BAB I
PENDAHULUAN

LATAR BELAKANG

World Wide Web (WWW) telah berubah dengan cepat dalam berbagai cara. Bahasa standar yang dipakai dalam WWW adalah bahasa HTML (Hyper Text Markup Language). Dengan adanya teknologi ini, kita dapat melihat dokumen yang kadang-kadang berada di bagian lain di dunia ini dengan hanya sebuah program sederhana dan kita akan banyak mendapatkan informasi yang dengan cepat dapat kita terima sesuai dengan yang kita perlukan.

Sekarang kita mengharapkan lebih dari internet. Kita mengharapkan sebuah situs web yang berisi dengan informasi yang banyak dan dalam tampilan yang menarik serta kita dapat mencari dokumen yang diperlukan dengan mudah. Untuk untuk mencapai hal tersebut kita memerlukan suatu situs web yang dinamis. Karena jika hanya mengandalkan situs web yang statis maka akan memerlukan pemeliharaan yang susah, sebagai contoh jika sebuah perusahaan ingin membuat suatu situs web yang berisi promosi produk sebanyak 1000 jenis produk maka dalam situs tersebut minimal kita harus membuat 1000 web statis yang berbeda (itu pasti merepotkan). Dengan memanfaatkan teknologi situs dinamis maka dalam situs tersebut hanya memerlukan halaman web yang lebih sedikit karena data produk disimpan dalam database.

Sebuah situs web yang modern sekarang bukan hanya sebuah web server, juga harus dapat menyertakan suatu jalan untuk menyimpan data dan melayani permintaan (seperti SQL query).

CONTOH KASUS

UNIKOM memiliki situs http://nilaionline.unikom.ac.id. Dalam situs tersebut seorang mahasiswa dapat melihat nilai yang didapatnya di semester sekarang. Seorang mahasiswa hanya perlu mengisikan NIM di form yang telah disediakan kemudian klik tombol pencarian nilai yang akan menampilkan data nilai mahasiswa tersebut. Data yang tersedia di situs tersebut akan selalu berubah tiap semesternya, oleh karena itu merupakan hal yang merepotkan jika kita membuat halaman web sebanyak banyak mahasiswa yang terdaftar (1 halaman web berisi nilai seorang

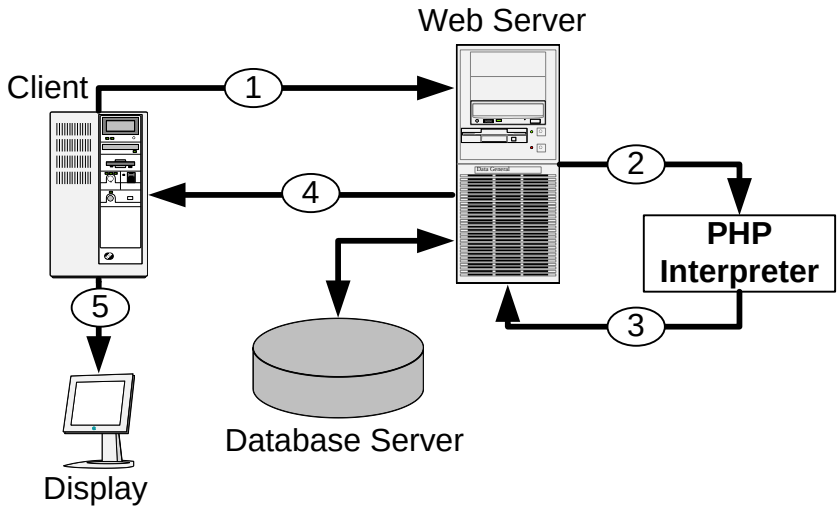
mahasiswa). Oleh karena itu data nilai mahasiswa di simpan dalam server database yang akan diupdate setelah nilai dihitung.

Akan tetapi jika kita menggunakan halaman web yang menggunakan HTML saja, akan sulit jika kita ingin mengakses suatu database, sehingga kita memerlukan suatu script yang dapat mengakses suatu database (dimana kita menyimpan data), tetapi harus dapat menghasilkan sebuah halaman HTML (karena kita memanfaatkan web).

Karena alasan diataslah PHP diciptakan. PHP memiliki fungsi-fungsi yang banyak yang dapat mengakses database.

CARA KERJA PHP

Adapun cara kerja PHP adalah :



Gambar 1. Alur kerja PHP

Proses-proses yang terjadi adalah :

1. Client me-request halaman web yang berisi script PHP.
2. Jika file yang di-request ditemukan, maka server akan meneruskannya ke PHP Interpreter (Penterjemah PHP) yang akan bekerja menghasilkan dokumen/halaman HTML berdasarkan script PHP. Jika dalam script tersebut terdapat permintaan terhadap database, maka akan terjadi proses query data ke database server.
3. Dokumen HTML hasil interpretasi oleh PHP Interpreter dikembalikan ke web server

4. Web Server mengirim dokumen HTML (Response) ke Client
5. Web Browser akan menterjemahkan dokumen HTML ke Display/Monitor.

Dari gambar di atas, dapat diambil kesimpulan bahwa untuk membangun sebuah situs yang dinamis dan interaktif, maka ada beberapa hal yang diperlukan yaitu Web Server, Database Server dan sebuah Script Interpreter (dalam hal ini PHP Interpreter).

BAB II
MEMULAI PHP

- **Persyaratan**
- **Program Pertama**
- **Mengirim Nilai Ke Variable**
- **Latihan**

PERSYARATAN

Seperti yang diterangkan di bab sebelumnya, jika ingin membangun sebuah situs dinamis maka ada 3 hal yang diperlukan yaitu Web Server, Database Server dan PHP Interpreter.

Web Server yang dapat digunakan diantaranya adalah Apache, Internet Information Services (IIS), Personal Web Server (PWS) dan lain-lain.

Databas Server yang dapat digunakan diantaranya adalah MySQL, Oracle, Interbase, Informix, SQL Server, PostgreSQL dan lain-lain.

Setiap aplikasi tersebut dapat diinstall secara terpisah. Akan tetapi ada sebuah paket yang didalamnya sudah menyatukan ketiga aplikasi tersebut yaitu WAMP yang kependekan dari Windows + Apache + MySQL + PHP. (Dapat ditemukan di CD Penunjang Perkuliahan).

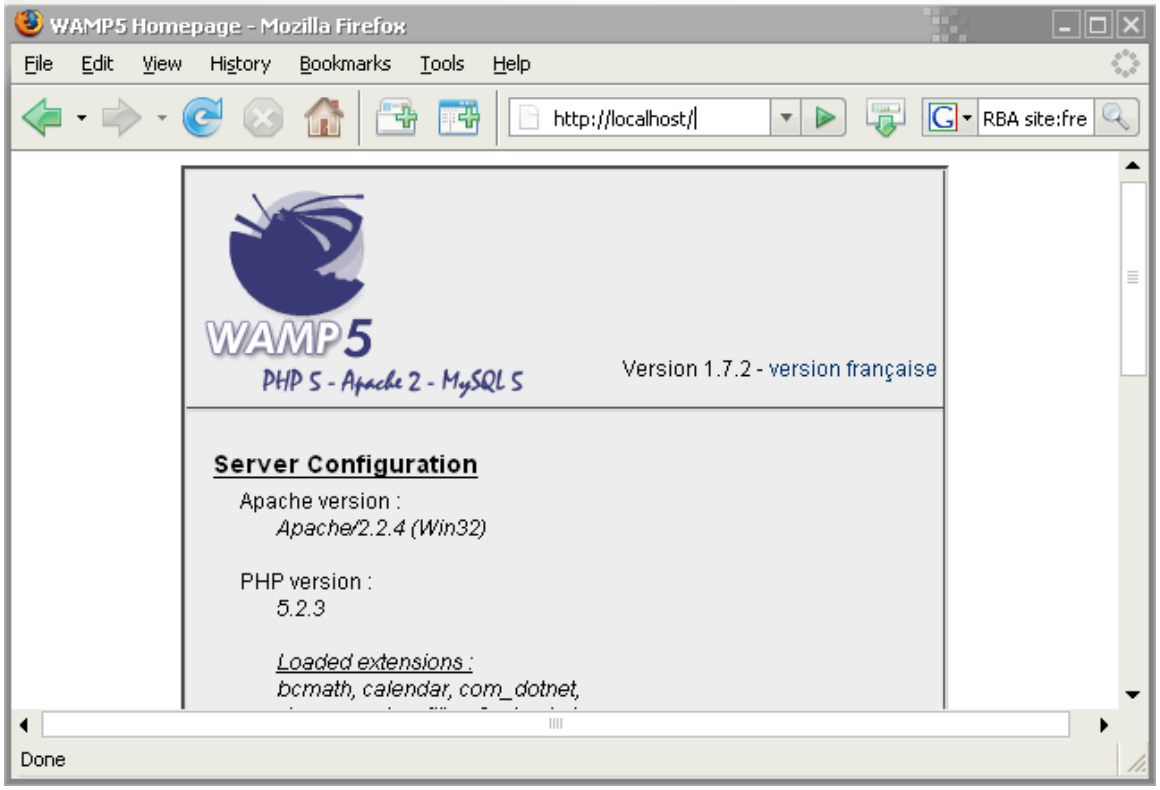
Oleh karena itu dalam komputer yang dimiliki harus terinstall minimal **WAMP**.

Untuk membuat suatu file PHP, program yang diperlukan adalah sebuah editor teks seperti Notepad, Ultra Edit, PHP Coder, PHP Editor atau Macromedia Dreamweaver.

Sekarang diasumsikan bahwa anda menggunakan WAMP. Untuk memulai pembuatan situs dinamis berbasis PHP, ada beberapa hal yang harus dikerjakan yaitu :

1. Web server harus dalam keadaan aktif caranya dengan mengklik icon WAMP yang ada di tray icon kemudian klik **Apache → Start/Resume Service**.

2. Untuk mengetes apakah web server sedang aktif, maka buka sebuah browser (Internet Explorer, Opera atau FireFox) kemudian ketik di toolbar address alamat berikut : http://localhost yang akan menampilkan halaman seperti di bawah ini.



3. Web site yang muncul di web browser sebenarnya berada di folder **c:\wamp\www**. Sehingga jika anda membuat file php dan ingin dapat dieksekusi, maka harus disimpan di folder tersebut. Anda dapat memindahkan folder web ke lokasi lain dengan syarat mengatur konfigurasi apache (yang ada pada file **httpd.conf**).
4. Untuk menjalankan suatu file php yang ada dalam c:\wamp\www adalah dengan menulis alamat file tersebut seperti http://localhost/namafile.php.

PROGRAM PERTAMA

Contoh kasus : Buatlah suatu halaman web yang dapat menampilkan bilangan 1 sampai 1000.

Solusi 1 : Dengan menggunakan HTML

```
<html>
<head>
       <title>Deret Angka</title>
</head>
<body>
    1<br>
    2<br>
    ...
    ...
    ...
    999<br>
    1000<br>
</body>
</html>
```

Jika solusi 1 dikerjakan, maka kita membutuhkan banyak baris. Tentunya akan merepotkan jika bilangan yang diinginkan adalah dari 1 sampai 100000. Jika kita mengandalkan bahasa HTML murni maka akan sulit.

Solusi 2 : Dengan menggunakan PHP

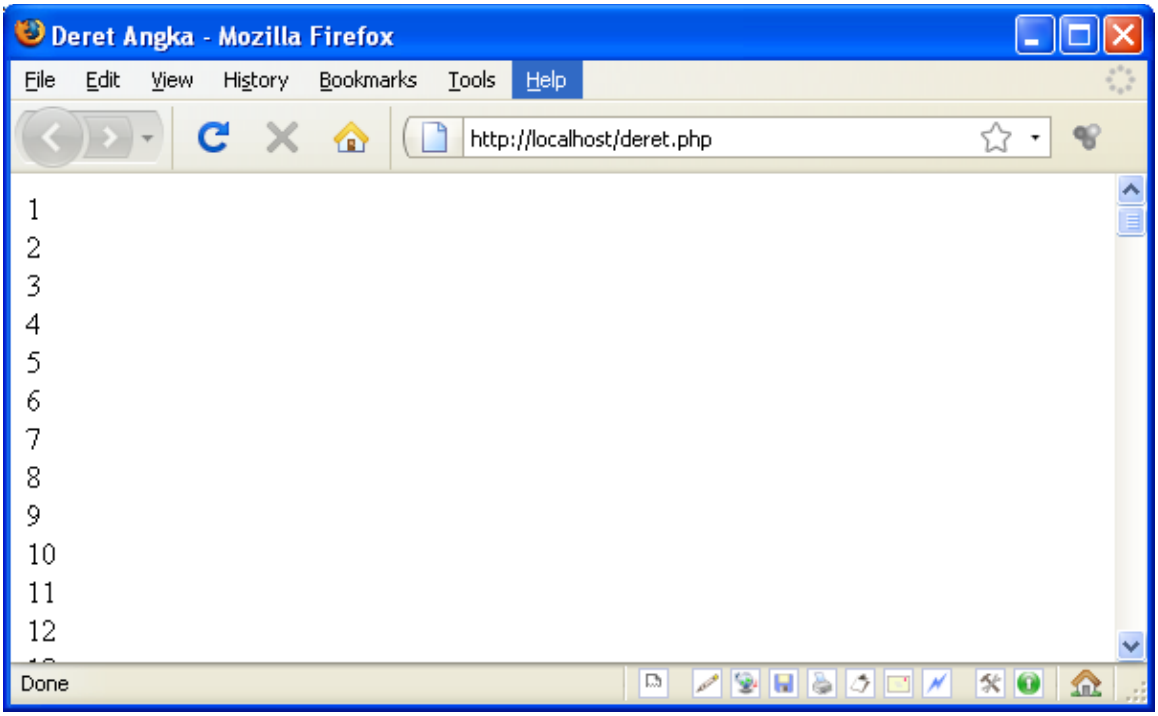
```
 1 <html>
 2 <head>
 3        <title>Deret Angka</title>
 4 </head>
 5 <body>
 6       <?php
 7            for($i=1;$i<=1000;$i++)
 8               echo "$i<br>";
 9       ?>
10 </body>
11 </html>
```

Script PHP disisipkan dalam sebuah file HTML.

Keterangan program :

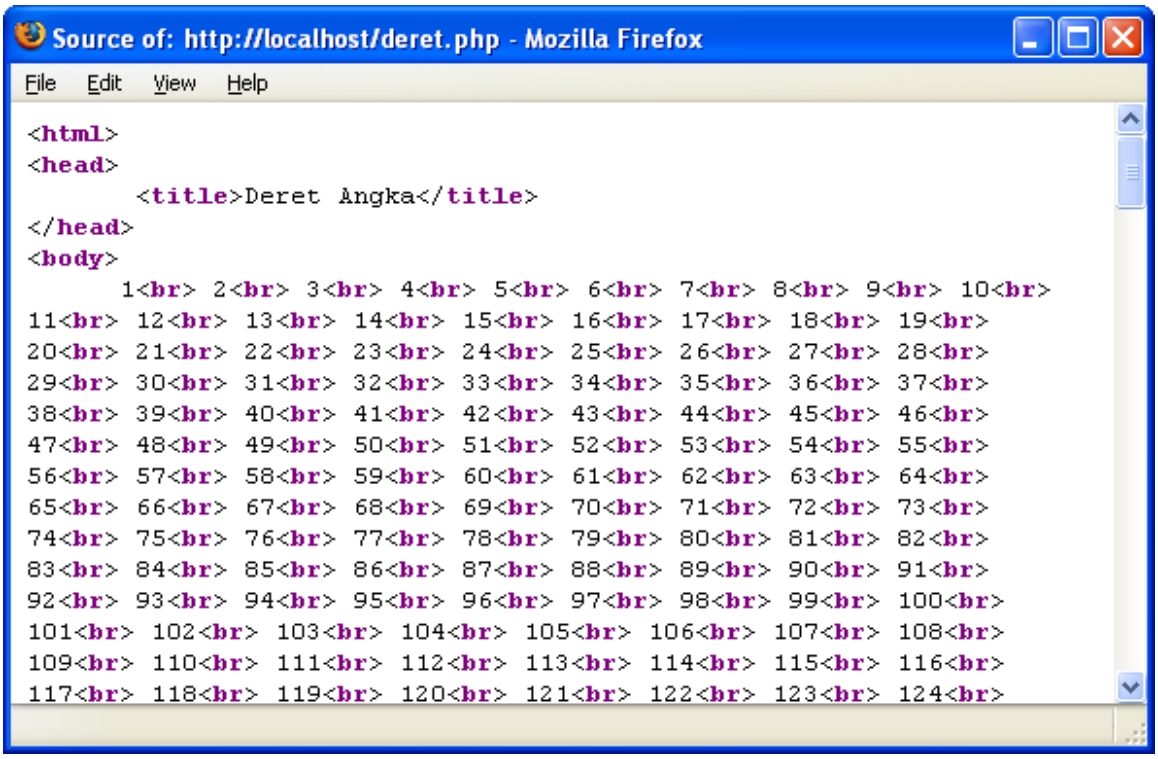
- Baris 1 – 5 : Awal dari HTML
- Baris 6 : Awal dari script php dimulai dengan tag **<?php**.
- Baris 7 – 8 : Perulangan dari 1 sampai 1000 untuk menghasilkan deret angka 1 sampai 1000.
- Baris 9 : Akhir dari script php diakhiri dengan tab **?>**.
- Baris 10 – 11 : Akhir dari HTML

Simpan fiel tersebut di dalam folder htdocs dengan nama file deret.php. Kemudian eksekusi file tersebut di web browser dengan alamat : http://localhost/deret.php yang akan menampilkan deret angka dari 1 sampai 1000 seperti gambar di bawah ini.



Hasil Eksekusi file deret.php

Untuk membuktikan bahwa sebuah script php menghasilkan dokumen html, maka pada web browser (contoh Internet Explorer) klik menu View → Source yang akan menampilkan sumber HTML dokumen tersebut seperti di bawah ini.



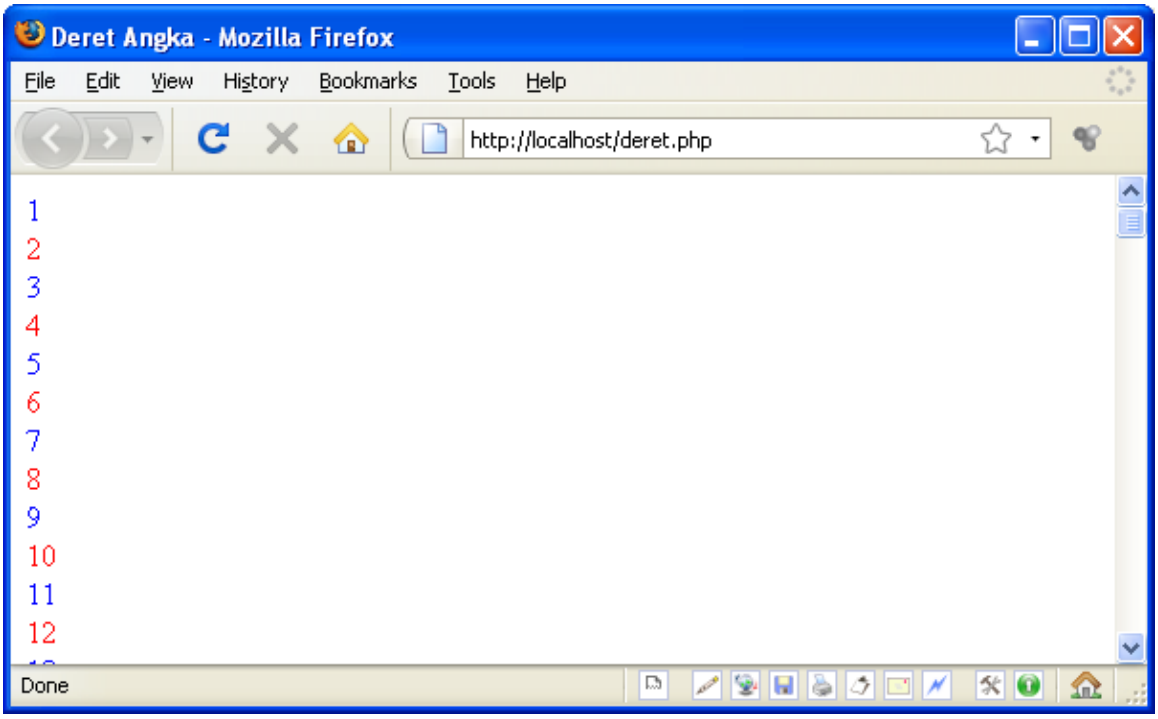
Kode sumber HTML hasil Interpreter PHP.

Sekarang kita modifikasi program di atas agar dapat menampilkan deret angka yang berwarna dengan ketentuan jika bilangan ganjil maka akan ditampilkan dengan warna merah dan jika bilangan tersebut genap maka akan ditampilkan dengan warna biru.

Untuk kasus diatas maka kita benar-benar memerlukan suatu percabangan (statement if), sehingga file deret.php akan berubah menjadi :

```
1  <html>
2  <head>
3         <title>Title here!</title>
4  </head>
5  <body>
6        <?php
7              for($i=1;$i<=1000;$i++)
8              {
9                   if($i % 2 == 0)
10                     echo "<font color=red>$i</font><br>";
11                  else
12                     echo "<font color=blue>$i</font><br>";
13             }
14       ?>
15 </body>
16 </html>
```

Jika file telah diedit, maka sebelum dieksekusi, file tersebut harus disimpan dulu. Kemudian tekan tombol **Refresh** di web browser agar mengeksekusi kembali file tersebut.



MENGIRIMKAN NILAI DARI LUAR PHP

Contoh kasus : File deret.php yang telah ditulis ingin diubah dimana awal perulangannya tidak selalu 1 dan akhir perulanggnya pun tidak ingin selalu 1000 tetapi bisa berapa saja. Oleh karena itu, angka 1000 pada perulangannya harus diganti dengan sesuatu yang bisa berubah yaitu variable sebut saja variable $akhir juga angka 1 diubah menjadi variable $awal. Sehingga file deret.php akan seperti di bawah ini.

```
1  <html>
2  <head>
3         <title>Title here!</title>
4  </head>
5  <body>
6        <?php
7             for($i=$_GET[awal];$i<=$_GET[akhir];$i++)
8             {
9                  if($i % 2 ==0)
10                    echo "<font color=red>$i</font><br>";
11                 else
12                    echo "<font color=blue>$i</font><br>";
```

```
13              }
14        ?>
15  </body>
16  </html>
```

Ada beberapa cara yang dipakai untuk mengirimkan sebuah nilai ke sebuah variable dalam bahasa PHP yaitu :

1. Melalui alamat URL

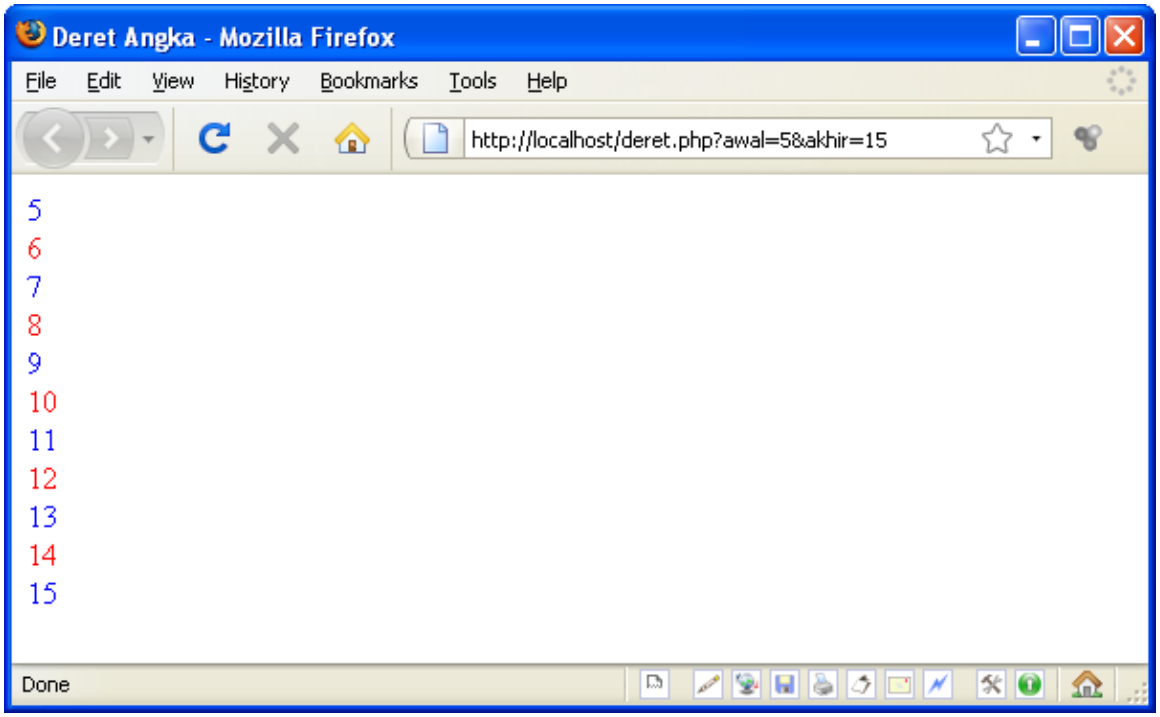
   Contoh : jika ingin mengisi nilai variable yang ada di file deret.php dengan nilai awal=5 dan akhir=15 adalah dengan menulis alamat URL di toolbar alamat browser sebagai berikut :

   http://localhost/deret.php?awal=5&akhir=15

   Keterangan :

   - Karakter ? digunakan sebagai pemisah antara nama file dan variable
   - Karakter & digunakan sebagai pemisah antar variable
   - Nama variable harus sama persis dengan nama variable yang tertulis di test1.php (case sensitive).

   Jika dijalankan, maka akan menghasilkan tampilan HTML seperti di bawah ini.



2. Melalui form dengan method get

   Jika ingin mengirim variable dari suatu form, maka langkah yang harus dilakukan adalah dengan membuat form HTML yang berisi textbox tempat mengisi variable awal dan akhir serta sebuah tombol Submit.

   Contoh form HTML.

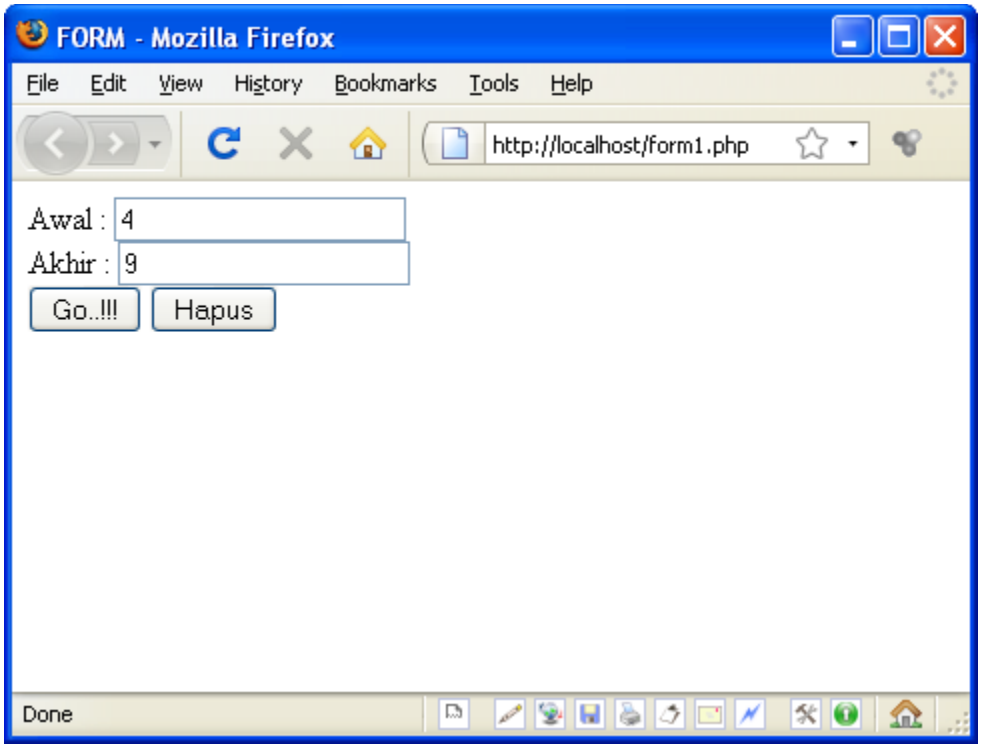
```
 1 <html>
 2 <head>
 3        <title>FORM</title>
 4 </head>
 5 <body>
 6    <form method=get action="deret.php">
 7    Awal  : <input type=text name=awal><br>
 8    Akhir : <input type=text name=akhir><br>
 9    <input type=submit value="Go..!!!">
10    <input type=reset value="Hapus">
11    </form>
12 </body>
13 </html>
```

   Simpan file ini dengan nama form1.php.

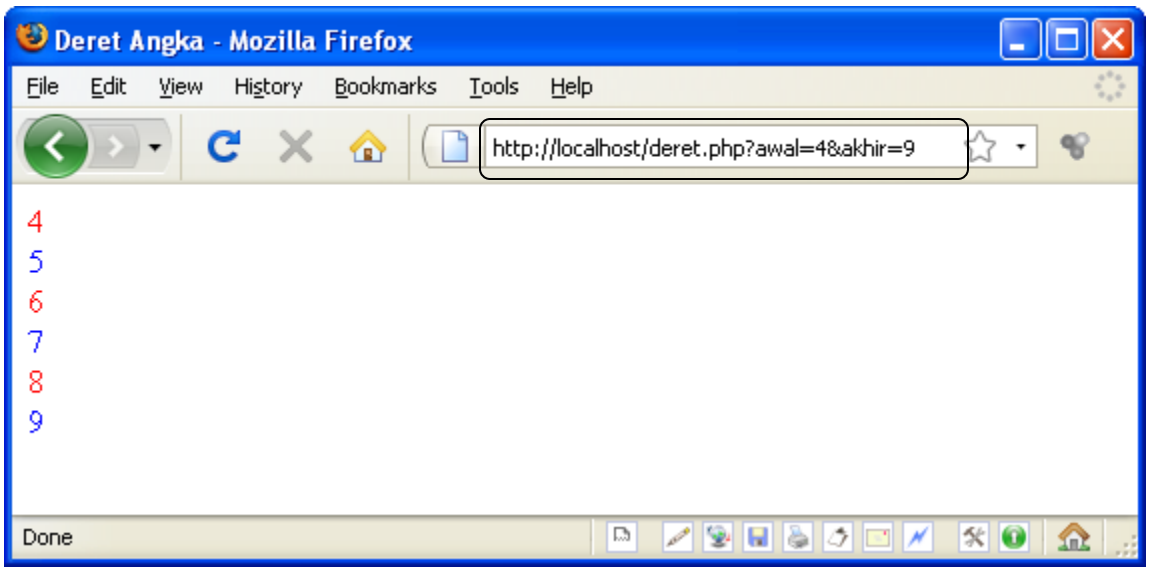
   Keterangan program :

   - Baris 6 : Definisi karakter dengan method=post dan jika tombol Submitnya diklik akan mengeksekusi file deret.php (property action).
   - Baris 7 : Pembuatan objek textbox dengan nama objek awal. Nama objek harus sama dengan nama variable dalam file phpnya.
   - Baris 8 : Pembuatan objek textbox dengan nama objek akhir.
   - Baris 9 : Pembuatan tombol submit dengan tulisan pada tombolnya adalah **Go..!!!.**
   - Baris 10 : Pembuatan tombol reset dengan tulisan **Hapus**.

   Eksekusi file tersebut dengan alamat : http://localhost/form1.php



Jika textbox awal diisi dengan 4 dan textbox akhir diisi dengan 9 kemudian tombol **Go..!!!** diklik maka akan menampilkan hasil seperti gambar di bawah ini.



Pengiriman data dengan menggunakan form bermethod get sebenarnya sama dengan pengisian data dengan URL. Perhatikan URL di address bar setelah tombol submit diklik.

3. Melalui form dengan method post

   Jika ingin mengirim variable dari suatu form, maka langkah yang harus dilakukan adalah dengan membuat form HTML yang berisi textbox tempat mengisi variable awal dan akhir serta sebuah tombol Submit.

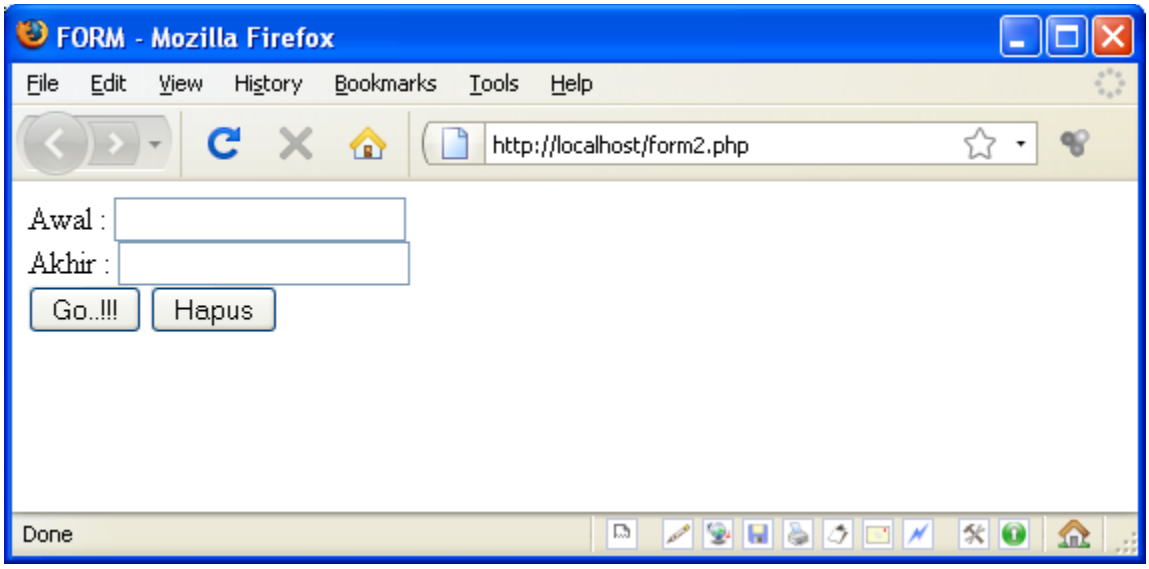
   Edit file form1.php, ganti method=get menjadi method=post dan action="deret.php" menjadi action="deret2.php". Kemudian save as file tersebut menjadi form2.php. Lihat source code di bawah ini.

```
 1 <html>
 2 <head>
 3        <title>FORM</title>
 4 </head>
 5 <body>
 6    <form method=post action="deret2.php">
 7    Awal  : <input type=text name=awal><br>
 8    Akhir : <input type=text name=akhir><br>
 9    <input type=submit value="Go..!!!">
10    <input type=reset value="Hapus">
11    </form>
12 </body>
13 </html>
```

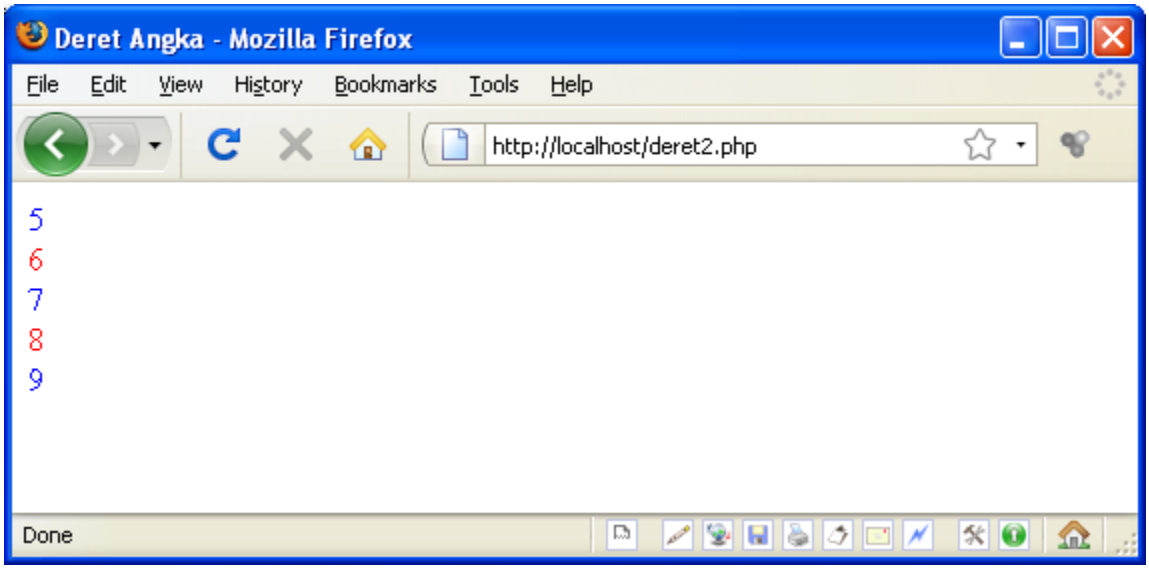
   Kemudian edit file deret.php dengan mengganti semua $_GET menjadi $_POST. Kemudian save as file ini menjadi deret2.php.

```
 1 <html>
 2 <head>
 3        <title>Title here!</title>
 4 </head>
 5 <body>
 6       <?php
 7            for($i=$_POST[awal];$i<=$_POST[akhir];$i++)
 8            {
 9                if($i % 2 ==0)
10                   echo "<font color=red>$i</font><br>";
11                else
12                   echo "<font color=blue>$i</font><br>";
13            }
14       ?>
15 </body>
16 </html>
```

   Eksekusi file tersebut dengan alamat : http://localhost/form2.php



Secara tampilan, form bermethod post dan form bermethod get tidak ada bedanya. Perbedaannya dapat dilihat ketika data telah disubmit. Perhatikan URL yang ada di address bar. Dengan menggunakan form bermethod post, isi variable yang dikirim tidak terlihat di address bar.



PROGRAM KEDUA

Contoh kasus : Buatlah suatu halaman web yang dapat melakukan perhitungan penjualan barang. Data yang diinputkan adalah Nama Barang, Harga Satuan, Quantit, Status Pelanggan (Pelanggan dan Bukan Pelanggan), dan Kota Pengiriman (Bandung, Jakarta dan Surabaya). Ketika data tersebut disubmit, maka lakukan perhitungan, sehingga dapat menampilkan informasi Subtotal, Diskon, Ongkos Kirim, dan Total.

Ketentuan :

1. Subtotal = Harga * Quantity
2. Diskon sebesar 10% dari Subtotal bagi pelanggan yang berstatus pelanggan
3. Ongkos Kirim :
    a. Kota Bandung = Rp. 10000
    b. Kota Jakarta = Rp. 20000
    c. Kota Surabaya = Rp. 30000
4. Total = Subtotal – Diskon + Ongkos Kirim

Adapun form untuk pengisian data dan contoh tampilan hasil perhitungan dapat dibuat seperti gambar di bawah ini.

PENJUALAN	
Nama Barang	Buku
Harga	5000
Quantity	10
Status	(•) Pelanggan ( ) Bukan Pelanggan
Kota	Jakarta
	Hitung Reset

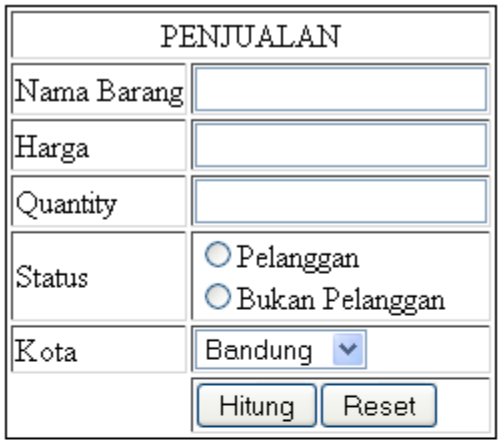
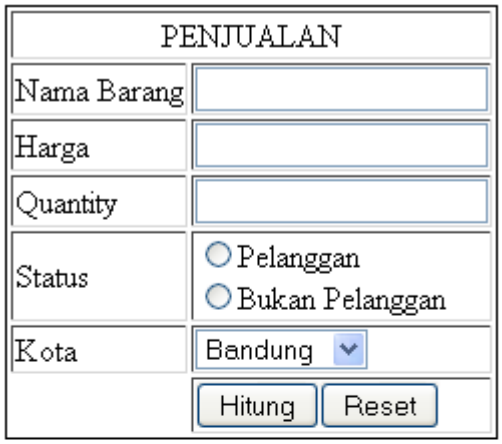
HASIL PERHITUNGAN	
Nama Barang	Buku
Harga	Rp. 5000
Quantity	10
Subtotal	Rp. 50000
Status	Pelanggan
Diskon	Rp. 5000
Ongkos Kirim	Rp. 20000 (Jakarta)
Total	Rp. 45000

Lihat Source Code :

1. Penjualan.php (Lihat file **penjualan.php.pdf**)

   Source code pertama untuk kasus penjualan.

   Hasil run :

Sebelum Submit data	Setelah Submit Data
	
	

PENJUALAN

Nama Barang	
Harga	
Quantity	
Status	○ Pelanggan ○ Bukan Pelanggan
Kota	Bandung
	Hitung Reset

HASIL PERHITUNGAN

Nama Barang	
Harga	Rp.
Quantity	
Subtotal	Rp. 0
Status	Bukan Pelanggan
Diskon	Rp. 0
Ongkos Kirim	Rp. ()
Total	Rp. 0

PENJUALAN

Nama Barang	
Harga	
Quantity	
Status	○ Pelanggan ○ Bukan Pelanggan
Kota	Bandung
	Hitung Reset

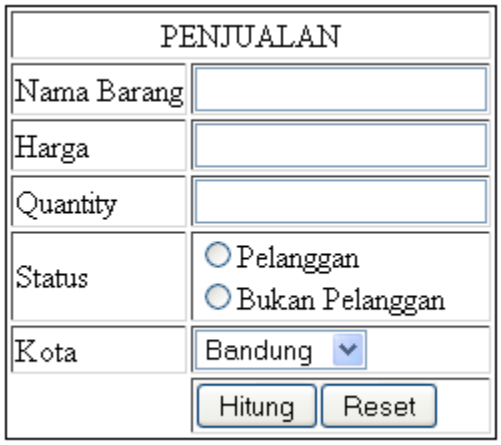
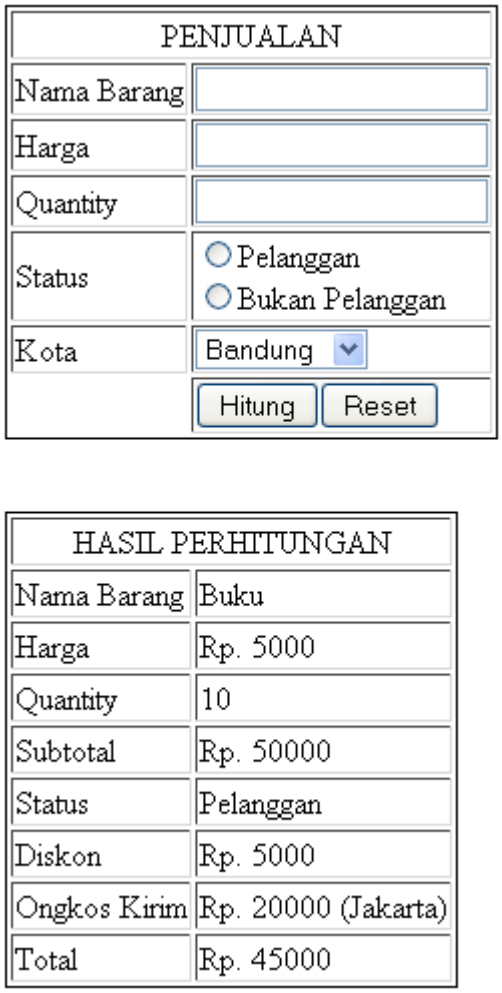
HASIL PERHITUNGAN

Nama Barang	Buku
Harga	Rp. 5000
Quantity	10
Subtotal	Rp. 50000
Status	Pelanggan
Diskon	Rp. 5000
Ongkos Kirim	Rp. 20000 (Jakarta)
Total	Rp. 45000

2. Penjualan2.php (Lihat file **penjualan2.php.pdf**)

   Penambahan perintah untuk mencegah tampilnya hasil perhitungan ketika user belum menekan tombol Hitung (submit).

   Hasil run :

Sebelum Submit data	Setelah Submit Data
	

3. Penjualan3.php (Lihat file **penjualan3.php.pdf**)

   Sama dengan Penjualan2.php, tetapi ditambah dengan kemampuan untuk menuliskan lagi data yang telah diisi di form. Jadi setelah disubmit, data akan tetap muncul seperti ketika sebelum disubmit. Untuk lebih jelas, bandingkan tampilan setelah submit data antara penjualan2.php dan penjualan3.php.

   Hasil run :

Sebelum Submit data	Setelah Submit Data

PENJUALAN

Nama Barang	
Harga	
Quantity	
Status	○ Pelanggan ○ Bukan Pelanggan
Kota	Bandung
	Hitung Reset

PENJUALAN

Nama Barang	Buku
Harga	5000
Quantity	10
Status	◉ Pelanggan ○ Bukan Pelanggan
Kota	Jakarta
	Hitung Reset

HASIL PERHITUNGAN

Nama Barang	Buku
Harga	Rp. 5000
Quantity	10
Subtotal	Rp. 50000
Status	Pelanggan
Diskon	Rp. 5000
Ongkos Kirim	Rp. 20000 (Jakarta)
Total	Rp. 45000

4. Penjualan4.php (Lihat file **penjualan4.php.pdf**)

   Sama dengan penjualan3.php, tetapi informasi yang berbentuk angka telah diformat dengan menggunakan perintah number_format.

   Hasil run :

Sebelum Submit data

PENJUALAN	
Nama Barang	
Harga	
Quantity	
Status	○ Pelanggan ○ Bukan Pelanggan
Kota	Bandung
	Hitung Reset

Setelah Submit Data

PENJUALAN	
Nama Barang	Buku
Harga	5000
Quantity	10
Status	◉ Pelanggan ○ Bukan Pelanggan
Kota	Jakarta
	Hitung Reset

HASIL PERHITUNGAN	
Nama Barang	Buku
Harga	Rp. 5.000
Quantity	10
Subtotal	Rp. 50.000
Status	Pelanggan
Diskon	Rp. 5.000
Ongkos Kirim	Rp. 20.000 (Jakarta)
Total	Rp. 45.000

LATIHAN

1. Buatlah suatu halaman web yang berguna untuk menampilkan tabel harga BBM (Bensin, Solar dan Minyak Tanah).

   Ketentuan :

   - Jika web dipanggil untuk pertama kalinya (belum diisi data), maka halaman web akan menampilkan daftar harga bbm dari 1 liter sampai 10 liter untuk semua jenis BBM.



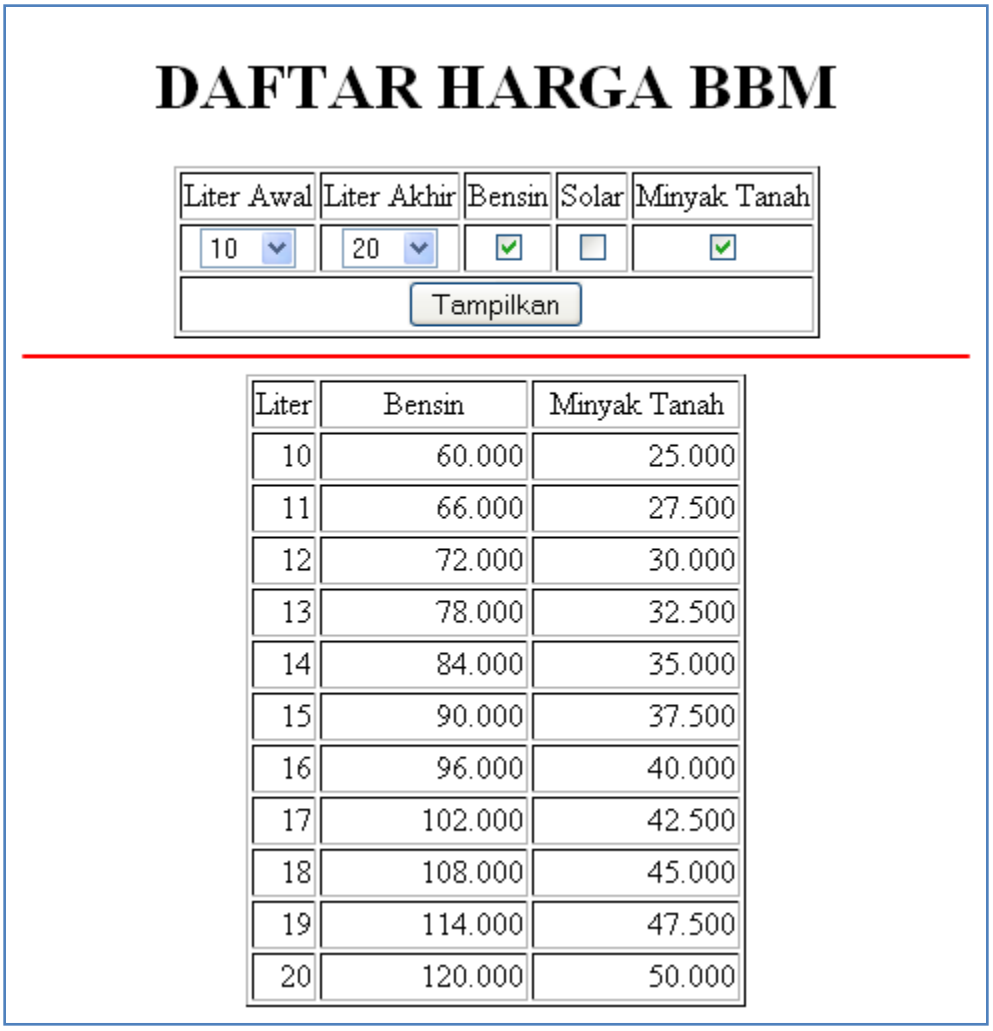
DAFTAR HARGA BBM

Liter Awal	Liter Akhir	Bensin	Solar	Minyak Tanah
1	1	☐	☐	☐

Tampilkan

Liter	Bensin	Solar	Minyak Tanah
1	6.000	5.500	2.500
2	12.000	11.000	5.000
3	18.000	16.500	7.500
4	24.000	22.000	10.000
5	30.000	27.500	12.500
6	36.000	33.000	15.000
7	42.000	38.500	17.500
8	48.000	44.000	20.000
9	54.000	49.500	22.500
10	60.000	55.000	25.000

- Jenis BBM yang ditampilkan harus sesuai dengan pilihan. Jika hanya memilih bensin dan minyak tanah, maka tabel hanya berisi harga bensin dan minyak tanah saja.



DAFTAR HARGA BBM

Liter Awal	Liter Akhir	Bensin	Solar	Minyak Tanah
10	20	☑	☐	☑
Tampilkan				

Liter	Bensin	Minyak Tanah
10	60.000	25.000
11	66.000	27.500
12	72.000	30.000
13	78.000	32.500
14	84.000	35.000
15	90.000	37.500
16	96.000	40.000
17	102.000	42.500
18	108.000	45.000
19	114.000	47.500
20	120.000	50.000

- Data yang telah diinputkan akan selalu ditampilkan kembali di halaman hasil.

- Jika terjadi kesalahan karena liter awal lebih besar dari liter akhir, maka hasil akan tetap ditampilkan.



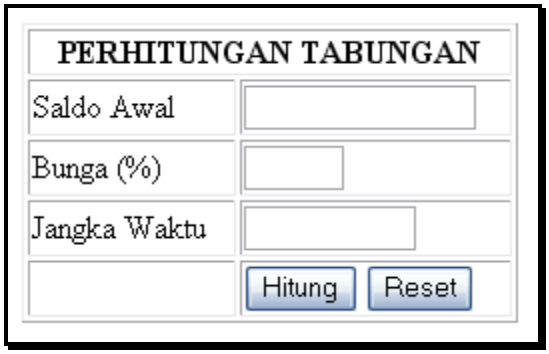
DAFTAR HARGA BBM

Liter Awal	Liter Akhir	Bensin	Solar	Minyak Tanah
10	5	☑	☑	☑
Tampilkan				

Liter	Bensin	Solar	Minyak Tanah
5	30.000	27.500	12.500
6	36.000	33.000	15.000
7	42.000	38.500	17.500
8	48.000	44.000	20.000
9	54.000	49.500	22.500
10	60.000	55.000	25.000

2. Buatlah suatu halaman web yang berguna untuk menghitung saldo tabungan berdasarkan saldo awal, bunga dan jangka waktunya.
   Formnya berbentuk :

Form2.php



Jika tombol hitung di klik, maka akan menghasilkan tampilan seperti di bawah ini.

HitungTabungan.php



DETAIL TABUNGAN

BULAN	SALDO
1	Rp. 110.000
2	Rp. 121.000
3	Rp. 133.100
4	Rp. 146.410
5	Rp. 161.051
6	Rp. 177.156
7	Rp. 194.872
8	Rp. 214.359
9	Rp. 235.795
10	Rp. 259.374

Setelah berhasil, anda rombak agar action dari file **form2.php** tidak ke **hitungtabungan.php** tetapi ke **form2.ph**p (action ke diri sendiri). Juga dibuat script agar ketika menampilkan hasil perhitungan form tempat mengisi data ditampilkan kembali dengan data sesuai data yang diisikan sebelumnya.

Solusi bisa ditemukan di halaman berikutnya.

Solusi :

a. File form2.php

```
<html>
<head><title>Perhitungan Tabungan</title></title></head>
<body>
<form method="post" action="hitungtabungan.php">
  <table width="248" border="1" align="center">
    <tr>
      <td colspan="2" align="center"><b>PERHITUNGAN TABUNGAN</b></td>
    </tr>
    <tr>
      <td>Saldo Awal </td>
      <td><input name=saldoawal type=text size="15"></td>
    </tr>
    <tr>
      <td>Bunga (%) </td>
      <td><input name=bunga type=text size="4"></td>
    </tr>
    <tr>
      <td>Jangka Waktu </td>
      <td><input name=waktu type=text size="10"></td>
    </tr>
    <tr>
      <td> </td>
      <td><input type=submit name=tbl_hitung value="Hitung">
      <input type=reset value="Reset"></td>
    </tr>
  </table>
</form>
</body>
</html>
```

b. File hitungtabungan.php

```
<html>
<head><title>Perhitungan Tabungan</title></head>
<body>
  <center>DETAIL TABUNGAN</center>
  <table border=1 width=300 align=center>
  <tr bgcolor=silver align=center>
          <td>BULAN</td><td>SALDO</td></tr>
  <?
        $saldo=$saldoawal;
        for($i=1;$i<=$waktu;$i++)
        {
            $saldo+=$bunga/100*$saldo;
              echo "<tr><td>$i</td><td>Rp. $saldo</td></tr>";
        }
  ?>
  </table>
</form>
</body>
</html>
```

c. Action ke diri sendiri

Jika action dari sebuah form ada dalam file yang sama, maka isi dari file action yang dulu (file hitungtabungan.php) harus dipindahkan ke file pertama (file form2.php). Juga property action formnya harus diganti ke file sendiri (form2.php). Sehingga isi form2.php adalah sebagai berikut :

```
<html>
<head><title>Perhitungan Tabungan</title></title></head>
<body>
<form method="post" action="hitungtabungan.php">
  <table width="248" border="1" align="center">
    <tr>
      <td colspan="2" align="center"><b>PERHITUNGAN TABUNGAN</b></td>
    </tr>
    <tr>
      <td>Saldo Awal </td>
      <td><input name=saldoawal type=text size="15"></td>
    </tr>
    <tr>
      <td>Bunga (%) </td>
      <td><input name=bunga type=text size="4"></td>
    </tr>
    <tr>
      <td>Jangka Waktu </td>
      <td><input name=waktu type=text size="10"></td>
    </tr>
    <tr>
      <td> </td>
      <td><input type=submit name=tbl_hitung value="Hitung">
      <input type=reset value="Reset"></td>
    </tr>
  </table>
</form>
<?
    if($tbl_hitung == "Hitung") // Jika tbl_hitung di klik/submit
    {
?>
      <center>DETAIL TABUNGAN</center>
      <table border=1 width=300 align=center>
      <tr bgcolor=silver align=center>
              <td>BULAN</td> <td>SALDO</td></tr>
      <?
         $saldo=$saldoawal;
         for($i=1;$i<=$waktu;$i++)
         {
             $saldo+=$bunga/100*$saldo;
               echo "<tr><td>$i</td><td>Rp. $saldo</td></tr>";
         }
      ?>
      </table>
<?
    }
?>
</body></html>
```

d. Menampilkan kembali isian form

Untuk menampilkan kembali isian form sebelumnya, maka property value dari textbox harus terisi dengan isian textbox sebelumnya.

Oleh karena itu, file form2.php akan menjadi seperti di bawah ini :

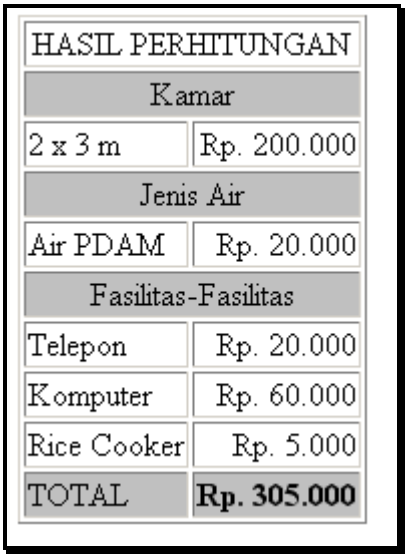
```
<html>
<head><title>Perhitungan Tabungan</title></title></head>
<body>
<form method="post" action="hitungtabungan.php">
  <table width="248" border="1" align="center">
    <tr>
      <td colspan="2" align="center"><b>PERHITUNGAN TABUNGAN</b></td>
    </tr>
     <tr>
       <td>Saldo Awal </td>
       <td><input name=saldoawal type=text
             value="<? echo $saldoawal;?>"></td>
     </tr>
     <tr>
       <td>Bunga (%) </td>
       <td><input name=bunga type=text value="<? echo $bunga;?>"></td>
     </tr>
     <tr>
       <td>Jangka Waktu </td>
       <td><input name=waktu type=text value="<? echo $waktu;?>"></td>
     </tr>
    <tr>
      <td> </td>
      <td><input type=submit name=tbl_hitung value="Hitung">
      <input type=reset value="Reset"></td>
    </tr>
  </table>
</form>
<?
    if($tbl_hitung == "Hitung") // Jika tbl_hitung di klik/submit
    {
?>
      <center>DETAIL TABUNGAN</center>
      <table border=1 width=300 align=center>
      <tr bgcolor=silver align=center>
             <td>BULAN</td> <td>SALDO</td></tr>
      <?
         $saldo=$saldoawal;
         for($i=1;$i<=$waktu;$i++)
         {
             $saldo+=$bunga/100*$saldo;
               echo "<tr><td>$i</td><td>Rp. $saldo</td></tr>";
         }
      ?>
      </table>
<?
    }
?>
</body></html>
```

3. Buatlah suatu halaman web yang berguna untuk menghitung biaya kost. Tampilan awalnya adalah :

Form Pengisian Data



Hasil Perhitungan



Input : Tipe Kamar, Air, dan Fasilitas yang terdiri dari Telepon, Televisi, Komputer dan Rice Cooker.

Ketentuan :

Tipe Kamar	Luas	Biaya Sewa
A	2 x 3 m	Rp. 200.000
B	3 x 3 m	Rp. 250.000
C	4 x 3 m	Rp. 300.000

Biaya Bulanan :

- Air PDAM : Rp. 20.000
- Air Tanah : Rp. 15.000
- Telepon : Rp. 20.000
- Televisi : Rp. 30.000
- Komputer : Rp. 60.000
- Rice Cooker : Rp. 5.000

Hitunglah biaya bulanan berdasarkan jenis kamar, jenis air dan fasilitas yang dipilihnya saja.

Solusi :

```
<html>
<head><title>Perhitungan Biaya Kost</title></head>
<body>
<form method="post" action="<? echo $PHP_SELF;?>">
  <table border="1">
    <tr>
      <td colspan="2" align="center"><b>PERHITUNGAN HARGA KOST </b></td>
    </tr>
    <tr>
      <td width="116">Tipe Kamar </td>
      <td width="174"><select name="tipe">
                        <option value="A">A. [2 x 3 m]</option>
                        <option value="B">B. [3 x 3 m]</option>
                        <option value="C">C. [4 x 3 m]</option>
                      </select>
        </td>
    </tr>
    <tr>
      <td>Air</td>
      <td><input name="rb_air" type="radio" value="PDAM">Air PDAM <br>
        <input name="rb_air" type="radio" value="Tanah">Air Tanah </td>
    </tr>
    <tr>
      <td>Fasilitas</td>
      <td><input name="cb_telepon" type="checkbox" value="Y">Telepon<br>
        <input name="cb_televisi" type="checkbox" value="Y">Televisi<br>
        <input name="cb_komputer" type="checkbox" value="Y">Komputer<br>
        <input name="cb_rice" type="checkbox" value="Y">Rice Cooker</td>
    </tr>
    <tr>
      <td> </td>
      <td><input type="submit" name="Submit" value="Hitung">
      <input type="reset" name="Submit2" value="Reset"></td>
    </tr>
  </table>
</form>
<?
   if($Submit)// Jika Tombol Submit (Hitung) Di Klik
   {
?>
    <hr>
       <table border=1>
       <tr><td colspan=2 align=center>HASIL PERHITUNGAN</td></tr>
       <tr bgcolor=silver><td colspan=2 align=center>Kamar</td></tr>
       <tr><td>
       <?
              if($tipe=="A")
              {
                  $luas="2 x 3 m";
                     $harga=200000;
              }
              else
              if($tipe=="B")
              {
                  $luas="3 x 3 m";
                     $harga=250000;
              }
```

```
58                else
59                if($tipe=="C")
60                {
61                    $luas="4 x 3 m";
62                       $harga=300000;
63                }
64                echo $luas;//menuliskan luas
65          ?></td><td align=right>Rp.
66                        <? echo number_format($harga,0,",",".");?></td></tr>
67          <tr bgcolor=silver><td align=center colspan=2>Jenis Air</td></tr>
68          <tr><td>Air <? echo $rb_air;?></td><td align=right>Rp.
69          <?
70              if($rb_air=="PDAM")
71                    $air=20000;
72                else
73                if($rb_air=="Tanah")
74                   $air=15000;
75                echo number_format($air,0,",",".");//menuliskan $air
76          ?></td></tr>
77       <?
78  if(($cb_telepon=="Y")or($cb_televisi=="Y")or($cb_komputer=="Y")or($cb_rice=="Y"))
79     echo "<tr bgcolor=silver>
80           <td align=center colspan=2>Fasilitas-Fasilitas</td></tr>";
81             if($cb_telepon=="Y")
82                {
83                    $telepon=20000;
84                       echo "<tr><td>Telepon</td><td align=right>Rp. ".
85                       number_format($telepon,0,",",".")."</td></tr>";
86                }
87             if($cb_televisi=="Y")
88                {
89                    $televisi=30000;
90                       echo "<tr><td>Televisi</td><td align=right>Rp. ".
91                       number_format($televisi,0,",",".")."</td></tr>";
92                }
93                if($cb_komputer=="Y")
94                {
95                    $komputer=60000;
96                       echo "<tr><td>Komputer</td><td align=right>Rp. ".
97                       number_format($komputer,0,",",".")."</td></tr>";
98                }
99             if($cb_rice=="Y")
100               {
101                   $rice=5000;
102                      echo "<tr><td>Rice Cooker</td><td align=right>Rp. ".
103                      number_format($rice,0,",",".")."</td></tr>";
104               }
105               $total=$harga+$air+$telepon+$televisi+$komputer+$rice;
106               echo "<tr bgcolor=silver><td>TOTAL</td><td align=right><b>Rp. ".
107               number_format($total,0,",",".")."</b></td></tr>";
108         ?>
109         </table>
110 <?
111    }
112 ?>
113 </body>
114 </html>
```

Keterangan :

BARIS	PENJELASAN
3	$PHP_SELF adalah suatu konstanta yang berisi nama file tersebut. Sehingga jika perintah tersebut dipanggil dalam file form.php, maka $PHP_SELF berisi nama file form.php. [Buktikan dengan view source di browser]. `action="<? echo $PHP_SELF;?>"` berarti action-nya ada di file sendiri.
9 - 17	Pembuatan objek pemilihan (select) dengan 3 option dengan nama objek **tipe**. Jika memilih "A. [2 x 3 m]", maka **tipe** bernilai "A", dan seterusnya.
18 - 22	Pembuatan objek radio button dengan 2 pilihan dengan nama objek **rb_air**. Jika memilih "Air PDAM", maka **rb_air** bernilai "PDAM" dan jika memilih "Air Tanah", maka **rb_air** bernilai "Tanah".
23 - 29	Pembuatan objek-objek pilihan fasilitas. Jenis objek yang dipakai adalah checkbox. Ada 4 checkbox yaitu cb_telepon, cb_televisi, cb_komputer, dan cb_rice. Setiap objek tersebut akan bernilai "Y" jika dipilih oleh user.
30 - 34	Pembuatan tombol submit dan reset. Tombol submit diberi nama **Submit** dengan tulisan "Hitung", sedangkan tombol reset diberi tulisan ""Reset".
37 - 40	Script PHP yang memeriksa apakah telah ada penekanan tombol **Submit.** Jika tombol **Submit** telah diklik, maka komputer akan mengeksekusi perintah yang ada di antara { [baris 40] dan } [baris 111] (tabel hasil perhitungan).
42	Pembuatan Tabel hasil perhitungan
43	Pembuatan judul tabel hasil perhitungan
44	Pembuatan baris yang berisi tulisan "Kamar". (baris 2 di tabel hasil).
47 - 63	Percabangan mencari **luas** kamar dan **harga** sewanya.
64	Penulisan **luas** kamar di cell sebelah kiri pada baris 3 dari tabel hasil.
65 - 66	Penulisan **harga** sewa di cell sebelah kanan pada baris 3 dan diformat seperti format penulisan uang dalam format Indonesia.
67	Pembuatan baris yang berisi tulisan "Jenis Air" (baris 3).
68	Penulisan jenis air di cell sebelah kiri.
69 - 76	Percabangan untuk menentukan biaya penggunaan air berdasarkan jenis airnya kemudian menuliskan hasilnya dalam format uang Indonesia (baris 75).
78 - 80	Percabangan yang berguna untuk menampilkan baris yang bertuliskan "Fasilitas-Fasilitas" pada tabel hasil perhitungan yang hanya akan muncul kalau salah satu atau lebih dari 4 pilihan fasilitas ada yang dipilih, sehingga operator logika yang dipakai adalah or (||).

BARIS	PENJELASAN
81 - 86	Percabangan yang memeriksa apakah checkbox **cb_telepon** dipilih oleh user. Jika objek **cb_telepon** bernilai "Y" (dipilih oleh user), maka biaya **telepon** adalah 20000 dan kemudian menampilkannya dalam format uang Indonesia (baris 84 - 85).
87 - 92	Percabangan yang memeriksa apakah checkbox **cb_televisi** dipilih oleh user. Jika objek **cb_televisi** bernilai "Y" (dipilih oleh user), maka biaya **televisi** adalah 30000 dan kemudian menampilkannya dalam format uang Indonesia (baris 90 - 91).
93 - 98	Percabangan yang memeriksa apakah checkbox **cb_komputer** dipilih oleh user. Jika objek **cb_komputer** bernilai "Y" (dipilih oleh user), maka biaya **komputer** adalah 60000 dan kemudian menampilkannya dalam format uang Indonesia (baris 96 - 97).
99 - 104	Percabangan yang memeriksa apakah checkbox **cb_rice** dipilih oleh user. Jika objek **cb_rice** bernilai "Y" (dipilih oleh user), maka biaya **rice** adalah 5000 dan kemudian menampilkannya dalam format uang Indonesia (baris 102 - 103).
105	Perhitungan **total** yang didapat dari penjumlahan dari variable **harga** sewa, **air**, **telepon**, **televisi**, **komputer** dan **rice** cooker.
106 - 107	Penulisan total dalam format uang Indonesia.
109	Tutup tabel hasil perhitungan
110 - 112	Tutup statement **if ($submit).**

VALIDASI DATA

Validasi data merupakan suatu proses yang sangat dilakukan untuk menjaga konsistensi data.

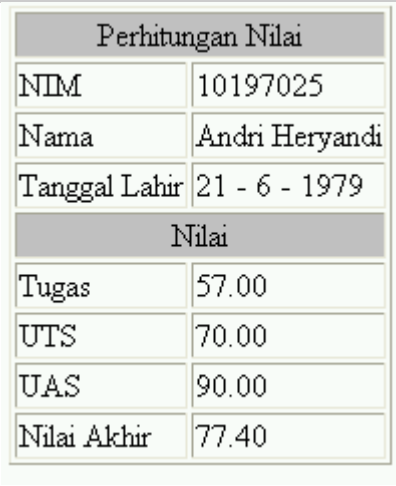
Contoh Kasus :

Buatlah suatu halaman web yang berguna untuk menghitung nilai seorang mahasiswa. Data yang diinputkan adalah : NIM, Nama, Tanggal Lahir, Nilai Tugas, Nilai UTS, dan Nilai UAS. Dari data-data tersebut dihasilkan Nilai Akhir dengan ketentuan 20% Tugas + 30% UTS + 50% UAS. Perhatikan gambar di bawah ini.

Form Pengisian Data Valid



Hasil Perhitungan Jika Data Valid



Form Pengisian Data Invalid



Hasil Jika Data Invalid

Proses perhitungan dibatalkan. Data yang anda isikan tidak valid. Kesalahan yang terjadi :

- 3 Huruf Pertama untuk IF harus 101 [S1], 107 [D3], 122 [D1]
- Tanggal Lahir harus diisi dengan lengkap
- Tanggal Lahir tidak sah
- Nilai UTS Hanya boleh angka

Kembali

Aturan-aturan :

Perhitungan hanya dapat dilakukan jika semua data telah valid.

Field	Ketentuan
NIM	▪ NIM harus 8 huruf ▪ NIM harus angka ▪ 3 huruf awal harus “101” untuk S1, “107” untuk D3 dan “122” untuk D1.
NAMA	▪ Awal dan akhir Nama tidak boleh ada spasi ▪ Nama minimal 2 huruf.
TANGGAL LAHIR	▪ Semua isian Tanggal Lahir (tanggal, bulan, dan tahun) harus diisi. ▪ Tanggal Lahir harus sesuai dengan ketentuan (valid).
TUGAS, UTS, UAS	▪ Data harus berupa angka maksimal 3 digit ▪ Semua digit harus berupa angka ▪ Nilai Data harus dalam range 0 sampai 100.

Solusi :

FormNilai.php

```
<html>
<head>
       <title>Perhitungan Nilai</title>
</head>
<body>
      <form method=post action="hitungnilai.php">
         <table border=1>
         <tr><td colspan=2 align=center bgcolor=silver>Perhitungan
Nilai</td></tr>
         <tr><td>NIM</td>
             <td><input type=text name=nim size=9 maxlength=8></td></tr>
         <tr><td>Nama</td>
             <td><input type=text name=nama size=25 maxlength=30></td></tr>
         <tr><td>Tanggal Lahir</td>
             <td>
                 <select name=tanggal>
                   <option value=0>Pilih</option>
                 <?
                      for($tgl=1;$tgl<=31;$tgl++)
                        echo "<option value=$tgl>$tgl</option>";
                 ?>
                 </select> -
                 <select name=bulan>
                    <option value=0>Pilih</option>
                    <option value=1>Januari</option>
                    <option value=2>Februari</option>
                    <option value=3>Maret</option>
                    <option value=4>April</option>
                    <option value=5>Mei</option>
                    <option value=6>Juni</option>
                    <option value=7>Juli</option>
```

```
32                      <option value=8>Agustus</option>
33                      <option value=9>September</option>
34                      <option value=10>Oktober</option>
35                      <option value=11>November</option>
36                      <option value=12>Desember</option>
37                   </select> -
38                   <select name=tahun>
39                      <option value=0>Pilih</option>
40                      <?
41                           for($thn=2000;$thn>=1950;$thn--)
42                              echo "<option>$thn</option>";
43                      ?>
44                   </select>
45               </td></tr>
46            <tr><td colspan=2 align=center bgcolor=silver>Nilai</td></tr>
47            <tr><td>Tugas </td>
48                <td><input type=text name=tugas size=4 maxlength=3></td></tr>
49            <tr><td>UTS </td>
50                <td><input type=text name=uts size=4 maxlength=3></td></tr>
51            <tr><td>UAS </td>
52                <td><input type=text name=uas size=4 maxlength=3></td></tr>
53            <tr bgcolor=silver><td> </td>
54               <td><input type=submit name=submit value="Hitung"></td></tr>
56            </table>
57         </form>
58   </body>
59   </html>
```

Keterangan Program :

BARIS	PENJELASAN
6	Pembuatan form yang akan melakukan action ke file hitungnilai.php.
11	Pembuatan objek textbox dengan nama **nim** dengan ukuran lebar 9 huruf dan maksimal isi adalah 8 huruf.
13	Pembuatan objek textbox dengan nama **nama** dengan ukuran lebar 25 huruf dengan maksimal isi adalah 30 huruf.
16 – 22	Pembuatan objek combobox (select) dengan nama **tanggal** dengan isi pilihan dari 0 sampai 31. Isian 0 digunakan untuk pemeriksaan apakah user memilih tanggal atau tidak. Jika objek ini bernilai 0 berarti user belum memilih tanggal.
23 - 37	Pembuatan objek combobox (select) dengan nama **bulan** dengan isian pilihan dari 0 sampai 12 tetapi bentuk pilihannya adalah Januari sampai Desember. Isian 0 digunakan untuk pemeriksaan apakah user memilih bulan atau tidak. Jika objek ini bernilai 0 berarti user belum memilih bulan.
38 - 44	Pembuatan objek combobox (select) dengan nama **tahun** dengan isian pilihan 0, serta tahun 1950 sampai 2000 tetapi terurut descending (dari besar ke kecil). Isian 0 digunakan untuk pemeriksaan apakah user memilih tahun atau tidak. Jika objek ini bernilai 0 berarti user belum mengisi tahun.

BARIS	PENJELASAN
48	Pembuatan objek textbox dengan nama **tugas** dengan lebar 4 huruf dan maksimal isian data adalah 3 digit/huruf.
50	Pembuatan objek textbox dengan nama **uts** dengan lebar 4 huruf dan maksimal isian data adalah 3 digit/huruf.
52	Pembuatan objek textbox dengan nama **uas** dengan lebar 4 huruf dan maksimal isian data adalah 3 digit/huruf.
54	Pembuatan objek tombol submit dengan nama objek **submit** dengan tulisan pada tombolnya adalah “Hitung”.

HitungNilai.php

```
 1  <html>
 2  <head>
 3         <title>Perhitungan Nilai</title>
 4  </head>
 5  <body>
 6           <?
 7               //AWAL VALIDASI DATA
 8               $ada_error=false;
 9               $pesan_error="Kesalahan yang terjadi :<br>";
10
11               $nim=trim($nim);
12
13               if(strlen($nim)<8)
14               {
15                   $pesan_error=$pesan_error."- Panjang NIM harus 8 Huruf<br>";
16                   $ada_error=true;
17               }
18               if(!ereg("[0-9]{8}",$nim))
19               {
20                   $pesan_error.="- NIM Hanya boleh angka<br>";
21                   $ada_error=true;
22               }
23
24
25               $jenjang=substr($nim,0,3);
26               if(($jenjang!="101")and($jenjang!="107")and($jenjang!="122"))
27               {
28                   $pesan_error=$pesan_error."- 3 Huruf Pertama untuk IF harus
29                                              101 [S1], 107 [D3], 122 [D1]<br>";
30                   $ada_error=true;
31               }
32
33               $nama=trim($nama);
34               if(strlen($nama)<2)
35               {
36                   $pesan_error=$pesan_error."- Nama Minimal 2 Huruf<br>";
37                   $ada_error=true;
38               }
39
```

```
40              if(($tanggal==0)or($bulan==0)or($tahun==0))
41              {
42                   $pesan_error.="- Tanggal Lahir harus diisi lengkap<br>";
43                   $ada_error=true;
44              }
45              if(!checkdate($bulan,$tanggal,$tahun))// if checkdate == false
46              {
47                   $pesan_error.="- Tanggal Lahir tidak sah<br>";
48                   $ada_error=true;
49              }
50
51              if(!ereg("[0-9]{1,3}",$tugas))
52              {
53                  $pesan_error.="- Nilai Tugas Hanya boleh angka<br>";
54                  $ada_error=true;
55              }
56              if(($tugas<0)||($tugas>100))
57              {
58                   $pesan_error.="- Nilai Tugas harus 0 s/d 100<br>";
59                   $ada_error=true;
60              }
61
62              if(!ereg("[0-9]{1,3}",$uts))
63              {
64                  $pesan_error.="- Nilai UTS Hanya boleh angka<br>";
65                  $ada_error=true;
66              }
67              if(($uts<0)||($uts>100))
68              {
69                   $pesan_error.="- Nilai UTS harus 0 s/d 100<br>";
70                   $ada_error=true;
71              }
72
73              if(!ereg("[0-9]{1,3}",$uas))
74              {
75                  $pesan_error.="- Nilai UAS Hanya boleh angka<br>";
76                  $ada_error=true;
77              }
78              if(($uas<0)||($uas>100))
79              {
80                   $pesan_error.="- Nilai UAS harus 0 s/d 100<br>";
81                   $ada_error=true;
82              }
83
84              if($ada_error)//if($ada_error==true)
85              {
86                    echo "Proses perhitungan dibatalkan. Data yang anda
87                          isikan tidak valid.<br>$pesan_error<br>
88                          <a href='javascript:history.back();'>Kembali</a>";
89              }
90              else
91              if(!$ada_error)//if($ada_error==false)
92              {
93          ?>
94      <table border=1>
95      <tr><td colspan=2 align=center bgcolor=silver>Perhitungan Nilai</td></tr>
96      <tr><td>NIM</td><td><? echo $nim;?></td></tr>
97      <tr><td>Nama</td><td><? echo $nama;?></td></tr>
```

```
 98      <tr><td>Tanggal Lahir</td>
 99          <td><? echo "$tanggal - $bulan - $tahun"; ?></td></tr>
100      <tr><td colspan=2 align=center bgcolor=silver>Nilai</td></tr>
101      <tr><td>Tugas </td><td><? echo number_format($tugas,2);?></td></tr>
102      <tr><td>UTS </td><td><? echo number_format($uts,2);?></td></tr>
103      <tr><td>UAS </td><td><? echo number_format($uas,2);?></td></tr>
104      <tr><td>Nilai Akhir </td>
105           <td><?
106                    $na=0.2*$tugas+0.3*$uts+0.5*$uas;
107                    echo number_format($na,2);
108               ?></td></tr>
109      </table>
110           <?
111              }//end of if(!$ada_error)
112           ?>
113  </body>
114  </html>
```

Keterangan Program :

BARIS	PENJELASAN
8	Penginisialisasian variable **$ada_error** dengan nilai **false** yang berguna untuk menyatakan bahwa belum ada error. Variable ini akan berubah nilai menjadi **true** jika ada sebuah kesalahan terjadi.
9	Penginisialisasian variable **$pesan_error**. Variable ini berguna untuk menyimpan pesan error apa saja yang terjadi. Variable ini akan ditampilkan di browser jika terjadi suatu kesalahan.
11	Variable **$nim** di-trim agar kalau ada spasi di sebelah kiri dan kanan (spasi yang tidak berguna) akan dihapus. Contoh jika **$nim** diisi “ 10197025 “ menjadi “10197025”.
13 - 17	Pemeriksaan apakah panjang string **$nim** kurang dari 8. Jika NIM kurang dari 8 huruf maka **$pesan_error** akan ditambah pesan “- Panjang NIM harus 8 Huruf” dan variable **$ada_error** menjadi bernilai **true** karena telah terjadi kesalahan.
18 - 22	Pemeriksaan apakah **$nim** berisi hanya angka saja sebanyak 8 huruf. Jika tidak (!) maka **$pesan_error** akan ditambah pesan “- NIM hanya boleh angka” dan variable **$ada_error** menjadi bernilai **true** karena telah terjadi kesalahan. Pemeriksaan string ini menggunakan suatu metode yang disebut **Regular Expression**. Pola **[0-9]{8}** mempunyai arti sekumpulan bilangan antara 0,1,2…9 yang harus terdiri dari 8 digit.
25 - 31	Pemeriksaan apakah $nim yang telah diinputkan merupakan nim yang sah untuk mahasiswa IF. Caranya pemeriksaannya adalah dengan mengambil 3 digit pertama dari nim (substr($nim,0,3)) dan dimasukan ke variable $jenjang, kemudian dilakukan percabangan jika $jenjang tidak sama dengan “101”, dan $jenjang tidak sama dengan “107” dan $jenjang tidak sama dengan “122” maka nim tersebut dianggap invalid. Jika

	nim tersebut invalid (di luar 101,107,122) maka **$pesan_error** akan ditambah pesan "- 3 Huruf pertama untuk IF adalah 101, 107 dan 122" dan variable **$ada_error** menjadi bernilai **true** karena telah terjadi kesalahan.
33	Variable **$nama** di-trim untuk menghilangkan spasi pada awal dan akhir nama.
34 - 38	Pemeriksaan apakah panjang dari variable nama kurang dari 2 huruf. Jika **$nama** kurang dari dua huruf maka **$pesan_error** akan ditambah pesan "- Nama Minimal 2 Huruf" dan variable **$ada_error** menjadi bernilai **true** karena telah terjadi kesalahan.
40 - 44	Pemeriksaan apakah **$tanggal** belum dipilih (0), atau **$bulan** belum dipilih (0) atau **$tahun** belum dipilih (0). Jika salah satu (or), masih mempunyai nilai 0 berarti pengisian tanggal lahir belum lengkap. Oleh karena itu maka **$pesan_error** akan ditambah pesan "- Tanggal lahir harus diisi dengan lengkap" dan variable **$ada_error** menjadi bernilai **true** karena telah terjadi kesalahan.
45 - 49	Pemeriksaan apakah tanggal lahir valid (sesuai dengan aturan penanggalan). Pemeriksaannya adalah dengan perintah **checkdate**. Fungsi ini menghasilkan nilai true bila tanggal yang dimasukan benar dan akan menghasilkan nilai false jika tanggal tidak sesuai dengan aturan (contoh tgl : 30 Februari 1999). Jika fungsi **checkdate** mempunyai nilai false (`if(!checkdate()`), maka **$pesan_error** akan ditambah pesan "- Tanggal Lahir tidak sah." dan variable **$ada_error** menjadi bernilai **true** karena telah terjadi kesalahan.
51 - 55	Pemeriksaan apakah isi variable **$tugas** berupa angka sebanyak 1 sampai 3 digit. Jika tidak maka **$pesan_error** akan ditambah pesan "- Nilai tugas hanya boleh angka. " dan variable **$ada_error** menjadi bernilai **true** karena telah terjadi kesalahan. Pemeriksaan ini dilakukan dengan teknik regular expression dengan pola pemeriksaan adalah **[0-9]{1,3}** yang artinya deretan 1 sampai 3 buah digit angka 0 – 9.
56 - 60	Pemeriksaan apakah isi **$tugas** tidak sah **($tugas<0)** atau **($tugas>100)**. Jika nilai **$tugas** tidak sah maka **$pesan_error** akan ditambah pesan "-Nilai tugas hanya boleh dari 0 s/d 100" dan variable **$ada_error** menjadi bernilai **true** karena telah terjadi kesalahan.
62 - 66	Pemeriksaan nilai **$uts**. Prosesnya sama dengan baris 51 – 55.
67 - 71	Pemeriksaan range nilai **$uts**. Prosesnya mirip dengan baris 56 – 60.
73 - 77	Pemeriksaan nilai **$uas**. Prosesnya mirip dengan baris 51 – 55.
78 - 82	Pemeriksaan range nilai **$uas**. Prosesnya mirip dengan baris 56 – 60

84 - 89	Pemeriksaan apakah variable **$ada_error** bernilai **true**. Jika benar maka itu menunjukan bahwa minimal ada 1 kesalahan terjadi, sehingga di browser akan muncul daftar error yang dimaksud dan juga menampilkan link yang bertuliskan Kembali untuk kembali ke pengisian data di form.
91 – 92	Percabangan jika variable **$ada_error** bernilai **false**, maka komputer akan menampilkan tabel perhitungan nilai yang dimulai dari baris 96 sampai 109. Percabangan ini dibuka dengan { di baris 92 dan ditutup dengan } di baris 111.
96	Menuliskan kembali data **$nim** yang diisi di form.
97	Menuliskan kembali data **$nama** yang diisi di form.
98 - 99	Menuliskan kembali data **$tanggal, $bulan** dan **$tahun** yang dipilih di form.
101	Menampilkan data **$tugas** yang diisi di form dengan format 2 angka desimal.
102	Menampilkan data **$uts** yang diisi di form dengan format 2 angka desimal.
103	Menampilkan data **$uas** yang diisi di form dengan format 2 angka desimal.
104 - 108	Menghitung Nilai Akhir (**$na**), kemudian menampilkannya dengan format 2 angka desimal.
110 - 112	Akhir dari `if (!$ada_error)`

TUGAS

Di file php sebelumnya, terdapat penggunaan Regular Expression. Tugas kamu adalah :

- Cari dokumentasi mengenai Regular Expression terutama aturan-aturannya.
- Bagaimana penggunaannya dalam PHP

Kemudian buatlah pola Regular Expression untuk pemeriksaan data sebagai berikut :

1. Data alamat e-mail.
2. Data NIM mahasiswa Teknik Informatika, dimana 3 huruf pertama harus 101 atau 107 atau 122.
3. Pemeriksaan apakah dalam suatu string terdapat kata XXX, tetapi tidak deretean X lebih dari 3 buah (XXXX, atau XXXXX).
4. Buatlah pola regular expression untuk pemeriksaan Kode Pos di Bandung, dimana angka pertama harus 4 dan 4 angka berikutnya harus angka semua.
5. Buat pula pola untuk nomor telepon dimana formatnya adalah Kode Area – No TELEPON. Dimana Kode area bisa 3 atau 4 gabungan angka dengan angka 0 sebagai angka pertama (seperti 022, 0265) kemudian diikuti dengan tanda – (minus) kemudian diikuti pula dengan no telepon yang terdiri dari 6 sampai 8 digit angka dimana angka pertama bukan 0. Contoh 021-12345667, atau 0265-740198
6. Mirip dengan No. 5 tetapi penulisan kode area boleh tidak dituliskan (penulisan no telepon local).
7. Pemeriksaan apakah suatu alamat email seseorang merupakan member dari server mail yahoo.com. Contoh jika email : andri@yahoo.com berarti member yahoo dan jika andri@unikom.ac.id berarti bukan member yahoo.
8. Pemeriksaan alamat apakah mengandung kata “Bali”. Contoh yang benar adalah “Jl. Bali no.7” atau “Restoran R.Aja Bali” tetapi bukan “Ci**bali**go Bandung” atau “Jl. **Bali**ng 2”.
9. Pemeriksaan NIM mahasiswa unikom yang tahun masuknya tahun 1999.
10. Pemeriksaan apakah dalam suatu string mengandung kata Bandung atau BDG.

Untuk menghindari tugas yang hanya Copy – Paste, maka tugas ini harus ditulis tangan di kertas folio bergaris.

TIPS : Google Keyword : **Regular Expression** atau **RegEx**