

## MODUL 5

### FUNGSI

---

Function / fungsi adalah sebuah model PHP untuk mengelompokkan program sehingga pada program tersebut akan dipecah menjadi bentuk program – program kecil yang akan dipanggil di dalam program utama. Cara pemanggilan fungsi cukup dengan menuliskan nama dari fungsi tersebut. Fungsi ini bisa dipanggil berulang – ulang dan dapat diletakkan di bagian manapun. Aturan pembuatan fungsi sama dengan aturan dalam pembuatan variabel. Berikut merupakan jenis-jenis fungsi:

#### Cara penulisan fungsi

```
function nama_fungsi(parameter1, parameter 2, parameter n){  
    statement;  
}
```

#### Jenis-Jenis Fungsi

##### 1. Fungsi dengan Parameter

Parameter digunakan sebagai input untuk fungsi yang selanjutnya diolah oleh fungsi tersebut. Jumlah parameter dari fungsi boleh lebih dari satu. Untuk memisahkan antar parameter digunakan tanda koma (,).

##### 2. Fungsi dengan Return Value

Sebuah fungsi juga dapat mengembalikan suatu nilai. Dalam sebuah fungsi hanya boleh terdapat satu return value. Nilai yang dikembalikan dapat berupa suatu bilangan real dan string.

##### 3. Fungsi TANPA Parameter dan Return Value

Sebuah fungsi yang tidak memiliki parameter dan return value biasanya digunakan untuk menampilkan suatu mekanisme atau tampilan, di mana mekanisme tersebut tidak memerlukan inputan dan nilai kembalian / return value.

## CONTOH PENERAPAN FUNGSI

---

### 1. Fungsi dengan parameter

```
<?php
function cetak_ganjil ($awal, $akhir) {
    for ($i=$awal; $i<$akhir; $i++) {
        if ($i%2 == 1) {
            echo "$i";
        }
    }
}

//pemanggilan fungsi
$a = 10;
$b = 50;
echo "<b>Bilangan ganjil dari $a sampai $b : </b><br>";
cetak_ganjil($a, $b);
?>
```

#### Penjelasan:

Di atas merupakan contoh penerapan fungsi untuk menampilkan bilangan ganjil sesuai dengan parameter yang dimasukkan. Parameter tersebut berfungsi untuk menentukan batasan bilangan ganjil yang akan ditampilkan.

### 2. Fungsi dengan return value

```
<?php
function luas_lingkaran ($jari) {
    return 3.14 * $jari * $jari;
}

//pemanggilan fungsi
$r = 10;
echo "Luas lingkaran dengan jari-jari $r = ";
echo luas_lingkaran($r);
?>
```

#### Penjelasan:

Di atas merupakan fungsi yang digunakan untuk menghitung luas lingkaran dengan menerapkan return value. Return value tersebut digunakan untuk mengembalikan nilai hasil perhitungan luas lingkaran, sehingga output dari pemanggilan fungsi tersebut berupa luas lingkaran.

### 3. Fungsi tanpa parameter dan return value

```
<?php
function head() {
    echo '
        <link rel="stylesheet" type="text/css" href="style/style.css">
        <link rel="stylesheet" type="text/css" href="style/font-awesome.css">
    ';
}

//pemanggilan fungsi
echo luas_lingkaran($r);
?>
```

#### Penjelasan:

Di atas merupakan penerapan fungsi tanpa parameter dan return value. Kegunaan dari contoh script di atas adalah untuk menambahkan komponen-komponen import data yang biasanya terdapat di dalam <head></head>. Komponen tersebut kemudian dijadikan satu dalam bentuk fungsi agar dapat dipanggil di halaman lain yang membutuhkan komponen-komponen tersebut. Dengan kata lain agar tidak terjadi duplikasi script di setiap halaman. Hal ini juga mempermudah kita dalam penambahan, pengurangan, maupun perubahan komponen, karena kita tidak perlu mengubahnya di setiap halaman, tapi hanya perlu mengubah di fungsi ini saja. Secara otomatis halaman lain akan mengikuti perubahan tersebut.

## Halaman "index.php"

```
1  <?php
2      session_start();
3      if(isset($_SESSION["user"])){
4          header("location:beranda.php");
5      }
6      include ("function.php");
7
8  ?>
9
10 <!DOCTYPE html>
11 <html>
12     <head><title>LOGIN</title></head>
13     <body>
14         <?php
15             if(isset($_GET["error"])){
16                 if($_GET["error"] == "wrong"){
17                     echo '<h3>Username dan Password salah!</h3>';
18                 }
19             }
20             if(isset($_GET["notif"])){
21                 if($_GET["notif"] == "logout"){
22                     echo '<h3>Berhasil logout!</h3>';
23                 }
24             }
25
26             if(isset($_POST['submit'])){
27                 echo do_login($_POST['username'], $_POST['password']);
28             }
29         ?>
30
31         <form action="<?php echo $_SERVER['PHP_SELF'];?>" method="post">
32             Username: <input type="text" name="username"><br >
33             Password: <input type="password" name="password"><br>
34             <input type="submit" name="submit" value="SIGN IN">
35         </form>
36
37     </body>
38 </html>
```

### Halaman "beranda.php"

```
1 <?php
2     session_start();
3     include("function.php");
4     echo check_login();
5 ?>
6
7 <!DOCTYPE html>
8 <html>
9     <head><title>praktik login session</title></head>
10    <body>
11        <h1>Selamat Datang!</h1>
12        <h3><a href="profil.php">Edit Profil</a></h3>
13        <h3><a href="logout.php">Logout!</a></h3>
14    </body>
15 </html>
```

### Halaman "function.php"

```
1 <?php
2     function do_login($username, $password){
3
4         //cek kondisi jika username dan password salah
5         if($username != "admin" || $password != "admin"){
6             header("location:index.php?error=wrong");
7         }
8
9         //cek kondisi jika username dan pasword benar
10        if($username == "admin" && $_POST["password"] == "admin"){
11            $_SESSION["user"] = $username;
12            $_SESSION["pass"] = $password;
13
14            header("location:beranda.php");
15        }
16    }
17
18    function check_login(){
19        //cek kondisi login session
20        if(!isset($_SESSION["user"])){
21            header("location:index.php");
22        }
23    }
24 ?>
```

## TUGAS

Sempurnakan tugas sebelumnya tentang pembuatan halaman web pada materi "perulangan" agar lebih dinamis dan modular dengan menambahkan FUNCTION untuk menampilkan elemen-elemennya.

😊SELAMAT MENGERJAKAN 😊