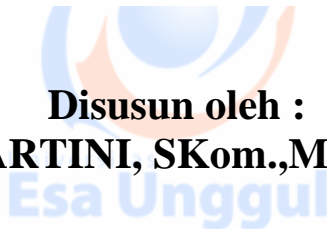


MODUL PRATIKUM - 09
PEMROGRAMAN BERBASIS WEB
(CCP119)



Disusun oleh :
KARTINI, SKom.,MMSI



LABORATORIUM KOMPUTER FAKULTAS ILMU
KOMPUTER
UNIVERSITAS ESA UNGGUL



MODUL PRATIKUM PBW 09

Struktur Program PHP Embed HTML Input-Output

A. TUJUAN

1. Praktikan mengetahui kegunaan Program PHP Embed HTML Input-Output dalam halaman web
2. Praktikan mengetahui penulisan Program PHP Embed HTML Input-Output
3. Praktikan dapat mengaplikasikan Program PHP Embed HTML Input-Output dalam tiap halaman web

B. DASAR TEORI

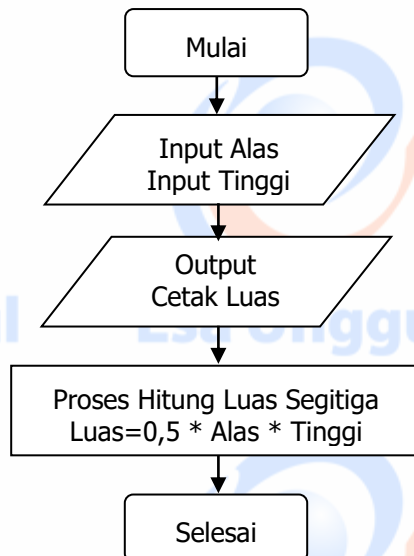
Dasar-dasar Struktur Program Input-Output

Struktur yang paling dasar dalam sebuah pemrograman adalah struktur input-output. Input merupakan interface untuk memasukkan data, kemudian data di olah dan selanjutnya hasil pengolahan ditampilkan pada komponen output.



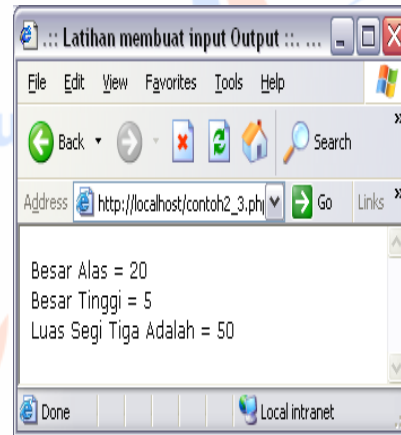
Input-Output Sederhana

Sebagai contoh, kita akan membuat program sederhana untuk menghitung luas segitiga berdasarkan flowchart berikut:



```
<html>
<head>
  <title>::: Latihan membuat input Output :::</title>
</head>
```

```
<body>
<font face=verdana size=2>
<?
//Mendeklarasikan data input
$Alas=20;
$Tinggi=5;
//Proses hitung luas segitiga
$luassegitiga=0.5*$Alas*
$Tinggi;
//Cetak hasil
echo "Beser Alas = $Alas <br>";
echo "Beser Tinggi = $Tinggi
<br>";
echo "Luas Segi Tiga Adalah = $luassegitiga<br>";
?>
</body></html>
```



Jika program tersebut dijalankan maka akan didapatkan hasil seperti pada gambar diatas.

Input-Output dengan Form

Dengan memanfaatkan form yang terdapat dalam HTML, kita dapat membuat program yang lebih interaktif. Sebagai contoh, program di contoh 2.3 akan modifikasi sehingga lebih interaktif.

```
<html>
<head>
<title>.: Menghitung Luas Segi Tiga :.</title></head>
<body>
<font face=verdana>
<center><h3><u>Menghitung Luas Segitiga</u></h3></center>
<form action=luassegitiga.php method=post>
<table align=center bgcolor=#cedadc bordercolor=#cfdbdd cellpadding=5>
<tr>
<td>Masukkan Alas</td>
<td>:</td>
<td><input type=text name=Alas size=10></td>
</tr>
<tr>
<td>Masukkan Tinggi</td>
<td>:</td>
<td><input type=text name=tinggi size=10></td>
</tr>
<tr>
<td colspan=3 align=center><input type=submit value=hitung></td>
</tr>
</table>
</form>
</body></html>
```

Ketika tombol HITUNG diklik, program memanggil file luassegitiga.php (perhatikan pada tag `<form action=luassegitiga.php method=post>`) yang akan digunakan untuk menampung hasil perhitungan. Maka buat lagi file dan beri nama luassegitiga.php dan simpan dalam satu direktori.

File luassegitiga.php

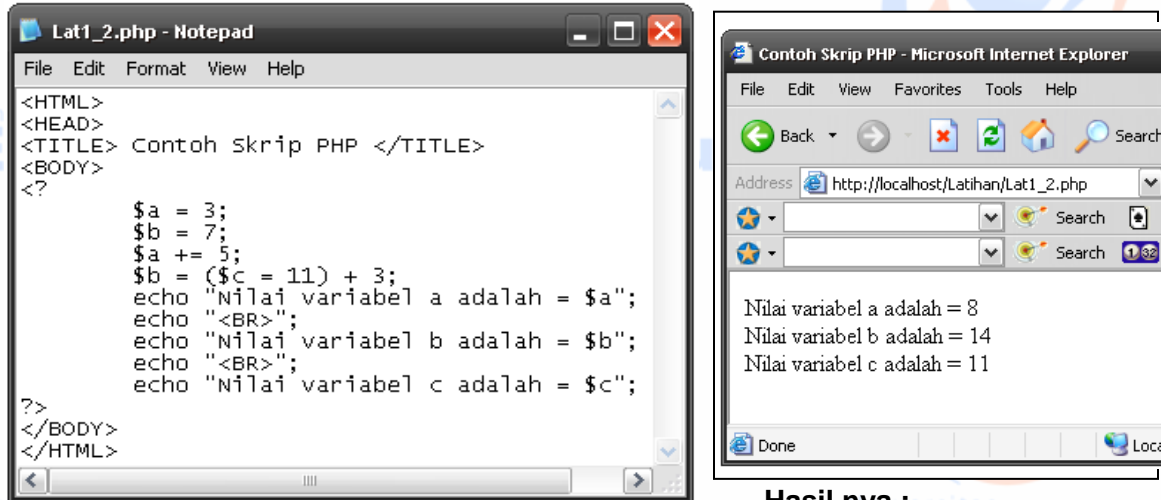
```
<html>
<head>
<title>.: Hasil Perhitungan Luas Segi Tiga :.</title></head>
<body>
<font face=verdana>
<center>
<h3><u>Luas Segitiga</u></h3>
</center>
<form action=contoh2_4.php method=post>
<?
    $luas=0.5*$alasan*$tinggi;
    echo "<table align=center bgcolor=#cedadc bordercolor=#cfdbdd cellpadding=0>";
    echo "<tr>";
    echo "<td>Alas</td>";
    echo "<td>:</td>";
    echo "<td>$alasan</td>";
    echo "</tr>";
    echo "<tr>";
    echo "<td>Tinggi</td>";
    echo "<td>:</td>";
    echo "<td>$tinggi</td>";
    echo "</tr>";
    echo "<tr>";
    echo "<td colspan=3 align=center><input type=submit value=ulang></td>";
    echo "</tr>";
    echo "</table>";
?>
</form>
</body></html>
```



Hasilnya adalah:

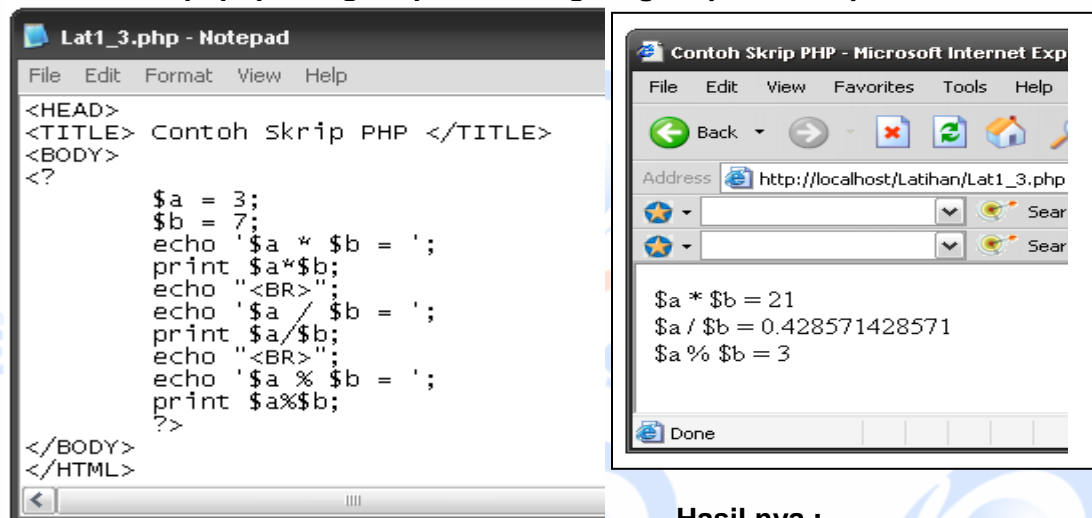
PRAKTEK 1

Lakukan script php meng output kan langsung tanpa form inputan berikut :



Hasil nya :

Lakukan script php meng output kan langsung tanpa form inputan berikut :

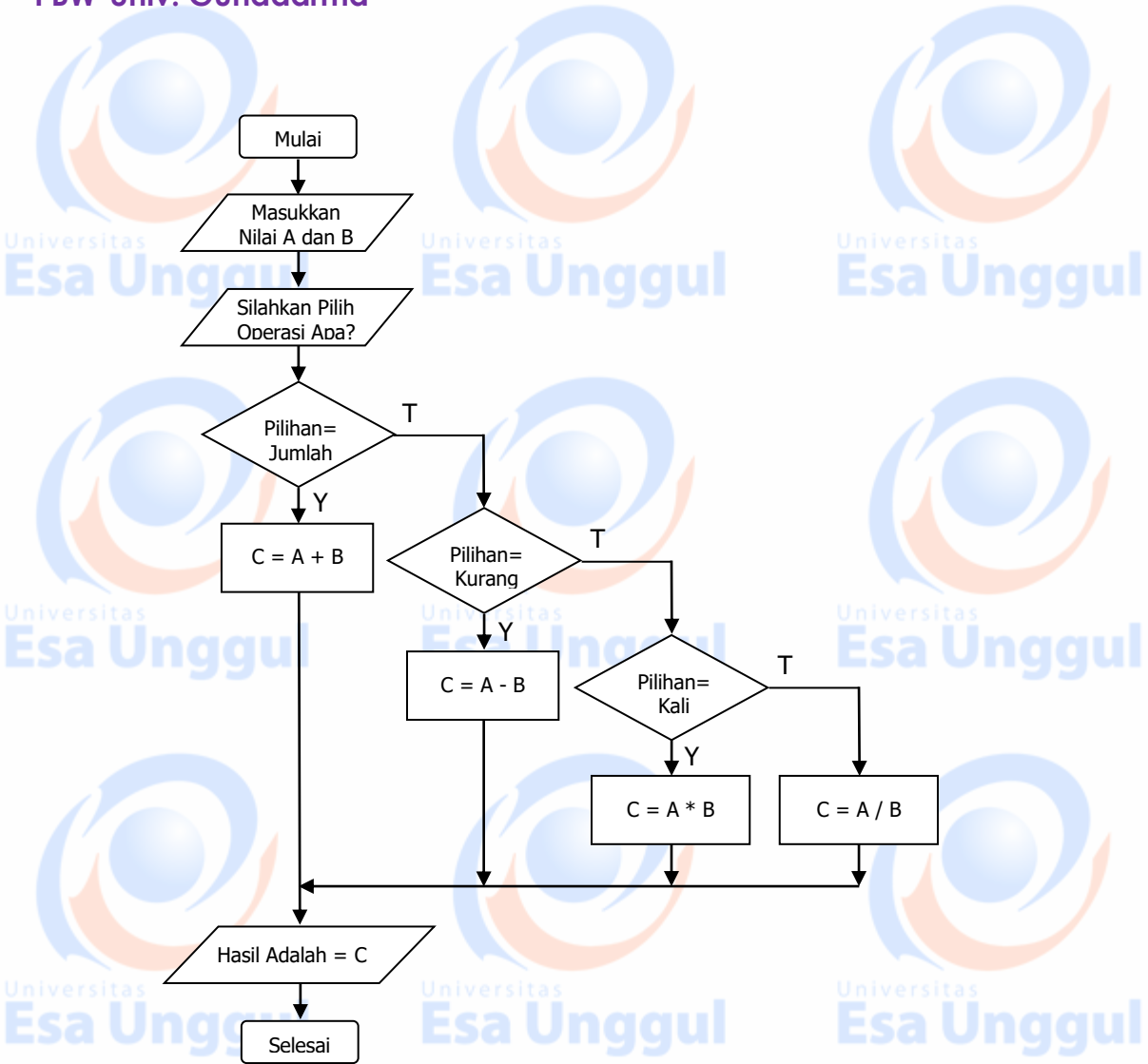


Hasil nya :

PRAKTEK 2

Contoh : 2. Membuat Program kalkulator sederhana

Program yang akan buat ini adalah program untuk melakukan beberapa operasi aritmatika, yaitu penjumlahan, pengurangan, perkalian dan pembagian. Setiap operasi yang dilakukan, program akan meminta masukan nilai dua operand yang akan dihitung dalam hal ini adalah A dan B. Program juga akan meminta masukan dari pemakai untuk memilih operasi apa yang akan dilakukan. Perhatikan flowchart berikut:



Dari flowchart tersebut, dapat kita buat programnya sebagai berikut:

```

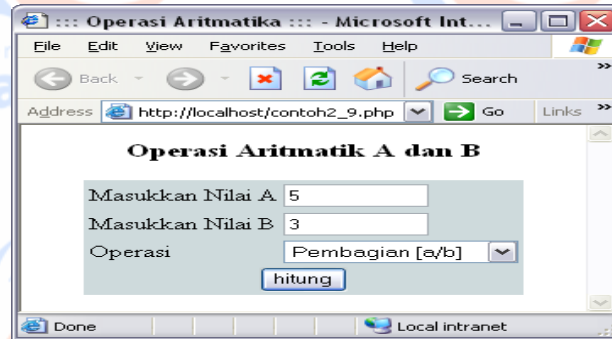
<html>
<head><title>::: Operasi Aritmatika :::</title></head>
<body>
<form action=operasi.php method=post>
<center><h3>Operasi Aritmatik A dan B</h3></center>
<table align=center bgcolor=#cedadc bordercolor=#cfdbdd>
<tr>
<td>Masukkan Nilai A </td>
<td><input type=text name=a size=10</td>
</tr>
<tr>
<td>Masukkan Nilai B </td>
<td><input type=text name=b size=10</td>
</tr>
<tr>
<td>Operasi</td>
<td>
<select name=operasi>

```

```
<option value=1>Penjumlahan [a+b]</option>
<option value=2>Pengurangan [a-b]</option>
```

```
<option value=3>
  Perkalian [a*b]</option>
<option value=4>
  Pembagian [a/b]</option>
</select>
</td></tr>
<tr><td colspan=2 align=center>
<input type=submit value=hitung></td>
</tr>
</table>
</form>
</body></html>
```

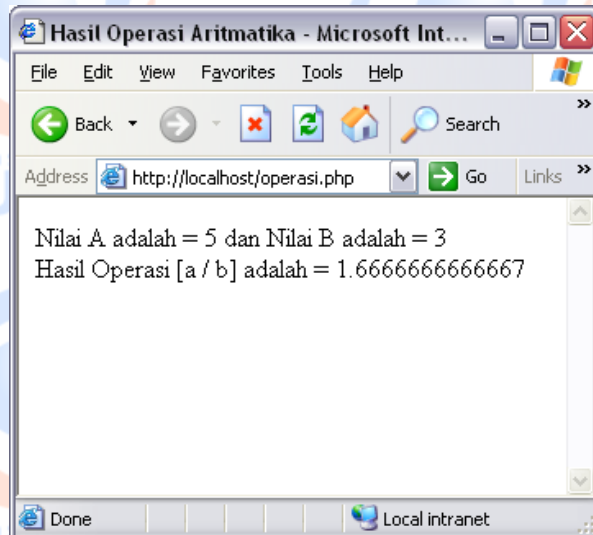
//Hasilnya :



Sekarang buat satu lagi program yang digunakan untuk memproses hasil setelah ada penekanan tombol hitung. File tersebut simpan dengan nama operasi.php.

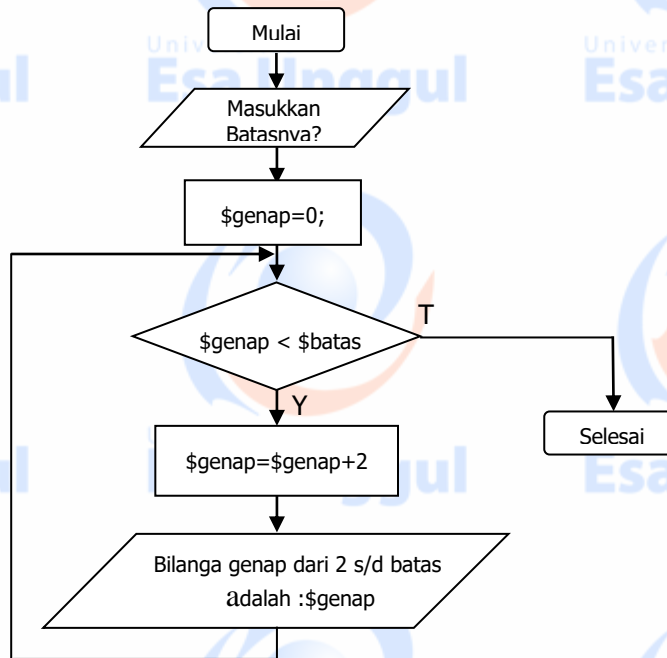
File operasi.php

```
<html>
<head><title>Hasil Operasi Aritmatika</title></head>
<body>
<?
  if ($operasi==1)
  {
    $c=$a+$b;
    $oper='[a + b]';
  }
  elseif ($operasi==2)
  //Hasilnya :
  {
    $c=$a-$b;
    $oper='[a - b]';
  }
  elseif ($operasi==3)
  {
    $c=$a*$b;
    $oper='[a * b]';
  }
  else
  {
    $c=$a/$b;
    $oper='[a / b]';
  }
  echo "Nilai A adalah = $a dan Nilai B adalah = $b<br>";
  echo "Hasil Operasi $oper adalah = $c<br>";
?>
</body></html>
```



Contoh 3 :

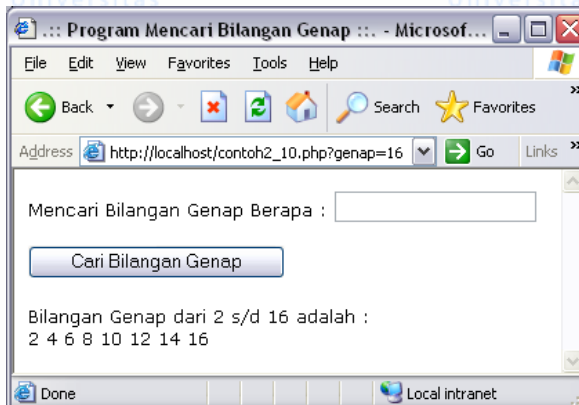
Perhatikan contoh berikut ini, Pada contoh tersebut, program digunakan untuk mencari bilangan genap dari 2 sampai dengan batas tertentu sesuai dengan masukan yang diberikan



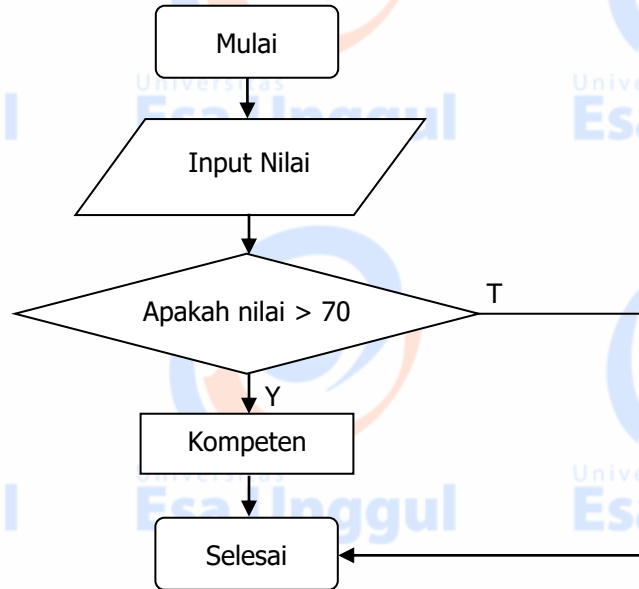
```

<html>
<head><title>::: Program Mencari Bilangan Genap :::</title></head>
<body>
<font face=verdana size=2>
<form>Mencari Bilangan Genap Berapa :
      <input type=text name=genap><br><br>
      <input type=submit value="Cari Bilangan Genap">
</form>
<?php
if (isset($genap))
{
    $bilgen=intval($genap);
    echo "Bilangan Genap dari 2 s/d $bilgen adalah : ";
    echo "<br>";
    $gen=0;
    while ($gen<$genap)
    {
        $gen=$gen+2;
        echo "$gen";
        echo " ";
    }
}
?>
</body></html>
  
```

//Hasilnya :



Contoh 4. :

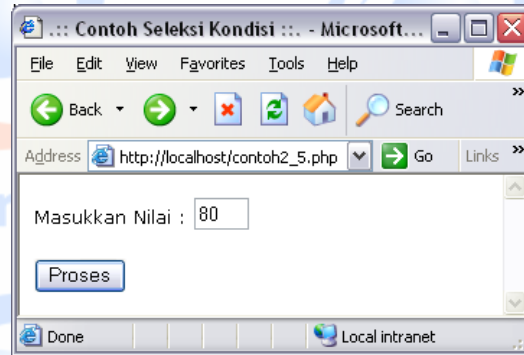


Script :

```

<html>
<head><title> Contoh Seleksi Kondis</title></head>
<body>
<font face=verdana size=2>
<form action=nilai.php
method=post>Masukkan Nilai :
<input type=text name=nilai
size=2><p>
<input type=submit value=Proses>
</form>
</body></html>
  
```

// Hasilnya :

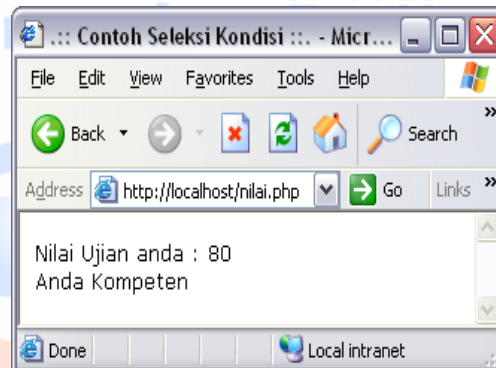


Untuk memproses file diatas, buat script berikut dan simpan dengan nama nilai.php

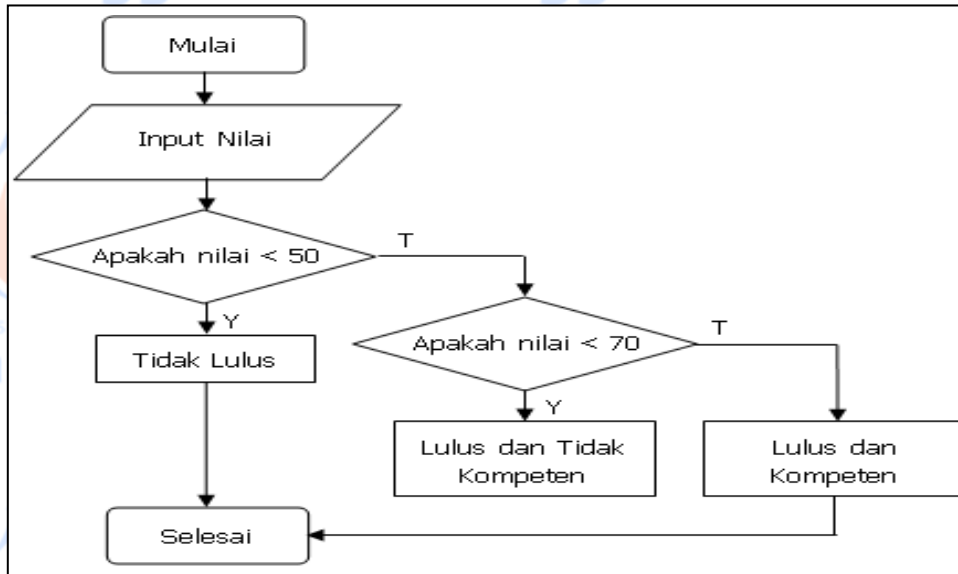
File nilai.php

```

<html>
<head><title> Contoh Seleksi Kondisi </title> </head>
<body>
<font face=verdana size=2>
<?
echo "Nilai Ujian anda : $nilai<br>";
if ($nilai>70)
{ echo "Anda Kompeten<br>"; }
?>
</body>
</html>
  
```



Contoh .5 :



Sebagai contoh, kita modifikasi file **nilai.php** pada contoh 2_5 dan simpan dengan nama **nilai2.php**

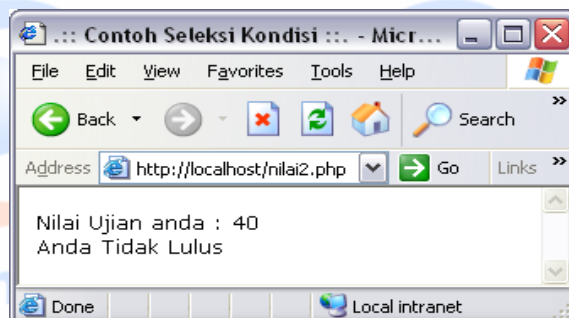
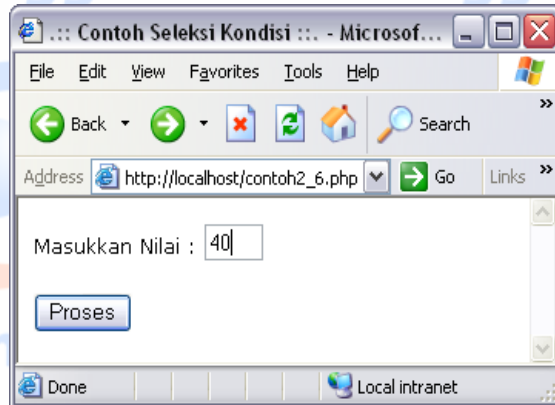
```

<html>
<head><title>:: Contoh Seleksi Kondisi ::</title></head>
<body>
<font face=verdana size=2>
<form action=nilai2.php method=post>
Masukkan Nilai :
<input type=text name=nilai
size=2><p>
<input type=submit value=Proses>
</form>
</body>
</html>
  
```

File nilai2.php:

```

<html>
<head>
<title>:: Contoh Seleksi Kondisi
::</title></head>
<body>
<font face=verdana size=2>
<?
  
```



```

echo "Nilai Ujian anda : $nilai<br>";
if ($nilai<50)
{ echo "Anda Tidak Lulus<br>"; }
elseif ($nilai<70)
{ echo "Anda Lulus Tapi Tidak Kompeten<br>"; }
else
{ echo "Selamat Anda Lulus dan Kompeten"; }
?>
</body></html>

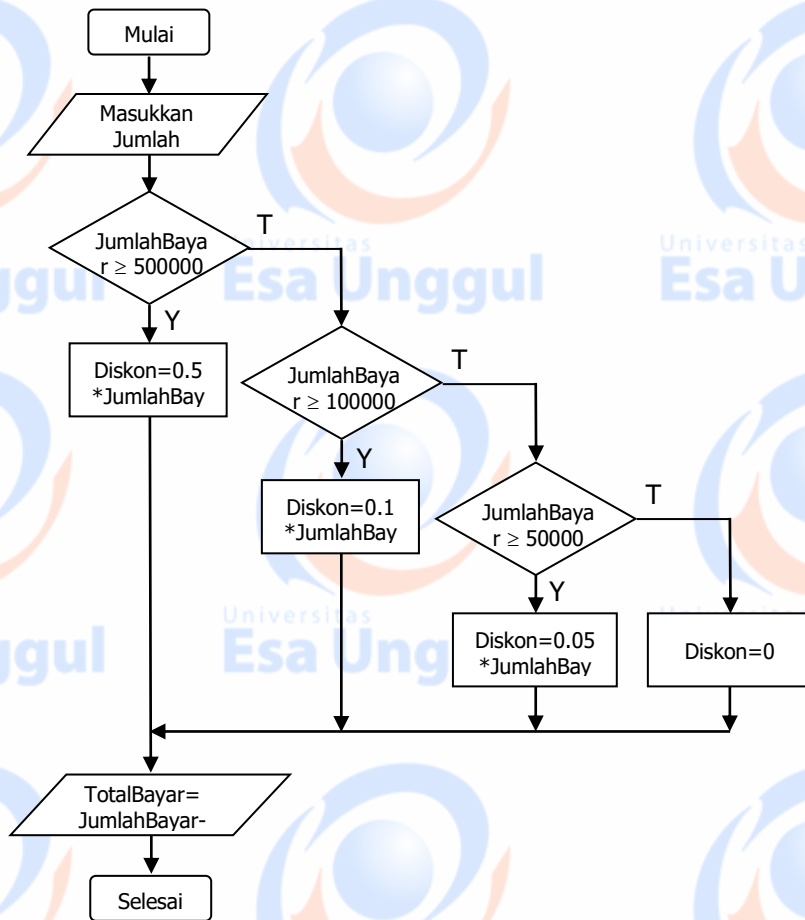
```

Hasilnya adalah seperti gambar di atas:

Contoh 6. : Membuat program penghitung discount

Program ini digunakan untuk mengetahui apakah pembeli dapat diskon atau tidak. Besar diskon tersebut mengikuti aturan sebagai berikut: Jika jumlah bayar ≥ 50.000 dapat diskon 5%, jika jumlah bayar ≥ 100.000 dapat diskon 10%, dan jika jumlah bayar ≥ 500.000 dapat diskon 50%. Selain kriteria tersebut, diskonnya adalah 0%.

Untuk mengerjakan program tersebut, perhatikan flowchart berikut:

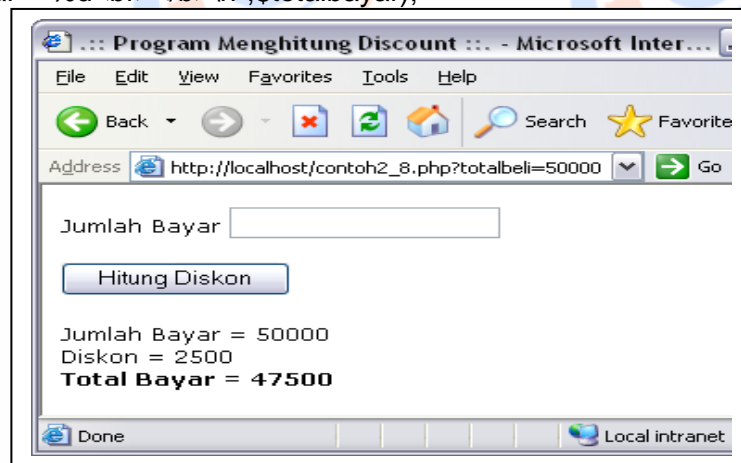


Dari flowchart tersebut dapat kita buat script programnya, sebagai berikut:

PBW-Univ. Gunadarma

```
<html>
<head>
<title>.: Program Menghitung Discount :.</title></head>
<body>
<font face=verdana size=2>
<form>Jumlah Bayar
<input type=text name=totalbeli><br><br>
<input type=submit value="Hitung Diskon">
</form>
<?php
if (isset($totalbeli))
{
    $toyar=intval($totalbeli);
    $diskon=0;
    if ($toyar>=500000)
        $diskon=(0.5*$toyar);
    elseif ($toyar>=100000)
        $diskon=(0.1*$toyar);
    elseif ($toyar>=50000)
        $diskon=(0.05*$toyar);
    else
        print("Maaf Tidak Ada Diskon <br>\n");
    printf("Jumlah Bayar = %d<br>\n",$toyar);
    printf("Diskon = %d<br>\n",$diskon);
    $totalbayar=$toyar-$diskon;
    printf("<b>Total Bayar = %d<br></b>\n", $totalbayar);
}
?>
</body>
</html>
```

Hasilnya :



Procedure dan Fungsi

Dalam pembuatan program sering kali dibutuhkan beberapa perintah yang digunakan berulang kali. Hal ini dapat dihindari dengan *subrutin*. Subrutin adalah sekumpulan perintah yang diberi nama dan dapat dipanggil sewaktu-waktu. Dalam pemrograman terdapat dua *subrutin*, yaitu prosedur dan fungsi. Standar penulisan fungsi adalah:

```
function nama_fungsi(argumen)
```



kode perintah

contoh:

```
function operasi_jumlah($x,$y)
{
    z=x+y;
    echo (z);
}
```

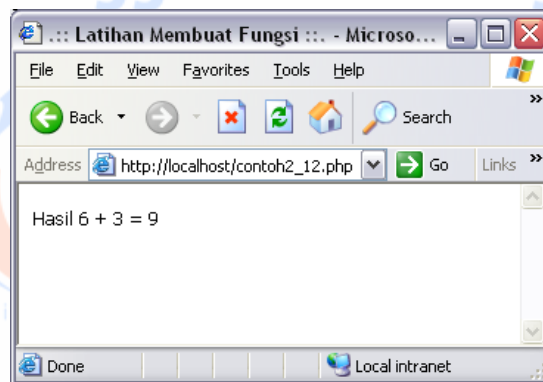
Beberapa hal yang perlu diperhatikan dalam pembuatan fungsi, yaitu:

- Nama fungsi tidak boleh sama dengan nama-nama fungsi yang sudah ada dalam PHP.
- Hanya boleh terdiri dari huruf, angka dan garis bawah
- Tidak boleh diawali dengan angka

Contoh :

```
<html>
<head>
    <title>:: Latihan Membuat Fungsi ::</title></head>
<body>
<font face=tahoma size=2>
<?>
```

```
//Fungsi yg dipanggil //Hasilnya :
function operasi_jumlah($a,$b)
{
    $c=$a+$b;
    echo ("Hasil $a + $b = $c");
}
//Program Utama
//Untuk Memanggil Fungsi
operasi_jumlah(6,3);
?>
</body>
</html>
```



REQUIRE

Function Require digunakan untuk membaca nilai variable dan fungsi-fungsi dari sebuah file lain.

Cara penulisan function Require adalah : **require(namafile);**

Function Require ini tidak dapat dimasukkan didalam suatu struktur looping misalnya while atau for. Karena hanya memperbolehkan pemanggilan file yang sama tersebut hanya sekali saja.

Contoh : nama file : contoh2_13.php

```
<?php
$a="Saya sedang belajar PHP";
function tulistebal($teks)
{
    echo("<b>$teks</b>");
}
?>
```

//Hasilnya :



Contoh :

```
<?php
require("contoh2_13.php");
tulistebal("Ini adalah tulisan tebal");
echo("<br>");
echo($a);
?>
```

INCLUDE

Function Include akan menyertakan isi suatu file tertentu. Include dapat diletakkan didalam suatu looping misalkan dalam statement for atau while.

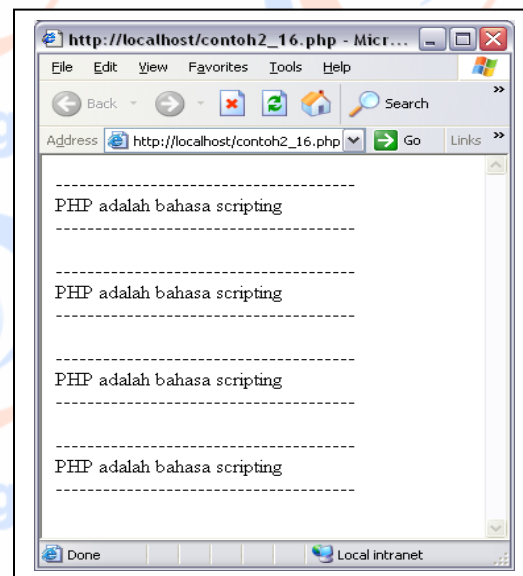
Contoh : nama file : contoh2_15.php

```
<?
echo("-----<br>");
echo("PHP adalah bahasa scripting<br>");
echo("-----<br>");
echo("<br>");
?>
```

Contoh :

```
<?php
for ($b=1; $b<5; $b++)
{ include("contoh2_15.php"); }
?>
```

Hasilnya adalah:

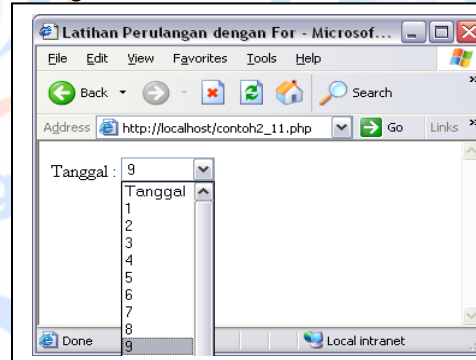


PRAKTEK

```
<html>
<head><title>Latihan Perulangan dengan For</title></head>
<body>
Tanggal :
<select name=tanggal>
<option value=0 selected>Tanggal
<?php
```

```
//Bentuk Pilihan tanggal 1 sampai dengan 31
for ($i=1;$i<32;$i++)
echo "<option value=$i>$i";
?>
</option>
</select>
</body></html>
```

Hasilnya :



ARRAY:

- o Array dalam PHP dideklarasikan dengan sangat luwes, misalnya :
`$nama=array("JOKO","BUDI","ARI");` maka indeks dimulai dari 0
`$nama=array(1=>"JOKO","BUDI","ARI");` maka indeks dimulai dari 1
- o sehingga program :

CONTOH 1 :

```
<?php
$nama=array(1=>"JOKO","BUDI","ATIK");
$i=1;
while($i<=3)
{print("<b> $nama[$i] ");
$i++;
}
?>
```

// Hasilnya :



CONTOH : 2

Contoh 1, SAMA DENGAN berikut penulisan script nya :

```
<?php
$nama["A"]="JOKO";
$nama["B"]="BUDI";
$nama["C"]="ATIK";
$huruf=array(1=>"A","B","C");
$i=1;
while($i<=3)
{ print("<b> $huruf[$i] isi : $nama[$huruf[$i]]<br> "); $i++; }
?>
```

INDEKS STRING DALAM ARRAY :

PHP juga memungkinkan kita membuat array 2 dimensi dengan indeks campuran dan dengan banyak elemen tiap barisnya tidak sama

```
Contoh : $tamu=array("senin">array("JOKO","ENDRO","DONO"),
"selasa">array("SITI","RINI"),
"rabu">array("MULYANTO"));
```

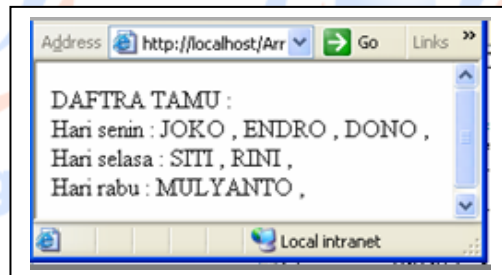
CONTOH 3 :

```
<?php
$tamu=array("senin">array("JOKO","ENDRO","DONO"),
"selasa">array("SITI","RINI"),
"rabu">array("MULYANTO"));
```

PBW-Univ. Gunadarma

```
print("DAFTAR TAMU :<BR>");  
while(list($indeks1,$nilai1)=each($tamu))  
{print("Hari $indeks1 : ");  
  while(list($indeks2,$nilai2)=each($nilai1))  
    print("$nilai2 , ");  
  print("<BR>");  
}  
?>
```

// Hasilnya :



Fungsi Mail

Digunakan untuk mengirimkan e-mail ke alamat e-mail tertentu.

Sintaks:

```
mail($penerima,$subject,$isi_email[, $header]);
```

Contoh:

```
$pengirim = "From: saya@email.com";  
$tujuan = "anonkuncoro@yahoo.com";  
$subject = "Pemberitahuan";  
$isi = "Ini adalah percobaan pengiriman e-mail dengan menggunakan PHP";  
mail($to,$subject,$isi,$pengirim);
```



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul