedang difokuskan oleh mouse.



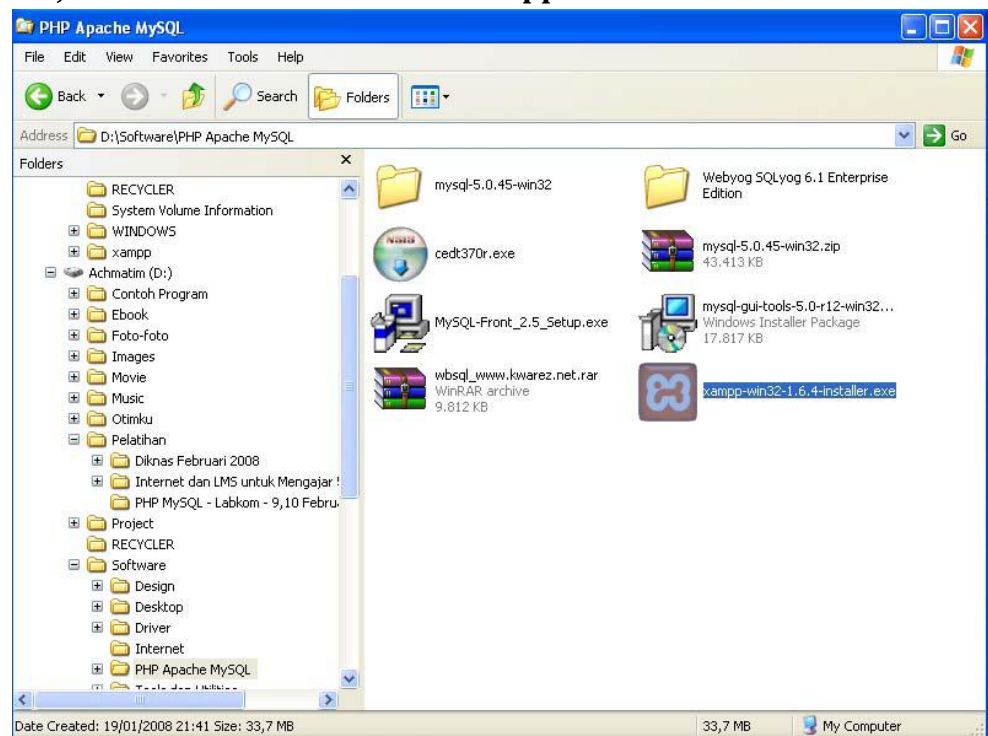
Gambar 6.4

PHP

Php merupakan singkatan dari **H**ypertext **P**reprocessor. PHP merupakan software yang bersifat open source yang dapat didownload digunakan secara gratis dan digunakan untuk Server side scripting yang bisa dijalankan di banyak platform (windows, linux, mac). Untuk mendukung PHP, dibutuhkan XAMPP. XAMPP adalah paket software lengkap yang sudah terdiri dari Apache, MySQL, PHP, PHPMyAdmin, dan tambahan pendukung lainnya), dan berjalan di atas sistem operasi Windows. Untuk mendownload XAMPP for Windows: <http://www.apachefriends.org/en/xampp-windows.html>

Proses Instalasi XAMPP

1. Jalankan file installer XAMPP **xampp-win32-x.x.x-installer.exe**.



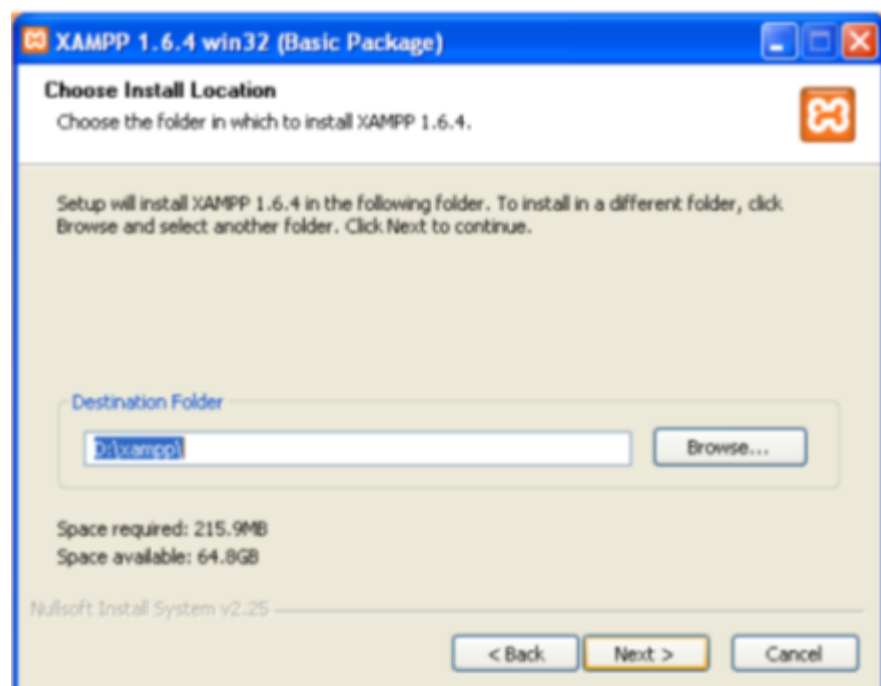
Gambar 7.1. File installer XAMPP

2. Akan ditampilkan window instalasi XAMPP. Pilih **Next >** untuk memulai proses instalasi.



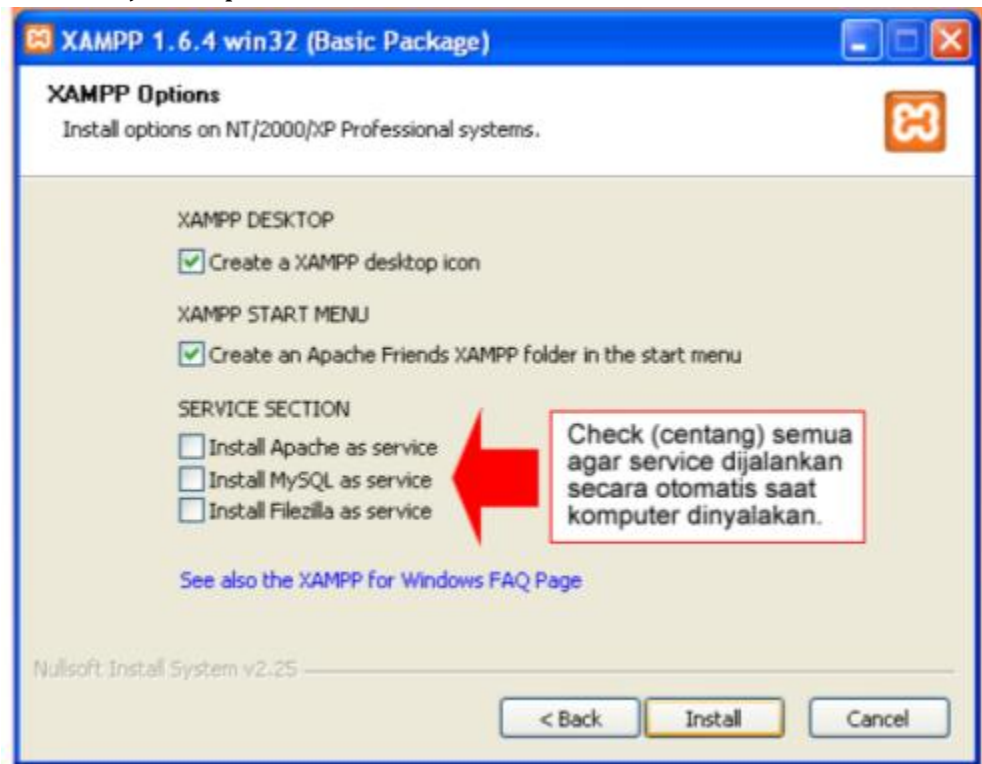
Gambar 7.2. Window Instalasi XAMPP

3. Akan ditampilkan window lokasi tujuan instalasi (*destination folder*), tentukan lokasi (folder) dimana XAMPP akan diinstall dengan menekan tombol **Browse....** Pilih tombol **Next >** untuk melanjutkan proses instalasi.



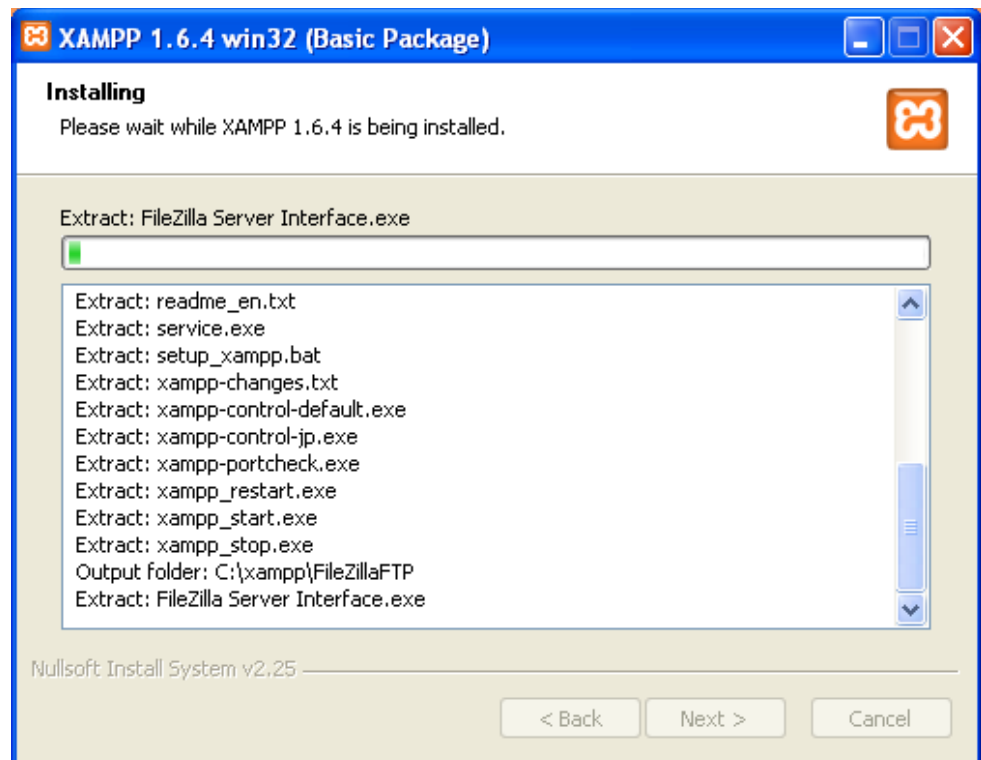
Gambar 7.3. Folder tujuan instalasi XAMPP

4. Selanjutnya akan ditampilkan window pilihan untuk kemudahan mengakses XAMPP nantinya. Tekan tombol Install untuk melanjutkan proses instalasi.



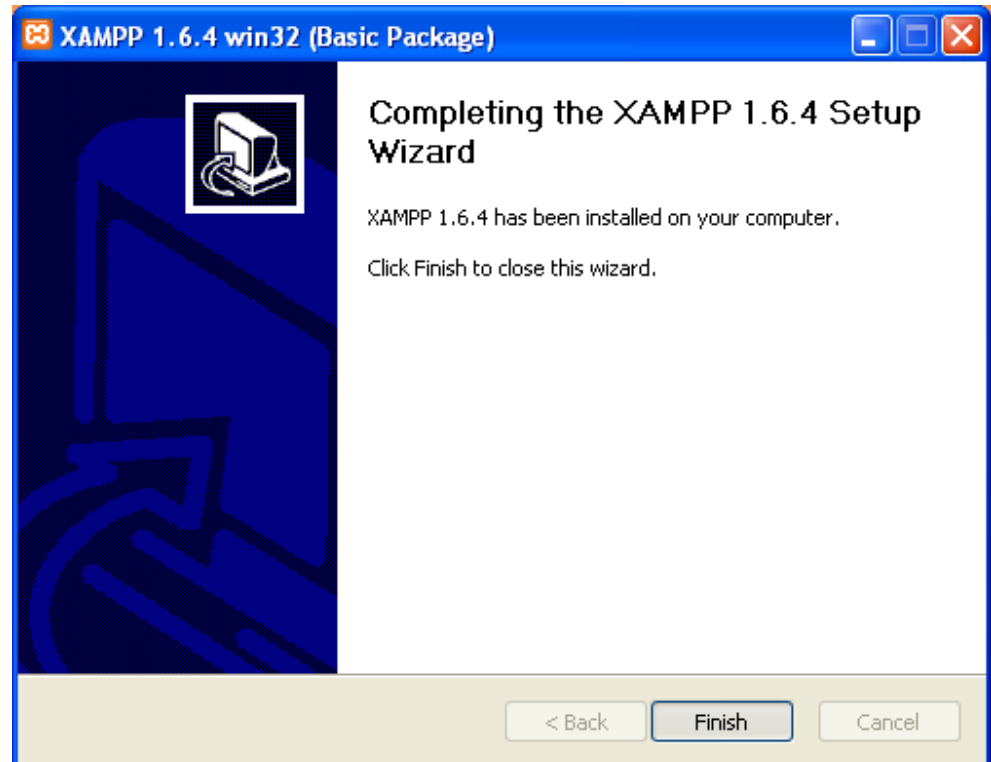
Gambar 7.4. Window pilihan XAMPP

5. Proses instalasi dimulai.



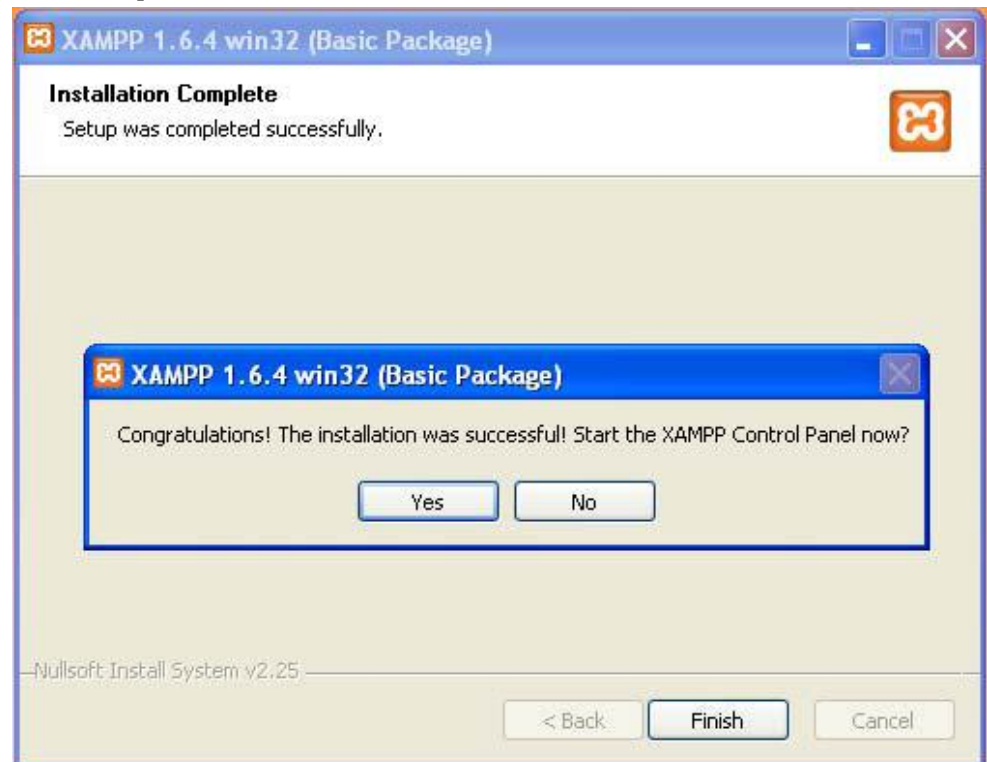
Gambar 7.5. Proses instalasi dimulai

6. Proses instalasi selesai. Klik **Finish** untuk menutup proses instalasi.



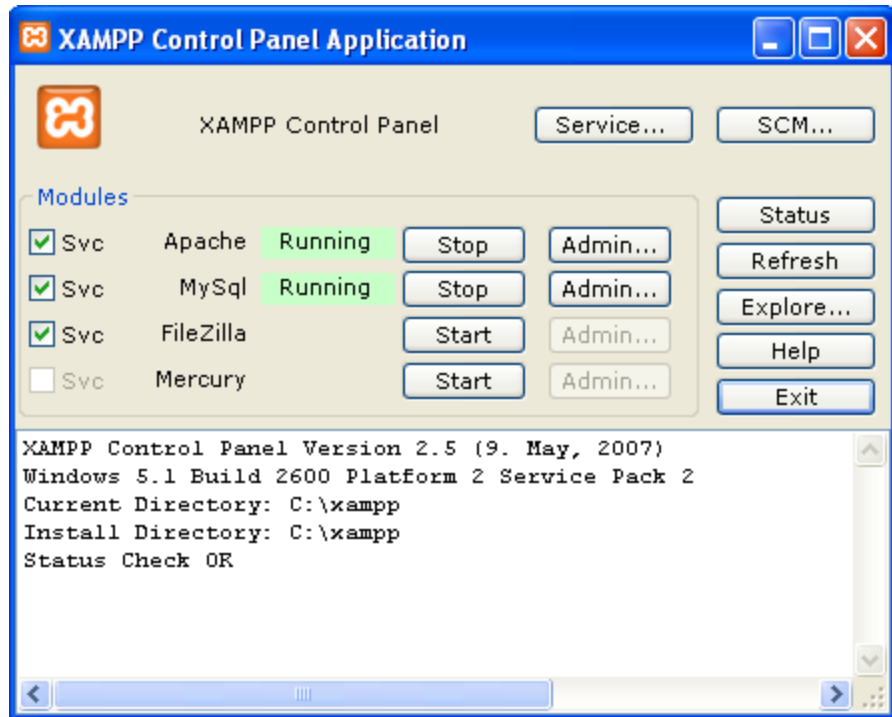
Gambar 7.6. Proses instalasi selesai

7. Jika proses instalasi berjalan dengan lancar maka akan ditampilkan window konfirmasi.



Gambar 7.7. Konfirmasi Instalasi Berhasil

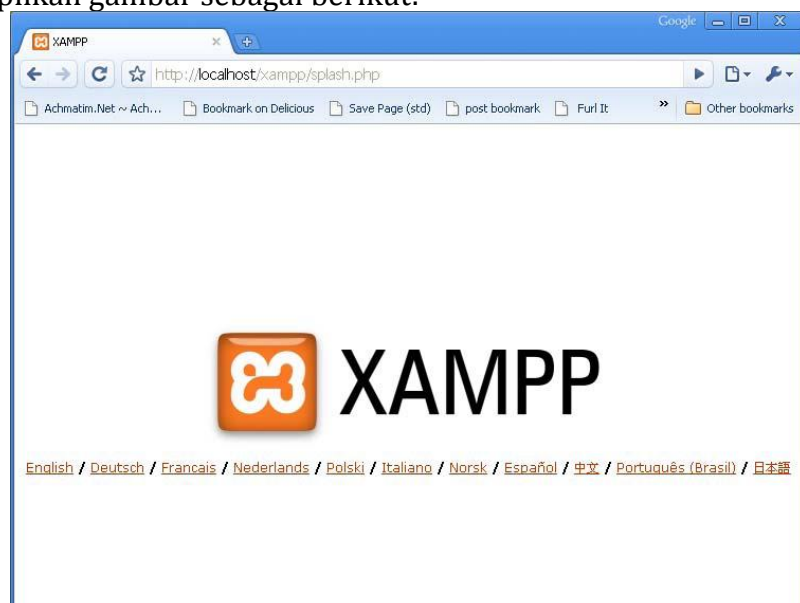
8. Kita dapat mengatur XAMPP dengan mengakses XAMPP Control Panel yang dapat diakses melalui menu di Program Files atau icon di taskbar.



Gambar 7.8. XAMPP Control Panel

Test Instalasi XAMPP di Browser

Untuk memastikan bahwa XAMPP beserta PHP, Apache dan MySQL berjalan dengan baik, bukalah browser dan ketikkan URL **http://localhost**. Jika semua sudah berjalan dengan baik, maka akan ditampilkan gambar sebagai berikut:



Gambar 7.9. Tampilan XAMPP di Browser

PHP

Pilih bahasa yang ingin digunakan dan akan ditampilkan halaman utama XAMPP sebagai berikut:



Gambar 7.10. Tampilan XAMPP di Browser

Memulai PHP

Untuk menjalankan file PHP, maka file PHP harus diletakkan di lokasi:

C:\xampp\htdocs

File PHP harus di simpan dengan extention .php. Untuk menjalankan file PHP (misalnya abc.php) dapat dilakukan dengan mengetikkan URL berikut pada browser:

http://localhost/abc.php

Penulisan Script PHP

Script PHP dimulai dengan `<?php` dan diakhiri dengan `?>`. Setiap statement pada PHP harus diakhir tanda `;` dan tanda `{` dan `}` digunakan untuk mengelompokkan sekumpulan statement.

```
<?php
    script php disini
?>
```

Contoh Penulisan :

```
<?php
    echo "Hello World";
?>
```

Fungsi dari “echo” adalah digunakan untuk menuliskan output dengan menggunakan php.

Komentar

PHP juga menyediakan fasilitas untuk menuliskan komentar. Pemberian komentar dalam PHP dapat dilakukan dengan menggunakan:

- `/* dan */`
- `// dan`
- `#`

Contoh:

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

Variable

Aturan penamaan variabel adalah:

- Nama variabel diawali dengan tanda `$`
- Panjang tidak terbatas
- Setelah tanda `$` boleh diawali oleh huruf atau *under-score* (`_`).
- Karakter berikutnya bisa terdiri dari huruf, angka, dan karakter
- Bersifat case-sensitive.
- Tidak perlu dideklarasikan.
- Tidak boleh mengandung spasi.

Contoh :

Benar

- `$_name`
- `$first_name`
- `$name3`
- `$name_3`

Salah

- `$3name`
- `$name?`
- `$first+name`
- `$first.name`
- `$first name`

Tipe Data

Pada PHP, tipe data variabel tidak didefinisikan oleh programmer, akan tetapi secara otomatis ditentukan oleh interpreter PHP. Tipe data dalam PHP sama seperti tipe data pada JavaScript.

Operator Aritmatika

Operator aritmatika digunakan untuk melakukan operasi aritmatika antara variabel. Misalkan bahwa `$y = 5`, tabel di bawah menjelaskan operator aritmatika:

Operator	Deskripsi	Contoh	Hasil x	Hasil y
+	Addition	$\$x = \$y + 2$	7	5
-	Subtraction	$\$x = \$y - 2$	3	5
*	Multiplication	$\$x = \$y * 2$	10	5
/	Division	$\$x = \$y / 2$	2.5	5
%	Modulus (division remainder)	$\$x = \$y \% 2$	1	5
++	Increment	$\$x = ++\y $\$x = \$y++$	6 5	6 6
--	Decrement	$\$x = --\y $\$x = \$y--$	4 5	4 4

Operator Assignment

Assignment operator digunakan untuk menempatkan nilai ke variabel. Misalkan $\$x = 10$ dan $\$y = 5$, tabel di bawah menjelaskan assignment operator:

Operator	Contoh	Sama Dengan	Hasil
=	$\$x = \y		$x = 5$
+=	$\$x += \y	$\$x = \$x + \$y$	$x = 15$
-=	$\$x -= \y	$\$x = \$x - \$y$	$x = 5$
*=	$\$x *= \y	$\$x = \$x * \$y$	$x = 50$
/=	$\$x /= \y	$\$x = \$x / \$y$	$x = 2$
%=	$\$x \% = \y	$\$x = \$x \% \$y$	$x = 0$

Operator Perbandingan

Operator perbandingan digunakan untuk menentukan kesetaraan atau perbedaan antara variabel. Misalkan $\$x = 5$, tabel di bawah menjelaskan operator perbandingan:

Operator	Deskripsi	Perbandingan	Hasil
==	Sama dengan	$\$x == 8$ $\$x == 5$	<i>Salah</i> <i>Benar</i>
===	Tepat sama dengan (nilai dan tipe)	$\$x === "5"$ $\$x === 5$	<i>Salah</i> <i>Benar</i>
!=	Tidak sama dengan	$\$x != 8$	<i>Salah</i>
!==	Tidak sama dengan (nilai dan tipe)	$\$x !== "5"$ $\$x !== 5$	<i>Benar</i> <i>Salah</i>

>	Lebih besar dari	$\$x > 8$	<i>Salah</i>
<	Lebih kecil dari	$\$x < 8$	<i>Benar</i>
>=	Lebih besar atau sama dengan	$\$x \geq 8$	<i>Salah</i>
<=	Lebih kecil atau sama dengan	$\$x \leq 8$	<i>Benar</i>

Operator Logika

Operator logika digunakan untuk menentukan logika antara variabel. Misalkan $\$x = 5$ dan $\$y = 3$, tabel di bawah menjelaskan operator logis:

Operator	Deskripsi	Contoh	Hasil
&&	and	$(\$x < 10 \ \&\& \ \$y > 1)$	<i>Benar</i>
	or	$(\$x == 5 \ \ \$y == 5)$	<i>Salah</i>
!	not	$! (\$x == \$y)$	<i>Benar</i>

Operator String

Operator string pada PHP digunakan untuk menggabungkan beberapa string menjadi sebuah string yang lebih panjang.

Contoh :

`nama = "Belajar" . "PHP";`

akan menghasilkan "BelajarPHP" pada variabel nama

Operator Kondisional

Operator kondisional memberikan nilai ke variabel didasarkan pada beberapa kondisi.

Syntax:

```
variableName=(kondisi)? nilai1: nilai2
```

contoh

```
 $\$status = (\$usia < 18) ? "Terlalu muda": "Cukup tua";$ 
```

Keterangan:

Jika variabel usia lebih kecil dari 18, status dari variabel nilai akan bernilai "Terlalu muda", jika nilai usia lebih besar dari 18, status akan bernilai "Cukup tua".

Statement Percabangan

JavaScript memiliki statement percabangan sebagai berikut:

- **If Statement** : digunakan untuk mengeksekusi baris kode hanya jika kondisi yang ditentukan benar.

Syntax:

```
if (kondisi)
{
    baris kode yang akan dieksekusi jika kondisi benar
}
```

Contoh :

```
<html>
<head>
</head>
<body>
<?php
$x="Good";
$y=12;
if ($y<12)
{
    $x=$x." day";
}
echo $x;

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

- **If...else Statement** : digunakan untuk mengeksekusi baris kode jika kondisi benar dan kode lain jika kondisi salah.

Syntax:

```
if (kondisi)
{
    baris kode yang akan dieksekusi jika kondisi benar
}
else
{
    baris kode yang akan dieksekusi jika kondisi salah
}
```

Contoh:

```
<html>
<head>
</head>
<body>
<?php
$x="Good";
$y=15;
if ($y<12) {
    $x=$x." day";
}
```



```

else{
    $x=$x." evening";
}

echo $x;
?>
</body>
</html>

```

- **If...else if...else** Statement : digunakan untuk untuk memilih salah satu dari banyak baris kode yang akan dieksekusi.

Syntax:

```

if (kondisi 1)
{
    baris kode yang akan dieksekusi jika kondisi 1 benar
}
else if ((kondisi 2)
{
    baris kode yang akan dieksekusi jika kondisi 2 benar
}
else
{
    baris kode yang akan dieksekusi jika kondisi 1 dan
    kondisi 2 salah
}

```

Contoh:

```

<html>
<head>
</head>
<body>
<?php

$x="Good";
$y=25;
if ($y<12)
{
    $x=$x." day";
}
else if ($y<24)
{
    $x=$x." evening";
}
else
{
    $x=$x." morning";
}

echo $x;

?>

</body>
</html>

```

- **Switch Statement** : digunakan untuk memilih salah satu dari banyak baris kode yang akan dieksekusi.

Syntax :

```
switch($n)
{
case 1:
    eksekusi baris kode 1
    break;
case 2:
    eksekusi baris kode 2
    break;
default:
    baris kode yang dieksekusi jika n tidak termasuk case 1 dan case 2.
}
```

Contoh :

```
<html>
<body>
<?php

$x = "Hari ini Hari";
$d = 1;
switch ($d)
{
    case 1:
        $x= $x." Senin";
        break;
    case 2:
        $x= $x." Selasa";
        break;
    case 3:
        $x= $x." Rabu";
        break;
    case 4:
        $x= $x." Kamis";
        break;
    case 5:
        $x= $x." Jumat";
        break;
    case 6:
        $x= $x." Sabtu";
        break;
    default:
        $x= $x." Minggu";
}

echo $x;

?>

</body>
</html>
```

Statement Perulangan

JavaScript memiliki statement perulangan sebagai berikut:

- **While Statement** : digunakan untuk menjalankan baris kode selama kondisi tertentu adalah benar.

Syntax:

```
while (Kondisi)
{
    baris kode yang akan dieksekusi
}
```

Contoh:

```
<html>
<body>

<?php
$x="";
$i=0;
while ($i<5)
{
    echo $i."<br>";
    $i++;
}
?>

</body>
</html>
```

- **Do ... While Statement** : merupakan varian dari while loop. Loop ini akan mengeksekusi baris kode sekali, sebelum memeriksa kondisi while. Jika kondisi benar, maka loop akan diulangi selama kondisi benar.

Sintax :

```
do
{
    baris kode yang akan dieksekusi
}
while (Kondisi);
```

Contoh:

```
<html>
<body>

<?php
$x="";
$i=0;
do{
    echo $i."<br>";
    $i++;
}while ($i<5);
?>

</body>
</html>
```

- **For Statement** : digunakan untuk loop baris kode beberapa kali.

Syntax :

```
for (pernyataan 1; pernyataan 2, pernyataan 3)
{
    blok kode yang akan dieksekusi
}
```

Dimana :

Pernyataan 1 : dijalankan sebelum loop (baris kode) dimulai.

Pernyataan 2 : mendefinisikan kondisi untuk menjalankan loop (baris kode).

Pernyataan 3 : dijalankan setiap kali setelah loop (baris kode) telah dieksekusi.

Contoh :

```
<html>
<body>

<?php

for ($i=1;$i<=6;$i++)
{
    echo "<h$i>Heading $i</h$i>";
}

?>

</body>
</html>
```

Funksi

Function juga dapat dibuat sendiri (*User-defined functions*). *User-defined functions* dalam PHP di didefinisikan di bagian tag <head> dan pemanggilan function untuk menjalankan function tersebut dapat dilakukan di tag <body>. Function ini dibagi menjadi 2:

- **Function Tanpa Parameter (Argumen)**

Syntax:

```
function namafunction()
{
    blok kode yang akan dieksekusi
}
```

Contoh :

```

<html>
<head>

<?php
function myFunction()
{
echo "Hello World!";
}
?>

</head>

<body>

<?php
myFunction();
myFunction();

?>

</body>
</html>

```

- **Function Dengan Parameter (Argumen)**

Syntax:

```

function namafunction(var1, Var2, ... )
{
    blok kode yang akan dieksekusi
}

```

Contoh :

```

<html>
<head>

<?php
function myFunction($a,$b)
{
    return $a*$b;
}
?>

</head>

<body>

<?php
echo myFunction(2,3)."<br>";
echo myFunction(3,1);
?>

</body>
</html>

```

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya.

1. Buat kode HTML dan PHP Untuk menampilkan Tabel Fibonansi berikut:

Tabel Fibonacci

Nilai	Hasil Fibonacci
1	1
2	1
3	2
4	3
5	5
6	8
7	13
8	21
9	34
10	55

2. Buat kode HTML dan PHP Untuk menampilkan berikut angka-angka berikut:

1
-2
3
-4
5
-6
7
-8
9
-10

3. Buat kode HTML dan PHP Untuk menampilkan berikut tabel berikut:

No	Judul Lowongan	Tanggal	Status

OBJEK-OBJEK PHP

Objek Untuk Memasukkan Data

Untuk merancang sebuah form inputan, setidaknya ada 3 (tiga) hal penting, yaitu :

1. **METHOD**

Method dari sebuah form menentukan bagaimana data inputan form dikirim. Method ini ada dua macam, yaitu GET dan POST. Method ini menentukan bagaimana data inputan dikirim dan diproses oleh PHP.

2. **ACTION**

Action dari sebuah form menentukan dimana data inputan dari form diproses. Jika action ini dikosongkan, maka dianggap proses form terjadi di halaman yang sama.

3. **SUBMIT BUTTON**

Submit button merupakan sebuah tombol (pada umumnya) yang berfungsi sebagai *trigger* pengiriman data dari form inputan. Jika tombol ini ditekan, maka data form akan dikirimkan (diproses) di halaman yang sudah ditentukan pada atribut action.

- **Objek Text**

Objek Text dapat digunakan untuk menginputkan data. Contoh penggunaannya dapat dilihat pada contoh berikut :

Contoh Program PHP

```
<html>
<head><title>Pengolahan Form</title>
</head>
<body>
<FORM ACTION="" METHOD="POST" NAME="input">
Nama Anda : <input type="text" name="nama"><br>
<input type="submit" name="Input" value="Input">
</FORM>
```

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

Proses pengolahan form yang dilakukan di halaman yang sama dengan form inputan maka atribut **action** pada tag **form** tidak perlu diisi (dikosongkan). Untuk Proses pengolahan form dilakukan di halaman yang terpisah dengan form inputannya, maka atribut **action** pada tag **form** harus diisi dengan alamat halaman tempat proses pengolahan form.

Contoh Program PHP

```
<!--simpan dengan nama ObjekText.html -->
<html>
<head><title>Pengolahan Form</title>
</head>
<body>
<FORM ACTION=" ObjekText.php" METHOD="POST" NAME="input">
Nama Anda : <input type="text" name="nama"><br>
<input type="submit" name="Input" value="Input">
</FORM>
</body>
</html>
```

```
<!--simpan dengan nama ObjekText.php -->
<?php
if (isset($_POST['Input'])) {
    $nama = $_POST['nama'];
    echo "Nama Anda : <b>$nama</b>";
}
?>
```

- **Objek Radio**

Objek radio adalah komponen yang digunakan untuk melakukan suatu pemilihan data. Karena Objek Radio merupakan Array , maka untuk mengakses satu tombol radio digunakan `radio[indeks]`. Objek radio juga mempunyai nilai `True` jika dipilih dan `False` jika tidak.

Contoh Program PHP

```
<!--simpan dengan nama ObjekRadio.html -->
<html>
<head><title>Latihan Dengan Objek Radio</title>
</head>
<body>
<FORM ACTION="ObjekRadio.php" METHOD="POST" NAME="input">
<H1> Objek Radio</H1><hr>
<input type="radio" value="D1" name="Lulusan">D1<br />
<input type="radio" value="D3" name="Lulusan">D3<br />
<input type="radio" value="S1" name="Lulusan">S1<br />
<hr>
<input type="submit" name="Input" value="Input">
</FORM>
</body>
</html>
```

```
<!--simpan dengan nama ObjekRadio.php -->
<?php
if (isset($_POST['Input'])) {
    $pil = $_POST['Lulusan'];
    echo "Pilihan anda : <b>$pil</b>";
}
?>
```

- **Objek Checkbox**

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

Contoh Program PHP

```
<!--simpan dengan nama ObjekCek.html -->

<html>
<head><title>Objek Checkbox</title></head>

<body>
<FORM ACTION="ObjekCek.php" METHOD="POST" NAME="input">
<H1>Objek Checkbox</H1><hr>
Nama Buah-buahan : <br />
<input type="checkbox" value="Anggur" name="buah01">Anggur
<br/>
<input type="checkbox" value="Jeruk" name="buah02">Jeruk
<br/>
<input type="checkbox" value="Mangga" name="buah03">Mangga
<hr>
<input type="submit" name="Input" value="Input">

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

```

<!--simpan dengan nama ObjekCek.php -->

<?php
if (isset($_POST['Input'])) {
    echo "Buah Favorit Anda adalah :<br>";
    if (isset($_POST['buah01'])) {
        echo "+ " . $_POST['buah01'] . "<br>";
    }
    if (isset($_POST['buah02'])) {
        echo "+ " . $_POST['buah02'] . "<br>";
    }
    if (isset($_POST['buah03'])) {
        echo "+ " . $_POST['buah03'] . "<br>";
    }
}
?>

```

- **Objek TextArea**

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

Contoh Program PHP

```

<!--simpan dengan nama ObjekArea.html -->

<html>
<head><title>Latihan Dengan Objek TextArea</title>
</head>
<body>
<FORM ACTION="ObjekArea.php" METHOD="POST" NAME="input">
<H1>Objek TextArea</H1><hr>
<h3>Keterangan :<h3><br>
<textarea name="Ket" rows="5" cols="30"></textarea>
<input type="submit" name="Input" value="Input">
</form>
</body>
</html>

```

```

<!--simpan dengan nama ObjekArea.php -->

<?php
if (isset($_POST['Input'])) {
    $ket = nl2br($_POST['Ket']);
    echo "<h3>Keterangan :<h3><br>";
    echo "<font color=blue><b>$ket</b></font>";
}
?>

```

- **Objek Select**

Objek Select menyimpan informasi tentang elemen form yang berupa daftar kotak.

Contoh Program PHP

```
<!--simpan dengan nama ObjekSelect.html -->

<html>
<head><title>Latihan Dengan Objek Select</title>
</head>
<body>
<FORM ACTION="ObjekSelect.php" METHOD="POST" NAME="input">
<H1>Objek Select</H1><hr>
<select name="Jurusan" Size="1">
<option value ="satu"> 1</option>
<option value ="dua"> 2</option>
<option value ="tiga"> 3</option>
</select>
<input type="submit" name="Input" value="Input">
</form>
</body>
</html>
```

```
<!--simpan dengan nama ObjekSelect.php -->

<?php
if (isset($_POST['Input'])) {
$film = $_POST['Jurusan'];
echo "Jurusan Anda adalah :
<font color=blue><b>$film</b></font>";
}
?>
```

Objek Array (Array Object)

Array adalah suatu variable yang dapat memuat beberapa nilai secara berurutan. pendeklarasian untuk array adalah:

```
$nama = Array(3)
```

Untuk mengisi ketiga elemen tersebut, dapat dilakukan dengan cara :

```
$nama[0] ="Dian"
$nama[1] ="Andri"
$nama[2]="Irawan"
```

Pendeklarasikan array juga dapat dilakukan sekaligus dengan mengisi elemen-elemennya.

```
$nama = Array("Dian","Andri","Irawan")
```

PHP menyediakan lebih dari 70 fungsi untuk manipulasi array. Fungsi-fungsi array dalam PHP bisa dilihat di alamat :

<http://ca.php.net/manual/en/ref.array.php>.

Mendeklarasikan atau mendefinisikan sebuah array, dapat dilakukan dengan cara *assosiatif* (array yang menggunakan index selain integer). Array *assosiatif* dapat dengan cara menyebutkan indexnya terlebih dahulu diikuti operator => dan diikuti value atau nilai elemennya.

```
$Umur=array("Dian"=>"25", "Andri"=>"37", "Irwan"=>"43");
```

Objek Tanggal (Date Object)

Objek Tanggal digunakan untuk memanipulasi tanggal dan waktu pada PHP. Untuk pendeklarasian Objek Tanggal adalah sebagai berikut :

```
$tgl = Date()
```

Objek String (String Object)

String adalah suatu objek yang merupakan kumpulan dari elemen karakter-karakter. Dalam Javascript string atau karakter harus diapit dengan tanda petik ganda("") atau tanda petik tunggal(').

Untuk mendeklarasikan Objek String adalah sebagai berikut:

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

PHP menyediakan lebih dari 90 fungsi untuk manipulasi string. Fungsi-fungsi string dalam PHP bisa dilihat di :

<http://ca.php.net/manual/en/ref.strings.php>.

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya.

1. Buat kode HTML dan PHP Untuk menampilkan Halaman Login berikut:

Login

Username :

Password :

Gambar 8.1

Keterangan:

Jika Login yang anda lakukan dengan username nama anda dan NIM sebagai password anda, maka tampilkan kata "Login Berhasil" dan Jika username dan password salah maka tampilkan kata "Lgin Gagal".

2. Buat kode HTML dan PHP Untuk menampilkan Halaman Form berikut:

Biodata

Nama

Nim

Email

Jenis Kelamin ☐ Laki-Laki ☐ Perempuan

Kegemaran ☐ Olahraga ☐ Musik ☐ Membaca ☐ Memasak ☐ Fotografi ☐ Otomotif

Jumlah Saudara

Komentar

Gambar 8.2

Keterangan:

Setelah mengisi data dalam form, dan tombol submit diklik maka kirimkan semua data tersebut ke "hasil.php" dengan tampilan seperti gambar 8.3.

Nama :Sukijo
 NIM: 0912321
 Email: y@yahoo.com
 Jenis Kelamin: Laki-lak
 Hobby:
 -Musik
 -Membaca
 Jumlah Saudara: tiga
 Komentar :
 Selesai

Gambar 8.3

3. Buat kode HTML dan PHP Untuk menampilkan Tabel berikut:

Negara	Bahasa
USA	Inggris
Spain	Spanyol
Brazil	Portugis
UK	Inggris
Mexico	Spanyol
Germany	Jerman
Colombia	Spanyol
Canada	Inggris
Russia	Rusia
Austria	Jerman
France	Perancis
Argentina	Spanyol

Keterangan:

Dengan fungsi Array, buat sebuah file PHP untuk menghasilkan Tabel Berikut.

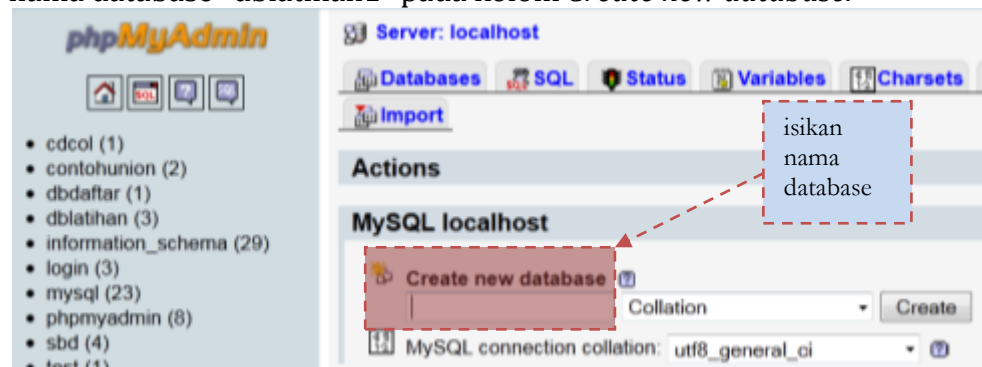
Bahasa	Jumlah Negara yang Menggunakan
Inggris	3
Spanyol	4
Portugis	1
Jerman	2
Rusia	1
Prancis	1

My SQL

MySQL adalah sebuah DBMS (Database Management System) di bawah lisensi General Public License (GPL) yang mampu menangani banyak query dengan kecepatan tinggi dalam satuan waktu tertentu. MySQL mendukung hampir semua OS (Operating System), selain itu bisa menampung database dalam skala besar. MySQL adalah database yang paling sering digunakan oleh web developer dalam pembuatan website.

Create Database

Untuk membuat database baru, ketikkan <http://localhost/phpmyadmin> di web browser. Kemudian ketikkan lah nama database “dlatihan1” pada kolom *Create new database*.



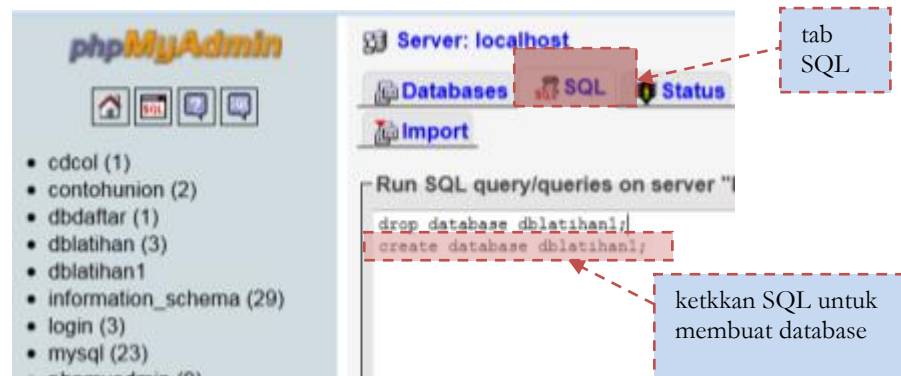
Gambar 9.1 tampilan phpmyadmin

Jika database berhasil dibuat maka di sebelah kiri layar, akan muncul nama database dengan nama “DBLatihan1”.



Gambar 9.2 Hasil create database ‘dlatihan1’

Database juga dapat dibuat menggunakan SQL (*Sintaks Query Language*), dengan memilih tab SQL dan mengetikkan sintaks Berikut:

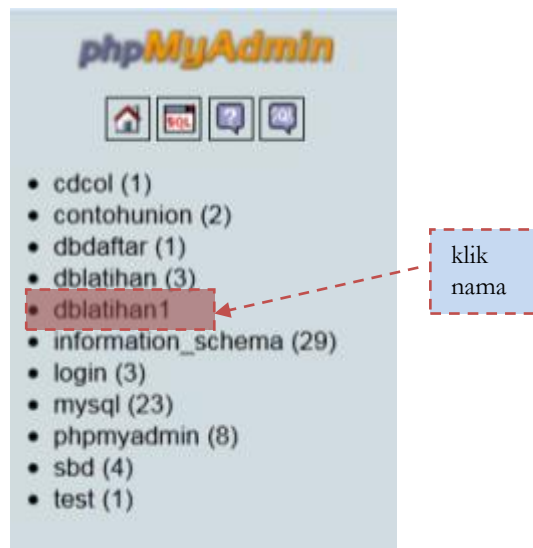


Gambar 9.3 create database 'dblatihan1' dengan SQL

Perintah/sintaks drop database digunakan untuk menghapus database. Pada kasus diatas, perintah drop perlu ditambahkan karena dalam database tidak diizinkan untuk membuat dua database dengan nama yang sama.

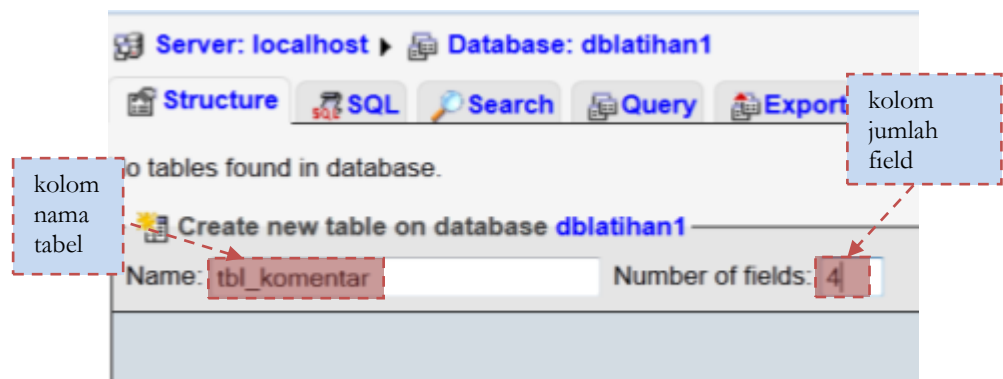
Create Tabel

Untuk membuat tabel pada database, terlebih dahulu pilih nama database yang akan digunakan. Misalkan untuk membuat tabel pada database 'dblatihan1', maka klik nama database 'dblatihan1' pada sebelah kiri layar.



Gambar 9.4 memilih database 'dblatihan1'

Setelah memilih database 'dblatihan1' maka untuk membuat tabel pada database 'dblatihan1', ketikkan nama tabel yang ingin dibuat pada kolom *name* dan berapa jumlah kolom yang dibutuhkan pada kolom *number of fields*.



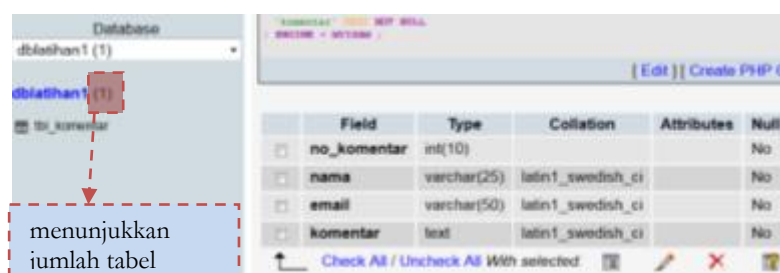
Gambar 9.4 membuat tabel pada 'dblatihan1'

Setelah membuat tabel 'tbl_komentar' pada database 'dblatihan1', maka selanjutnya adalah mengisi nama dari tiap *field* dan tipe data dari *field* tersebut.

Field	Type	Length/Values ¹
no_komentar	INT	10
nama	VARCHAR	25
email	VARCHAR	50
komentar	TEXT	

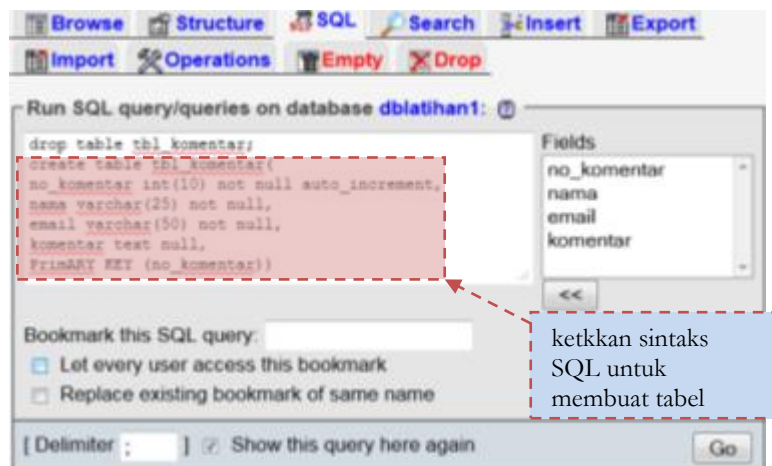
Gambar 9.5 field tabel 'tbl_komentar'

Masuk nama field dan type serta length sesuai gambar 9.5. Jika tabel komentar sukses dibuat, maka hasilnya akan muncul seperti gambar 9.6.



gambar 9.6 hasil create table

Untuk membuat tabel dengan menggunakan sintaks query pada database 'dblatihan1', gunakan sintaks create table seperti pada gambar 9.7.



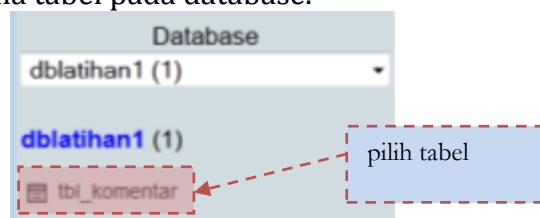
Gambar 9.7 create table 'tbl_komentar' dengan SQL

Perintah/sintaks drop table digunakan untuk menghapus table. Pada kasus diatas, perintah drop perlu ditambahkan karena dalam database tidak diizinkan untuk membuat dua table dengan nama yang sama.

Perintah Auto_Increment pada SQL digunakan untuk tipe field integer yang secara otomatis akan bertambah nilainya jika terjadi penambahan row pada table dimana field tersebut berada.

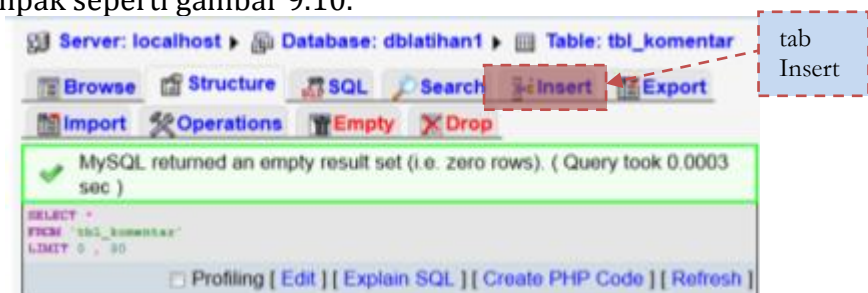
Insert, Select Tabel

Untuk memasukkan data ke dalam record tabel, pertama yang harus dilakukan adalah memilih tabel yang ingin diisikan datanya dengan mengklik nama tabel pada database.



Gambar 9.8 pemilihan table 'tbl_komentar'

Setelah meng-klik tabel tersebut, pilih tab insert. maka hasilnya akan tampak seperti gambar 9.10.



Gambar 9.9 pemilihan tab Insert

Field	Type	Function	Null	Value
no_komentar	int(10)			
nama	varchar(25)			Edy
email	varchar(50)			mr.edy@mmail.com
komentar	text		<input checked="" type="checkbox"/>	Terima Kasih atas kunjungannya

Go

Gambar 9.10 Hasil pemilihan tab Insert

Masukkan value sesuai dengan data diatas berdasarkan tipe tiap-tiap fieldnya sesuai gambar 9.10, dan klik tombol Go. Maka setelah meng-klik tombol go, tabel tbl_komentar telah memiliki 1 record. Field pada no_komentar tidak perlu diisi karena pada field tersebut ada penggunaan Auto_Increment.

Untuk meng-insert dengan perintah SQL, bisa dilakukan dengan memasukkan skript perintah SQL yang ada pada tab SQL.

Run SQL query/queries on database dblatihan1: ?

```
Insert Into tbl_komentar
values ("","Ardi","ardymail.com","Terima Kasih
atas informasinya")
```

Fields

- no_komentar
- nama
- email
- komentar

Bookmark this SQL query:

☐ Let every user access this bookmark ☐ Replace existing bookmark of same name

[Delimiter :] ☒ Show this query here again Go

Gambar 9.11 SQL Insert

Jika berhasil maka , tabel tbl_komentar telah memiliki 2 record. Untuk melihat isi dari semua record yang telah di-insert ke dalam tbl_komentar, gunakan SQL select berikut pada tab SQL.

Run SQL query/queries on database dblatihan1: ?

```
SELECT * FROM tbl_komentar
```

Fields

- no_komentar
- nama
- email
- komentar

Bookmark this SQL query:

☐ Let every user access this bookmark ☐ Replace existing bookmark of same name

[Delimiter :] ☒ Show this query here again Go

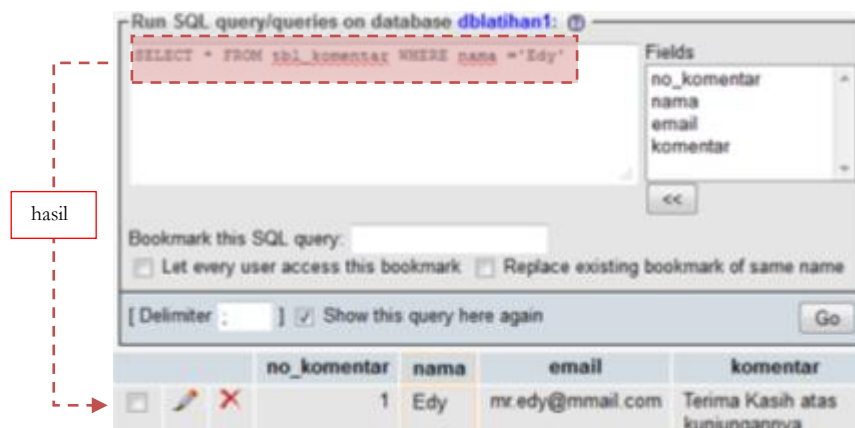
Gambar 9.12 SQL Select

Hasil yang muncul akan tampil seperti gambar 9.13

		no_komentar	nama	email	komentar
<input type="checkbox"/>			1 Edy	mr.edy@mmail.com	Terima Kasih atas kunjungannya
<input type="checkbox"/>			2 Ardi	ard@ymail.com	Terima Kasih atas informasinya

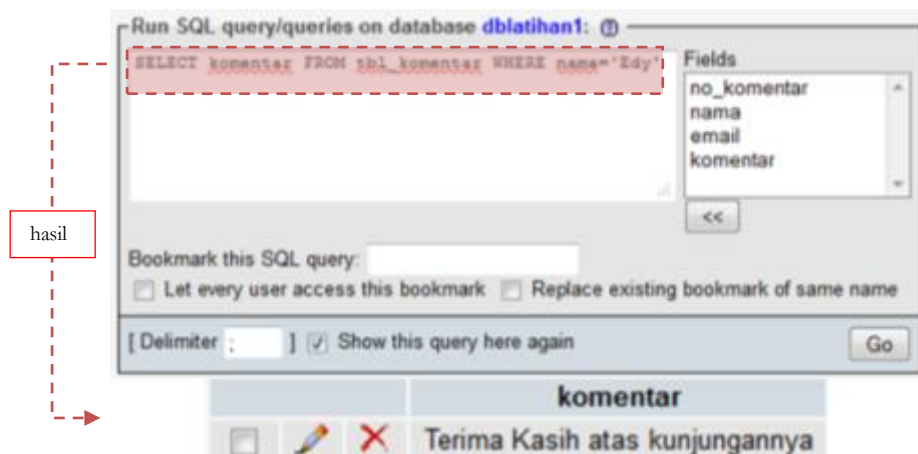
Gambar 9.12 Hasil SQL Select

Untuk menampilkan semua record yang bernama Edy dapat digunakan SQL berikut:



Gambar 9.13 SQL Select dan Hasil SQL Select (1)

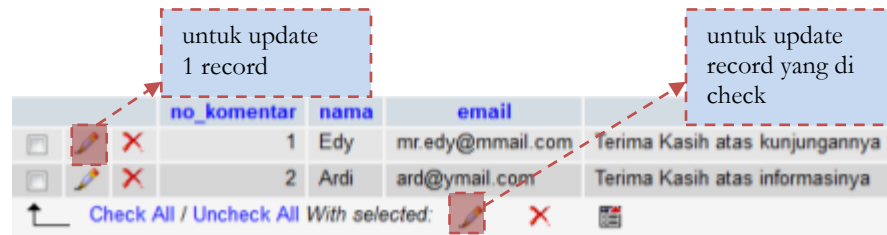
Untuk menampilkan record komentar yang bernama Edy dapat digunakan SQL berikut:



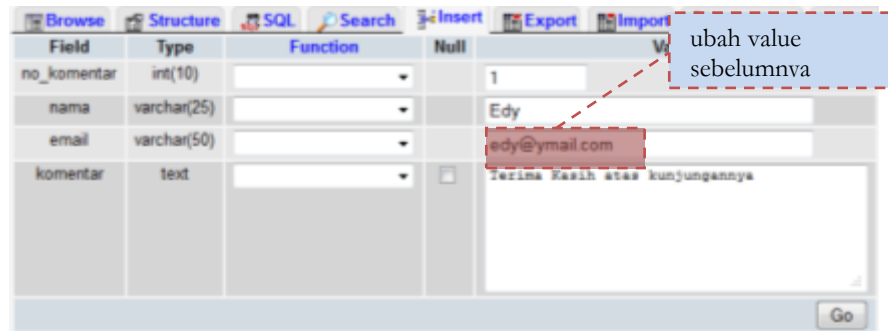
Gambar 9.14 SQL Select dan Hasil SQL Select (2)

Update dan Delete Tabel

Untuk melakukan update terhadap record yang telah disimpan pada 'tbl_komentar', dapat dilakukan dengan meng-klik tombol . Tombol tersebut digunakan untuk melakukan perubahan terhadap record yang dipilih. Dengan mengklik tombol tersebut maka akan muncul tampilan seperti gambar 2.16.



Gambar 9.15 Tombol update record



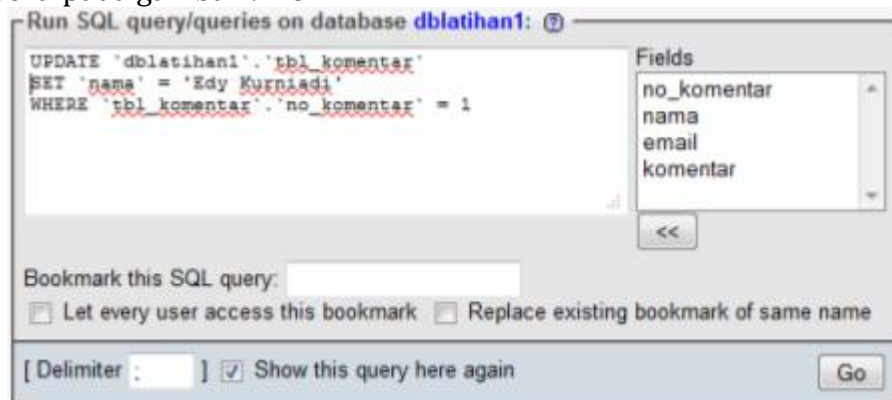
Gambar 9.16 Tampilan update record

Setelah mengubah nilai value yang sebelumnya “mr.edy@mmail.com” menjadi “edy@ymail.com” dan meng-klik tombol go, maka isi tabel tbl_komentar akan berubah seperti pada gambar 9.17.

	no_komentar	nama	email	komentar
<input type="checkbox"/>	1	Edy	edy@ymail.com	Terima Kasih atas kunjungannya
<input type="checkbox"/>	2	Ardi	ard@ymail.com	Terima Kasih atas informasinya

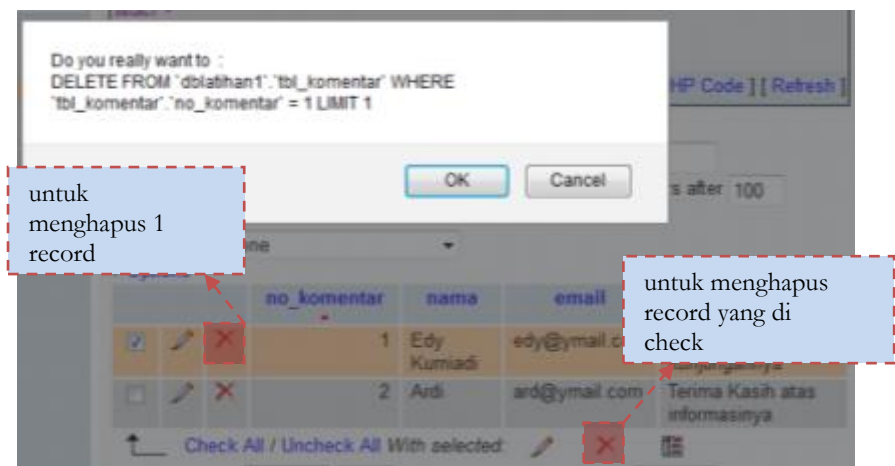
Gambar 9.17 Hasil update record

Untuk update record pada tbl_komentar menggunakan SQL pada database ‘dblatihan1’, gunakan perintah SQL update pada tab SQL seperti pada gambar 9.18.



Gambar 9.18 SQL update

Untuk melakukan delete terhadap record yang telah disimpan pada ‘tbl_komentar’, dapat dilakukan dengan meng-klik tombol . Tombol tersebut digunakan untuk melakukan penghapusan terhadap record yang dipilih. Dengan mengklik tombol tersebut maka akan muncul tampilan seperti gambar 2.19.



Gambar 9.20 Tampilan saat hapus record

Jika pada alert Delete form dipilih “Ok”, maka isi tabel tbl_komentar akan berubah seperti pada gambar 9.21.

	no_komentar	nama	email	komentar
<input type="checkbox"/>	2	Ardi	ard@gmail.com	Terima Kasih atas informasinya

Gambar 9.21 Hasil delete record

Untuk delete record pada tbl_komentar menggunakan SQL pada database ‘dblatihan1’, gunakan perintah SQL delete pada tab SQL seperti pada gambar 9.22.



Gambar 9.22 SQL Delete

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya

Buat database dengan nama db_perpustakaan dan buat/rancang tabel berikut ini :

tbl_anggota			
nomor_anggota	Nama_anggota	Alamat	No_telepon
1	Agnes	Medan	88884444
2	Bima	Medan	22221111
3	Yunita	Jakarta	11118888

tbl_staff			
Nomor_pegawai	Nama_pegawai	jabatan	Alamat
100	Mali	staff	Jl. Thamrin
200	Via	staff	Jl. Thamrin
300	Sinta	Manager	Jl. Asia

tbl_buku			
kode_buku	Judul_buku	Tahun	Pengarang_buku
111	Pemrograman Web	2011	Agustaf
222	Desai Web	2012	Irma
333	PHP	2011	Isma

tbl_peminjaman						
nomor_anggota	Nama_anggota	Judul_buku	kode_buku	Tanggal_pinjam	Tanggal_kembali	Denda
1	Agnes	Pemrograman Web	111	25/04/2013	27/04/2013	0
2	Bima	Desai Web	222	25/04/2013	27/04/2013	0
3	Yunita	PHP	333	25/04/2013	27/04/2013	0

Keterangan:

- Tentukan tipe data masing-masing tabel sesuai data yang dimasukkan

Mengakses MySQL Dengan PHP

MySQL adalah salah satu database server yang dapat diakses dengan PHP. PHP mendukung SQL untuk melakukan insert, update, dan delete terhadap record-record dalam database MySQL.

Secara umum, akses database melalui tiga tahapan:

1. Koneksi ke database (persiapan)
2. Query atau permintaan data (operasi)
3. Menutup koneksi

Koneksi ke Database

Untuk melakukan koneksi ke MySQL gunakan perintah berikut :

```
mysql_connect(servername, username,password);
```

atau

```
$koneksi = mysql_connect(servername, username,password);
```

dimana :

servername :Opsional. Menentukan server tujuan koneksi. Nilai default adalah "localhost"

username :Opsional. Menentukan nama pengguna untuk login. Nilai default adalah "root"

password :Opsional. Menentukan password untuk login dengan. Nilai default adalah ""

contoh:

```
<!--simpan dengan nama Config.php -->

<?php
$koneksi=mysql_connect("localhost","root","");
if(!$koneksi)
    echo "koneksi gagal: ".mysql_error();
else
    echo "koneksi Berhasil"
?>
```

Fungsi dari `mysql_error()` adalah untuk mengembalikan deskripsi kesalahan dari operasi MySQL terakhir.

Menutup Koneksi ke Database

Untuk menutup koneksi ke MySQL gunakan perintah berikut :

```
mysql_close();
```

atau

```
mysql_close($koneksi);
```

contoh:

```
<!--simpan dengan nama CloseConection.php -->

<?php
include("config.php");
mysql_close($koneksi);
if (!$koneksi)
{
    echo "<br>Gagal menutup koneksi ".mysql_error();
}
    echo "<br>Berhasil menutup koneksi";
?>
```

Memilih Database

Sebelum menggunakan database MySQL menggunakan PHP, database tersebut harus dipilih terlebih dahulu . Fungsi untuk memilih database adalah:

```
mysql_select_db ( databasename )
```

contoh:

```
<!--simpan dengan nama SelectDB.php -->

<?php
include("config.php");
$db=mysql_select_db (dblatihan1);
if (!$db){
    echo "<br>Gagal Memilih Database ".mysql_error();
}
else{
    echo "<br>Berhasil Memilih Database";
}

mysql_close($koneksi);
?>
```

Menjalankan Perintah SQL

Untuk mengoperasikan perintah-perintah SQL agar dapat memanipulasi database gunakan perintah berikut :

```
mysql_query(Query);
```

contoh:

```
<!--simpan dengan nama QueryDB.php -->

<?php
include("config.php");
$db=mysql_select_db (dblatihan1);
$query="INSERT INTO
tbl_komentar(`no_komentar`,`nama`,`email`,`komentar`)
VALUES('' , 'Agus', 'aguss@ymail.com', 'Terima Kasih')";
mysql_query($query);
mysql_close($koneksi);
?>
```

Mengambil Nilai Dari Record

Terdapat beberapa cara untuk mengambil nilai dari sebuah record:

- mysql_fetch_array()
- mysql_fetch_assoc()
- mysql_fetch_object()
- mysql_fetch_row()

a. mysql_fetch_array()

mysql_fetch_array() mengembalikan fungsi baris dari record sebagai array asosiatif atau array numerik.

contoh:

```
<!--simpan dengan nama fetch_array.php -->

<?php
include("config.php");
$db=mysql_select_db (dblatihan1);
$query = "SELECT * from tbl_komentar where no_komentar=1";
$result = mysql_query($query);
print_r(mysql_fetch_array($result));
mysql_close($koneksi);
?>
```

hasil :

```
Array (
    [0] => 1
    [no_komentar] => 1
    [1] => Edy Kurniadi
    [nama] => Edy Kurniadi
    [2] => edy@ymail.com
    [email] => edy@ymail.com
    [3] => Terima Kasih
    [komentar] => Terima Kasih
)
```

b. `mysql_fetch_assoc()`

`mysql_fetch_assoc()` mengembalikan fungsi baris dari record sebagai array asosiatif.

contoh:

```
<!--simpan dengan nama fetch_assoc.php -->

<?php
include("config.php");
$db=mysql_select_db (dblatihan1);
$query = "SELECT * from tbl_komentar WHERE no_komentar=1";
$result = mysql_query($query);
print_r(mysql_fetch_assoc($result));
mysql_close($koneksi);
?>
```

hasil :

```
Array (
    [no_komentar] => 1
    [nama] => Edy Kurniadi
    [email] => edy@ymail.com
    [komentar] => Terima Kasih
)
```

c. `mysql_fetch_object()`

`mysql_fetch_object()` mengembalikan fungsi baris dari record sebagai obyek.

contoh:

```
<!--simpan dengan nama fetch_object.php -->

<?php
include("config.php");
$db=mysql_select_db (dblatihan1);
$query = "SELECT * from tbl_komentar";
$result = mysql_query($query);
while ($row = mysql_fetch_object($result))
{
    echo $row->nama.$row->email . "<br />";
}
mysql_close($koneksi);
?>
```

hasil :

```
Edy Kurniadi edy@ymail.com
agus aguss@ymail.com
Mirza Mirza123@ymail.com
```

d. `mysql_fetch_row()`

`mysql_fetch_row()` mengembalikan fungsi baris dari record sebagai array numerik.

contoh:

```
<!--simpan dengan nama fetch_row.php -->

<?php
include("config.php");
$db=mysql_select_db (dblatihan1);
$query = "SELECT * from tbl_komentar WHERE no_komentar=1";
$result = mysql_query($query);
print_r(mysql_fetch_row($result));
mysql_close($koneksi);
?>
```

hasil :

```
Array (
    [0] => 1
    [1] => Edy Kurniadi
    [2] => edy@ymail.com
    [3] => Terima Kasih
)
```

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya

1. Dari db_perpustakaan, tampilkan semua record untuk setiap tabel yang ada pada db_perpustakaan sehingga hasilnya seperti gambar 10.1

TABEL ANGGOTA PERPUSTAKAAN			
nomor_anggota	nama_anggota	alamat	no_telepon
1	Agnes	Medan	88884444
2	Bima	Medan	22221111
3	Yunita	Jakarta	11118888

TABEL STAF PERPUSTAKAAN			
nomor_pegawai	nama_pegawai	jabatan	alamat
100	Mali	staff	jl. Thamrin
200	Via	staff	jl. Thamrin
300	Sinta	manager	jl. Asia

TABEL BUKU PERPUSTAKAAN			
kode buku	judul buku	tahun	pengarang Buku
111	Pemrograman Web	2011	Agustaf
2222	Desain Web	2012	Irma
333	PHP	2011	Ismi

TABEL PEMINJAMAN BUKU PERPUSTAKAAN						
nomor anggota	nama anggota	judul buku	tgl pinjam	tgl kembali	Denda	kode buku
1	Agnes	Pemrograman Web	2013-04-25	2013-04-27	0	111
2	Bima	Desain Web	2013-04-25	2013-04-27	0	222
3	Yunita	PHP	2013-04-25	2013-04-27	0	333

Gambar 10.1

2. Buat sebuah form untuk menghapus isi tabel staff sesuai Nomor_pegawai seperti gambar 10.2



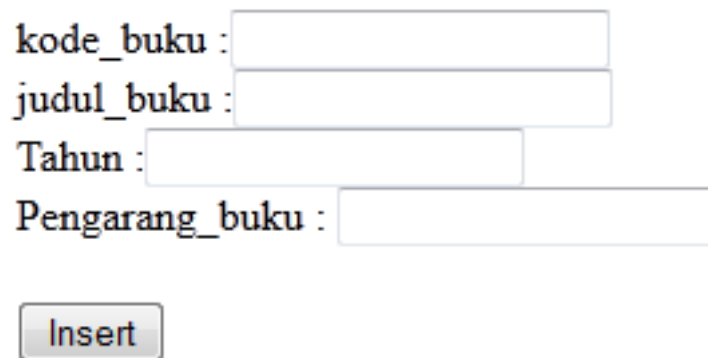
Nomor Pegawai :

Gambar 10.2

Keterangan:

Jika nomor pegawai ditemukan pada tabel staff, maka baris dari nomor pegawai yang sama akan dihapus dari tabel staff.

3. Buat sebuah form untuk mengisi tabel `tbl_buku` seperti gambar 10.3



kode_buku :

judul_buku :

Tahun :

Pengarang_buku :

Gambar 10.3

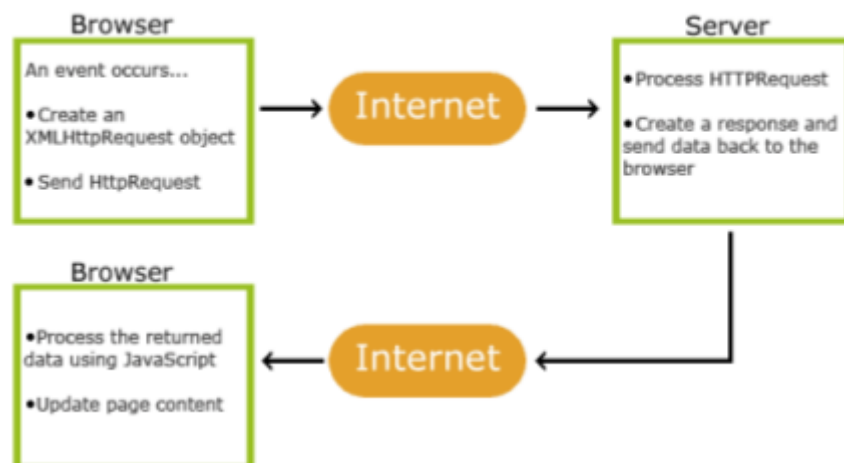
Keterangan:

Setiap data yang dimasukkan akan disimpan pada tabel buku pada database `db_perpustakaan`

AJAX

AJAX adalah singkatan dari Asynchronous JavaScript and XML. AJAX bekerja dengan berbasis Javascript dan request HTTP. Dengan menggunakan AJAX, pengembangan web dimungkinkan untuk bekerja secara asynchronous. Artinya pengiriman dan penerimaan data dari user ke server tidak perlu me-load ulang seluruh halaman tetapi hanya dilakukan pada bagian web yang hendak diubah.

How AJAX Works



Dengan AJAX, Javascript dapat langsung berkomunikasi dengan server dengan menggunakan objek Dengan objek ini, javascript dapat melakukan transaksi data dengan server web, tanpa harus me-reloading halaman web tersebut secara keseluruhan.

XMLHttpRequest

Semua modern browser mendukung objek XMLHttpRequest (untuk IE5 dan IE6 menggunakan XMLHttpRequest). Objek XMLHttpRequest digunakan untuk pertukaran data dengan server di belakang layar.

Dalam XMLHttpRequest, terbagi menjadi 4 tahapan yaitu:

- XMLHttpRequest Create Object
- XMLHttpRequest Request
- XMLHttpRequest readyState

- XHR Response

1. XHR Create Object

Sintaks untuk menciptakan objek XMLHttpRequest:

```
variable=new XMLHttpRequest();
```

Untuk versi lama dari Internet Explorer (IE5 dan IE6) menggunakan Object ActiveX:

```
variable=new ActiveXObject("Microsoft.XMLHTTP");
```

contoh:

```
<html>
<head>
<script type="text/javascript">
var objxhr;
if (window.ActiveXObject){
    try{
        objxhr = new ActiveXObject("Microsoft.XMLHTTP");
    }
    catch(e){
        alert(e);
    }
}
else{
    try{
        objxhr = new XMLHttpRequest();
    }
    catch (e){
        alert(e);
    }
}
</script>
</head>
<body>
<div id="myDiv"></div>
</body>
</html>
```

2. XHR Request

Metode `open()` dan `send()` dari object XMLHttpRequest digunakan untuk mengirim permintaan ke server

```
open(method,url,async);
send();
```

dimana :

Method	Deskripsi
open(method,url,async)	Menentukan jenis permintaan, URL, dan jika permintaan harus ditangani asynchronous atau tidak. <i>method</i> : jenis permintaan GET atau POST <i>url</i> : lokasi dan nama file pada server <i>async</i> : true (asynchronous) atau false

	(synchronous)
send(string)	Mengirim permintaan ke server. <i>string</i> : Kusus untuk method POST

Metode GET lebih sederhana dan lebih cepat daripada metode POST. Namun, permintaan POST dilakukan ketika:

- memperbarui file atau database di server
- Mengirim sejumlah besar data ke server (POST tidak memiliki keterbatasan ukuran)
- Mengirim input pengguna (yang dapat berisi karakter tidak dikenal), dimana metode POST aman daripada GET

contoh:

```
<html>
<head>
<script type="text/javascript">
var objxhr;
if (window.ActiveXObject){
    try{
        objxhr = new ActiveXObject("Microsoft.XMLHTTP");
    }
    catch(e){
        alert(e);
    }
}
else{
    try{
        objxhr = new XMLHttpRequest();
    }
    catch (e){
        alert(e);
    }
}
objxhr.open("GET","Data.txt");
objxhr.send(null);
</script>
</head>
<body>
<div id="myDiv"></div>
</body>
</html>
```

3. XHR readyState

Ketika permintaan dikirim ke server, ada beberapa tindakan yang dapat dilakukan berdasarkan respon dari server dengan menggunakan event:

```
onreadystatechange = function()
```

Onreadystatechange dipicu setiap kali terjadi perubahan pada readyState. Properti readyState menyangg status XMLHttpRequest. Dua sifat penting dari objek XMLHttpRequest adalah:

Properti	Deskripsi
readyState	Menyandang status XMLHttpRequest. mempunyai nilai antara 0-4: 0: Permintaan tidak diinisialisasi 1: Koneksi server terbentuk 2: Permintaan diterima 3: Memproses permintaan 4: Permintaan selesai dan respon siap diberikan
status	200: "OK" 404: Page not found

contoh:

```
<html>
<head>
<script type="text/javascript">
var objxhr;
if (window.ActiveXObject){
    try{
        objxhr = new ActiveXObject("Microsoft.XMLHTTP");
    }
    catch(e){
        alert(e);
    }
}
else{
    try{
        objxhr = new XMLHttpRequest();
    }
    catch (e){
        alert(e);
    }
}
objxhr.open("GET","Data.txt");
objxhr.send(null);

objxhr.onreadystatechange = function(){
if (objxhr.readyState==4){
    //XHR Response
}
};
</script>
</head>
<body>
<div id="myDiv"></div>
</body>
</html>
```

4. XHR Response

ResponseText atau properti responseXML digunakan untuk mendapatkan respon dari server dari objek XMLHttpRequest.

Properti	Deskripsi
responseText	mendapatkan respon data sebagai string

responseXML	mendapatkan respon data sebagai data XML
--------------------	--

contoh:

```
<html>
<head>
<script type="text/javascript">
var objxhr;
if (window.ActiveXObject){
    try{
        objxhr = new ActiveXObject("Microsoft.XMLHTTP");
    }
    catch(e){
        alert(e);
    }
}
else{
    try{
        objxhr = new XMLHttpRequest();
    }
    catch (e){
        alert(e);
    }
}
objxhr.open("GET","Data.txt");
objxhr.send(null);

objxhr.onreadystatechange = function(){
if (objxhr.readyState==4){
    var x = document.getElementById("myDiv");
    x.innerHTML = objxhr.responseText;
}
};
</script>
</head>
<body>
<div id="myDiv"></div>
</body>
```

Untuk menampilkan hasil baris kode diatas, buat sebuah file teks Data.txt dengan isi sebagai berikut:

```
<h1> Selamat Anda Berhasil Melakukan permintaan ke
Server</h1>
```

AJAX XML FILE

XML adalah markup language yang sama seperti HTML yang dirancang untuk membawa data, bukan untuk menampilkan data. Fungsinya adalah untuk menyimpan data dalam bentuk yang terstruktur. XML menggunakan tag-tag yang diciptakan sendiri untuk pendeskripsian dan penyimpanan data (disimpan dalam .xml). Syarat-syarat dalam penulisan XML adalah:

1. Semua Elemen XML Harus Memiliki Tag Penutup

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

```
<br />
```

2. Tag XML bersifat Case Sensitive

```
<Message>Salah</message>
<message>Benar</message>
```

3. Elemen XML yang Bersarang Harus Berpasangan dengan Benar

```
<b><i>This text is bold and italic</b></i>
<b><i>This text is bold and italic</i></b>
```

4. Dokumen XML Harus Memiliki Root

```
<root>
  <child>
    <subchild>.....</subchild>
  </child>
</root>
```

5. Nilai Atribut XML Harus diberi tanda kutip

```
<note date="12/11/2007">
  <to>Tove</to>
  <from>Jani</from>
</note>
```

Contoh:

```
<!-- Simpan dengan nama DaftarBuku.xml -->
<?xml version="1.0" encoding="ISO-8859-1"?>
<daftar>
  <buku>
    <judul bahasa='English'>Web Programming</judul>
    <kategori>Web</kategori>
    <penerbit>ABC Printing</penerbit>
    <pengarang>Smith</pengarang>
    <harga>75000</harga>
  </buku>
  <buku>
    <judul bahasa='Indonesia'>Jaringan Komputer</judul>
    <kategori>Networking</kategori>
    <penerbit>Andi Offset</penerbit>
    <pengarang>Budi SKom</pengarang>
    <harga>40000</harga>
  </buku>
  <buku>
    <judul bahasa='Indonesia'>Belajar C#</judul>
    <kategori>Programming</kategori>
    <penerbit>Andi Offset</penerbit>
    <pengarang>Joni MSc</pengarang>
    <harga>50000</harga>
  </buku>
</daftar>
```

AJAX

```
<judul bahasa='Indonesia'>HTML dan CSS</judul>
<kategori>Web</kategori>
<penerbit>Elex</penerbit>
<pengarang>Thomas</pengarang>
<harga>40000</harga>
</buku>
<buku>
  <judul bahasa='Indonesia'>Routing dan Switching
</judul>
  <kategori>Networking</kategori>
  <penerbit>Elex</penerbit>
  <pengarang>Joni MSc</pengarang>
  <harga>60000</harga>
</buku>
</daftar>
```

XML DOM memperlakukan dokumen XML dalam bentuk tree (pohon) sehingga Untuk membaca data-data dalam format XML dapat dilakukan seperti menggunakan DOM HTML dengan menggunakan responseXML.

contoh:

```
<html>
<head>
<script type="text/javascript">
var objxhr;
function createobject()
{
    if (window.ActiveXObject)
        objxhr = new ActiveXObject("Microsoft.XMLHTTP");
    else
        objxhr = new XMLHttpRequest();
}

function tampil()
{
    createobject();
    objxhr.open("GET", "DaftarBuku.xml");
    objxhr.send(null);
    objxhr.onreadystatechange = show;
}

function show()
{
    if (objxhr.readyState==4){
        var x = objxhr.responseXML.documentElement;
        var y = x.getElementsByTagName("buku");
        txt = "<table
border=1><tr><td>Judul</td><td>Bahasa</td><td>Kategori</td>
<td>Penerbit</td><td>Pengarang</td><td>Harga</td></tr>";

        for (i=0;i<y.length;i++){
            txt += "<tr>";
            y1 = y[i].getElementsByTagName("judul");
            txt+= "<td>"+y1[0].firstChild.data+ "</td>";
            txt+= "<td>"+y1[0].getAttribute("bahasa")+"</td>";
            y1 = y[i].getElementsByTagName("kategori");
```

```

        txt += ("<td>" + y1[0].firstChild.data + "</td>");
        y1 = y[i].getElementsByTagName("penerbit");
        txt += ("<td>" + y1[0].firstChild.data + "</td>");
        y1 = y[i].getElementsByTagName("pengarang");
        txt += ("<td>" + y1[0].firstChild.data + "</td>");
        y1 = y[i].getElementsByTagName("harga");
        txt += ("<td>" + y1[0].firstChild.data + "</td>");
        txt += "</tr>";
    }
    txt += "</table>";
    var x = document.getElementById("isi");
    x.innerHTML = txt;
}
</script>
</head>
<body>
<input type="button" value="Tampilkan Table"
onclick="javascript:tampil()" />
<div id="isi"></div>
</body>
</html>

```

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya

1. Buat kode HTML, JavaScript, dan Ajax untuk menghasilkan tampilan seperti gambar 11.1 dan 11.2

Keterangan:

- a. Sediakan terlebih dahulu sebuah file teks yang isinya adalah data barang yang ditempatkan pada sebuah tabel seperti gambar 11.2 (gunakan tag <table>, <tr>, <th>, dan <td> Html untuk pembuatan tabel).
- b. Jika tombol Tampilkan Daftar Barang pada gambar 11.1 diklik, maka baca tabel barang dari file teks yang telah disiapkan sebelumnya dan tampilkan tabel tersebut di bawah tombol Tampilkan Daftar Barang seperti gambar 11.2.

Tampilkan Daftar Barang

Gambar 11.1

Tampilkan Daftar Barang

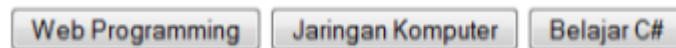
Kode Brg	Nama Brg	Harga
LAP-001	Laptop Toshiba	10.000.000
LAP-002	Laptop Acer	8.000.000
PC-001	Komputer DELL	5.000.000
PC-002	Komputer IBM	6.000.000

Gambar 11.2

2. Buat kode HTML, JavaScript, dan Ajax untuk menghasilkan tampilan seperti gambar 11.3, 11.4, 11.5, dan 11.6

Keterangan:

- Siapkan file XML berikut `DaftarBuku.xml` (seperti contoh diatas)
- Jika tombol Web Programming diklik maka keterangan mengenai buku Web Programming akan ditampilkan seperti gambar 11.4
- Jika tombol Jaringan Komputer diklik maka keterangan mengenai buku Jaringan Komputer akan ditampilkan seperti gambar 11.5
- Jika tombol Belajar C# diklik maka keterangan mengenai buku Belajar C# akan ditampilkan seperti gambar 11.6



Gambar 11.3

Web Programming Jaringan Komputer Belajar C#

Judul	Bahasa	Kategori	Penerbit	Pengarang	Harga
Web Programming	English	Web	ABC Printing	Smith	75000

Gambar 11.4

Web Programming Jaringan Komputer Belajar C#

Judul	Bahasa	Kategori	Penerbit	Pengarang	Harga
Jaringan Komputer	Indonesia	Networking	Andi Offset	Budi SKom	40000

Gambar 11.5

AJAX

Web Programming		Jaringan Komputer		Belajar C#	
Judul	Bahasa	Kategori	Penerbit	Pengarang	Harga
Belajar C#	Indonesia	Programming	Andi Offset	Joni MSc	50000

Gambar 11.6

AJAX DATABASE

Ajax dapat digunakan untuk mengakses Database dengan menggunakan PHP. Dengan Ajax, aplikasi web dapat mengirim data dan mengambil data dari database server secara asynchronous (di latar belakang) tanpa mengganggu tampilan halaman. Data dapat diambil dengan menggunakan objek XMLHttpRequest.

Contoh AJAX Database Select

Program ini menunjukkan cara melakukan menampilkan tabel user sesuai yang dipilih. Simpan file php berikut dengan nama SelectAjax.php

```
<html>
<head>
<script>
function showUser(str)
{
  if (str=="")
  {
    document.getElementById("hasil").innerHTML="<b>Hasilnya
akan tampil disini.</b>";
    return;
  }
  if (window.XMLHttpRequest)
  {
    xmlhttp=new XMLHttpRequest();
  }
  else
  {
    xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
  }
  xmlhttp.onreadystatechange=function()
  {
    if (xmlhttp.readyState==4)
    {
      document.getElementById("hasil").innerHTML=xmlhttp.responseText;
    }
  }
  xmlhttp.open("GET","SelectUser.php?pilihan="+str,true);
  xmlhttp.send();
}
```



```

</script>

<form>
<select name="users" onchange="showUser(this.value)">
<option value="">Select a person:</option>
<option value="1">Edy Kurniadi</option>
<option value="2">Agus</option>
<option value="3">Mirja</option>
</select>
</form>
<br>
<div id="hasil"><b>Hasilnya akan tampil disini.</b></div>

</body>
</html>

```

Simpan file php berikut dengan nama SelectUser.php

```

<?php
$pilihan=$_GET["pilihan"];
$koneksi=mysql_connect("localhost","root","");
if (!$koneksi){
    die('Could not connect: ' . mysql_error());
}
$db=mysql_select_db (dblatihan1);

$sql="SELECT * FROM tbl_komentar WHERE no_komentar =
'".$pilihan."'";
$result = mysql_query($sql);

echo "<table border='1'><tr>
<th>no_komentar</th>
<th>Nama</th>
<th>E-mail</th>
<th>Komentar</th>
</tr>";

$row = mysql_fetch_object($result);
echo "<tr><td>".$row->no_komentar."</td><td>".$row->
nama."</td><td>".$row->email ."</td><td>".$row->
komentar."</td></tr>";

echo "</table>";

mysql_close($koneksi);
?>

```

Contoh AJAX Database Insert

Program ini menunjukkan cara melakukan menambahkan isi tabel sesuai yang dimasukkan pada form. Simpan file php berikut dengan nama InsertAjax.php

```

<html>
<head>
<script>
function Input()
{
var nama = document.getElementById("nama").value;
var email = document.getElementById("email").value;
var komentar = document.getElementById("komentar").value;
if (window.XMLHttpRequest)
{
xmlhttp=new XMLHttpRequest();
}
else
{
xmlhttp=new ActiveXObject("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function()
{
if (xmlhttp.readyState==4)
{

document.getElementById("hasil").innerHTML+xmlhttp.responseText;
}
}
xmlhttp.open("GET","InsertUser.php?nama="+nama+"&email="+email+"&komentar="+komentar,true);
xmlhttp.send();
}
</script>
</head>
<body>

<form>
<table>
<tr>
<td>Nama </td><td>:<input type="text" id="nama" /><td>
</tr>
<tr>
<td>Email</td><td>:<input type="text" id="email" /></td>
</tr>
<tr>
<td>Komentar</td><td>:<input type="text" id="komentar" /></td>
</tr>
</table>
<button type="button" onclick="Input()"><b>Hasilnya akan tampil disini.</b></button>
<div id="hasil"></div>
</form>
</body>
</html>

```

Simpan file php berikut dengan nama InsertUser.php

```

<?php
$nama=$_GET["nama"];
$email=$_GET["email"];
$komentar=$_GET["komentar"];
if($nama!=""&&$email!=""&&$komentar!="") {
    $koneksi=mysql_connect("localhost","root","");
    if (!$koneksi)
    {
        die('Could not connect: ' . mysql_error());
    }
    $db=mysql_select_db(dblatihan1);
    $sql= "INSERT INTO tbl_komentar VALUES
('','$nama','$email','$komentar')";
    $result = mysql_query($sql);
    if($result){
        $sql="SELECT * FROM tbl_komentar";
        $result = mysql_query($sql);
        echo "<table border='1'><tr>
<th>no_komentar</th>
<th>>Nama</th>
<th>E-mail</th>
<th>Komentar</th>
</tr>";

        while($row = mysql_fetch_object($result)){
            echo "<tr><td>".$row->no_komentar."</td><td>".$row->nama."</td><td>".$row->email."</td><td>".$row->komentar."</td></tr>";
        }
        echo "</table>";
        mysql_close($koneksi);
    }
    else{
        echo "query salah";
    }
}
else{
    echo "data tidak lengkap";
}
?>

```

Contoh AJAX Database Delete

Program ini menunjukkan cara melakukan penghapusan isi tabel sesuai yang dipilih. Simpan file php berikut dengan nama DeleteAjax.php

```

<html>
<head>
<script>
function showUser(str)
{
    if (window.XMLHttpRequest)
    {
        xmlhttp=new XMLHttpRequest();
    }
}

```

```

else
{
    xmlhttp=new XMLHttpRequest("Microsoft.XMLHTTP");
}
xmlhttp.onreadystatechange=function()
{
    if (xmlhttp.readyState==4)
    {

document.getElementById("hasil").innerHTML+xmlhttp.responseText;
    }
}
xmlhttp.open("GET","DeleteUser.php?pilihan="+str,true);
xmlhttp.send();
}
</script>
</head>
<body onload="showUser()">

<form name="frm">
<select name="users">
<?php

$koneksi=mysql_connect("localhost","root","");
if (!$koneksi)
{
    die('Could not connect: ' . mysql_error());
}

$db=mysql_select_db (dblatihan1);
$sql="SELECT * FROM tbl_komentar";
$result = mysql_query($sql);
while($row = mysql_fetch_object($result)){
    echo "<option value='".$row->no_komentar."'>".$row->no_komentar."</option>";
}
echo "</select>";

mysql_close($koneksi);
?>
<button type="button"
onclick="showUser(document.frm.users.value)">Hapus</button>
</form>
<br>
<div id="hasil"><b>Hasilnya akan tampil disini.</b></div>

</body>
</html>

```

Simpan file php berikut dengan nama InsertUser.php

```

<?php
$pilihan=$_GET["pilihan"];
$koneksi=mysql_connect("localhost","root","");
if (!$koneksi)
{
    die('Could not connect: ' . mysql_error());
}

$db=mysql_select_db (dblatihan1);
if($pilihan!=""){
    $sql="DELETE FROM tbl_komentar WHERE no_komentar =
    '". $pilihan ."'";
    $result = mysql_query($sql);
}
$sql="SELECT * FROM tbl_komentar";
$result = mysql_query($sql);

echo "<table border='1'><tr>
<th>no_komentar</th>
<th>>Nama</th>
<th>E-mail</th>
<th>Komentar</th>
</tr>";

while($row = mysql_fetch_object($result)){
    echo "<tr><td>". $row->no_komentar."</td><td>". $row->
    nama."</td><td>". $row->email . "</td><td>". $row->
    komentar."</td></tr>";
}
echo "</table>";

mysql_close($koneksi);
?>

```

Kegiatan Mandiri.

Lakukan kegiatan mandiri berikut. Hasil Kegiatan Mandiri dikumpulkan ke Scele. Laporan Kegiatan ditulis dalam LogBook dan dikumpulkan pada pertemuan berikutnya

1. Buat sebuah database yang mempunyai tabel user berikut:

user	password
skj110	1234
dki117	77777
Sinsin	214214
Boy77	1357
Bo	333

Buat kode HTML, JavaScript, dan Ajax untuk menghasilkan tampilan seperti gambar 12.1

User :

Password :

gambar 12.1

Keterangan:

- Jika User dan Password yang dikirimkan sesuai dengan User dan Password yang disimpan pada database, maka hasilnya akan tampak seperti gambar 12.2.
- Jika User dan Password yang dimaukan tidak sesuai dengan User dan Password yang disimpan pada database, maka hasilnya akan tampak seperti gambar 12.3.

Akses ditolak!

User : sari

Password :

Gambar 12.2

Akses diterima :)

User : skj110

Password :

Gambar 12.3

- Buat kode HTML, JavaScript, dan Ajax untuk menghasilkan tampilan seperti gambar 12.4

Nama :

Alamat :

No.Telepon :

<input type="button" value="Tambah Data"/>		<input type="button" value="Reset"/>		
Nomor Anggota	Nama Anggota	Alamat	No.Tlpn	Hapus
2	Bima	Medan	22221111	<input type="button" value="Hapus"/>
1	Agnes	Medan	88884444	<input type="button" value="Hapus"/>
3	Yunita	Jakarta	11118888	<input type="button" value="Hapus"/>

Gambar 12.4

Keterangan:

- Data yang dimasukkan ke dalam form akan disimpan ke dalam database jika tombol “Tambah Data” diklik. Secara langsung tabel yang menampilkan semua data akan langsung berubah (tanpa refresh halaman).

AJAX

- b. Tombol hapus digunakan untuk menghapus data pada baris yang diinginkan. Tanpa refresh halaman, tabel akan berubah setelah proses penghapusan.