

MODUL PRAKTIKUM

PEMROGRAMAN WEB



**UNIVERSITAS
TUNAS PEMBANGUNAN**

Patriotisme, Kepelaporan, Kemandirian

DISUSUN OLEH :

MOH. ERKAMIM, S.Kom., M.Kom

PROGRAM STUDI D4 SISTEM INFORMASI KOTA CERDAS

FAKULTAS TEKNIK

**UNIVERSITAS TUNAS PEMBANGUNAN
SURAKARTA**

HALAMAN PENGESAHAN

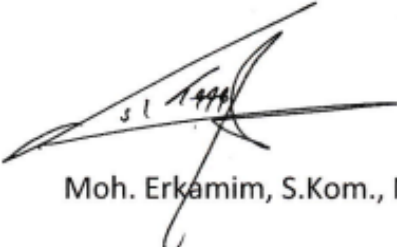
Matakuliah : Pemrograman Web
Kode Dokumen : VSI3317/02/2022
Jenis Bahan Ajar : Modul Praktikum
Program Studi : D4 Sistem Informasi Kota Cerdas
Fakultas : Teknik
Perguruan Tinggi : Universitas Tunas Pembangunan Surakarta
Penyusun : Moh. Erkamim, S.Kom., M.Kom

Modul ini disusun sebagai bahan ajar atau petunjuk praktikum dalam mata kuliah **Pemrograman Web.**

Mengesahkan
Dekan Fakultas Teknik



Disahkan
Surakarta, 8 Agustus 2022
Menyetujui
Ketua Program Studi
D4 Sistem Informasi Kota Cerdas



Moh. Erkamim, S.Kom., M.Kom

DAFTAR ISI

HALAMAN PENGESAHAN	ii
DAFTAR ISI.....	iii
TATA TERTIB.....	iv
MODUL I <i>HYPertext MARKUP LANGUAGE</i> (HTML).....	1
MODUL II <i>CASCADING STYLE SHEETS</i> (CSS)	11
MODUL III STRUKTUR DASAR PHP	22
MODUL IV PERNYATAAN KONDISI	31
MODUL V PERNYATAAN PERULANGAN.....	36
MODUL VI ARRAY.....	41
MODUL VII FORM.....	47
MODUL VIII FUNCTION	59
MODUL IX SESSION DAN COOKIES.....	62
MODUL X DATABASE DALAM PHP	69
MODUL XI LANJUTAN DATABASE DALAM PHP	77
MODUL XII LANJUTAN DATABASE DALAM PHP.....	84
MODUL XIII PEMROGRAMAN BERORIENTASI OBYEK DALAM PHP	95

TATA TERTIB
Praktikum Menggunakan Laboratorium Pemrograman
Program Studi Sistem Informasi Kota Cerdas
Fakultas Teknik Universitas Tunas Pembangunan Surakarta

Praktikum Laboratorium Pemrograman adalah salah satu kegiatan penting dalam Program Studi Sistem Informasi Kota Cerdas. Agar praktikum berjalan dengan lancar dan efisien, serta menjaga keselamatan dan kedisiplinan, berikut adalah tata tertib yang harus diikuti oleh seluruh mahasiswa yang mengikuti praktikum ini:

1. Pendaftaran dan Presensi:
 - ✓ Setiap mahasiswa wajib mendaftar sebelum memulai praktikum.
 - ✓ Mahasiswa harus mencatat kehadiran mereka dalam buku presensi yang telah disediakan.
2. Pakaian:
 - ✓ Menggunakan pakaian yang sesuai dengan aturan laboratorium.
 - ✓ Tidak diperkenankan mengenakan sandal jepit, sandal tidur, atau pakaian yang tidak pantas.
3. Perlengkapan:
 - ✓ Pastikan membawa alat tulis, buku catatan, dan laptop (jika diperlukan) sesuai dengan petunjuk dosen.
 - ✓ Dilarang membawa makanan atau minuman ke dalam laboratorium.
4. Kedisiplinan:
 - ✓ Mahasiswa diharapkan tiba tepat waktu sesuai jadwal praktikum.
 - ✓ Dilarang berbicara keras atau mengganggu mahasiswa lain selama praktikum berlangsung.
 - ✓ Matikan telepon seluler atau atur ke mode senyap selama praktikum.
5. Keselamatan:
 - ✓ Ikuti semua petunjuk keamanan yang diberikan oleh dosen.
 - ✓ Tidak diperkenankan mencolokkan atau mencabut kabel-kabel tanpa izin.
 - ✓ Hindari mengganggu peralatan laboratorium yang tidak relevan dengan praktikum yang sedang berlangsung.
6. Penggunaan Perangkat Keras dan Perangkat Lunak:
 - ✓ Gunakan perangkat keras dan perangkat lunak laboratorium dengan bijak.
 - ✓ Jangan mencoba mengakses atau mengubah perangkat keras atau perangkat lunak laboratorium tanpa izin.
7. Tugas dan Proyek:
 - ✓ Serahkan tugas atau proyek sesuai dengan jadwal yang telah ditentukan oleh dosen.
 - ✓ Jangan melakukan plagiarisme atau menyalin pekerjaan orang lain.

8. Perilaku Etis:

- ✓ Hormati hak dan privasi mahasiswa dan dosen lain.
- ✓ Jangan menyebarkan informasi pribadi mahasiswa atau dosen tanpa izin.

9. Keteraturan Ruang:

- ✓ Setelah selesai praktikum, pastikan meja dan area kerja Anda dalam keadaan bersih dan rapi.
- ✓ Pastikan semua perangkat yang digunakan telah dimatikan sebelum meninggalkan laboratorium.

10. Sanksi:

- ✓ Pelanggaran terhadap tata tertib ini dapat mengakibatkan sanksi, seperti penurunan nilai atau pembatalan hak untuk mengikuti praktikum.

Semua mahasiswa diharapkan mematuhi tata tertib ini untuk menjaga keamanan, ketertiban, dan kelancaran praktikum laboratorium pemrograman.

MODUL I *HYPertext MARKUP LANGUAGE* (HTML)

A. TUJUAN :

1. Mahasiswa dapat mengenal dasar-dasar HTML
2. Mahasiswa dapat menguasai dasar-dasar HTML
3. Mahasiswa dapat mengenal dan memahami blok program HTML
4. Mahasiswa dapat memahami setiap object sebagai dasar pembuatan interface pada halaman browser
5. Mahasiswa dapat membuat halaman web sederhana dengan HTML

B. DASAR TEORI

HTML merupakan singkatan dari HyperText Markup Language adalah script untuk menyusun dokumen-dokumen Web. Dokumen HTML disimpan dalam format teks reguler dan mengandung tag-tag yang memerintahkan web browser untuk mengeksekusi perintah-perintah yang dispesifikasikan.

1. Penamaan Dokumen HTML

Penamaan dokumen dapat dilakukan pada saat melakukan penyimpanan, pada saat mengetikkan nama dile, maka file diberikan ekstensi .html, contoh : namafilename.html.

2. Struktur Dokumen HTML

Dokumen HTML mempunyai tiga buah tag utama yang membentuk struktur dari dokumen HTML yaitu HTML, HEAD dan BODY. Tag HTML berfungsi untuk menyatakan suatu dokumen HTML, tag HEAD berfungsi untuk memberikan informasi tentang dokumen HTML dan tag BODY berfungsi untuk menyimpan informasi atau data yang akan ditampilkan dalam dokumen HTML. Elemen yang terdapat pada sebuah dokumen HTML terbagi atas dua bagian yaitu Head dan Body.

a. Bagian Head

Bagian header dari dokumen HTML diapit dengan tag `<HEAD>....</HEAD>` di dalam bagian ini biasanya dimuat tag `<TITLE>....</TITLE>` yang menampilkan judul dari halaman pada browser.

b. Bagian Body

Dokumen body diapit dengan tag `<BODY>....</BODY>`, bagian body digunakan untuk menampilkan text, image, link dan semua yang akan ditampilkan pada web page.

Contoh Penulisan Struktur HTML :

```
<!doctype html>
<html>
<head>
    <title>Disini Judul Dokumen HTML </title>
</head>
<body>
```

Disini Penulisan Informasi Web

`</body>`

`</html>`

Penjelasan :

`<! Doctype html>`

Adalah tag awal dari setiap dokumen HTML, tag ini berfungsi untuk menginformasikan pada browser bahwa dokumen yang sedang dibuka adalah dokumen HTML.

`<html>.....</html>`

Awal dari isi dokumen HTML, semua kode HTML yang dibuat akan ditulis dalam tag ini.

`<head>.....</head>`

Tag head digunakan untuk menyimpan informasi tentang dokumen HTML.

`<title>.....</title>`

Salah satu contoh informasi yang terdapat didalam tag HEAD adalah title. Title akan menampilkan teks pada judul browser atau pada tab browser.

`<body>.....</body>`

Merupakan tag pembuka dari badan dokumen HTML.

3. Elemen dan Tag Pada Dokumen HTML

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

Dokumen HTML terdiri dari elemen-elemen yang saling berkaitan untuk membentuk suatu dokumen HTML yang utuh, antara lain elemen HTML, elemen Head, elemen Body yang masing-masing dapat diapit tanda yang disebut delimiter(`<....>`).

Elemen yang telah diapit dengan tanda delimiter disebut dengan Tag.

Contoh penulisan :

`<nama elemen>`

Misalnya : **`<html>`**

Perlu diingat juga bahwa pada setiap elemen atau tag harus selalu diakhiri atau ditutup dengan menuliskan kembali tag dan menambahkan tanda garis miring `"/`”.

Cara penulisan :

`<nama elemen>`

Misalnya : **`</html>`**

Tag HTML dasar

Berikut adalah tag HTML dasar :

Tag	Deskripsi
<code><html></code>	Mendefinisikan sebuah dokumen HTML
<code><body></code>	Mendefinisikan bagian body dokumen HTML
<code><h1> sampai <h6></code>	Mendefinisikan header 1 sampai 6
<code><p></code>	Mendefinisikan paragraph
<code>
</code>	Menambahkan line break (spasi baru)
<code><hr></code>	Membuat garis datar (horizontal)
<code><!--></code>	Mendefinisikan komentar

a. Heading dan Paragraph

Heading digunakan untuk memberikan penjudulan pada suatu dokumen HTML. Untuk melakukan format heading maka menggunakan tag `<h1>....</h1>` untuk judul utama dan untuk judul sub bab dapat menggunakan tag `<h2>....</h2>` sampai dengan `<h6>....</h6>`. Untuk membuat sebuah paragraph dalam dokumen HTML maka digunakan tag `<p>...</p>`.

b. Link

Link atau biasa disebut dengan anchor (pengait) dapat dibuat dengan menambahkan tag `<a>....` serta atribut href agar link tersebut dapat menyimpan alamat web yang akan dituju ketika link diklik. Sehingga sintaks link adalah :

```
<a href='alamat web yang dituju'></a>
```

c. List

Dalam dokumen HTML suatu daftar atau list adalah suatu kumpulan item tertentu, daftar item dapat ditulis secara berurutan atau tidak. Jenis list dalam HTML adalah :

- *Ordered list*
Ordered list dapat dibuat dengan menggunakan tag `....` dan untuk setiap listnya menggunakan tag `....`.
- *Unordered list*
Unordered list dapat dibuat dengan menggunakan tag `....` dan untuk setiap listnya menggunakan tag `....`.

d. Image

Dalam dokumen HTML kita dapat memuat dan menampilkan gambar. Untuk memuat gambar menggunakan tag `` untuk menentukan file gambar maka digunakan atribut src. Sehingga sintaks image adalah :

```
<img src ='nama file gambar.jpg'>
```


e. Form

Sebuah form dalam dokumen HTML dapat memiliki bermacam kontrol, mulai dari text input, combo box, button dan lain sebagainya. Pembuatan sebuah form diawali dengan menggunakan tag :

```
<form>....</form>
```

Berikut beberapa kontrol form yang biasa digunakan dalam HTML:

- **Label**

Label digunakan untuk memberikan keterangan pada setiap input yang ada.

Sintaks form label adalah :

```
<label for='nama'>keterangan input</label>
```

- **Text**

Kontrol input ini dapat diisi dengan teks yang memiliki panjang kata terbatas, biasa digunakan dalam form pencarian, nama dan lainnya. Sintaks form text adalah :

```
<input type='text' name='nama'>
```

- **Combobox**

Combobox adalah kontrol yang memiliki pilihan ketika diklik. Sintaks form combobox adalah :

```
<select>
```

```
<option value="name value 1"></option>
```

```
<option value="name value 2"></option>
```

```
<option value="name value 3"></option>
```

```
</select>
```

f. Tabel

Untuk menampilkan data dengan tipe tabel dalam HTML, dapat menggunakan tag :

```
<table>...</table>
```

Sedangkan untuk menampilkan baris tabel maka digunakan tag :

```
<tr>...</tr>
```

Sedangkan untuk menentukan banyaknya kolom maka menggunakan tag :

```
<td>...</td>
```

g. DIV

Penggunaan tag <div> adalah sebagai container atau pembatas satu struktur HTML dengan struktur lainnya. dalam struktur HTML terdapat beberapa bagian antara lain :

- **Header**

Bagian kepala website yang berisi logo, nama, slogan, menu dan lainnya. Sintaks

<div> untuk footer adalah :

```
<div id='header'>.....</body>
```

- **Content**

Berisi isi website itu sendiri, contoh content berisi postingan terbaru atau informasi lainnya. sintaks <div> untuk content adalah :

```
<div id='content'>.....</body>
```

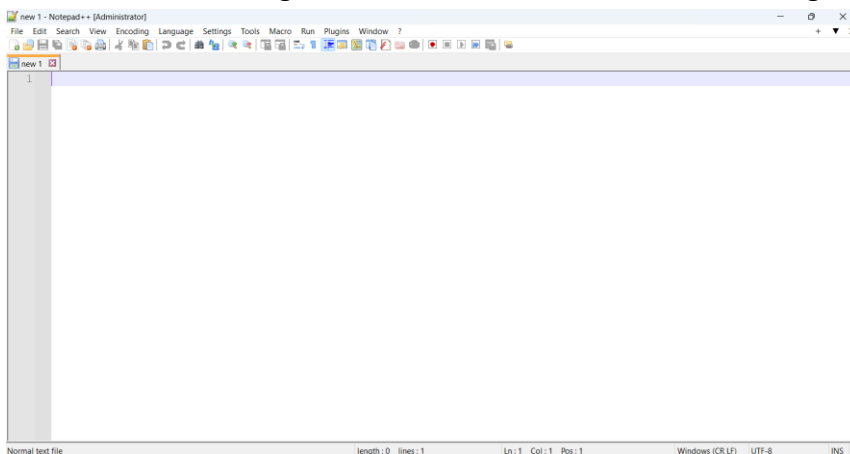
- **Sidebar**
Berada disamping content dan biasanya berisi iklan, kategori artikel dan widget dan lainnya. Sintaks <div> untuk sidebar adalah :
<div id='sidebar'>.....</body>
- **Footer**
Bagian kaki dari website yang pada umumnya diisi dengan tag website tersebut, misalnya copyright©2018 by me. Sintaks <div> untuk footer adalah :
<div id='footer'>.....</body>

C. KEGIATAN PRAKTIKUM

1. Membuat Heading dan Paragraph

Langkah-langkah praktikum

- Buka direktori **C > xampp > htdocs**, kemudian buat folder dan beri nama **"praktikum_web"**
- Buka editor **Notepad++** atau **Sublime Text**
- Buatlah file baru dengan membuka **menu file > new** atau dengan shortcut **ctrl + N**



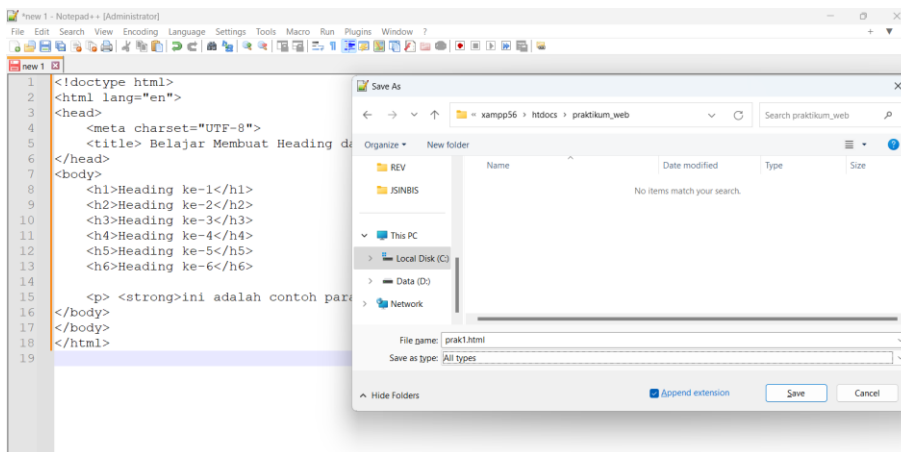
- Tuliskan coding berikut:

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title> Belajar Membuat Heading dan Paragraph</title>
</head>
<body>
  <h1>Heading ke-1</h1>
  <h2>Heading ke-2</h2>
  <h3>Heading ke-3</h3>
  <h4>Heading ke-4</h4>
  <h5>Heading ke-5</h5>
  <h6>Heading ke-6</h6>

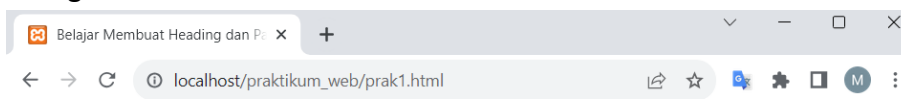
  <p> <strong>ini adalah contoh paragraph</strong></p>
```

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

- Simpan coding yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > praktikum_web**, dan beri nama file dengan **“prak1.html”** atau pilih **“Hypertext Markup Language File”** pada **Save as type**, klik **Save**.



- Buka **Xampp** kemudian aktifkan **Apache dan MySQL**, dengan menekan tombol **Start** pada modul **Apache dan MySQL**
- Lakukan kompilasi program melalui browser dengan mengetikkan <http://localhost/namafolder/namafile> , yaitu http://localhost/praktikum_web/prak1.html Maka pada browser akan muncul hasil sebagai berikut:



Heading ke-1

Heading ke-2

Heading ke-3

Heading ke-4

Heading ke-5

Heading ke-6

ini adalah contoh paragraph

2. Praktikum membuat List, salin koding dibawah ini :

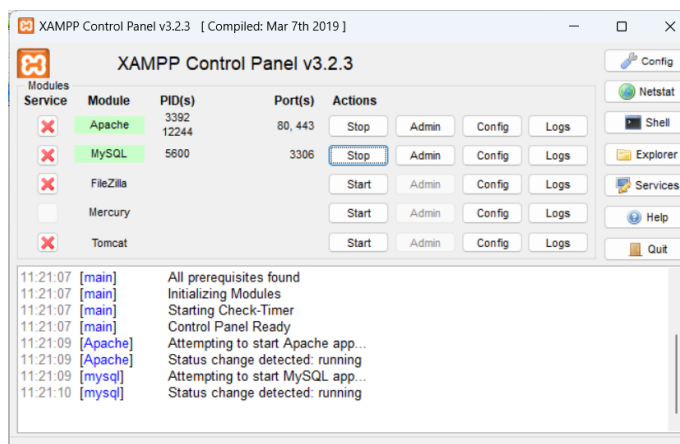
```
<!doctype html>
<html>
<head>
  <title>List</title>
</head>
```

```

<body>
  <h3>Membuat Unordered List</h3>
  <strong><i><u>Bahan-bahan Memasak Telur Goreng</u></i></strong>
  <ul>
    <li>Telur 3 butir</li>
    <li>Margarin (1 sdm)</li>
    <li>Batang Daun bawang</li>
    <li>Bawang merah (2 butir)</li>
    <li>cabai merah keriting(2 buah)</li>
    <li>lada 1/4 sdt</li>
    <li>garam 1/4 sdt</li>
  </ul>
  <h3>Membuat Ordered List</h3>
  <strong><i><u>Cara Memasak Telor Goreng</u></i></strong>
  <ol>
    <li>Campur semua bahan di atas</li>
    <li>Kocok semua bahan hingga merata</li>
    <li>Panaskan margarin</li>
    <li>Goreng dan dadar hingga kuning keemasan (matang)</li>
    <li>Selesai dan masakan di hidangkan</li>
  </ol>
</body>
</html>

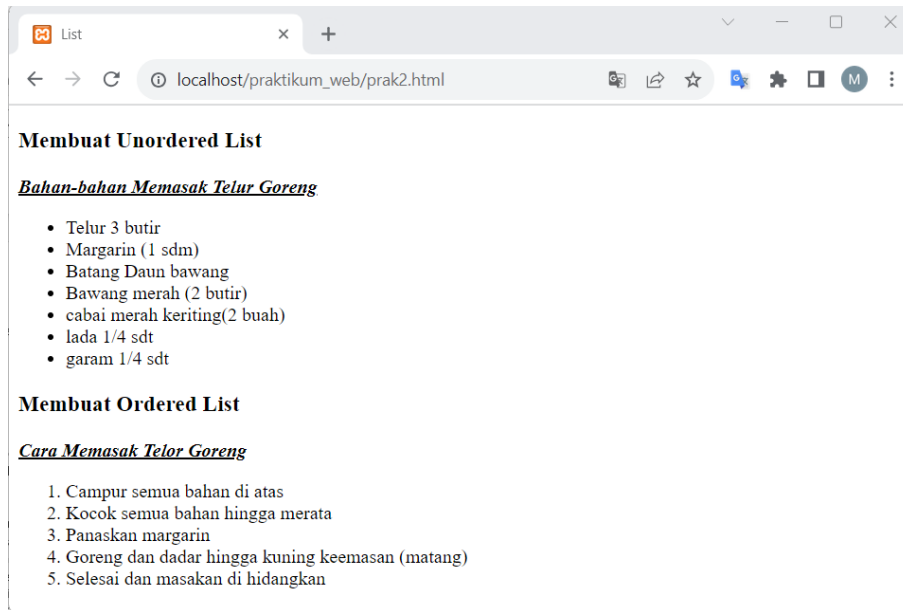
```

- Simpan coding yang telah dituliskan dengan membuka menu file > save as... pilih lokasi penyimpanannya pada direktori C > xampp > htdocs > praktikum_web, dan beri nama file dengan “prak2.html” atau pilih “Hypertext Markup Language File” pada Save as type, klik Save.
- Buka Xampp kemudian aktifkan Apache dan MySQL, dengan menekan tombol Start pada modul Apache dan MySQL.



- Lakukan kompilasi program melalui browser dengan mengetikkan `http://localhost/namafolder/namafile` , yaitu

http://localhost/praktikum_web/prak2.html Maka pada browser akan muncul hasil sebagai berikut:



3. Praktikum gambar dan tabel, salin koding dibawah ini:

```
<!doctype html>
<html>
<head>
  <title>Image & Table (Harga Laptop)</title>
</head>
<body>
  <h3>Daftar Barang (Merk dan Type)</h3>
  <table border="1">
    <caption>Tabel Harga Produk</caption>
    <tr>
      <th>Kode Produk</th>
      <th>Nama Produk</th>
      <th>Harga</th>
      <th>Gambar</th>
    </tr>
    <tr>
      <td>L02301</td>
      <td>Laptop Dell 7100</td>
      <td>400.000</td>
      <td></td>
    </tr>
    <tr>
      <td>L02302</td>
      <td>Laptop Asus A405</td>
      <td>1700.000</td>
    </tr>
  </table>
</body>
</html>
```

```

        <td></td>
</tr>
<tr>
<td>L02303</td>
<td>Laptop HP X100</td>
<td>2000.000</td>
<td></td>
</tr>
<tr>
<td>L02304</td>
<td>Laptop Lenovo G302</td>
<td>2000.000</td>
<td></td>
</tr>
</table>
</body>
</html>

```





- Simpan *coding* yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > praktikum_web**, dan beri nama file dengan **"prak3.html"** atau pilih **"Hypertext Markup Language File"** pada Save as type, klik *Save*.
- Lakukan kompilasi program melalui browser dengan mengetikkan <http://localhost/namafolder/namafile>, yaitu http://localhost/praktikum_web/prak3.html Maka pada browser akan muncul hasil sebagai berikut:

Image & Table (Harga Laptop) x +

localhost/praktikum_web/prak...

Daftar Barang (Merk dan Type)

Tabel Harga Produk

Kode Produk	Nama Produk	Harga	Gambar
L02301	Laptop Dell 7100	9.000.000	
L02302	Laptop Asus A405	8.500.000	
L02303	Laptop HP X100	8.000.000	
L02304	Laptop Lenovo G302	7.500.000	

D. TUGAS

Buatlah tampilan website Foodcourt dengan ketentuan:

- Header berisikan Nama foodcourt, Gambar
- Konten berisikan daftar menu makanan, harga dan gambar makanan dalam bentuk tabel
- Link untuk mengakses Foodcourt
- Footer yang berisikan Copyright nama anda

MODUL II CASCADING STYLE SHEETS (CSS)

A. TUJUAN :

1. Mahasiswa dapat mengenal dasar-dasar CSS
2. Mahasiswa dapat menguasai dasar-dasar CSS
3. Mahasiswa dapat memahami setiap object sebagai dasar pembuatan interface pada halaman browser
4. Mahasiswa dapat membuat halaman web sederhana dengan CSS

B. DASAR TEORI

CSS adalah kependekan dari *Cascading Style Sheet*, berfungsi untuk memperindah tampilan HTML atau menentukan bagaimana elemen HTML ditampilkan, seperti menentukan posisi, merubah warna teks atau background dan lainnya. Berikut adalah sintak dasar dari CSS :



Penjelasan :

1. Selector

Selector adalah elemen/tag HTML yang ingin diberi style. Kita dapat menuliskan langsung nama tag yang ingin diberi style tanpa perlu menambahkan tanda <>. Pada contoh kode CSS diatas kita akan memberi style pada seluruh tag *h1* yang terdapat dalam file HTML.

Jika tag HTML yang ingin diberi style memiliki ID, dapat menuliskan nama ID tersebut dengan diawali tanda kress (#). Contoh :

#header

Dan jika tag yang diberi style memiliki Class, maka penulisan selector bisa dilakukan dengan tanda titik (.) diikuti dengan nama class. Contoh :

.artikel

Jika hanya menuliskan satu selector, seperti contoh kode CSS di atas, maka seluruh tag *h1* yang terdapat dalam file HTML akan memiliki style yang sama. Bagaimana jika kita hanya ingin memberi style pada tag *h1* yang *hanya terdapat di dalam Class* artikel. Maka penulisan selectornya seperti berikut :

.artikel h1

Kode tersebut akan memerintahkan pada browser untuk memberi style pada tag h1 yang hanya terdapat di dalam *class artikel* (atau - h1 yang merupakan child dari class artikel).

Kita dapat memilih lebih dari satu tag untuk penghematan kode CSS. Misalnya ketika anda memiliki dua atau lebih tag dengan warna background yang sama, dari pada menuliskan kode seperti ini :

```
h1{ background-color: #666666; }
{ background-color: #666666; }
a { background-color:
```

kita dapat menggabungkan selector dengan menambahkan tanda koma pada nama tag yang ingin diberi style.

```
h1, p, a { background-color: #666666; }
```

2. Property dan Value

Property adalah sifat-sifat yang ingin diterapkan pada selector, seperti warna text, warna background, jarak antar elemen, garis pinggir dan lainnya. Untuk memberikan nilai value pada property digunakan tanda titik dua (:). Setiap property diakhiri dengan tanda koma (;).

a) Penulisan CSS

Ada 3 cara penulisan kode CSS yaitu inline, internal, dan external.

- **Inline**

Penulisan kode CSS dengan metode inline ini bisa dilakukan langsung pada tag yang ingin diberi style dengan menggunakan atribut style, seperti contoh berikut:

```
<h1 style="color : red"> Judul Web </h1>
```

- **Internal**

Penulisan kode CSS dengan metode internal yaitu ditulis didalam tag style yang ditempatkan pada tag head, seperti contoh berikut :

```
<html>
<head>
  <title>Judul HTML</title>
  <style>
    h1 {
      color : red;
    }
  </style>
</head>
...
```

- **External**

Metode external yaitu dengan membuat file CSS dan dipanggil didalam tag head. File CSS memiliki ekstensi misalnya *namafile.css*, seperti contoh berikut :

```
<html>
  <head>
    <title>Judul HTML</title>
    <link rel="stylesheet" href="namafile.css"/>
  </head>
  ...
```

b) Margin

Property margin digunakan untuk memberikan jarak antar elemen/tag HTML. Misalnya jarak antar *<div>* dan tag-tag HTML lainnya. Sebagai contoh, buatlah file HTML dengan kode berikut:

```
<div class='box'>
  Box ke satu
</div>
<div class='box'>
  Box ke dua
</div>
```

Lalu tambahkan tag style dalam tag head, dan ketikkan kode CSS berikut :

```
.box{
  background-color:green;
  color: yellow;
  width:100px;
  height:100px;
  margin-bottom:50px;
}
```

Berikut adalah tampilan yang akan anda dapatkan :



Jarak antar box ke satu dan box ke dua sedikit berjauhan, karena kita menambahkan margin bawah (`margin-bottom`) sebesar 50px. Untuk melihat perbedaan ketika kita memberikan margin atau tidak, anda dapat menghapus property `margin-bottom` dan perhatikan perbedaannya. Anda juga dapat memberikan jarak dari kanan (`right`), kiri (`left`) dan atas (`top`). Misalnya :

```
margin-top:20px;
margin-right:30px;
margin-bottom:50px;
margin-left:40px;
```

c) **Padding**

Property *padding* digunakan untuk memberikan jarak antara konten elemen. Property *padding* digunakan untuk memberikan ruang di dalam border di sekeliling atas, kanan, bawah dan kiri isi/elemen. Seperti halnya margin, property *padding* dapat ditulis langsung dengan cara disingkat (Shorthand) untuk keempat sisi tersebut, atau bisa satu per satu (*Individual Sides*) yang dibagi menjadi *padding-top*, *padding-right*, *padding-bottom*, dan *padding-left*. Penulisan nilai pada padding sama seperti dengan margin. Contoh penulisan padding :

```
div {
    padding-top: 50px;
    padding-right: 30px;
    padding-bottom: 50px;
    padding-left: 80px;
}
```

d) **Border**

Property *border* digunakan untuk memberikan garis pinggir pada elemen, setidaknya ada tiga property yang harus anda set, yaitu `color`, `style` dan `width`.

```
border-color : black;
border-style : solid;
border-width : 5px;
```

e) **Background**

Background atau latar belakang adalah sebuah elemen utama untuk menentukan bagaimana sebuah obyek ditampilkan. Ada beberapa properti dalam background seperti berikut :

- ***background-color***

Properti `background-color` ini adalah untuk menentukan warna background dari sebuah element. Contoh penggunaannya yaitu seperti dibawah ini :

```
body {  
    background-color: #b0c4de;  
}
```

Di dalam CSS, warna ditentukan oleh:

HEX value – seperti “#ff0000”

RGB value – seperti “rgb(255,0,0)”

Nama Warna – seperti “red”

Untuk detail dan nama warna yang bisa digunakan bisa dicek disini.

- **background-image**

Properti background-color ini adalah untuk menentukan background berupa gambar. Contoh penggunaan nya yaitu :

```
body {  
    background-image: url(“paper.gif”);  
    background-color: #cccccc;  
}
```

- **background-repeat**

Secara default, properti background-repeat mengulang gambar secara horizontal dan vertikal. Gambar dapat diulang dengan menggunakan repeat-x atau repeat-y.

```
body {  
    background-image: url(“paper.gif”);  
    background-repeat: repeat-y;  
}
```

- **background-attachment**

Properti ini menentukan bagaimana gambar terlihat, apakah fixed (tidak bergerak ketika discroll) atau scroll (bergerak ketika discroll). Value yang dapat digunakan yaitu scroll (mengikuti elemen), fixed (tidak mengikuti elemen), local, initial, dan inherit. Contoh penggunaannya yaitu:

```
body {  
    background-image: url(‘w3css.gif’);  
    background-repeat: no-repeat;  
    background-attachment: fixed;  
}
```

- **background-position**

Digunakan menentukan posisi background, value yang dapat digunakan antara lain : *left top, left center, left bottom, right top, right center, right bottom, center top, center center, center bottom*

Contoh penggunaannya yaitu seperti dibawah ini :

```
body {  
    background-image: url('smiley.gif');  
    background-repeat: no-repeat;  
    background-attachment: fixed;  
    background-position: center;  
}
```

f) Positioning (static, relative, Absolut, Fixed)

Dalam CSS kita pun dapat memposisikan (*Positioning*) elemen sesuai kehendak kita. Hal ini membuat desain web dengan CSS lebih fleksibel karena kita bisa menempatkan elemen-elemen HTML sesuai dengan keinginan kita atau tuntutan desain web. Beberapa position dalam CSS yaitu:

- **Static**

Secara default, seluruh elemen/tag HTML memiliki positioning static. Artinya kita belum bisa/tidak dapat menentukan posisi elemen tersebut. Ketika kita ingin merubah posisi dari elemen tersebut maka harus memberikan `position: relative` dan selanjutnya dapat menggunakan property `top`, `right`, `bottom`, dan `left` untuk menentukan posisi elemen yang bersangkutan. Sebagai contoh perhatikan kode HTML dan CSS berikut :

```
<div id='box'></div>
```

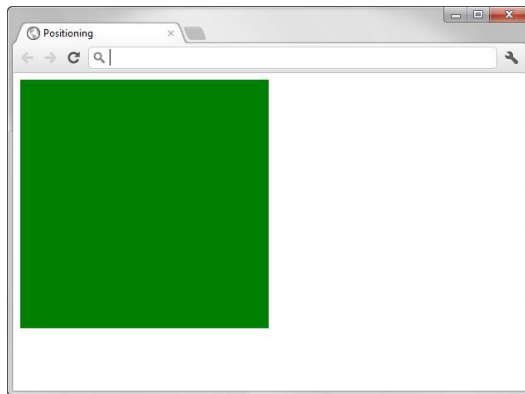
Berikut kode CSSnya.

```
#box{  
    width : 300px;  
    height : 300px;  
}
```

Ketika tidak memberikan nilai pada property position, maka secara default, `#box` memiliki `position: static`, jadi anda tidak dapat menggunakan property `top`, `right`, `bottom` dan `left` untuk menentukan posisi `#box` tersebut :

```
#box{  
    background : green;  
    width : 300px;  
    height : 300px;  
    top : 100px;  
    left : 200px;  
}
```

Maka hasilnya sebagai berikut :

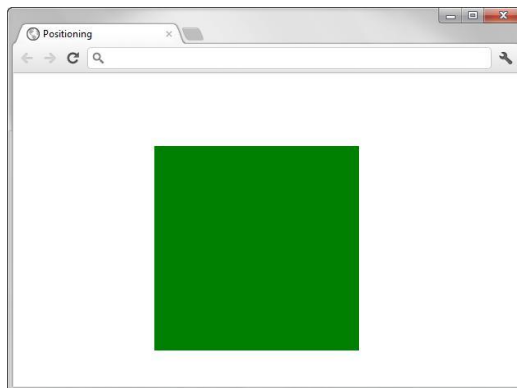


- **Relative**

Sekarang cobalah untuk menambahkan `position:relative` pada `#box` dan perhatikan bahwa box tersebut bergeser 100px dari atas dan 200px dari kiri.

```
#box{  
    background : green;  
    width : 300px;  
    height : 300px;  
    top : 100px;  
    left : 200px;  
    position:relative;  
}
```

Maka hasilnya sebagai berikut :



- **Absolut**

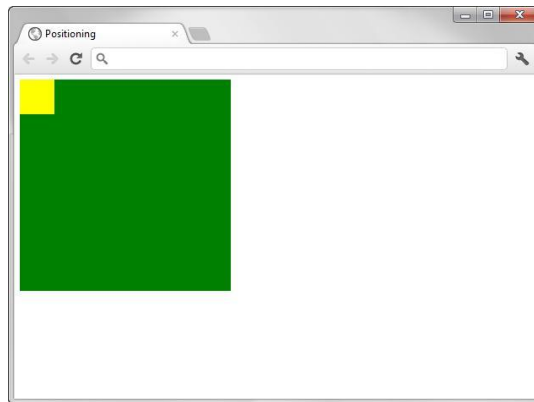
Penambahan `position:absolute` akan membuat elemen tersebut terlepas dari aturan penempatan elemen dan dibatasi oleh parent yang memiliki positioning relative. Buatlah markup HTML seperti berikut :

```
<div id='box'>  
    <div id='box2'></div>  
</div>
```

Lalu berilah style seperti berikut :

```
#box{
    width:300px;
    height:300px;
    background:green;
}
#box2{
    width:50px;
    height:50px;
    background:yellow;
}
```

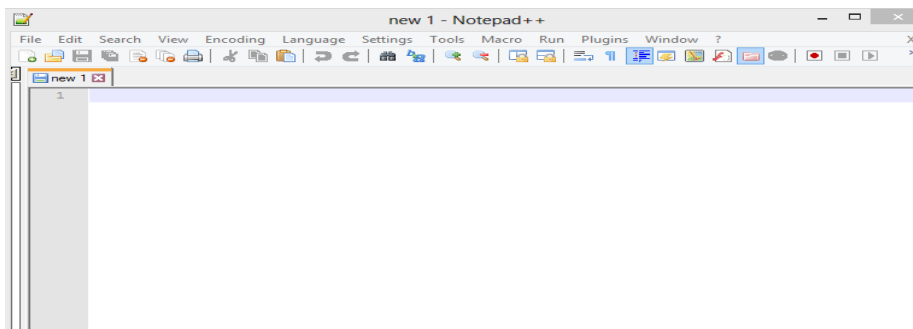
Maka hasilnya sebagai berikut :



C. KEGIATAN PRAKTIKUM

Langkah-langkah praktikum

- Buka direktori **C > xampp > htdocs**, kemudian buat folder dan beri nama "**belajarCSS**"
- Buka editor **Notepad++** atau **Sublime Text**
- Buatlah file baru dengan membuka menu **file > new** atau dengan *shortcut* **ctrl + N**



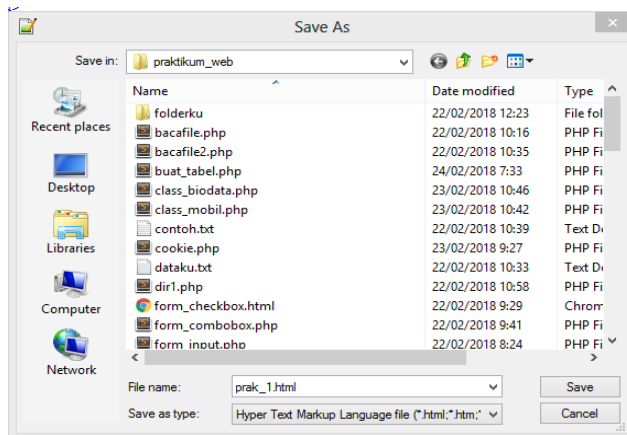
- Tuliskan *coding* berikut

```

<html>
<head>
  <title></title>
  <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>
  <div id="box">
    <div id="header" align="center">
      <h1>Universitas Tunas Pembangunan Surakarta</h1>
      Jl. Balekambang Lor No.1, Manahan, Surakarta
    </div>
    <div id="menubar" align="center">
      <a href="" class="teksmenu">Home</a> |
      <a href="">Beranda</a> |
      <a href="">Produk</a>
    </div>
    <div id="konten">
<p> Universitas Tunas Pembangunan atau biasa kita kenal dengan nama UTP
adalah salah satu universitas swasta terbaik di Indonesia yang berlokasi di
Kota Surakarta, Jawa Tengah. Universitas ini dikelola oleh Yayasan
Perguruan Tinggi Tunas Pembangunan Surakarta. Dengan lokasi kampus
yang strategis & aksesibilitas yang mudah membuat perkuliahan dapat
dijangkau dengan cepat dan efisien. UTP Surakarta sendiri memiliki 11 Prodi
unggulan dan 2 Prodi Vokasi. Pada tahun 2020 UTP Surakarta membuka
Program Vokasi baru D3 Teknologi Pemeliharaan Pesawat & Sarjana
Terapan Sistem Informasi Kota Cerdas yang pertama di Indonesia </p>
</div>
    <div id="footer" align="center">
      Web Programming I <br> 2023
    </div>
  </div>
</body>
</html>

```

- Simpan *coding* yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > belajarCSS**, dan beri nama file dengan "**index.html**" atau pilih "Hypertext Markup Language File" pada Save as type, klik *Save*.



- Buatlah file baru dengan membuka menu **file > new** atau dengan *shortcut* **ctrl + N**, kemudian salin koding dibawah ini :

html, body{

```
margin: 0;
padding: 0;
}
```

p{

```
text-align: justify;
}
```

#box{

```
width: 1000px;
height: auto;
margin: auto;
}
```

#header{

```
background-color: #dfd5d5;
padding: 20px;
}
```

#menubar{

```
background-color: #e4e4e4;
margin: auto;
}
```

.teksmenu{

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

#konten{

```
padding: 20px;
```

}

#footer{

```

background-color: #999494;
padding: 20px;
}

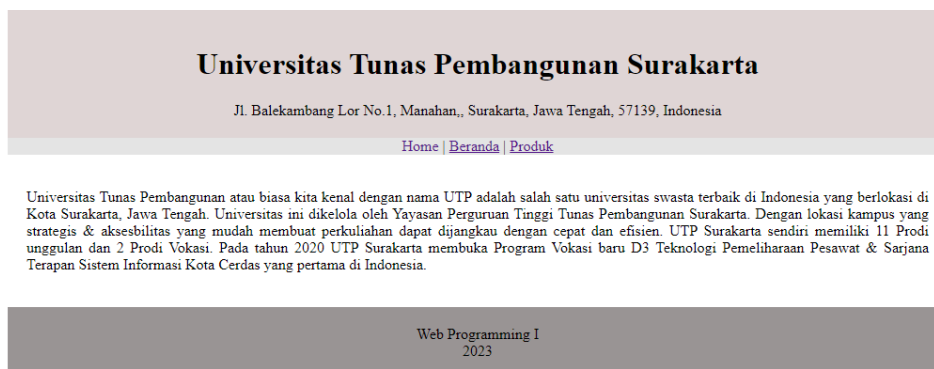
```

- Simpan *coding* yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > belajarCSS**, dan beri nama file dengan **“style.css”** atau pilih **“Cascade Style Sheets File”** pada Save as type, klik *Save*.
- Buka *Xampp* kemudian aktifkan *Apache* dan *MySQL*, dengan menekan tombol *Start* pada modul *Apache* dan *MySQL*.



- Lakukan kompilasi program melalui browser dengan mengetikkan http://localhost/namafolder/namafile_, yaitu http://localhost/praktikum_web/index.html

Maka pada browser akan muncul hasil sebagai berikut:



D. TUGAS PRAKTIKUM

Gunakan CSS untuk memperindah tampilan tugas pada modul 1

MODUL III STRUKTUR DASAR PHP

(Pertemuan 3)

A. TUJUAN :

1. Mahasiswa dapat memahami struktur dasar php dan dapat mengaplikasikannya dalam pembuatan program sederhana.
2. Mahasiswa dapat mengetahui blok program php dan memulai pembuatan program php.
3. Mahasiswa dapat memahami dan mengaplikasikan pembuatan variabel di PHP.
4. Mahasiswa dapat memahami penggunaan tipe data yang ada di PHP dan mampu mengaplikasikannya.
5. Mahasiswa dapat memahami dan mengaplikasikan pembuatan konstanta

B. DASAR TEORI

Bahasa pemrograman PHP adalah bahasa pemrograman yang bekerja dalam sebuah web server. Script-script yang telah dibuat harus tersimpan dalam sebuah server dan dieksekusi atau diproses dalam server tersebut. Penggunaan program PHP memungkinkan sebuah website menjadi lebih interaktif dan dinamis.

Adapun struktur dasar dalam PHP akan dijelaskan sebagai berikut :

1. Blok Program PHP

Pada Pemrograman PHP, penulisan sintak dalam PHP harus diapit oleh simbol sebagai berikut:

- `<? dan ?>`
- `<?PHP dan ?>`

Simbol-simbol di atas berfungsi sebagai tanda bahwa sintak tersebut ialah termasuk bahasa pemrograman PHP.

Contoh :

```
<? echo "Selamat Belajar PHP"; ?>
```

atau,

```
<?PHP echo "Selamat Belajar PHP"; ?>
```

2. Variabel

Variable adalah sebuah bentuk pendeklarasian suatu nama yang memiliki *value*(nilai), data atau informasi dengan tipe data tertentu. Setiap pendeklarasian variabel pada pemrograman PHP diawali dengan tanda \$ (dolar). Bentuk penulisan pendeklarasian variabel ialah sebagai berikut:

```
$(nama variabel) = (nilai variabel);
```

Misal

```
$nama = "Joni";  
$nilai = 90;
```

3. Tipe Data

Ada beberapa tipe data yang dikenal dalam script PHP, diantaranya antara lain :

- **String**

String adalah sebuah tipe data yang terdiri dari kata, bisa berupa kata tunggal ataupun kalimat. Penulisan string harus diapit oleh tanda petik tunggal („ ‘) atau tanda petik ganda (“ ”).

- **Integer**

Integer adalah tipe data yang berisi bilangan bulat, bukan bilangan desimal. Besar range integer dalam pemrograman PHP ialah dari **-2,147,483,648** sampai **+2,147,483,647** pada platform 32 bit. Apabila data ada di luar kisaran tersebut, maka PHP akan secara otomatis mengkonversi data tersebut dari integer ke **floating point**.

- **Floating Point**

Floating Point ialah tipe data yang berisi bilangan pecahan atau bilangan desimal. Kisaran data floating adalah antara **1.7E-308** sampai **1.7E+308**.

- **Array**

Array ialah tipe data yang mengandung beberapa data di dalamnya dan di-index atau dibaca berdasarkan data numeric atau string.

- **Object**

Tipe data object bisa berupa bilangan, variabel, ataupun fungsi. Tipe data tersebut dapat membantu programmer untuk membuat sebuah program. Data itu dapat disertakan dalam program sehingga meringkas beberapa fungsi dan dapat memperkecil ukuran file. Semakin kecil ukuran file, semakin singkat waktu yang dibutuhkan untuk mengakses file tersebut.

Dalam script PHP, tipe data suatu variabel tidak didefinisikan oleh programmer, akan tetapi akan secara otomatis dikonversikan sesuai karakternya.

4. Konstanta

Konstanta adalah sebuah variabel yang memiliki nilai yang tidak berubah-ubah. Dalam mendeklarasikan konstanta tidak memerlukan tanda \$ (dolar) akan tetapi menggunakan **define()** dengan penulisan:

define({nama konstanta},{nilai konstanta})

Contoh :

```
define("Nama", "Joni");
```

```
define("Nilai", 90);
```

maka dari pendefinisian di atas konstanta **Nama** memiliki *value* (nilai) **Indra** dan **Nilai** memiliki *value* (nilai) **90**.

5. Operator

Operator adalah simbol atau tanda yang jika diletakkan pada dua buah operand dapat menghasilkan sebuah hasil. Penggunaan operator dalam bahasa pemrograman berfungsi untuk memanipulasi nilai dari sebuah variabel agar lebih interaktif dan dinamis. Adapun operator yang dikenal dalam PHP ialah sebagai berikut:

Operasi	Operator	Contoh	Keterangan
Matematika	*	\$a * \$b	\$a dikalikan dengan \$b
	/	\$a / \$b	\$a dibagi dengan \$b
	+	\$a + \$b	\$a ditambah dengan \$b
	-	\$a - \$b	\$a dikurangi dengan \$b
	%	\$a % \$b	Sisa hasil \$a dibagi \$b
Increment dan Decrement	++	++\$a	\$a ditambah satu, lalu operasi dijalankan
	++	\$a ++	Operasi dijalankan, lalu \$a ditambah satu
	--	--\$a	\$a dikurangi satu, lalu operasi dijalankan
	--	\$a--	\$a dikurangi satu, lalu operasi dijalankan
String	.	\$a. \$b	Penggabungan String \$a dan \$b
Perbandingan	>	\$a > \$b	True jika \$a lebih besar dari \$b
	<	\$a < \$b	True jika \$a lebih kecil dari \$b
	>=	\$a >= \$b	True jika \$a lebih besar atau sama dengan \$b
	<=	\$a <= \$b	True jika \$a lebih kecil atau sama dengan \$b
	==	\$a == \$b	True jika \$a sama dengan \$b
	!=	\$a != \$b	True jika \$a tidak sama dengan \$b
Logika	And atau && Atau	(\$a And \$b) Atau	Menghasilkan nilai True jika \$a dan \$b bernilai Benar

		(\$a && \$b)	
	Or atau 	(\$a or \$b) Atau (\$a \$b)	Menghasilkan nilai True jika \$a atau \$b bernilai Benar atau keduanya bernilai benar
	xor	(\$a xor \$b)	Menghasilkan nilai True jika salah satu dari \$a
	!	!\$a	Menghasilkan nilai True jika \$a bernilai salah

6. Komentar

Komentator adalah script PHP yang tidak akan dibaca ketika dijalankan. Komentator berfungsi sebagai dokumentasi dari script PHP atau penjelasan terhadap script yang kita tulis.

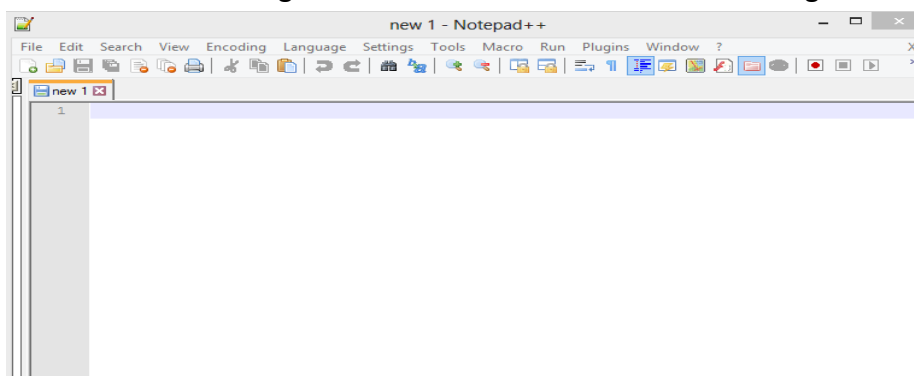
Dalam menulis komentar dapat digunakan beberapa simbol antara lain :

```
/* dan */
//
#
/* {komentar} */
// {komentar}
# {komentar}
```

C. KEGIATAN PRAKTIKUM

Langkah-langkah praktikum

1. Buka direktori **C > xampp > htdocs**, kemudian buat folder dan beri nama **"praktikum_web"**
2. Buka editor **Notepad++** atau **Sublime Text**
3. Buatlah file baru dengan membuka menu **file > new** atau dengan *shortcut* **ctrl + N**



4. Tuliskan *coding* berikut

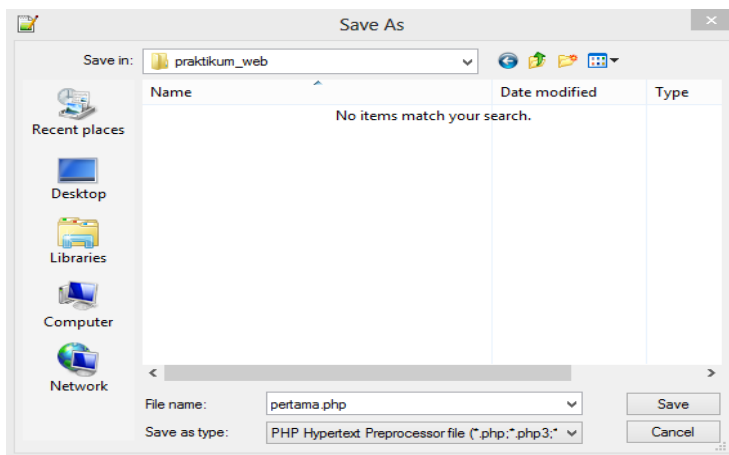
```
<!doctype html>
<html lang="en">
```

```

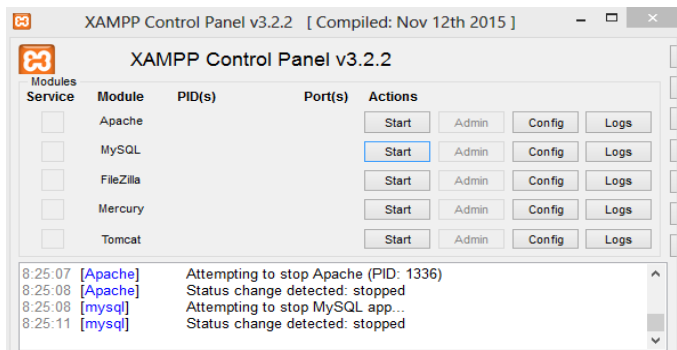
<head>
  <meta charset="UTF-8">
  <title>Belajar PHP</title>
</head>
<body>
  <?php
    // tanda untuk memberikan komentar satu baris
    echo "<h1>Selamat Datang.</h1>";
    echo "Program PHPku yang pertama<br>";
    /*
      tanda untuk memberikan komentar
      komentar ini lebih dari satu baris
    */
  ?>
</body>
</html>

```

5. Simpan *coding* yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > praktikum_web**, dan beri nama file dengan **"pertama.php"** atau pilih "PHP Hypertext Preprocessor File" pada Save as type, klik *Save*.



6. Buka *Xampp* kemudian aktifkan *Apache* dan *MySQL*, dengan menekan tombol *Start* pada modul *Apache* dan *MySQL*.

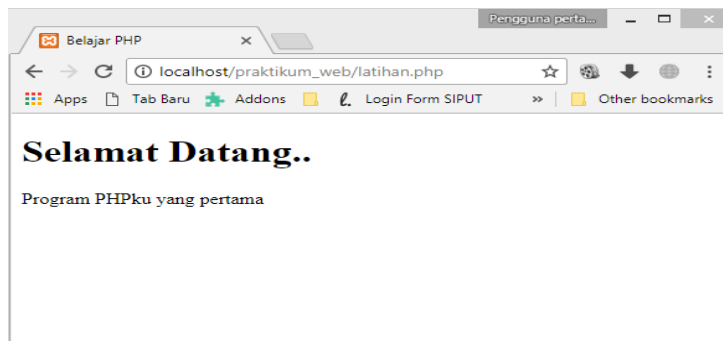


7. Lakukan kompilasi program melalui browser dengan mengetikkan

http://localhost/namafolder/namafile_, yaitu

http://localhost/praktikum_web/latihan.php

Maka pada browser akan muncul hasil sebagai berikut:



8. Selesai

D. LATIHAN PRAKTIKUM

1. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini

:

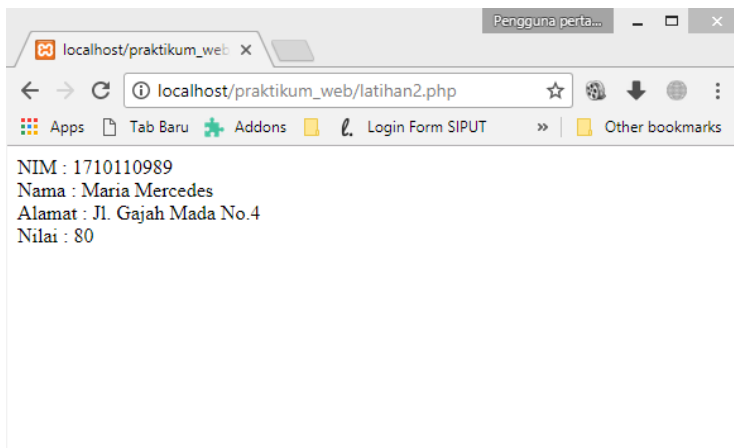
```
<?php
    $nim = '1710110989';
    $nama = 'Maria Mercedes;
    $alamat = 'Jl. Gajah Mada No.4';
    $nilai = 80;

    echo "NIM : ", $nim, "<br>";
    echo "Nama : ", $nama, "<br>";
    echo "Alamat : ", $alamat, "<br>";
    echo "Nilai : $nilai";
?>
```

Simpan program tersebut dengan memilih menu *File* lalu *Save As* pada notepad++, simpan file dengan nama **biodata.php** tersebut di dalam folder

htdocs/praktikum_web pada file sistem XAMPP dan jalankan aplikasi XAMPP apabila belum aktif, lalu klik tombol start pada modul Apache dan MySQL. Setelah itu buka browser anda dan jalankan program yang telah di buat tersebut dengan cara menetik nama file program contoh <http://localhost/praktikumweb/latihan2.php> dan jalankan.

Setelah selesai menetik coding di atas, maka simpan dengan nama **latihan2.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



2. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

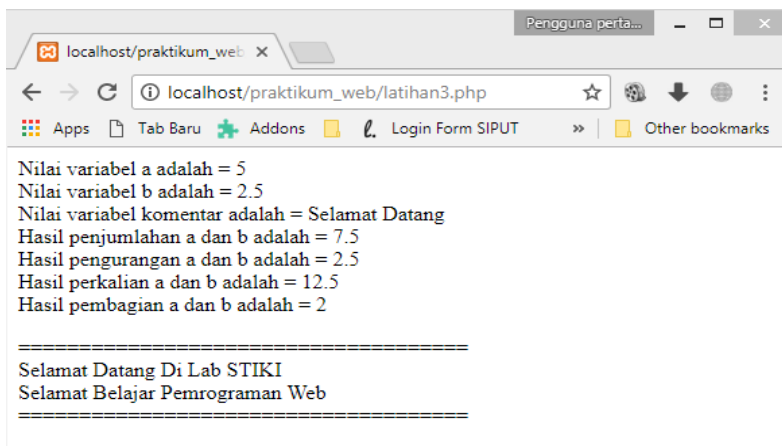
```
<?php
$a="5";
$b="2.5";
$komentar="Selamat Datang";
echo ("Nilai variabel a adalah = $a <br>");
//variabel bertipe integer
echo ("Nilai variabel b adalah = $b <br>");
//variabel bertipe real
echo ("Nilai variabel komentar adalah = $komentar<br>");
//variabel bertipe string
$tambah= $a + $b;
//rumus pengurangan
$kurang = $a - $b;
//rumus perkalian
$kali = $a * $b;
//rumus pembagian
$bagi = $a / $b;
```

```

echo ("Hasil penjumlahan a dan b adalah = $tambah <br>");
echo ("Hasil pengurangan a dan b adalah = $kurang<br>");
echo ("Hasil perkalian a dan b adalah = $kali <br>");
echo ("Hasil pembagian a dan b adalah = $bagi <br>");
$nama = "STIKI";
$garis= "===== ";
echo "<p>";
echo $garis."<br>";
echo $komentar. " Di Lab ". $nama. "<br>Selamat Belajar Pemrograman Web <br>";
echo $garis."<br>";
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **latihan3.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



E. TUGAS

1. Jika ada seorang nasabah bank yang menabung di bank X dengan saldo awal Rp.2000.000,-. Bank X memberikan kebijakan bunga 3% perbulan dari saldo awal tabungan. Hitunglah jumlah saldo akhir nasabah tersebut setelah 11 bulan. Lengkapi script berikut ini untuk mengerjakan :

```

<?php
    $saldoAwal = 1000000;
    $bunga = 0.03;
    $bulan = 11;
    $saldoAkhir = .....;// lengkapi pada perhitungan $saldo akhir
    echo "Saldo akhir setelah ".$bulan." bulan adalah : Rp. ".$saldoAkhir. ",-";
?>

```

2. Buatlah program (dalam satu file) untuk menghitung bangun ruang kerucut dengan rumus seperti berikut:

- Luas alas = πr^2
- Luas permukaan = $\pi r^2 + \pi r s$

MODUL IV PERNYATAAN KONDISI

A. TUJUAN

1. Mahasiswa dapat memahami dan membuat pernyataan kondisi dalam bahasa pemrograman php dan dapat mengaplikasikannya dalam pembuatan program php.
2. Mahasiswa dapat memahami dan mengaplikasikan pernyataan kondisi If dalam pemrograman PHP.
3. Mahasiswa dapat memahami dan mengaplikasikan pernyataan kondisi If-Else dalam pemrograman PHP.
4. Mahasiswa dapat memahami dan mengaplikasikan pernyataan kondisi Switch-Case dalam pemrograman PHP.

B. DASAR TEORI

Dalam pemrograman PHP terdapat beberapa pernyataan kondisi yang berfungsi sebagai percabangan apabila terdapat beberapa kondisi yang berbeda. Adapun pernyataan kondisi tersebut antara lain:

1. Pernyataan kondisi if

Fungsi pernyataan kondisi if ialah sebagai pernyataan dalam memilih salah satu operasi percabangan yang akan dilaksanakan sesuai syarat atau pernyataan tertentu yang telah ditetapkan.

Adapun bentuk penulisan kondisi if dalam pemrograman PHP ialah sebagai berikut:

```
If (syarat /pernyataan){  
    Operasi;  
}
```

Pada formula diatas menjelaskan bahwa apabila kondisi sesuai dengan syarat / pernyataan maka operasi di dalam formula akan dieksekusi dan apabila kondisi tidak sesuai maka formula akan dilewati tanpa mengeksekusi operasi yang terdapat di dalamnya.

2. Pernyataan kondisi if-Else

Fungsi dari pernyataan kondisi if-Else hampir sama dengan kondisi if yang telah dijelaskan di atas, akan tetapi pada kondisi if-Else digunakan untuk operasi percabangan yang terdiri dari dua atau lebih pernyataan yang berbeda. Adapun bentuk penulisan kondisi if-Else dalam pemrograman PHP, ialah sebagai berikut:

```
If (syarat /pernyataan 1){  
    Operasi1;
```

```

}else if (syarat / pernyataan 2){
    Operasi2;
} else if (.....){
    .....
}else {
    Operasi x;
}

```

Pada contoh di atas menjelaskan bahwa apabila kondisi sesuai dengan syarat / pernyataan 1 maka jalankan Operasi1, jika tidak maka akan dilanjutkan kesyarat/pernyataan 2 apabila kondisi sesuai maka Operasi 2 akan dieksekusi, dan apabila tidak sesuai maka akan dilanjutkan sampai kondisi yang ada sesuai dengan syarat/pernyataan sehingga apabila tidak terdapat pernyataan yang sesuai maka program akan mengeksekusi Operasi x.

3. Pernyataan kondisi Switch-Case

Fungsi dari pernyataan kondisi Switch-Case ialah membandingkan variabel yang memiliki beberapa nilai yang berbeda. Fungsi ini hampir sama dengan pernyataan kondisi if hanya saja kondisinya ditulis berulang-ulang. Adapun bentuk penulisan kondisi Switch-Case dalam pemrograman PHP, ialah sebagai berikut:

```

Switch ($variabel_kondisi)
{
    Case "nilai 1";
    Operasi 1;
    Break;
    Case "nilai 2";
    Operasi 2;
    Break;
    .....
}

```

Pada formula di atas menjelaskan bahwa apabila value/nilai dari variabel \$variabel_kondisi sesuai dengan nilai 1 maka operasi 1 dijalankan, dan jika tidak maka apabila value/nilai dari variabel \$variabel_kondisi sesuai dengan nilai 2 maka operasi 2 dijalankan dan seterusnya.

C. KEGIATAN PRAKTIKUM

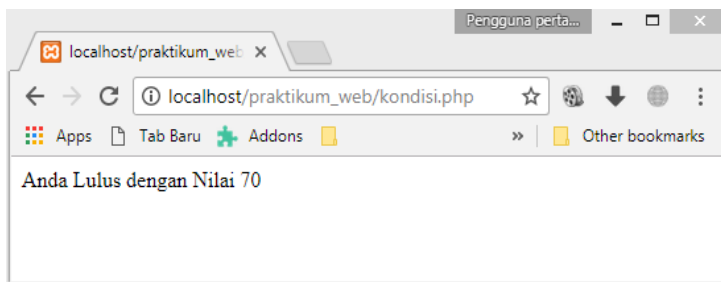
1. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:
<?php

```

$nilai = 70;
if ($nilai >= 60) {
    echo "Anda Lulus dengan Nilai $nilai";
}
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **kondisi.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



Pada program di atas apabila nilai lebih besar atau sama dengan 60 maka akan muncul string “anda lulus dengan nilai (70)” apabila nilai tidak lebih besar atau sama dengan 60 maka di browser tidak akan muncul apa-apa.

2. Buka notepad++, buat file baru **New** atau *Ctrl+N* , kemudian Salin coding di bawah ini:

```

<?php
    $username = "Roman Picisan";
    if ($username = "Roman Picisan")
        echo "Anda Berhasil Login, Roman";
    else
        echo "Username Anda Salah";
?>

```

Setelah selesai mengetik koding diatas, maka simpan dengan nama **kondisi_1.php**.

3. Buka notepad++, buat file baru **New** atau *Ctrl+N* , kemudian Salin coding di bawah ini:

```

<?php
    $nilai = 80;
    if($nilai >= 90){
        echo "Nilai Memuaskan";
    }elseif ($nilai >= 80){

```

```
        echo "Nilai Bagus";
    }elseif($nilai >=50){
        echo "Nilai Kurang";
    }else{
        echo "Nilai Sangat Kurang";
    }
?>
```

Setelah selesai mengetik koding diatas, maka simpan dengan nama **kondisi_2.php**.

4. Buka notepad++, buat file baru **New** atau *Ctrl+N* , kemudian Salin coding di bawah ini:

```
<?php
    $bln = date("M");
    switch($bln)
    {
    case "Jan" : $namaBln = "Januari";
    break;
    case "Feb" : $namaBln = "Februari";
    break;
    case "Mar" : $namaBln = "Maret";
    break;
    case "Apr" : $namaBln = "April";
    break;
    case "May" : $namaBln = "Mei";
    break;
    case "Jun" : $namaBln = "Juni";
    break;
    case "Jul" : $namaBln = "Juli";
    break;
    case "Aug" : $namaBln = "Agustus";
    break;
    case "Sep" : $namaBln = "September";
    break;
    case "Oct" : $namaBln = "Oktober";
    break;
    case "Nov" : $namaBln = "Nopember";
    break;
    case "Dec" : $namaBln = "Desember";
```

```

        break;
    }
    echo "Nama bulan sekarang adalah : ".$namaBln;
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **kondisi_3.php** dan jalankan di browser.

D. TUGAS PRAKTIKUM

1. Terdapat suatu kondisi dimana pada tempat fotocopy kampus, harga fotocopy per lembar disesuaikan dengan berapa jumlah fotocopy, dengan ketentuan sebagai berikut.
 - Jika fotocopy kurang dari 100 lembar maka harganya Rp. 150,-/lembar
 - Jika fotocopy sebanyak 100-200 lembar dapat harga Rp. 100,-/lembar
 - Tapi jika fotocopy lebih dari 200 maka harganya Rp. 80,-/lembar. Berapa biaya harus dibayar jika asumsi pelanggan akan fotocopy sejumlah 158 lembar? . Buat script dengan menggunakan kondisi If-Else untuk menampilkan biaya fotocopy.
2. Buatlah keterangan dari grade nilai dengan menggunakan If Else dan Switch-Case!

| Hari | Mata Kuliah | Keterangan |
|----------|-------------|---------------|
| 90 – 100 | A | Baik Sekali |
| 76 -89 | B | Baik |
| 60 - 75 | C | Cukup |
| 50 - 59 | D | Kurang |
| 0 – 49 | E | Sangat Kurang |

3. Dengan menggunakan Switch-Case, buatlah script untuk membaca bulan saat ini dan tampilkan jumlah hari dalam bulan tersebut.

MODUL V PERNYATAAN PERULANGAN

A. TUJUAN

1. Mahasiswa dapat memahami dan membuat pernyataan perulangan dalam bahasa pemrograman php dan dapat mengaplikasikannya dalam pembuatan program php.
2. Mahasiswa dapat memahami dan mengaplikasikan pernyataan perulangan For dalam pemrograman PHP.
3. Mahasiswa dapat memahami dan mengaplikasikan pernyataan perulangan While dalam pemrograman PHP.

B. DASAR TEORI

Dalam pemrograman PHP terdapat tiga kondisi perulangan (looping) dimana perulangan tersebut berfungsi sebagai sintak untuk melakukan operasi / proses yang berulang. Adapun kondisi perulangan tersebut antara lain:

1. Perulangan While

Fungsi dari perulangan while ialah untuk menjalankan operasi secara berulang-ulang hingga ditemukan kondisi untuk menghentikannya. Adapun penulisan kondisi perulangan while pada pemrograman PHP ialah sebagai berikut:

```
while (Kondisi){  
    Operasi;  
}
```

Pada formula di atas dapat dijelaskan bahwa apabila kondisi bernilai false maka eksekusi operasi sehingga kondisi yang telah ditetapkan menjadi true. Pada formula while program akan mengecek kondisi terlebih dahulu apakah false atau true, apabila false maka operasi di jalankan dan apabila true operasi berhenti dijalankan.

2. Perulangan For

Fungsi dari perulangan for ialah untuk menjalankan operasi secara berulang-ulang sesuai dengan jumlah perulangan yang telah ditentukan. Adapun penulisan kondisi perulangan for pada pemrograman php adalah:

```
for (nilai_awal; nilai_akhir; penambahan atau pengurangan nilai){  
    Operasi;  
}
```

Pada formula di atas dapat dijelaskan pada perulangan for program akan menset nilai sesuai dengan nilai_awal lalu mengecek nilai tersebut apakah telah mencapai nilai_akhir, apabila tidak maka operasi akan dijalankan setelah itu akan terjadi

pengubahan pada nilai apakah itu ditambahkan atau dikurangi kemudian dilakukan kembali proses dari pengecekan dan seterusnya hingga nilai mencapai ke nilai_akhir, apabila nilai mencapai nilai_akhir maka proses perulangan akan dihentikan.

3. Perulangan Do-While

Fungsi dari perulangan do-while hampir sama dengan perulangan while, perbedaan antara keduanya terletak pada urutan prosesnya saja. Pada perulangan do-while proses dimulai dari menjalankan operasi lalu pengecekan kondisi, sedangkan pada perulangan while proses dimulai dari melakukan pengecekan kondisi lalu menjalankan operasi. Adapun penulisan kondisi perulangan do-while pada pemrograman PHP ialah sebagai berikut:

```
do {  
    Operasi;  
} while(Kondisi);
```

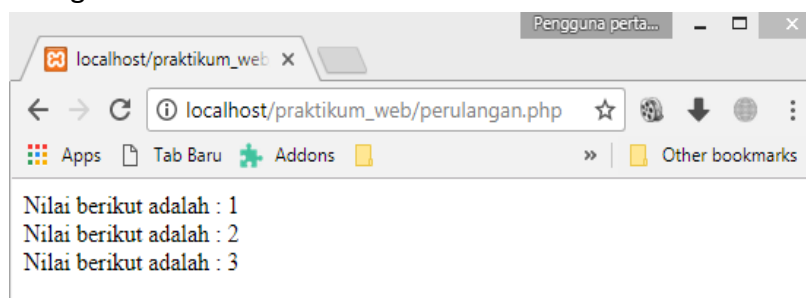
Dari formula diatas dapat dijelaskan program akan menjalankan operasi lalu mengecek apakah kondisi true atau false, apabila false maka proses akan diulang kembali sehingga kondisi menjadi true dan proses do-while berhenti.

C. KEGIATAN PRAKTIKUM

1. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```
<?php  
    $bila = 1;  
    while($bil <= 3){  
        echo "Nilai Sekarang adalah $bil <br>";  
        $bil++;  
    }  
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **perulangan.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



2. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```
<?php
    $bil = 1;
    while ($bil <= 3)
    {
        echo $bil. "<br />";
        $bil++;
    }
    echo "Nilai bilangan sekarang adalah:". $bil;
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **perulangan_2.php** dan jalankan di browser.

3. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```
<?php
    $bil = 5;
    while ($bil <= 100)
    {
        if ($bil % 10 == 0) echo $bil. "<br />";
        $bil++;
    }
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **perulangan_3.php** dan jalankan di browser.

4. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```
<?php
    $bil = 5;
    while ($bil <= 100)
    {
        if ($bil % 10 == 0) echo $bil. "<br />";
        $bil++;
    }
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **perulangan_3.php** dan jalankan di browser.

5. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```

<?php
    $bilangan = 0;
    for ($a=1; $a<=5; $a++){

        echo "Bilangan Sekarang adalah $bilangan <br>"; $bilangan = $bilangan +
        $a;
    }
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **perulangan_5.php** dan jalankan di browser.

6. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```

<?php
    for ($a = 1; $a <=3; $a++)
    {
        for ($b = 1; $b <= 2; $b++)
        {
            echo "Nilai a = ".$a. " Nilai b = ".$b. "<br />";
        }
    }
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **perulangan_5.php** dan jalankan di browser.

7. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```

<?php
    $d = 1;
    do{
        echo "Perulangan ke-$d <br>";
        $d++;
    }while($d <= 5);
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **perulangan_6.php** dan jalankan di browser.

D. TUGAS

1. Buatlah program menggunakan pernyataan perulangan untuk menampilkan bilangan bulat 5 s/d 100 yang berkelipatan 10.
2. Buatlah program menggunakan pernyataan perulangan untuk menjumlahkan bilangan 2 s/d 50.
3. Buatlah program menggunakan pernyataan perulangan untuk mencari banyaknya bilangan bulat mulai dari 3 s/d 127 yang merupakan kelipatan 6.

MODUL VI ARRAY

A. TUJUAN

1. Mahasiswa dapat memahami dan mengaplikasikan array dalam PHP dalam pemrograman web.
2. Mahasiswa dapat mengetahui cara pendeklarasian array menggunakan bahasa pemrograman PHP dan mengaplikasikannya dalam pembuatan web.
3. Mahasiswa dapat memahami dan melakukan pengolahan data array menggunakan bahasa pemrograman PHP.
4. Mahasiswa dapat mengetahui dan menggunakan fungsi-fungsi array yang ada dalam bahasa pemrograman PHP.

B. DASAR TEORI

Array merupakan suatu tipe data yang terstruktur dan dapat digunakan untuk menyimpan data yang memiliki tipe data yang sama. Array berfungsi untuk mengidentifikasi variabel yang memiliki jumlah data yang lebih dari satu, tapi memiliki nama variabel yang sama. Bagian yang menyusun array disebut elemen array, yang masing-masing elemen dapat diakses melalui index array. Index array dapat berupa bilangan integer atau string.

1. Deklarasi Array

Cara pendeklarasian sebuah array ialah sebagai berikut :

```
$nama_variabel = array();  
key1 => value,  
key2 => value,  
key3 => value,
```

Penjelasan :

komponen array terdiri dari pasangan kunci (key) dan nilai (value). *Key* adalah penunjuk posisi dimana *value* disimpan. Tanda panah (=>) untuk memberikan nilai kepada *key*.

Pengaksesan nilai dari array dapat dilakukan dengan menggunakan kombinasi *\$nama_variabel* dan nilai-*key*, dengan penulisan sebagai berikut :

```
$nama_variabel[key];
```

2. Jenis-jenis Array

- **Array berindeks (*indexed array*)**

Array berindeks adalah *array* yang diindeks menggunakan angka/nomor maupun karakter yang sifatnya berurutan. Ada dua cara membuat array berindeks yaitu indeks yang diberikan secara otomatis (selalu dimulai dari indeks ke-0), contohnya:

```
$mobil = array("Volvo", "BMW", "Toyota");
```

Atau bisa dengan indeks yang diberikan secara manual, contohnya :

```
$mobil[0] = "Volvo";  
$mobil[1] = "BMW";  
$mobil[2] = "Toyota";
```

- **Array asosiatif (*Assosiative Array*)**

Array asosiatif adalah array yang diindeks berdasarkan nama tertentu. Ada dua cara membuat array asosiatif, contohnya :

```
$umur = array("MangJorge"=>"29","Jennifer"=>"28","Romeo"=>"33");  
Atau  
$umur['MangJorge'] = "29";  
$umur['Jennifer'] = "28";  
$umur['Romeo'] = "33";
```

3. Fungsi-fungsi Array

Dalam pemrograman PHP terdapat banyak fungsi-fungsi yang dapat digunakan untuk memanipulasi data array. Adapun di antaranya ialah:

- **Array_push()**

Berfungsi untuk menambah isi dari sebuah data array, yang memungkinkan isi data array bisa disusun dengan susunan yang baru.

- **Array_merge()**

Berfungsi untuk menggabungkan data array dengan beberapa array sekaligus.

- **Array_shift()**

Berfungsi untuk menghapus data pertama dari sebuah array.

- **Sort()**

Berfungsi untuk mengurutkan isi data array

- **Count()**

Berfungsi untuk menghitung jumlah element/data dari sebuah array. Untuk fungsi-fungsi array, lebih lengkapnya dapat anda lihat pada website PHP di link

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

4. Perulangan dalam Array

Perulangan dapat digunakan untuk mengakses elemen array. Perulangan dalam array ditujukan apabila isi data array banyak maka akan lebih efektif untuk menggunakan perulangan dalam mengakses elemen array sehingga data yang ada dalam array akan ditampilkan seluruhnya. Perulangan yang paling sering digunakan adalah perulangan *Foreach*. Contoh sintaks perulangan array :

```
foreach ($umur as $data => $nama)
```

Penjelasan :

foreach digunakan untuk mengakses semua elemen yang ada di dalam array *\$umur*. Pada setiap elemen yang sedang dibaca, kunci (key) atau nama indeks array akan

ditampung dalam variabel *\$data*, sedangkan nilainya ditampung ke dalam variabel *\$nama*.

5. Mengurutkan (*Sorting*) Array

Salah satu fungsi array yaitu adalah *sorting* yaitu mengurutkan elemen array. Dalam mengurutkan array juga terdapat beberapa fungsi-fungsi antara lain :

- **Sort ()**

Fungsi *sort ()* akan mengurutkan array berdasarkan nilai elemennya. Dalam proses pengurutan ini, yang diurutkan hanya nilainya saja. Contoh sintaks pengurutan *sort ()* yaitu :

```
sort($nama_array);
```

- **Asort ()**

Contoh sintaks pengurutan *asort ()* yaitu :

```
asort ( $nama_array);
```

- **Ksort ()**

Digunakan untuk mengurutkan elemen array berdasarkan kunci (key) dan indeksnya. Contoh sintaks pengurutan *ksort ()* yaitu :

```
ksort ($nama_array);
```

- **Rsort ()**

Digunakan untuk mengurutkan array secara *descending*. Contoh sintaks pengurutan *rsort ()* yaitu:

```
rsort ($nama_array);
```

- **Arsort ()**

Contoh sintaks pengurutan *arsort ()* yaitu :

```
arsort ($nama_array);
```

- **Krsort ()**

Contoh sintaks pengurutan *krsort ()* yaitu :

```
Sort ($nama_array);
```

C. KEGIATAN PRAKTIKUM

1. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```
<?php
$kelompok1 =
array("Andi","Budi","Chika","David","Erna"); echo "Nama Kelompok 1 :<br />";
    for ($i=0; $i<=4; $i++){
        echo " + ",$kelompok1[$i],"<br/>";
    }

echo"<br /><br />";
```



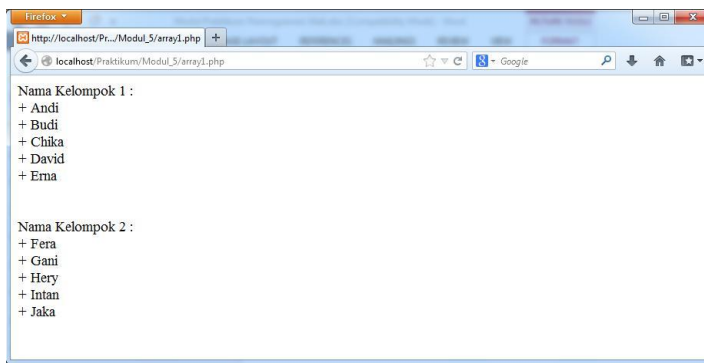
```

$kelompok2[] = "Fera";
$kelompok2[] = "Gani";
$kelompok2[] = "Hery";
$kelompok2[] = "Intan";
$kelompok2[] = "Jaka";

echo "Nama Kelompok 2 :<br />";
    for ($i=0; $i<=4; $i++){
        echo " + ",$kelompok2[$i],"<br/>";
    }
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **array1.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



2. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```

<?php
$array1 = array ("Arman","Bayu","Feri"); echo"array awal ialah:<br/>";
    for ($i=0; $i<count($array1); $i++){
        echo"data ke-" , $i, " : ", $array1[$i], "<br/>";
    }
    echo "<br/>Setelah ditambahkan \"Herni, Gita & Dewi\" menggunakan
    fungsi array_push():<br/>";
array_push($array1,"Herni","Gita","Dewi"); for ($i=0; $i<count($array1); $i++){
    echo"data ke-" , $i, " : ", $array1[$i], "<br/>";
}

    echo "<br/>Setelah bagian awal array dihapus menggunakan fungsi
    array_shift():<br/>";
array_shift($array1);

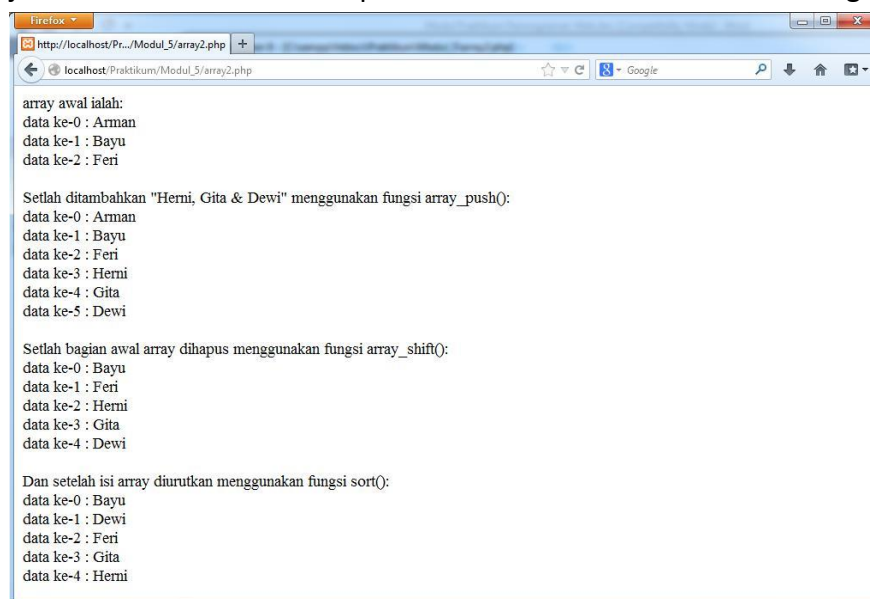
```

```

        for ($i=0; $i<count($array1); $i++){
            echo"data ke-" , $i, " : " , $array1[$i], "<br/>";
        }
echo "<br/>Dan setelah isi array diurutkan menggunakan fungsi sort():<br/>";
sort($array1);
        for ($i=0; $i<count($array1); $i++){
            echo"data ke-" , $i, " : " , $array1[$i], "<br/>";
        }
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **array2.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



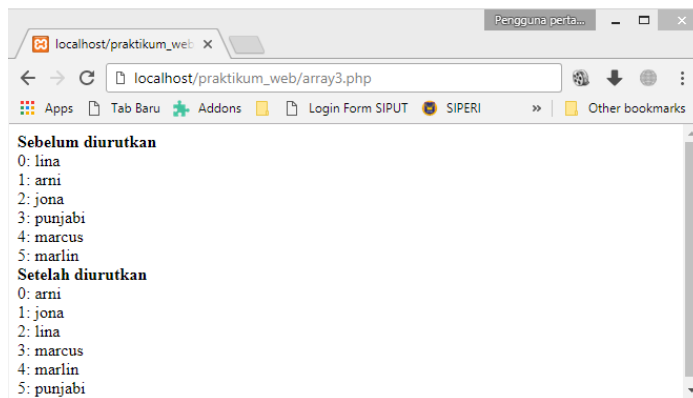
3. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian salin koding dibawah ini:

```

<?php
    $pegawai = array("lina", "arni", "jona", "punjabi", "marcus", "marlin");
    echo "<strong>Sebelum diurutkan</strong><br />";
    foreach($pegawai as $data => $nama){
        echo "$data: $nama". "<br />";
    }
sort($pegawai);
    echo "<strong>Setelah diurutkan</strong><br />";
    foreach($pegawai as $data => $nama){
        echo "$data: $nama". "<br />";
    }
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **array3.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



```
Sebelum diurutkan
0: lina
1: arni
2: jona
3: punjabi
4: marcus
5: marlin
Setelah diurutkan
0: arni
1: jona
2: lina
3: marcus
4: marlin
5: punjabi
```

D. TUGAS

1. Buatlah array asosiatif ,kemudian lakukan semua fungsi pengurutan.
2. Cari dan gunakan 5 fungsi array (selain yang di atas) dan implementasikan 5 fungsi array tersebut ke array yang telah anda buat!

MODUL VII FORM

A. TUJUAN

1. Mahasiswa dapat memahami dan mengaplikasikan form dalam pemrograman web dan dapat melakukan pengolahan data menggunakan bahasa pemrograman PHP.
2. Mahasiswa dapat memahami dan mengaplikasikan form input type text, radio, check box, combo box, text area dan melakukan pengolahan inputan data menggunakan bahasa pemrograman PHP.

B. DASAR TEORI

Dalam pemrograman web form ialah suatu tools dalam membantu interaksi aplikasi web dengan user, dengan adanya form user dapat memanipulasi jalannya sistem sesuai dengan keinginan. Dalam form, dibutuhkan pembuka dan penutup tag form untuk menampung untuk menampung input yang ada. Adapun pembuka dan penutup tag form ialah sebagai berikut:

```
<form action="action.php" method="GET/POST">  
  {Jenis input}  
</form>
```

Pada form biasanya dibutuhkan minimal dua atribut yaitu action dan method. Action berisi halaman atau file tujuan untuk memproses inputan dalam form, sedangkan method berisi tentang metode pengiriman data yang akan digunakan (GET atau POST).

Terdapat dua metode pengiriman data yang digunakan dalam form, yaitu metode GET dan POST. Metode GET berarti input user dikirim ke halaman atau file action melalui address bar, sedangkan metode POST berarti input user tidak dikirim melalui address bar.

Adapun beberapa jenis input pada form, diantaranya ialah sebagai berikut:

1. Form Input Type Text

Text berate input merupakan string biasa satu bari. Adapun syntak type text ialah:

```
<input type = "text" name="nama_input_text"/>
```

2. Form Input Type Radio

Type radio ialah tipe inputan yang memberikan user pilihan, namun user hanya dapat memilih salah satu saja dari beberapa pilihan tersebut. Biasanya digunakan untuk pilihan agama, jenis kelamin dan lainnya yang bersifat diskrit. Adapun syntaknya type radio ialah:

```
<input type="radio" name="JK" value="Pria" /> Pria </b>
```

3. Form Input Type Check Box

Type check box ialah tipe inputan yang memberikan user pilihan, namun user dapat memilih lebih dari satu pilihan atau tidak memilih sama sekali. Adapun contoh sintak type check box ialah:

```
<input type="checkbox" name="hobby" value="Joging" /> Joging </b>
```

4. Form Input Type Combo Box

Combo box biasanya digunakan untuk membuat user memilih salah satu dari banyak pilihan, fungsi combo box sama dengan radio akan tetapi tampilan menunya yang berbeda. Adapun contoh sintak type combo box ialah:

```
<select name="hari">
  <option value="senin">Senin</option>
  <option value="selasa">Selasa</option>
  <option value="rabu">Rabu</option> <option
  value="kamis">Kamis</option> <option
  value="jumat">Jumat</option> <option
  value="sabtu">Sabtu</option> <option
  wvalue="minggu">Minggu</opton>
</select>
```

5. Form Input Type Text Area

Type text area biasanya digunakan untuk kalimat-kalimat yang cukup panjang seperti alamat. Adapun contoh sintak type text area ialah:

```
<textarea name="alamat"></textarea>
```

6. Mengupload file

Pada pemrograman berbasis website seperti PHP fasilitas upload file sangat lumrah dibutuhkan, file-file yang disimpan dalam server dari proses upload dapat membantu dalam sebuah website. Adapun fungsi-fungsi yang dapat digunakan untuk mengupload file ialah:

- **is_uploaded_file()**

Berfungsi untuk untuk konfirmasi ada atau tidaknya proses terjadinya upload file.

Contoh penulisan:

```
is_uploaded_file($_FILES [„file“][„tmp_name“]);
```

- **move_uploaded_file()**

Dimana fungsi dari fungsi ini ialah untuk melakukan pengcopy-an pada path distionasion ke bagian server. Contoh:

```
$dirktori = „upload/“;
```

```
$uploadfile = $dirktori . $_FILES[„file“][„tmp_name“];
```

```
move_uploaded_file($_FILES [„file“][„tmp_name“], $uploadfile);
```

pada sintak di atas fungsi **move_uploaded_file** ialah memindahkan file ke dalam direktori atau folder yang sudah ditentukan contoh folder **upload** yang telah disediakan.

Adapun variabel gloabal \$_files upload yang dapatdigunakan antara lain:

Variabel	Keterangan
<code>\$_FILES[„\$file_nama“][„type“]</code>	Menampilkan tipe file
<code>\$_FILES[„\$file_nama“][„size“]</code>	Menampilkan ukuran file dalam byte
<code>\$_FILES[„\$file_nama“][„tmp_name“]</code>	Menampilkan nama temporary file
<code>\$_FILES[„\$file_nama“][„error“]</code>	Menampilkan kode error pada upload

* \$file_name adalah nama variabel yang bererisi file yang diupload.

C. KEGIATAN PRAKTIKUM

1. Program Form Input Text, Salin coding di bawah ini:

```
<html>
  <head>
    <title>Form Input Text</title>
  </head>

  <body>
    <form id="form_text" name="form_text" method="POST"
      action="Proses_text.php">

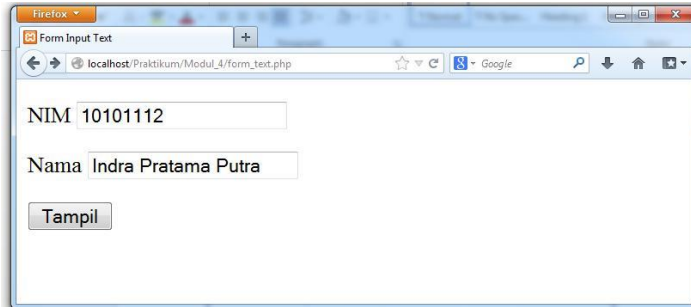
      <p>NIM <input type="text" name="nim" id="nim" /></p> <p>Nama <input
      type="text" name="nama" id="nama"/></p> <p><input type="submit"
      name="Button" id="Button"
      value="Tampil"/></p>
    </form>
  </body>
</html>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **form_text.php** dan buka file baru dan salin coding di bawah ini :

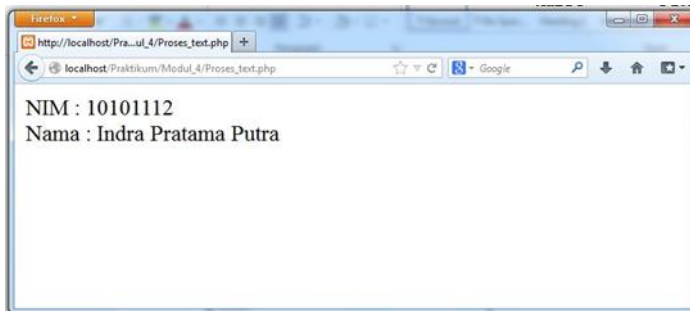
```
<?php
  $nim = $_POST['nim'];
  $nama = $_POST['nama'];
  echo "NIM : $nim <br>";
```

```
    echo "Nama : $nama <br>";  
?>
```

Simpan file baru tersebut dengan nama **roses_text.php** dan letakkan di folder yang sama dengan **form_text.php**. Lalu jalankan **form_text.php** di browser, Maka pada browser akan muncul hasil sebagai berikut:



Bila tombol tampil di tekan, maka muncul hasil:



2. Program Form Input Radio, Salin Koding dibawah ini :

```
<!doctype html>  
<html lang="en">  
  <head>  
    <meta charset="UTF-8">  
    <title>Kalkulator sederhana</title>  
  </head>  
  <body>  
    <form id="form_text" name="form_text" method="POST"  
      action="proses_hitung.php">  
      <table width="400" cellspacing="0" cellpadding="3">  
        <tr>  
          <td>Angka 1</td>  
          <td>: <input type="text" name="angka1" /></td>  
        </tr>  
        <tr>  
          <td>Operator</td>
```

```

        <td><input type="radio" name="operator" value="tambah"> +
        <input type="radio" name="operator" value="kurang"> -
        <input type="radio" name="operator" value="kali"> *
        <input type="radio" name="operator" value="bagi"> /
        <input type="radio" name="operator" value="modulo"> % </td>
    </tr>
    <tr>
        <td>Angka 2</td>
        <td>: <input type="text" name="angka2" /></td>
    </tr>
    <tr>
        <td></td>
        <td><br><input name="submit" type="submit" value="Hitung">
        <input name="Reset" type="reset" value="CLEAR"></td>
    </tr>
</table>
</form>
</body>
</html>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **kalkulator.html** dan buka file baru dan salin coding di bawah ini :

```

<?php
    $angka1=$_POST['angka1'];
    $angka2=$_POST['angka2'];
    $operator=$_POST['operator'];

    if($operator=="tambah"){
        $hasil=$angka1+$angka2;
    }else if ($operator=="kurang"){
        $hasil=$angka1-$angka2;
    }else if ($operator=="kali"){
        $hasil=$angka1*$angka2;
    }else if ($operator=="bagi" && $angka2!=0){
        $hasil=$angka1/$angka2;
    }else if ($operator=="bagi" && $angka2==0){
        $hasil="<b>Error Pembagi 0</b>";
    }else if ($operator=="modulo" && $angka2!=0){
        $hasil=$angka1%$angka2;
    }

```

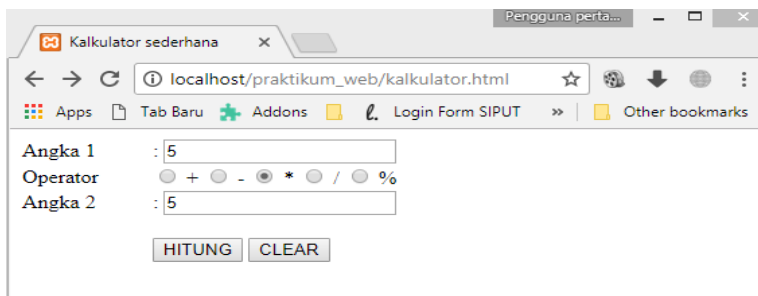


```

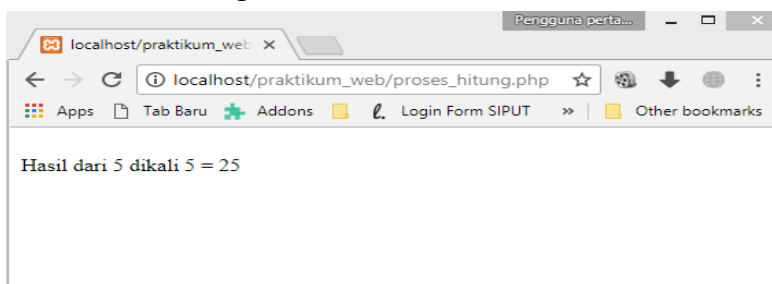
}else if ($operator=="modulo" && $angka2==0){
    $hasil="<b>Error Pembagi 0</b>";
}
echo"<br> Hasil dari $angka1 di$operator $angka2 = $hasil";
?>

```

Simpan file baru tersebut dengan nama **proses hitung.php** dan letakkan di folder yang sama dengan **kalkulator.html**. Lalu jalankan **kalkulator.html** di browser, Maka pada browser akan muncul hasil sebagai berikut:



Bila tombol Hitung di tekan, maka muncul hasil :



3. Program Form Input CheckBox, Salin koding dibawah ini:

```

<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Form Input Checkbox</title>
  </head>
  <body>
    <p>Pilih Game yang disukai</p>
    <FORM NAME="form_checkbox" METHOD="POST"
    ACTION="proses_checkbox.php">
      <input type="checkbox" name="game1" value="Dota 2"> Dota 2<br>
      <input type="checkbox" name="game2" value="Mobile Legend"> Mobile
      Legend<br>

```

```



```

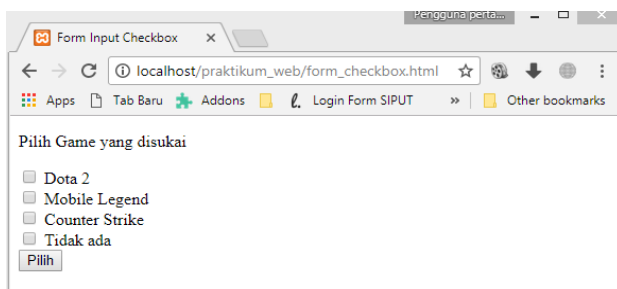
Setelah selesai mengetik coding di atas, maka simpan dengan nama **form_checkbox.html** dan buka file baru dan salin coding di bawah ini:

```

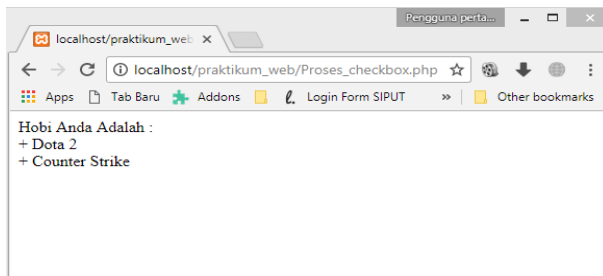
<?php
echo "Hobi Anda Adalah :<br>";
if (isset($_POST['game1'])) {
    echo "+ " . $_POST['game1'] . "<br>";
}
if (isset($_POST['game2'])) {
    echo "+ " . $_POST['game2'] . "<br>";
}
if (isset($_POST['game3'])) {
    echo "+ " . $_POST['game3'] . "<br>";
}
if (isset($_POST['game4'])) {
    echo "+ " . $_POST['game4'] . "<br>";
}
?>

```

Simpan file baru tersebut dengan nama **proses_checkbox.php** dan letakkan di folder yang sama dengan **form_checkbox.html**. Lalu jalankan **form_checkbox.html** di browser, Maka pada browser akan muncul hasil sebagai berikut:



Bila tombol Proses di tekan, maka muncul hasil:



4. Program Form Input ComboBox, salin koding dibawah ini :

```
<html>
  <head>
    <title>Form Input ComboBox</title>
  </head>
  <body>
    <p>Binatang apa yang Anda suka?</p>
    <form id="form_combobox" name="form_combobox" method="POST"
      action="Proses_combobox.php">

      <select name="favorit">
        <option value="Kucing">Kucing</option>
        <option value="Anjing">Anjing</option>
        <option value="Burung">Burung</option>
        <option value="Sapi">Sapi</option>
        <option value="Kambing">Kambing</option>
      </select>
      <p></p>
      <p><input type="submit" name="button" value="Tampil"/></p>
    </form>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **form_combobox.php** dan buka file baru dan salin coding di bawah ini :

```
<?php
  $binatang = $_POST['favorit'];
  echo "Binatang yang anda sukai ialah $binatang";
?>
```

Simpan file baru tersebut dengan nama **proses_combobox.php** dan letakkan di folder yang sama dengan **form_combobox.php**. Lalu jalankan **form_combobox.php** di browser, Maka pada browser akan muncul hasil sebagai berikut:



Bila tombol Proses di tekan, maka muncul hasil :



5. Program Form Input TextArea, buka file baru dan salin koding dibawah ini:

```
<html>
<head>
    <title>Form Input Text Area</title> </head>
<body>
    <p>Masukkan komentar atau pendapat anda!</p>
    <form id="form_area" name="form_area" method="POST"
    action="Proses_area.php">
    <textarea name="komen"></textarea><br/>
    <p></p>
    <p><input type="submit" name="button" value="Tampil"/></p>
    </form>
</body>
</html>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **form_radio.php** dan buka file baru dan salin coding di bawah ini :

```
<?php
$komentar = $_POST['komen'];
echo "Komentar atau Pendapat Anda :<br/>"; echo "$komentar";
?>
```

Simpan file baru tersebut dengan nama **proses_radio.php** dan letakkan di folder yang sama dengan **form_radio.php**. Lalu jalankan **form_radio.php** di browser, Maka pada browser akan muncul hasil sebagai berikut:



Bila tombol Proses di tekan, maka muncul hasil



6. Program Upload File, salin koding dibawah ini :

```
<html>
<head>
    <title>Form Upload File</title>
</head>
<body>
    <form enctype="multipart/form-data" METHOD="POST"
    ACTION="proses_upload.php">
    <p>Pilih File : <input type="file" name="file"></p> <input type="submit"
    name="Button" value="Upload">
    </form>
</body>
</html>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **form_upload.php** dan buka file baru dan salin coding di bawah ini :

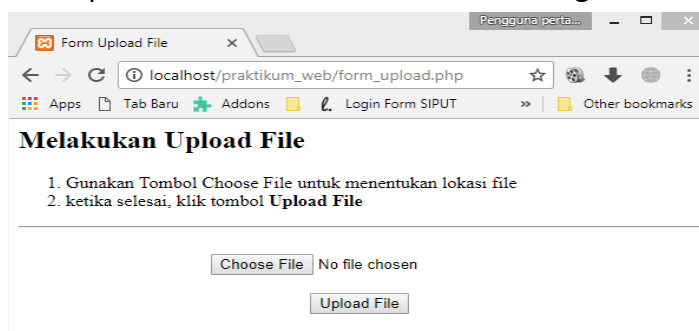
```
<?php
$dir_upload = "files/";
$uploadfile = $dir_upload . $_FILES['file']['name'];
if (is_uploaded_file($_FILES['file']['tmp_name'])) {
    $kirim = move_uploaded_file ($_FILES['file']['tmp_name'], $uploadfile);
```

```

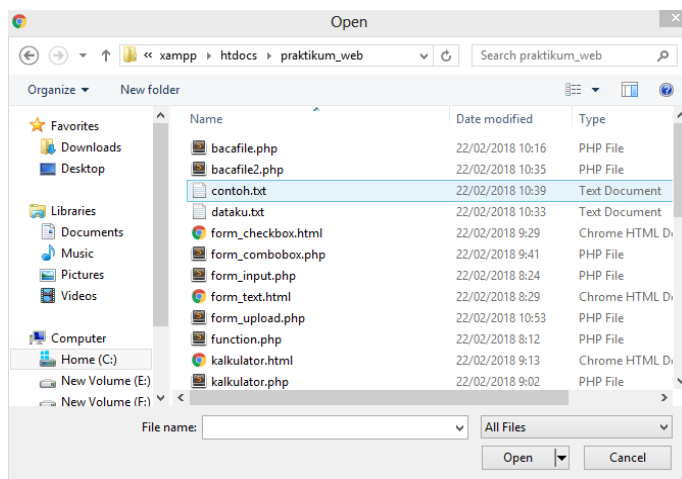
if ($kirim) {
echo "File berhasil diupload ke server pada folder <b>$uploadfile</b>";
} else {
echo "File gagal diupload";
echo "error : ", $_FILES['file']['error'];
}
}
?>

```

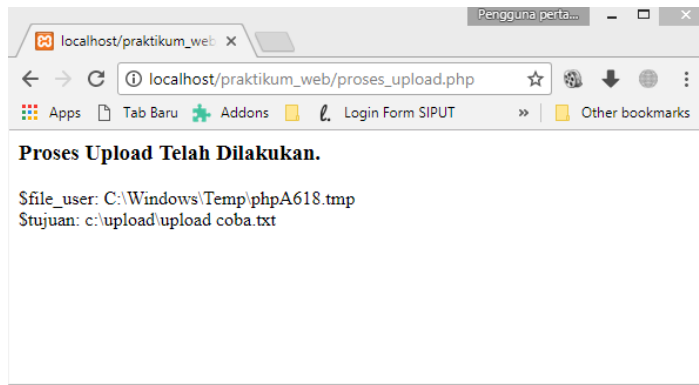
Simpan file baru tersebut dengan nama **proses_upload.php** dan letakkan di folder yang sama dengan **form_upload.php**. Lalu jalankan **form_upload.php** di browser, Maka pada browser akan muncul hasil sebagai berikut:



Untuk mencari file yang akan diupload kita harus menekan tombol choose file... , setelah menekan tombol choose file... maka akan muncul jendela windows untuk memilih file yang akan diupload.



Setelah memilih file, tekan tombol Upload file. Maka akan muncul hasil seperti berikut:



D. TUGAS

1. Buatlah program tampil biodata menggunakan 5 form input di atas!
2. Buatlah form login sederhana!

MODUL VIII FUNCTION

A. TUJUAN :

1. Mahasiswa dapat memahami dan mengaplikasikan fungsi dalam pembuatan web menggunakan bahasa pemrograman PHP.
2. Mahasiswa dapat mengetahui cara pendeklarasian array menggunakan bahasa pemrograman PHP dan mengaplikasikannya dalam pembuatan web.
3. Mahasiswa dapat memahami dan melakukan pengolahan data array menggunakan bahasa pemrograman PHP
4. Mahasiswa dapat mengetahui dan menggunakan fungsi-fungsi array yang ada dalam bahasa pemrograman PHP.

B. DASAR TEORI

Fungsi (*function*) adalah kumpulan beberapa statement yang dibuat untuk menyelesaikan suatu tugas tertentu yang dapat dipanggil sesera berulang. Fungsi juga digunakan untuk mencegah script dieksekusi pada saat halaman web diload, karena untuk mengeksekusinya kita harus memanggil fungsi tersebut.

1. Deklarasi fungsi

Untuk mendeklarasikan fungsi terdapat beberapa hal yang harus diperhatikan :

- a. **nama fungsi** (nama dari fungsi yang berfungsi untuk memanggil fungsi tersebut untuk dieksekusi)
- b. **parameter** (nilai tertentu yang dimasukkan ke dalam fungsi. dapat juga diisi dengan variabel)
- c. **baris perintah** (perintah yang dijalankan saat fungsi dipanggil)

contoh:

```
function nama_fungsi (parameter1, parameter2,.....){  
    baris_perintah;  
}
```

2. Fungsi tanpa parameter

Pada saat mendeklarasikan sebuah fungsi adakalanya kita dapat mendeklarasikan tanpa memberikan parameter.

Contoh :

```
function nama_saya (){  
    echo "Indra Saputra";  
}
```

3. Fungsi dengan parameter

Saat kita mendeklarasikan sebuah fungsi yang memiliki parameter, maka parameter tersebut menjadi sebuah inputan yang akan diproses dalam fungsi tersebut.

Contoh:

```
Function tambah ($x, $y){  
    $z = $x +$ y; echo $z;  
}
```

C. KEGIATAN PRAKTIKUM

1. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```
<?php  
    function tampil_ganjil(){  
        for ($i =1; $i<=10; $i++){  
            if ($i % 2 == 1){  
                echo "$i ";  
            }  
        }  
    }  
    //pemanggil fungsi  
  
    echo "Bilangan ganjil dari 1-10 :<br/>"; tampil_ganjil();  
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **function.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:

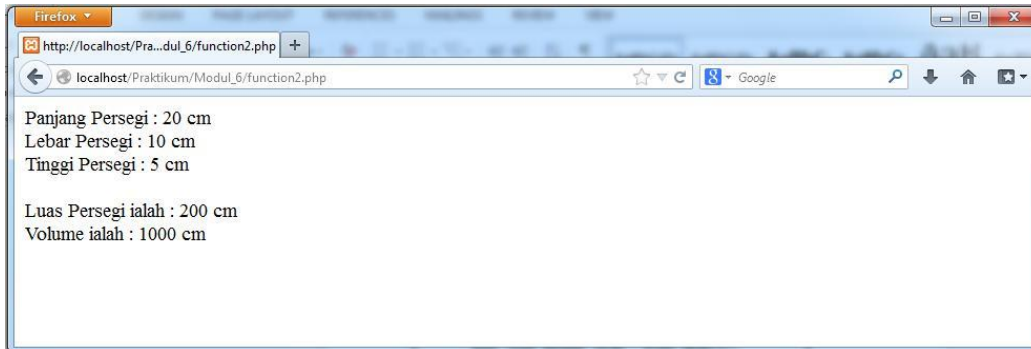


2. Buka notepad++, buat file baru **New** atau *Ctrl+N*, kemudian Salin Koding Dibawah ini:

```
<?php  
    function volume($luas,$tinggi){  
        $volume = $luas * $tinggi;  
        echo "Volume ialah : $volume cm";  
    }  
  
    function luas_persegi($p, $l, $t){  
        $luas = $p * $l;  
        echo "Luas Persegi ialah : $luas cm<br/>"; volume($luas, $t);  
    }  
  
    $panjang = 20;  
    $lebar = 10;  
    $tinggi = 5;
```

```
echo "Panjang Persegi : $panjang cm<br />";  
echo "Lebar Persegi : $lebar cm<br/>";  
echo "Tinggi Persegi : $tinggi cm<br/><br/>"; luas_persegi($panjang, $lebar,  
$tinggi);  
?>
```

Setelah selesai mengetik *coding* di atas, maka simpan dengan nama **function1.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



D. TUGAS PRAKTIKUM

1. Buatlah script untuk menghitung jumlah bilangan ganjil antara 10 sampai 123456. Gunakan function untuk mengecek apakah suatu bilangan termasuk bilangan ganjil atau tidak.

MODUL IX SESSION DAN COOKIES

A. TUJUAN :

1. Mahasiswa dapat memahami dan mengaplikasi penggunaan session dan cookies dalam pembuatan web menggunakan bahasa pemrograman PHP.
2. Mahasiswa dapat memahami dan mengaplikasikan pembuatan, penggunaan dan penghapusan session dalam pemrograman PHP.
3. Mahasiswa dapat memahami dan mengaplikasikan pembuatan, penggunaan dan penghapusan cookies dalam pemrograman PHP.

B. DASAR TEORI

1. *Session*

Session adalah sebuah tools dalam pemrograman web yang memungkinkan server menyimpan data informasi user secara semi-permanen, dimana selama masa tertentu informasi tersebut akan disimpan di server. Biasanya session digunakan dalam proses login sebuah website, misal saat kita ingin melihat suatu halaman pada suatu website yang mengharuskan kita untuk login sebagai user terlebih dahulu, dalam proses login akan terjadi pembuatan sebuah session yang akan dipakai user pada setiap pembukaan halaman pada website tersebut. Pada saat pembukaan suatu halaman di website tersebut maka session tersebut akan diperiksa, apabila session benar maka halaman yang ingin dibuka tersebut akan tampil dan apabila session salah atau tidak terbaca maka website akan menampilkan kembali halaman login dan meminta user untuk melakukan login kembali.

Dengan session, sebuah halaman pada suatu website dapat dibuka oleh lebih dari satu user yang berbedan karena setiap user memiliki session yang berbeda.

FUNGSI	KETERANGAN
SESSION_START()	Untuk memulai/membuat session.
SESSION_DESTROY()	Untuk mengakhiri/menghapus session.
SESSION_ID()	Untuk mengambil atau menentukan identitas sebuah session (session id).

2. *Cookies*

Hampir sama dengan session, cookies adalah data yang berisi tentang informasi user yang dikirim oleh server ke clien dan disimpan di remout browser client sehingga memudahkan penelusuran identitas user. Adapun fungsi yang digunakan untuk membuat cookiest ialah `setcookie()`, dan penulisannya ialah:

Setcookie("Nama", "Isi", "Usia");

Dimana Nama adalah nama dari cookies tersebut sedangkan isi adalah data yang akan dimasukkan kedalam cookies tersebut dan usia adalah berapa lamakah cookies tersebut akan aktif. Untuk usia cookies dibaca dalam detik, misal apabila kita memasukan nilai 10 maka cookies akan aktif selama 10 detik dan akan terhapus apabila telah telah mencapai waktu tersebut. Dan untuk menghapus cookies, kita hanya perlu untuk menset cokies tersebut denga isi dan usia yang kosong.

C. KEGIATAN PRAKTIKUM

1. Program Login Session, salin koding dibawah ini :

```
<?php
    session_start();
    if(isset($_POST['Login'])){
        if(($_POST['nama']=="" && ($_POST['pass']=="")) { echo "User Name dan
        Password masih kosong"; session_destroy();
            }else{
                // user = "johan" & password = "bengbeng123"
                if((($_POST['nama']=="johan")and ($_POST['pass']=="bengbeng123"))){
                    $_SESSION['login']=1;
                    $_SESSION['username']=$_POST['nama'];
                }

                if ((isset($_SESSION['login']) and ($_SESSION['login']==1)){
                    header("location: submit_formlogin.php");
                    exit();
                }
            }
        }
?>

<html>
<head>
    <title>Session</title>
</head>
<body>
    <form name="session" method="POST" action=""> <p>Form Login</p>
    <p>User Name <input type="text" name="nama"/></p> <p>Password
    <input type="password" name="pass" /></p> <input type="submit"
    name="Login" value="Login" />
    </form>
</body>
</html>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **proses_login.php** dan buka file baru dan salin coding di bawah ini :

```
<?php
session_start();
if ( (isset($_GET['aksi'])) and ($_GET['aksi']=="logout") ){
    session_destroy();
    header("refresh;3;proses_radio.php");
}
if(isset($_SESSION['login'])){
    $nama = $_SESSION['username'];
    echo"<center>";
```

```

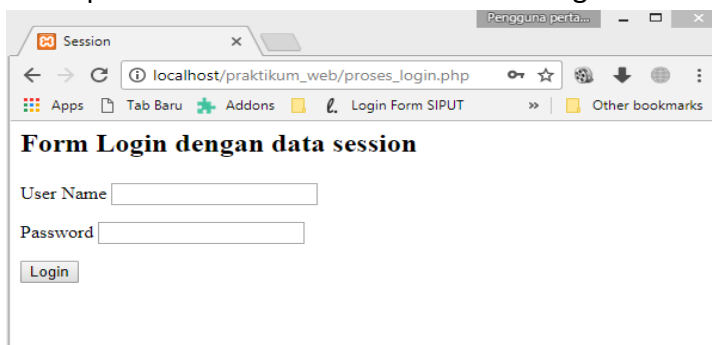
        echo"<p>Selamat Datang <b>".$nama."</b></p>";
        echo"<p>Berikut ini menu navigasi anda</p>";
        echo"<p><a href='menu1.php'>Menu 1&nbsp;</a>
        <a href='menu2.php'>Menu 2&nbsp;</a>
        <a href='menu3.php'>Menu 3&nbsp;</a></p>";
?>

<a href="?aksi=logout">Logout</a> </body>
</center>

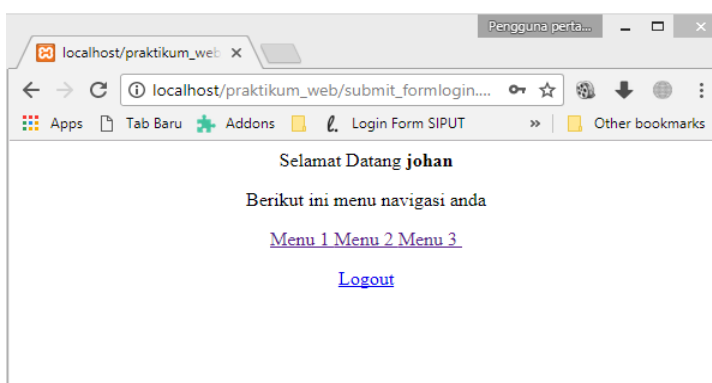
<?php
    }else{
        header("location:proses_login.php");
    }
?>

```

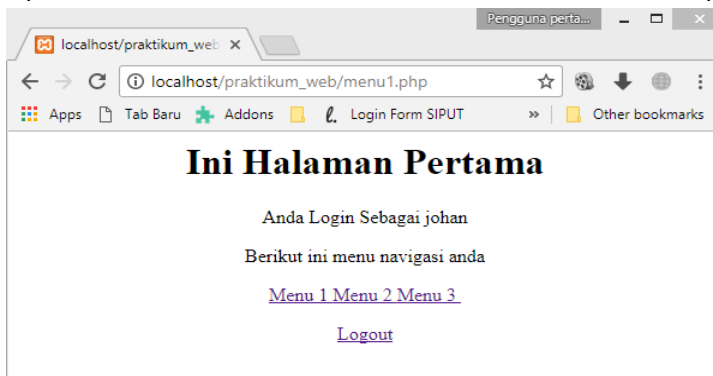
Simpan file baru tersebut dengan nama **submit_formlogin.php** dan letakkan di folder yang sama dengan **proses_login.php**. Lalu jalankan **proses_login.php** di browser, Maka pada browser akan muncul hasil sebagai berikut:



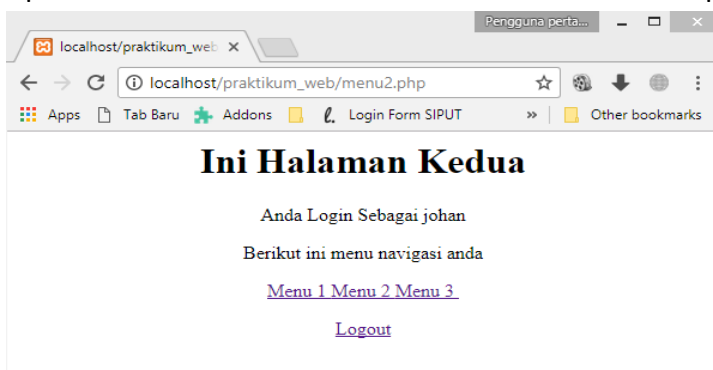
Setelah itu masukan "johan" pada user name dan "bengbeng123" pada password, kemudian akan muncul :



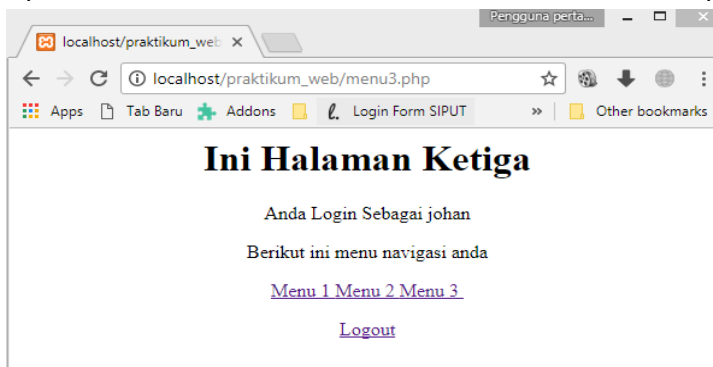
Apabila kita menekan Menu 1 maka akan muncul tampilan :



Apabila kita menekan Menu 2 maka akan muncul tampilan :



Apabila kita menekan Menu 3 maka akan muncul tampilan :



Apabila kita menekan "Logout" maka kita akan diarahkan kembali ke tampilan form login.

2. Program Cookies, salin koding dibawah ini :

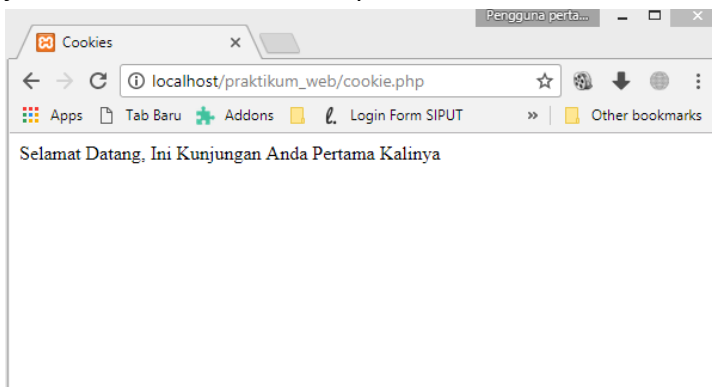
```
<?PHP
    $expire = time()+10;
    setcookie('netter',1,$expire);
?>
<html>
<head>
<title>Cookies</title>
</head>
```

```

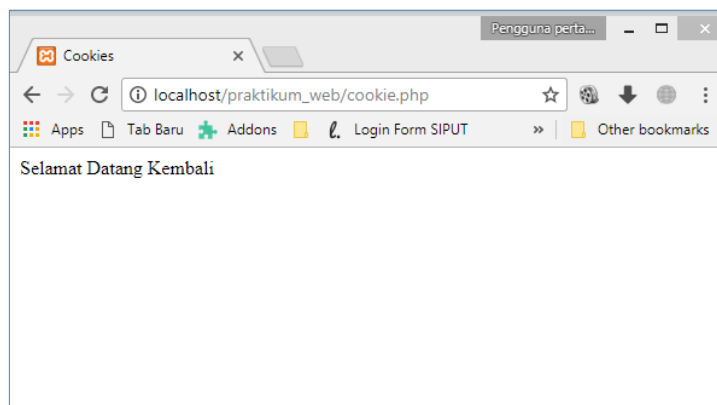
<body>
<?PHP
    if(isset($_COOKIE['netter'])){
        echo "Selamat Datang Kembali";
    }else{
        echo "Selamat Datang, Ini Kunjungan Anda Pertama Kalinya";
    }
?>
</body>
</html>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama cookie.php dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



Apabila kita merefresh browser kembali sebelum 10 detik maka akan muncul :



D. TUGAS

1. Buatlah autentikasi user dari suatu aplikasi web menggunakan konsep *SESSION*. Adapun username-username yang boleh mengakses aplikasi ini adalah

USERNAME	PASSWORD
ALI	password1
BONA	password2

CHARLIE	password3
DEDE	password4
EMON	password5

Username dan password di atas harus disimpan dalam sebuah ARRAY.

2. Buatlah aplikasi seperti berikut:

Terlebih dahulu buatlah sebuah desain halaman web dengan *layout* seperti di bawah ini:

TITLE
KONTEN
FOOTER

Selanjutnya, bagi-bagilah layout tersebut ke dalam modul header.php dan footer.php Untuk halaman utama, tampilkan form login untuk user.

TITLE
<p style="text-align: center;">Silakan Login</p> <p>Username <input type="text"/></p> <p>Password <input type="password"/></p> <p style="text-align: center;">[SUBMIT]</p>
FOOTER

Selanjutnya bila login berhasil tampilkan halaman berikut ini (munculkan link navigasi) :

<p>TITLE</p> <p>Username : XXX</p>
<p>Link 1 Link 2 Link 3 Logout</p> <p>Anda telah berhasil login</p>
<p>FOOTER</p>

Keterangan:

- Login dikatakan berhasil bila password yang dimasukkan oleh seorang user melalui form **login sama dengan password** yang tersimpan dalam aplikasi (dalam hal ini password yang ada dalam array).
- XXX adalah username yang sukses login.

Untuk setiap halaman Link 1, 2, dan 3 berikut ini tampilannya

<p>TITLE</p> <p>Username : XXX</p>
<p>Link 1 Link 2 Link 3 Logout</p> <p>Ini adalah isi dari halaman Link 1</p>
<p>FOOTER</p>

Catatan:

1. Aplikasi yang dibuat harus bisa handle apabila login salah, misalnya muncul keterangan "Password yang dimasukkan salah" bila username terdaftar namun passwordnya salah, atau "Username tidak terdaftar" bila usernamenya tidak terdaftar dalam aplikasi.
Bila password login salah atau user tidak terdaftar, maka menu link navigasi tidak ditampilkan
2. Aplikasi yang dibuat harus bisa handle adanya penyusup yang ingin langsung masuk ke halaman Link 1, 2 dan 3 tanpa melalui login (autentifikasi).
3. Aplikasi yang dibuat harus bisa handle proses logout.

MODUL X DATABASE DALAM PHP

A. TUJUAN :

1. Mahasiswa dapat memahami dan mengaplikasi pembuatan web yang menggunakan database MySQL.
2. Mahasiswa dapat memahami dan melakukan koneksi database MySQL dengan web menggunakan bahasa pemrograman PHP
3. Mahasiswa dapat menampilkan record data menggunakan bahasa pemrograman PHP
4. Mahasiswa dapat menambah, mengubah dan menghapus record data dalam database menggunakan bahasa pemrograman PHP.

B. DASAR TEORI

Penyimpanan data yang fleksibel dan mudah diakses sangatlah dibutuhkan dalam website yang interaktif dan dinamis. Jenis database yang sangat populer dan digunakan pada banyak website di internet sebagai bank data adalah MySQL. Dalam PHP terdapat beberapa fungsi-fungsi yang dapat digunakan untuk berinteraksi dengan MySQL yaitu Create Read Update Delete (CRUD).

1. Koneksi PHP dan MySQL

Dalam php untuk melakukan koneksi ke database MySQL kita dapat menggunakan fungsi **Mysql_connect()** agar database yang terdapat pada MySQL dapat diakses melalui PHP, adapun bentuk penulisannya ialah:

Mysqli_connect(nama_host, nama_user, password);

Dimana host ialah sebagai alamat letak database tersimpan sedangkan user adalah nama user untuk mengakses database dan password adalah password yang digunakan untuk verifikasi keamanan database. Biasanya kita juga dapat menggunakan koneksi default ke computer local dengan memasukkan "localhost" pada host, serta untuk user kita masukan "root" dan mengkosongkan password, contoh:

Mysqli_connect("localhost","root","");

• Mysqli_query()

Ini adalah fungsi yang dapat membantu kita untuk memasukkan query perintah untuk memberikan perintah ke database MySQL, seperti membuat dan menghapus tabel, menambah, mengupdate dan menghapus data pada tabel. Adapun bentuk penulisannya ialah:

Mysqli_query(perintah);

Dimana perintah adalah string yang berisi query perintah dalam database MySQL.

2. Menambah, mengubah, dan menghapus data

Dalam PHP tidak memiliki fungsi khusus untuk menangani field-field beserta data-datanya sehingga untuk memanipulasi field digunakan bentuk penulisan perintah dari MySQL itu sendiri selaku database management sistem. Pemberian sebuah perintah ke MySQL dapat dioperasikan menggunakan fungsi **mysqli_query()**.

Adapun penulisannya ialah:

Mysqli_query(perintah);

Keterangan :

perintah adalah string yang berisi query perintah dalam database MySQL Sebelum memasuki materi memadipulasi data dalam field, terlebih dahulu kita harus mengetahui bagai mana cara untuk menciptakan sebuah database, tabel dan field dengan menggunakan fungsi mysqli_query(). Adapun perintah query yang digunakan untuk menciptakan database dan tabel baru ialah:

- **Pembuatan database baru**

Untuk membuat database baru, kita dapat menggunakan perintah query SQL sebagai berikut:

CREATE DATABASE nama_db;

Dimana nama_db ialah nama yang akan diberikan pada database yang akan diciptakan.

- **Pembuatan tabel baru**

Untuk membuat tabel baru , kita dapat menggunakan perintah query SQL sebagai berikut :

Create tabel nama_tabel(field1, field2, ...);

Dimana nama_tabel ialah nama yang akan diberikan untuk tabel yang akan dibuat. Sedangkan field1, field2, ... ialah field-field yang akan dimiliki oleh tabel yang akan dibuat.

Contoh:

Create tabel mahasiswa(nim char(7), nama varchar(20), nilai int(),...);

Dari query di atas membuat perintah untuk membuat tabel bernama mahasiswa dengan field nim bertipe data char dengan panjang 7 karakter, nama bertipe data varchar dengan panjang 20 karakter, nilai bertipe data integer dan seterusnya. Setelah membuat sebuah database dan tabel baru maka kita dapat memanipulasi isi tabel tersebut dengan fungsi mysqli_query() dan perintah query sebagai berikut:

- **Insert into**

Insert into adalah Perintah query untuk menambah data pada tabel tertentu. Adapun bentuk penulisannya ialah:

INSERT INTO Nama_Tabel (field1, field2,...) VALUES (data1, data2, ...);

Dimana Nama_Tabel ialah nama dari tabel yang akan diisi oleh data tersebut, **sedangkan** field1, field2,... ialah nama field atau kolom dari tabel yang akan diisi data baru, sedangkan data1, data2,... ialah nilai atau data yang akan dimasukkan ke dalam field atau kolom, contoh:

INSERT INTO mahasiwa (nim, nama, nilai, ...) VALUES ("1112134", "Dewi", 90, ...);

- **Update**

Update adalah perintah query SQL untuk memodifikasi atau merubah data pada database.

Adapun bentuk penulisannya ialah:

UPDATE Nama_Tabel SET field1 = nilai_baru, field2 = nilai_baru, ... WHERE syarat;

Dimana Nama_Tabel ialah nama tabel dari data yang akan diupdate, sedangkan **field1**, field2, ... adalah nama field atau kolom dari data yang akan dirubah, sedangkan nilai_baru ialah data yang akan menggantikan data yang akan diganti dan syarat ialah ketentuan yang harus dipenuhi dalam mengubah data. Contoh :

UPDATE mahasiswa SET nama = 'Ayu', nilai = 95 ... WHERE nim = '1112134';

- **Delete from**

Delete from adalah perintah query SQL untuk menghapus data pada database. Adapun

bentuk penulisannya ialah :

DELETE FROM Nama_Tabel WHERE syarat;

Dimana nama_tabel adalah nama tabel dari data yang akan dihapus, sedangkan syarat ialah ketentuan data yang akan dihapus. Contoh:

DELETE FORM mahasiswa WHERE nim = '1112134';

3. Menampilkan data

Untuk menampilkan data dari database MySQL, kita dapat menggunakan fungsi query **SELECT** untuk memilih apa saja data yang akan ditampilkan. Adapun bentuk penulisannya ialah:

SELECT field1, field2, ... FORM nama_tabel WHERE syarat;

Dimana field1, field2, ... adalah nama field atau kolom dalam tabel yang datanya akan ditampilkan, sedangkan nama_tabel adalah nama tabel yang akan ditampilkan datanya, dan syarat ialah ketentuan data yang akan ditampilkan. Contoh:

SELECT nama, nilai, ... FROM mahasiswa WHERE nim='1112134';

Selain menggunakan query **SELECT**, kita juga dapat menggunakan fungsi **mysqli_fetch_row()** atau **mysqli_fetch_array()** yang berfungsi untuk mengambil data dari hasil query **SELECT** secara baris per baris. Pengambilan pertama adalah baris data yang paling atas dan data yang diambil berbentuk array. Adapun bentuk penulisannya ialah :

mysqli_fetch_row(variabel); dan

mysqli_fetch_array(variabel);

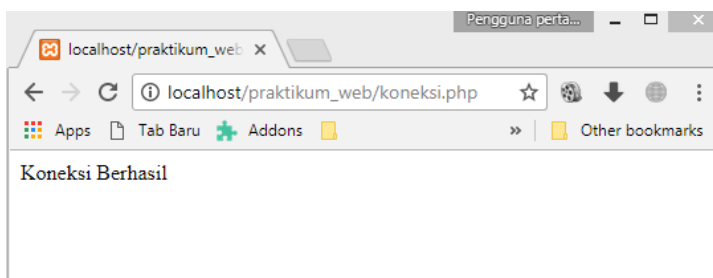
Dimana variabel adalah variabel yang berisi hasil pelaksanaan perintah query untuk mengakses database.

C. KEGIATAN PRAKTIKUM

1. Program koneksi MySQL, salin koding dibawah ini :

```
<?php
//Koneksi Ke MySQL
$conn = mysqli_connect('localhost','root',''); if(!$conn){
echo "Koneksi Gagal";
}else{
echo "Koneksi Berhasil";
}
?>
```

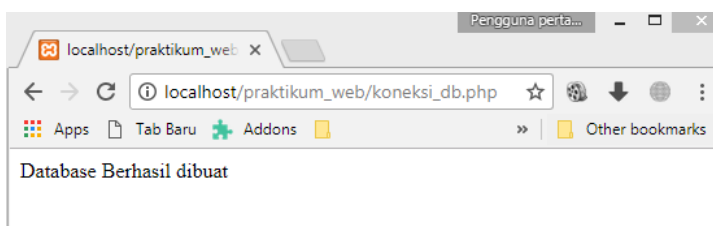
Setelah selesai mengetik coding di atas, maka simpan dengan nama koneksi.php dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



2. Program koneksi MySQL, salin koding dibawah ini :

```
<?php
$conn = mysqli_connect('localhost','root','');
if(mysqli_connect_errno()){
echo "koneksi ke sever gagal";
}
$sql = "CREATE DATABASE db_kampus";
if(mysqli_query($conn, $sql))
{ echo "Database Berhasil dibuat";
} else{ echo "Gagal membuat Database :".mysqli_error($conn); }
mysqli_close($conn);
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama koneksi_db.php dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



3. Program Membuat Tabel, salin koding dibawah ini :

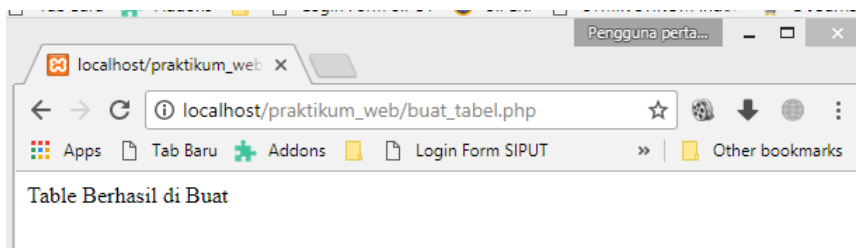
```
<?php
    $dbname='db_kampus';
    $host='localhost';
    $password="";
    $username='root';

    //Koneksi Ke MySQL
    $cnn = mysqli_connect($host,$username,$password,$dbname);

    //Membuat Koneksi
    if(!$cnn){ die("Koneksi Failed : ".mysqli_connect_error()); }
    //Membuat Tabel
    $sql ="CREATE TABLE mahasiswa (
        nim CHAR(10) NULL,
        nama VARCHAR(25) Null,
        alamat VARCHAR(50) Null,
        telp VARCHAR(15) Null,
        constraint pk_dasiswa primary key(nim))";

    if(mysqli_query($cnn, $sql)){
        echo "Table Berhasil di Buat";
    } else {
        echo "Table Gagal di Buat :".mysqli_error($cnn); }
    mysqli_close($cnn);
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama buat_tabel.php dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



4. Program Tambah Data, salin koding dibawah ini :

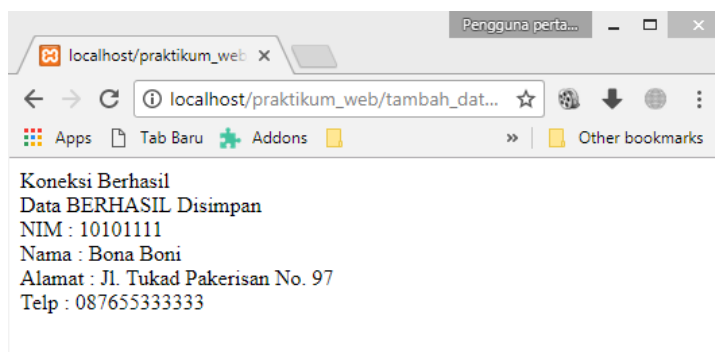
```
<?php
    $cnn = mysqli_connect('localhost','root','');
    if(!$cnn){
        echo "Koneksi Gagal";
    }else{
        echo "Koneksi Berhasil<br/>";
        mysqli_select_db($cnn,"db_kampus");
    }
    $nim = "10101111";
```

```

$nama = "Bona Boni";
$alamat = "Jl. Tukad Pakerisan No. 97";
$telp = "087655333333";
$query = "select * from mahasiswa";// memilih mengakses db_kampus
$sql = "INSERT INTO mahasiswa (nim,nama,alamat,telp) VALUES
('$nim','$nama','$alamat','$telp)";
$hasil = mysqli_query($cnn,$sql);
if(mysqli_query($cnn,$sql)){
    echo "Data GAGAL Disimpan <br/>";
}else{
    echo "Data BERHASIL Disimpan <br/>";
    echo "NIM : $nim <br/>";
    echo "Nama : $nama <br/>";
    echo "Alamat : $alamat <br/>";
    echo "Telp : $telp <br/>";
}
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama tambah_data.php dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



5. Program Update Data, salin koding dibawah ini :

```

<?php
$cnn = mysqli_connect('localhost','root','','db_kampus');
if(!$cnn){
    echo "Koneksi Gagal <br/>";
}else{
    echo "Koneksi Berhasil <br/>";
}

mysqli_select_db($cnn,"db_kampus");

$sql = "UPDATE mahasiswa set nama='Rona Roni', telp = '081657775987', alamat =
'Jalan Gowokwaru' where nim = '10101111'";

$update = mysqli_query($cnn,$sql);

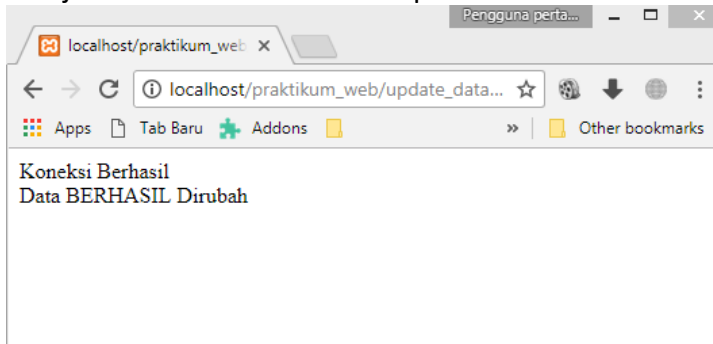
```

```

if (!$update){
    echo "Data GAGAL Dirubah <br/>";
}else{
    echo "Data BERHASIL Dirubah <br/>";
}

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama update_data.php dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



6. Program Menampilkan Data, salin koding dibawah ini :

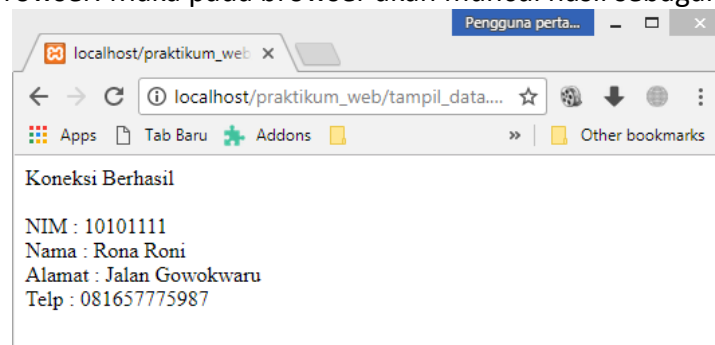
```

<?php
    $cnn = mysqli_connect('localhost','root','','db_kampus');
    if(!$cnn){
        echo "Koneksi Gagal <br/>";
    }else{
        echo "Koneksi Berhasil <br/>";
    }

    mysqli_select_db($cnn,"db_kampus");
    $sql = "SELECT nim, nama, alamat, telp FROM mahasiswa;";
    $tampil = mysqli_query($cnn,$sql);
    while ($data = mysqli_fetch_array($tampil)){ echo "NIM : ".
    $data['nim']."<br/>"; echo "Nama : ". $data['nama']."<br/>"; echo "Alamat
    : ". $data['alamat']."<br/>"; echo "Telp : ". $data['telp']."<br/>"; echo
    "<br/>";
    }
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama tampil.php dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



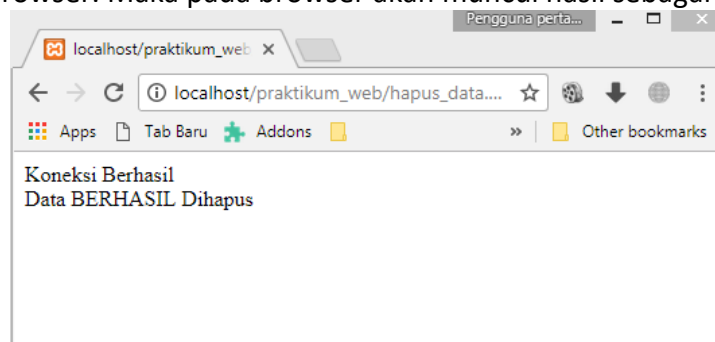
7. Program Delete Data, salin koding dibawah ini :

```
<?php
    $cnn = mysqli_connect('localhost','root','','db_kampus');
    if(!$cnn){
        echo "Koneksi Gagal <br/>";
    }else{
        echo "Koneksi Berhasil <br/>";
    }

    mysqli_select_db($cnn,"db_kampus");
    $sql = "DELETE FROM mahasiswa where nim ='10101111'";

    $hapus = mysqli_query($cnn,$sql);
    if (!$hapus){
        echo "Data GAGAL Dihapus <br/>";
    }else{
        echo "Data BERHASIL Dihapus <br/>";
    }
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama delete.php dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



D. TUGAS

1. Buatlah CRUD sederhana untuk menampilkan 5 nama teman anda di kelas

MODUL XI LANJUTAN DATABASE DALAM PHP

A. TUJUAN :

1. Mahasiswa dapat memahami dan mengaplikasi pembuatan web yang menggunakan database MySQL.
2. Mahasiswa dapat memahami dan melakukan koneksi database MySQL dengan web menggunakan bahasa pemrograman PHP
3. Mahasiswa dapat menampilkan record data menggunakan bahasa pemrograman PHP
4. Mahasiswa dapat menambah, mengubah dan menghapus record data dalam database menggunakan bahasa pemrograman PHP.

B. KEGIATAN PRAKTIKUM

1. Buka direktori **C > xampp > htdocs**, kemudian buat folder dan beri nama **“bukutamu”**.
2. Program koneksi MySQL, salin koding dibawah ini :

```
<?php
$dbname='db_tamu';
$host='localhost';
$password="";
$username='root';

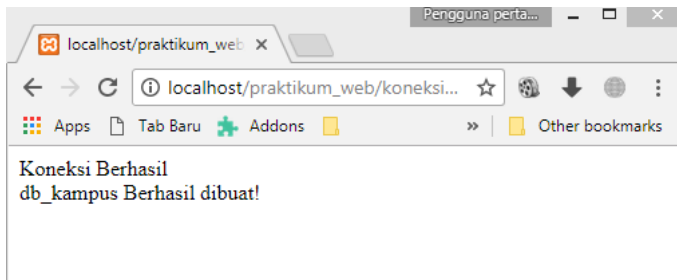
//Koneksi Ke MySQL
$conn = mysqli_connect($host,$username,$password,$dbname);
if(mysqli_connect_errno()){
    echo "Koneksi Gagal.";
    exit();
}
?>
```

Simpan coding yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > bukutamu**, dan beri nama file dengan **“koneksi.php”**, klik Save.

3. Membuat database, salin koding dibawah ini :

```
<?php
$conn = mysqli_connect('localhost','root');
if(mysqli_connect_errno()){
    echo"koneksi ke sever gagal";
}
$sql = "CREATE DATABASE db_tamu";
if(mysqli_query($conn, $sql))
    { echo "Database Berhasil dibuat";
      } else{ echo "Gagal membuat Database :".mysqli_error($conn); }
mysqli_close($conn);
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **buat_db.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



4. Program Membuat Tabel, salin koding dibawah ini :

```
<?php
    $dbname='db_tamu';
    $host='localhost';
    $password='';
    $username='root';
    //Koneksi Ke MySQL
    $cnn = mysqli_connect($host,$username,$password,$dbname);
    //Membuat Koneksi
    if(!$cnn){
        die("Koneksi Failed : ".mysqli_connect_error()); }
    //Membuat Tabel
    $sql ="CREATE TABLE bukutamu (
        id integer not null auto_increment,
        nama varchar(25),
        email varchar(50),
        url varchar(100),
        komentar text,
        constraint pk_bukutamu primary key(id)
    )";
    if (mysqli_query($cnn, $sql)){echo "Table Berhasil di Buat";
    } else {
        echo "Table Gagal di Buat :".mysqli_error($cnn); }
    mysqli_close($cnn);
?>
```

Setelah selesai membuat script di atas, maka simpan dengan nama **buat_tb.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:

5. Membuat Buku Tamu, salin koding dibawah ini :

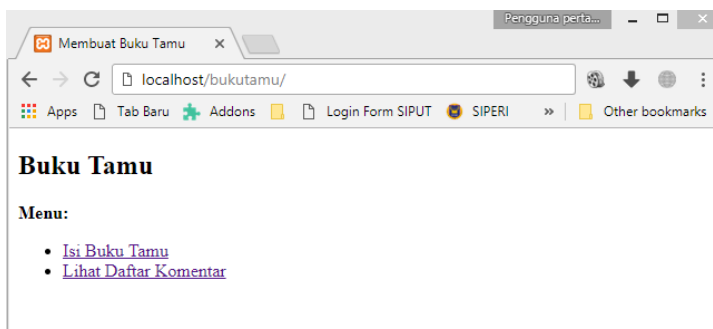
```
<!doctype html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <title>Membuat Buku Tamu</title>
    </head>
    <body>
```

```

<h2>Buku Tamu</h2>
<strong>Menu:</strong>
  <ul>
    <li> <a href="tambah_data.php">Isi Buku Tamu</a></li>
    <li><a href="tampil_data.php">Lihat Daftar Komentar
      </a></li>
    </ul>
</body>
</html>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **index.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



6. Tambah buku tamu, salin koding dibawah ini :

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Membuat Buku Tamu</title>
</head>
<body>
  <h2>Form Pencatatan Buku Tamu</h2>
  Silahkan Masukkan Komentar Anda
  <br/><br/>
  <form action="proses_tambah.php" method="post">
  <table>
    <tr>
      <td>Nama</td>
      <td><input type="textfield" name="nama" maxlength="50" size="50" /></td>
    </tr>
    <tr>
      <td>Email</td>
      <td><input type="textfield" name="email" maxlength="50" size="50" /></td>
    </tr>
    <tr>
      <td>URL</td>

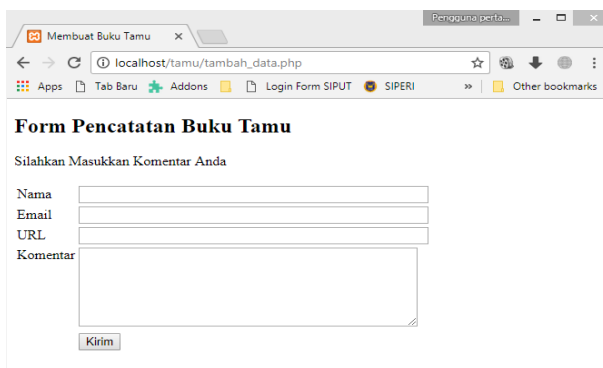
```

```

<td><input type="textfield" name="url"maxlength="100" size="50" /></td>
</tr>
<tr>
<td valign="top">Komentar</td>
<td><textarea name="komentar" cols="50" rows="6"> </textarea>
</td>
</tr>
<td>&nbsp;</td>
<td><input type="submit" name="kirim" value="Kirim"> </td>
</tr>
</table>
</form>
</body>
</html>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **tambah_data.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



7. Proses Menambahkan Data, salin coding dibawah ini :

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Membuat Buku Tamu</title>
</head>
<body>
  <?php
    if(isset($_POST["kirim"])){
      $nama = $_POST["nama"];
      $email = $_POST["email"];
      $url = $_POST["url"];
      $komentar = $_POST["komentar"];
      require("koneksi.php");
      $sql="insert into bukutamu(nama,
      email, ". "url,komentar)". "values('$nama','$email','$url', ".'"$komentar')";

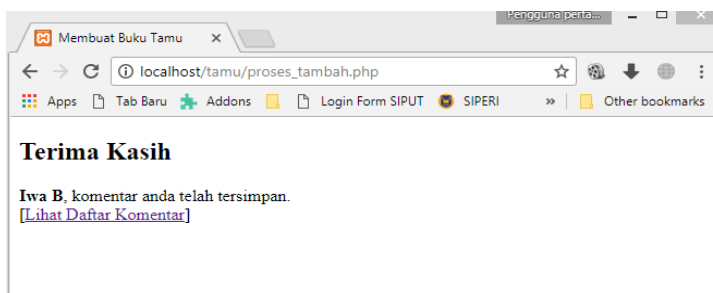
```

```

mysqli_query($conn,$sql);
$num=mysqli_affected_rows($conn);
if ($num > 0){
    ?>
    <h2>Terima Kasih</h2>
    <strong><?php echo $nama; ?></strong>,
    komentar anda telah tersimpan.<br/>
    [<a href="tampil_data.php">Lihat Daftar Komentar</a>]
    <br/> <br/>
    <?php
}else{
    ?>
    <hr>error<h2>
    Proses pencatatan bukutamu gagal. Silahkan Ulangi!
    <br/>
    [<a href="tambah_data.php">Kembali ke Form Pencatatan Buku
    Tamu</a>]
    <?php
}
}
?>
</body>
</html>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **proses_tambah.php** dan isikan form pencatatan buku tamu, kemudian klik kirim, maka pada browser akan muncul hasil sebagai berikut:



8. Menampilkan Daftar Komentar Tamu, salin koding dibawah ini :

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Membuat Buku Tamu</title>
</head>
<body>
    <h2>Daftar Komentar dalam Buku Tamu</h2>
    <?php
    require("koneksi.php");

```

```

$sql="select nama,email,url,komentar"." from bukutamu";
$result=mysqli_query($conn,$sql);
$num=mysqli_num_rows($result);
if($num > 0){
?>
<table>
  <tr bgColor="#ddcc45">
    <th width="140">Nama</th>
    <th width="200">Emal</th>
    <th width="200">URL</th>
    <th width="250">Komentar</th>
  </tr>
  <?php
  while(list($nama,$email,$url,$komentar)=mysqli_fetch_array($result)){
  ?>
  <tr>
    <td Valign="top"><?php echo $nama; ?></td>
    <td Valign="top"><?php echo $email; ?></td>
    <td Valign="top"><a href="?php echo $url; ?>"><?php echo $url; ?></a> </td>
    <td Valign="top"><?php echo $komentar; ?></td>
  </tr>
  <?php
  }
  ?>
</table>
<?php
  }else{
  ?>
  <i>Belum ada komentar.</i>
  <?php
  }
?>
<br/><br/>
<div align="center">
[<a href="tambah_data.php">Isi Buku Tamu</a>]
</div>
</body>
</html>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **tampil_data.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



C. TUGAS PRAKTIKUM

1. Modifikasi Halaman CRUD Siswa dengan HTML dan CSS, serta gunakan form untuk tambah, ubah dan hapus, implementasikan HTML dan CSS.
2. Tugas Project

MODUL XII LANJUTAN DATABASE DALAM PHP

A. TUJUAN :

1. Mahasiswa dapat memahami dan mengaplikasi pembuatan web yang menggunakan database MySQL.
2. Mahasiswa dapat memahami dan melakukan koneksi database MySQL dengan web menggunakan bahasa pemrograman PHP.
3. Mahasiswa dapat menampilkan record data menggunakan bahasa pemrograman PHP.
4. Mahasiswa dapat menambah, mengubah dan menghapus record data dalam database menggunakan bahasa pemrograman PHP.

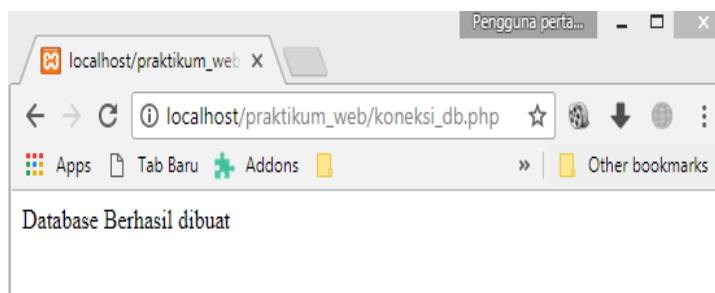
B. KEGIATAN PRAKTIKUM

Membuat CRUD data siswa

1. Buka direktori **C > xampp > htdocs**, kemudian buat folder dan beri nama **“siswa”**.
2. Membuat database **db_siswa**, salin koding dibawah ini :

```
<?php
    $conn = mysqli_connect('localhost','root');
    if(mysqli_connect_errno()){
        echo"koneksi ke sever gagal";
    }
    $sql = "CREATE DATABASE db_siswa";
    if(mysqli_query($conn, $sql))
        { echo "Database Berhasil dibuat";
        } else{ echo "Gagal membuat Database
        :".mysqli_error($conn); }
    mysqli_close($conn);
?>
```

Simpan coding yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > siswa**, dan beri nama file dengan **“buat_db.php”**, klik Save. Jalankan di browser, pada browser akan muncul hasil sebagai berikut:



3. Membuat koneksi mysql, salin koding dibawah ini :

```
<?php
    $host = "localhost"; // Nama hostnya
    $username = "root"; // Username
    $password = ""; // Password (Isi jika menggunakan password)
    $database = "db_siswa"; // Nama databasenya
    $connect = mysqli_connect($host, $username, $password, $database); //
    Koneksi ke MySQL
?>
```

Simpan coding yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > siswa**, dan beri nama file dengan **"koneksi.php"**, klik Save.

4. Membuat tabel datasiswa, salin koding dibawah ini :

```
<?php
$dbname='db_kampus';
$host='localhost';
$password='';
$username='root';
    //Koneksi Ke MySQL
    $cnn = mysqli_connect($host,$username,$password,$dbname);

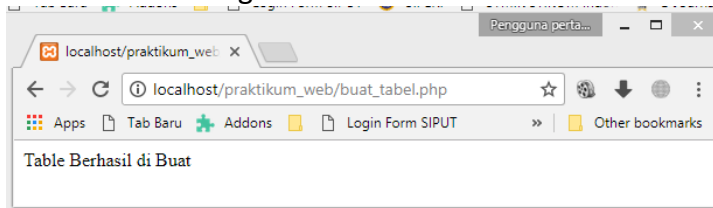
    //Membuat Koneksi
    if(!$cnn){
        die("Koneksi Failed : ".mysqli_connect_error()); }

    //Membuat Tabel
    $sql ="CREATE TABLE datasiswa (
        nim CHAR(10) NULL,
        nama VARCHAR(25) Null,
        alamat VARCHAR(50) Null,
        telp VARCHAR(15) Null,
        constraint pk_datasiswa primary key(nim)
    )";

    if (mysqli_query($cnn, $sql)){
        echo "Table Berhasil di Buat";
    } else {
        echo "Table Gagal di Buat :".mysqli_error($cnn); }
    mysqli_close($cnn);
?>
```

Simpan coding yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > siswa**, dan beri nama

file dengan “**buat_tabel.php**”, klik Save. Jalankan di browser, pada browser akan muncul hasil sebagai berikut:



5. Membuat halaman index, salin koding dibawah ini :

```
<html>
  <head>
    <title>CRUD Data Siswa</title>
  </head>
  <body>
    <h1>Data Siswa</h1>
    <a href="form_simpan.php">Tambah Data</a><br><br>
    <table border="1" width="100%">
      <tr>
        <th>Foto</th>
        <th>NIS</th>
        <th>Nama</th>
        <th>Jenis Kelamin</th>
        <th>Telepon</th>
        <th>Alamat</th>
        <th colspan="2">Aksi</th>
      </tr>

      <?php
        // Load file koneksi.php
        include "koneksi.php";

        $query = "SELECT * FROM datasiswa"; // Query untuk menampilkan semua
        data siswa
        $sql = mysqli_query($connect, $query); // Eksekusi/Jalankan query dari
        variabel $query

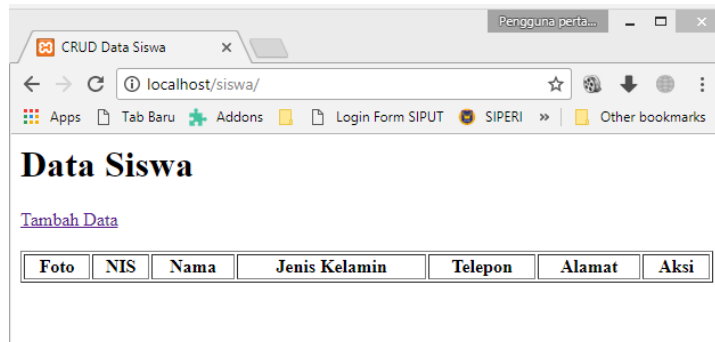
        while($data = mysqli_fetch_array($sql)){
          // Ambil semua data dari hasil eksekusi $sql
          echo "<tr>";
          echo "<td><img src='images/'.".$data['foto']."' width='100' height='100'></td>";
          echo "<td>".$data['nis']."</td>";
          echo "<td>".$data['nama']."</td>";
          echo "<td>".$data['jenis_kelamin']."</td>";
          echo "<td>".$data['telp']."</td>";
          echo "<td>".$data['alamat']."</td>";
          echo "<td><a href='form_ubah.php?nis=".$data['nis']."'>Ubah</a></td>";
          echo "<td><a href='proses_hapus.php?nis=".$data['nis']."'>Hapus</a></td>";
```

```

        echo "</tr>";
    }
?>
</table>
</body>
</html>

```

Simpan coding yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > siswa**, dan beri nama file dengan **"index.php"**, klik Save. Jalankan di browser, pada browser akan muncul hasil sebagai berikut:



6. Membuat form tambah data, salin koding dibawah ini :

```

<html>
<head>
    <title>CRUD Data Siswa</title>
</head>
<body>
    <h1>Tambah Data Siswa</h1>
    <form method="post" action="proses_simpan.php" enctype="multipart/form-data">
    <table cellpadding="8">
    <tr>
        <td>NIS</td>
        <td><input type="text" name="nis"></td>
    </tr>
    <tr>
        <td>Nama</td>
        <td><input type="text" name="nama"></td>
    </tr>
    <tr>
        <td>Jenis Kelamin</td>
        <td>
            <input type="radio" name="jenis_kelamin" value="Laki-laki"> Laki-laki
            <input type="radio" name="jenis_kelamin" value="Perempuan"> Perempuan
        </td>
    </tr>
    <tr>
        <td>Telepon</td>

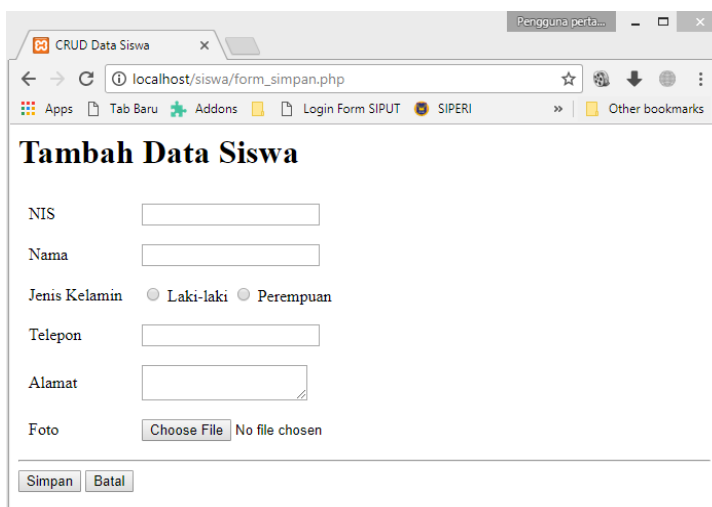
```

```

        <td><input type="text" name="telp"></td>
</tr>
<tr>
        <td>Alamat</td>
        <td><textarea name="alamat"></textarea></td>
</tr>
<tr>
        <td>Foto</td>
        <td><input type="file" name="foto"></td>
</tr>
</table>
<hr>
<input type="submit" value="Simpan">
<a href="index.php"><input type="button" value="Batal"></a>
</form>
</body>
</html>

```

Simpan coding yang telah dituliskan dengan membuka menu **file > save as...** pilih lokasi penyimpanannya pada direktori **C > xampp > htdocs > siswa**, dan beri nama file dengan **"index.php"**, klik Save. Jalankan di browser atau klik tambah data, pada browser akan muncul hasil sebagai berikut:



7. Membuat proses simpan, salin koding dibawah ini :

```

<?php
// Load file koneksi.php
include "koneksi.php";
// Ambil Data yang Dikirim dari Form
$nis = $_POST['nis'];
$nama = $_POST['nama'];
$jenis_kelamin = $_POST['jenis_kelamin'];
$telp = $_POST['telp'];
$alamat = $_POST['alamat'];

```

```

$foto = $_FILES['foto']['name'];
$tmp = $_FILES['foto']['tmp_name'];

// Rename nama fotonya dengan menambahkan tanggal dan jam upload
$fotobaru = date('dmYHis').$foto;

// Set path folder tempat menyimpan fotonya
$path = "images/".$fotobaru;

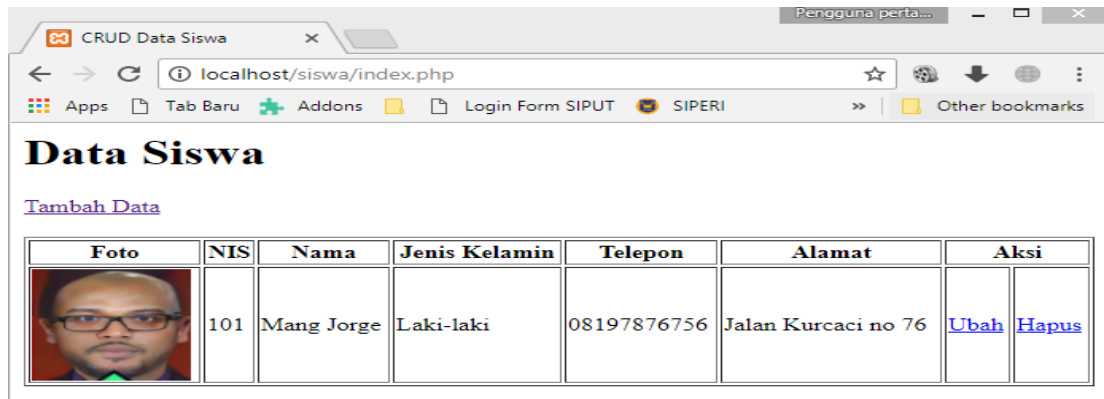
// Proses upload
if(move_uploaded_file($tmp, $path)){ // Cek apakah gambar berhasil diupload
atau tidak

// Proses simpan ke Database
$query = "INSERT INTO datasiswa VALUES('".$nis."', '".$nama."',
".$jenis_kelamin."', '".$stelp."', '".$alamat."', '".$fotobaru."')";
$sql = mysqli_query($connect, $query); // Eksekusi/ Jalankan query dari variabel
$query

if($sql){ // Cek jika proses simpan ke database sukses atau tidak
// Jika Sukses, Lakukan :
header("location: index.php"); // Redirect ke halaman index.php
}else{
// Jika Gagal, Lakukan :
echo "Maaf, Terjadi kesalahan saat mencoba untuk menyimpan data ke
database.";
echo "<br><a href='form_simpan.php'>Kembali Ke Form</a>";
}
}else{
// Jika gambar gagal diupload, Lakukan :
echo "Maaf, Gambar gagal untuk diupload.";
echo "<br><a href='form_simpan.php'>Kembali Ke Form</a>";
}
?>

```

Simpan coding pada direktori **C > xampp > htdocs > siswa**, dan beri nama file dengan **“proses_simpan.php”**, klik Save. Jalankan di browser atau klik simpan, pada browser akan muncul hasil sebagai berikut:



8. Membuat form ubah, salin koding dibawah ini :

```

<html>
<head>
  <title>CRUD Data Siswa</title>
</head>
<body>
  <h1>Ubah Data Siswa</h1>
  <?php
  // Load file koneksi.php
  include "koneksi.php";

  // Ambil data NIS yang dikirim oleh index.php melalui URL
  $nis = $_GET['nis'];

  // Query untuk menampilkan data siswa berdasarkan NIS yang dikirim
  $query = "SELECT * FROM datasiswa WHERE nis='".$nis."'";
  $sql = mysqli_query($connect, $query); // Eksekusi/Jalankan query dari variabel $query
  $data = mysqli_fetch_array($sql); // Ambil data dari hasil eksekusi $sql
  ?>

  <form method="post" action="proses_ubah.php?nis=<?php echo $nis; ?>"
  enctype="multipart/form-data">
  <table cellpadding="8">
  <tr>
    <td>Nama</td>
    <td><input type="text" name="nama" value="<?php echo $data['nama']; ?>"></td>
  </tr>
  <tr>
    <td>Jenis Kelamin</td>
    <td>
      <?php
      if($data['jenis_kelamin'] == "Laki-laki"){
        echo "<input type='radio' name='jenis_kelamin' value='laki-laki'
        checked='checked'> Laki-laki";
        echo "<input type='radio' name='jenis_kelamin' value='perempuan'>
        Perempuan";
      }
    </td>
  </tr>
  </table>
  </form>
  
```

```

    }else{
    echo "<input type='radio' name='jenis_kelamin' value='laki-laki'> Laki-
laki";
    echo "<input type='radio' name='jenis_kelamin' value='perempuan'
checked='checked'> Perempuan";
    }
    ?>
</td>
</tr>
<tr>
    <td>Telepon</td>
    <td><input type="text" name="telp" value="<?php echo $data['telp']; ?>"></td>
</tr>
<tr>
    <td>Alamat</td>
    <td><textarea name="alamat"><?php echo $data['alamat']; ?></textarea></td>
</tr>
<tr>
    <td>Foto</td>
    <td>
    <input type="checkbox" name="ubah_foto" value="true"> Ceklis jika ingin
mengubah foto<br>
    <input type="file" name="foto"> </td>
</tr>
</table>
<hr>
<input type="submit" value="Ubah">
<a href="index.php"><input type="button" value="Batal"></a>
</form>
</body>
</html>

```

Simpan coding pada direktori **C > xampp > htdocs > siswa**, dan beri nama file dengan **“form_ubah.php”**, klik Save.

9. Membuat proses ubah, salin koding dibawah ini :

```

<?php
// Load file koneksi.php
include "koneksi.php";

// Ambil data NIS yang dikirim oleh form_ubah.php melalui URL
$nis = $_GET['nis'];

// Ambil Data yang Dikirim dari Form
$nama = $_POST['nama'];
$jenis_kelamin = $_POST['jenis_kelamin'];
$telp = $_POST['telp'];

```



```

$alamat = $_POST['alamat'];

// Cek apakah user ingin mengubah fotonya atau tidak
if(isset($_POST['ubah_foto'])){ // Jika user menceklis checkbox yang ada di form
ubah, lakukan :
    // Ambil data foto yang dipilih dari form
    $foto = $_FILES['foto']['name'];
    $tmp = $_FILES['foto']['tmp_name'];

    // Rename nama fotonya dengan menambahkan tanggal dan jam upload
    $fotobaru = date('dmYHis').$foto;

    // Set path folder tempat menyimpan fotonya
    $path = "images/".$fotobaru;

    // Proses upload
    if(move_uploaded_file($tmp, $path)){
        // Cek apakah gambar berhasil diupload atau tidak
        // Query untuk menampilkan data siswa berdasarkan NIS yang dikirim
        $query = "SELECT * FROM datasiswa WHERE nis='".$nis."'";
        $sql = mysqli_query($connect, $query);
        // Eksekusi/Jalankan query dari variabel $query
        $data = mysqli_fetch_array($sql);
        // Ambil data dari hasil eksekusi $sql

        // Cek apakah file foto sebelumnya ada di folder images
        if(is_file("images/".$data['foto'])) // Jika foto ada
            unlink("images/".$data['foto']);
        // Hapus file foto sebelumnya yang ada di folder images
        // Proses ubah data ke Database
        $query = "UPDATE datasiswa SET nama='".$nama."',
        jenis_kelamin='".$jenis_kelamin."', telp='".$telp."',
        alamat='".$alamat."', foto='".$fotobaru.'" WHERE nis='".$nis.'";
        $sql = mysqli_query($connect, $query); // Eksekusi/ Jalankan query
        dari variabel $query

        if($sql){ // Cek jika proses simpan ke database sukses atau tidak
            // Jika Sukses, Lakukan :
            header("location: index.php"); // Redirect ke halaman index.php
        }else{
            // Jika Gagal, Lakukan :
            echo "Maaf, Terjadi kesalahan saat mencoba untuk menyimpan
            data ke database.";
            echo "<br><a href='form_ubah.php'>Kembali Ke Form</a>";
        }
    }
}
// Jika gambar gagal diupload, Lakukan :

```

```

        echo "Maaf, Gambar gagal untuk diupload.";
        echo "<br><a href='form_ubah.php'>Kembali Ke Form</a>";
    }
} else { // Jika user tidak menceklis checkbox yang ada di form ubah, lakukan :

// Proses ubah data ke Database
$query = "UPDATE datasiswa SET nama='".$nama."',
jenis_kelamin='".$jenis_kelamin."', telp='".$telp."', alamat='".$alamat.'" WHERE
nis='".$nis.'";
$sql = mysqli_query($connect, $query); // Eksekusi/ Jalankan query dari variabel
$query

if($sql){ // Cek jika proses simpan ke database sukses atau tidak
    // Jika Sukses, Lakukan :
    header("location: index.php"); // Redirect ke halaman index.php
} else {
    // Jika Gagal, Lakukan :
    echo "Maaf, Terjadi kesalahan saat mencoba untuk menyimpan data ke
database.";
    echo "<br><a href='form_ubah.php'>Kembali Ke Form</a>";
}
}
?>

```

Simpan coding pada direktori **C > xampp > htdocs > siswa**, dan beri nama file dengan **“proses_ubah.php”**, klik Save. Jalankan di browser atau klik aksi ubah, pada browser akan muncul hasil sebagai berikut:

The screenshot shows a web browser window with the title 'CRUD Data Siswa'. The address bar shows 'localhost/siswa/form_ubah.php?nis=101'. The main content area has the heading 'Ubah Data Siswa' and a form with the following elements:

- Nama:** Text input field containing 'Mang Jorge'.
- Jenis Kelamin:** Radio button group with 'Laki-laki' selected and 'Perempuan' unselected.
- Telepon:** Text input field containing '08197876756'.
- Alamat:** Text input field containing 'Jalan Kurcaci no 76'.
- Foto:** A checkbox labeled 'Ceklis jika ingin mengubah foto' which is unchecked, and a 'Choose File' button next to it showing 'No file chosen'.

At the bottom of the form are two buttons: 'Ubah' and 'Batal'.

10. Melakukan proses hapus, salin koding dibawah ini :

```
<?php
// Load file koneksi.php
include "koneksi.php";

// Ambil data NIS yang dikirim oleh index.php melalui URL
$nis = $_GET['nis'];

// Query untuk menampilkan data siswa berdasarkan NIS yang dikirim
$query = "SELECT * FROM datasiswa WHERE nis='".$nis."'";
$sql = mysqli_query($connect, $query); // Eksekusi/Jalankan query dari
variabel $query
$data = mysqli_fetch_array($sql); // Ambil data dari hasil eksekusi $sql

// Cek apakah file fotonya ada di folder images
if(is_file("images/".$data['foto'])) // Jika foto ada
    unlink("images/".$data['foto']); // Hapus foto yang telah diupload dari
folder images

// Query untuk menghapus data siswa berdasarkan NIS yang dikirim
$query2 = "DELETE FROM datasiswa WHERE nis='".$nis."'";
$sql2 = mysqli_query($connect, $query2); // Eksekusi/Jalankan query dari
variabel $query

if($sql2){ // Cek jika proses simpan ke database sukses atau tidak
    // Jika Sukses, Lakukan :
    header("location: index.php"); // Redirect ke halaman index.php
}else{
    // Jika Gagal, Lakukan :
    echo "Data gagal dihapus. <a href='index.php'>Kembali</a>";
}
?>
```

Simpan coding pada direktori **C > xampp > htdocs > siswa**, dan beri nama file dengan **“proses_hapus.php”**, klik Save. Jalankan di browser atau klik aksi hapus, pada browser akan muncul halaman index dan data siswa kosong, sehingga anda bisa melakukan kembali proses tamba, ubah data.

C. TUGAS PRAKTIKUM

1. Buatlah Login untuk Siswa
2. Modifikasi Halaman CRUD Siswa dengan HTML dan CSS, serta gunakan form untuk tambah, ubah dan hapus, implementasikan HTML dan CSS.
3. Tugas Project

MODUL XIII PEMROGRAMAN BERORIENTASI OBYEK DALAM PHP

A. TUJUAN :

1. Mahasiswa dapat memahami pemrograman berorientasi objek dalam pemrograman PHP dan dapat mengaplikasikannya dalam pembuatan web.
2. Mahasiswa dapat memahami object dan class dan dapat mengaplikasikannya dalam pembuatan web menggunakan bahasa pemrograman PHP
3. Mahasiswa dapat memahami properti dan method yang ada di pemrograman PHP berorientasi objek dan dapat mengaplikasikannya dalam pembuatan web.

B. DASAR TEORI

Pemrograman berorientasi objek atau object-oriented programming (OOP) adalah pemrograman yang didasarkan pada objek dunia nyata. Objek-objek dalam OOP memiliki proses yang berbeda-beda. Adapun konsep dasar OOP ialah:

- **Encapsulation** (pengkapsulan) adalah penyatuan antara proses-proses yang terjadi dan sejumlah data yang dikelola kedalam suatu bentuk kelompok tertentu.
- **Inheritance** (pewaris) adalah merupakan konsep dimana suatu objek memiliki sejumlah perilaku yang dimiliki oleh objek lain dalam suatu hirarki tertentu.
- **Polymorphism** adalah konsep dimana beberapa objek dapat memiliki metode yang sama dengan aksi yang berbeda.

1. Object dan Class

Object biasanya adalah suatu benda yang dikenai pekerjaan bisa berupa orang, rumah, benda dan lain sebagainya. Object juga bisa berupa benda abstrak seperti account bank, file, data atau variabel yang dapat diproses. Contoh:

```
$andi = new nama_class();
```

Sedangkan class adalah penjelasan atau deskripsi dari object, andaikan bila object adalah sebuah variabel maka class adalah sebuah tipe data. Di dalam class, terdapat penjelasan tentang suatu object termasuk properties yang dimiliki serta kelakuan atau method yang bisa dilakukan oleh object. Dalam pemrograman PHP class dideskripsikan sebagai berikut:

```
Class nama_class
{
    //Sintak deklarasi dan definisi properties dan method
}
```

2. Properti dan Method

Setiap objek memiliki properties yang kadang disebut juga dengan atribut. Contoh sebuah mobil memiliki warna, merk, jenis dan sebagainya. Property di letakkan di dalam class, dimana properties dinyatakan dengan sebuah variabel dan diawali

dengan visibility keyword (seperti “public”, “protected”, “private”) atau kata “var”, misal \$warna, \$merk, \$jenis dan sebagainya. Contoh:

```
Class Mobil{
    Var $warna;
    Var $merk;
    Var $jenis;
}
```

Sedangkan method merupakan suatu kemampuan yang dimiliki sebuah objek untuk melaksanakan tugas, contoh sebuah mobil dapat bergerak maju, bergerak mundur, mengisi bensin dan sebagainya. Contoh:

```
Class Mobil{
    Var $bensin;

    function maju(){
        $this->bensin = $this->bensin - 2; // menseting properti
    }

    function mundur(){
        $this->bensin = $this->bensin - 1; // menseting properti
    }

    function isibensin($tambah){
        $this->bensin = $this->bensin + $tambah;
    }
    function sisabensin(){
        return $this->bensin;
    }
}
```

Untuk memanggil method dari sebuah kelas, kita hanya perlu menggunakan tanda ->. Contoh :

```
$mobil1 = new Mobil
$mobil1->isibensin(6); // memanggil method isibensin()
$mobil1->maju(); // memanggil method maju()
$mobil1->mundur(); // memanggil method mundur()
```

Dengan adanya pemrograman berorientasi objek atau OOP dalam PHP, kita dapat lebih mudah berbagi teknik pemrograman. Kita dapat membuat suatu class dan programmer lain dapat dengan mudah menggunakannya tanpa perlu mengetahui proses jalannya class tersebut.

C. KEGIATAN PRAKTIKUM

1. Program Class Mobil, salin koding dibawah ini :

```
<?php
class Mobil{
    var $bensin;

    function maju(){

        //aksi maju menghabiskan 2 liter
        $this->bensin = $this->bensin - 2;
    }

    function mundur(){

        //aksi mundur menghabiskan 1 liter
        $this->bensin = $this->bensin - 1;
    }

    function isibensin($tambah){

        $this->bensin = $this->bensin + $tambah;
    }

    function sisabensin(){
        return $this->bensin;
    }
}

$mobil1 = new Mobil();
$mobil1->isibensin(6);

echo "Mobil melakukan isi bensin. Sisa bensin : ",$mobil1->sisabensin(),"
Liter<br/>";
$mobil1->maju();

echo "Mobil melakukan aksi Maju. Sisa Bensin : ",$mobil1->sisabensin(),"
Liter<br/>";

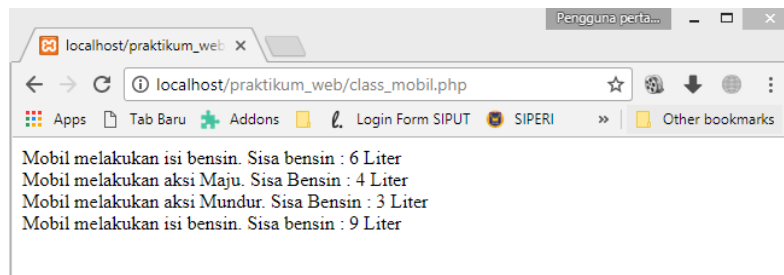
$mobil1->mundur();

echo "Mobil melakukan aksi Mundur. Sisa Bensin : ",$mobil1->sisabensin(),"
Liter<br/>";

$mobil1->isibensin(6);

echo "Mobil melakukan isi bensin. Sisa bensin : ",$mobil1->sisabensin(),"
Liter<br/>";
?>
```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **class_mobil.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



2. Program Class Biodata, salin koding dibawah ini :

```
<?php
class Biodata{
    var $nama;

    function setnama($namamhs){
        $this->nama = $namamhs;
    }

    function getnama(){
        echo "Nama : ", $this->nama;
    }

    function setnim($nimmhs){
        $this->nim = $nimmhs;
    }

    function getnim(){
        echo "NIM : ", $this->nim;
    }

    function setalamat($alamatmhs){
        $this->alamat = $alamatmhs;
    }

    function getalamat(){
        echo "Alamat : ", $this->alamat;
    }

    function settgllahir($tgllahirmhs){
        $this->tgllahir = $tgllahirmhs;
    }

    function gettgllahir(){
```

```

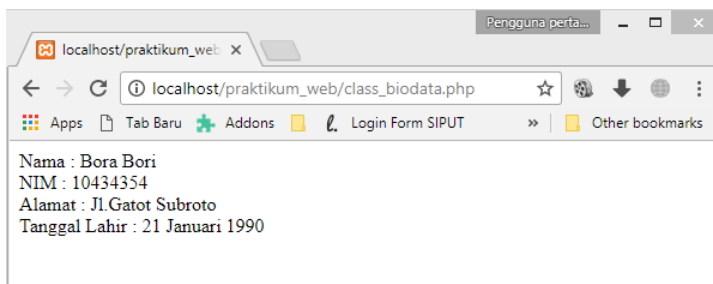
        echo "Tanggal Lahir : ", $this->tglahir;
    }

}

$mhs1 = new Biodata();
$mhs1->setnama("Dwi Putri Rahayu");
echo $mhs1->getnama()."</br>";
$mhs1->setnim("10101259");
echo $mhs1->getnim()."</br>";
$mhs1->setalamat("Jl.Raya Pemogan");
echo $mhs1->getalamat()."</br>";
$mhs1->settglahir("21 Januari 1992");
echo $mhs1->gettglahir()."</br>";
?>

```

Setelah selesai mengetik coding di atas, maka simpan dengan nama **class_biodata.php** dan jalankan di browser. Maka pada browser akan muncul hasil sebagai berikut:



D. TUGAS

Buatlah program yang didalamnya terdapat class untuk membuat form input sederhana mengenai daftar mahasiswa dikelas anda.