Bab 11 Teknik Modularitas

Konsep modularitas dalam programming diperlukan untuk mempermudah dalam pengorganisasian script/program. Adapun prinsip dari konsep ini adalah meletakkan beberapa perintah yang menjalankan suatu tugas khusus ke dalam modul atau file script tersendiri. Setiap kali modul tersebut dibutuhkan, kita hanya menyisipkan modul tersebut ke dalam script yang sedang dibuat dengan cara memanggilnya.

Adapun cara memanggil modul dengan menggunakan perintah

include "namafilemodul";

Perhatikan gambaran penggunaan konsep modularitas berikut ini:

Misalkan Anda membuat modul dan disimpan dalam script bernama **modulku.php**. Selanjutnya Anda sedang membuat script tertentu katakanlah diberinama file **scriptku.php**. Kebetulan dalam scriptku.php Anda membutuhkan modulku.php, maka skema yang terjadi adalah:



Dari skema di atas, jelas tampak bahwa bila Anda memberikan perintah include "modulku.php"; dalam scriptku.php, maka akan identik dengan bila Anda menyisipkan script yang ada dalam modulku.php ke dalam scriptku.php secara manual.

Contoh:

Dalam contoh ini kita akan menggunakan konsep modularitas untuk membuat script operasi aritmatika penjumlahan dan pengurangan dari 2 buah bilangan. Untuk input bilangannya disimpan dalam modul tersendiri, misalkan dinamakan bilangan.php. Selanjutnya modul ini akan diincludekan ke dalam script penjumlahan dan pengurangan.

osthan

bilangan.php

<?php

\$bil1 = 10; \$bil2 = -5;

?>

jumlah.php

<?php

```
include "bilangan.php";
```

```
$hasil = $bil1 + $bil2;
echo "Hasil penjumlahannya adalah : ".$hasil;
```

?>

Bentuk script jumlahkan.php di atas akan identik dengan script berikut ini

<?php

\$bil1 = 10; \$bil2 = -5;

\$hasil = \$bil1 + \$bil2; echo "Hasil penjumlahannya adalah : ".\$hasil;

kurang.php

<?php

include "bilangan.php";

\$hasil = \$bil1 - \$bil2;

echo "Hasil pengurangannya adalah : ".\$hasil;

?>

Script kurang.php di atas akan identik dengan

<?php

\$bil1 = 10; \$bil2 = -5;

\$hasil = \$bil1 - \$bil2; echo "Hasil pengurangannya adalah : ".\$hasil;

?>

Penerapan Teknik Modularitas

Teknik modularitas dapat dipergunakan dalam pengaturan layout halaman web supaya lebih mudah. Sebelum Anda mengenal teknik modularitas, apa yang Anda lakukan bila diminta membuat halaman web sejumlah 20 buah dengan desain layout yang sama, namun kontennya berbeda? Tentu Anda akan membuat satu buah desain template untuk layout, lalu Anda tinggal mengganti kontennya saja.

OK.. itu mudah, namun bagaimana seandainya dari 20 buah halaman web tersebut Anda ingin ubah desainnya? Tentu Anda harus mengubah semua halaman satu persatu. Bagaimana bila terdapat 100 buah halaman web dan kesemuanya Anda ingin ubah desainnya? Wah repot ya kalau satu persatu.

Nah... di sinilah kita akan gunakan teknik modularitas. Dengan teknik modularitas ini, Anda dapat mengubah desain layout dari banyak halaman web dengan satu langkah saja secara cepat.

Perhatikan contoh berikut ini:

Misalkan kita buat desain layout untuk halaman web kita sebagai berikut:

```
<html>
<head>
<title>Halaman Webku</title>
</head>
<body background="yellow">
<h1><font face="arial">...</font></h1>
<font face="verdana"> ...</font>
</body>
</html>
```

ada Ar

Nah... misalkan kita akan membuat 3 halaman web menggunakan layout di atas. Maka caranya, kita cari blok dalam HTML layout yang nantinya **akan diisi dengan konten yang berbeda-beda** untuk masing-masing halaman.

```
<html>
<head>
<title>Halaman Webku</title>
</head>
<body bgcolor="yellow">
<h1><font face="arial">...</font></h1>
<font face="verdana">...</font>
</body>
</html>
```

Keterangan:

Bagian yang diblok di atas merupakan bagian tag yang nantinya akan diisi dengan konten yang berbeda-beda untuk setiap halamannya.

Langkah selanjutnya, kita pindahkan tag HTML mulai bagian paling atas dari layout sampai dengan sebelum tag yang dicetak tebal. Tag-tag tersebut kita pindahkan ke dalam modul tersendiri, misalkan dinamakan **header.php**

header.php

```
<html>
<head>
<title>Halaman Webku</title>
</head>
<body bgcolor="yellow">
```

Kemudian pindahkan tag setelah bagian yang dicetak tebal dalam layout sampai dengan tag terakhir ke dalam modul tersendiri dan simpan dengan nama **footer.php**.

footer.php

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

Nah... selanjutnya untuk setiap halaman yang ingin Anda buat, berikan include "header.php" dan include "footer.php";

hal1.php

<?php include ''header.php''; ?>

<h1>Ini Halaman 1</h1> Ini isi halaman 1. Ini isi halaman 1

```
<?php
include ''footer.php'';
?>
```

hal2.php

```
<?php
include ''header.php'';
?>
```

```
<h1><font face="arial">Ini Halaman 2</font></h1>
<font face="verdana">Ini isi halaman 2. Ini isi halaman 2</font>
```

```
<?php
include ''footer.php'';
?>
```

hal3.php

```
<?php
include ''header.php'';
?>
```

```
<h1><font face="arial">Ini Halaman 3</font></h1>
<font face="verdana">Ini isi halaman 3. Ini isi halaman 3</font>
```

<?php include ''footer.php''; ?>

Nah... dengan demikian, kita akan peroleh 3 halaman web yang pasti memiliki layout yang sama. Terus... bagaimana bila kita ingin mengubah layoutnya? Misal kita ubah warna background menjadi hijau. Yang Anda lakukan hanyalah mengedit modul header.php karena pengaturan warna background terdapat pada modul tersebut.

200 1

header.php

```
<html>
<head>
<title>Halaman Webku</title>
</head>
<body bgcolor=''green''>
```

Secara ajaib ketiga halaman web tadi akan otomatis backgroundnya berubah menjadi hijau semuanya.

Selain cara di atas, Anda juga bisa menggunakan cara yang lain dalam pengaturan layout. Misalkan konfigurasi pengaturan layout seperti jenis font, dan warna background Anda letakkan dalam modul tersendiri.

config.php

```
<?php
$warnaBackground = "yellow";
$jenisFontHeading = "arial";
$jenisFontParagraf = "verdana";
?>
```

header.php

```
<?php
include "config.php";
?>
```

<html>

```
<head>
<title>Halaman Webku</title>
```

```
</head>
```

<body bgcolor="<?php echo \$warnaBackground; ?>">

footer.php

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

hal1.php

```
<?php
include ''header.php'';
?>
```

<h1><font face="<?php echo \$jenisFontHeading; ?>">Ini Halaman 1</h1><font face="<?php echo \$jenisFontParagraf; ?>">Ini isi halaman 1. Ini isi halaman 1

```
<?php
include ''footer.php'';
?>
```

hal2.php

```
<?php
include ''header.php'';
?>
```

```
<h1><font face="<?php echo $jenisFontHeading; ?>">Ini Halaman 2</font></h1><font face="<?php echo $jenisFontParagraf; ?>">Ini isi halaman 2. Ini isi halaman 2</font>
```

```
<?php
include ''footer.php'';
?>
```

hal3.php

```
<?php
include ''header.php'';
?>
```

```
<h1><font face="<?php echo $jenisFontHeading; ?>">Ini Halaman 3</font></h1><font face="<?php echo $jenisFontParagraf; ?>">Ini isi halaman 3. Ini isi halaman 3</font>
```

```
<?php
include "footer.php";
?>
```

Nah... dengan demikian, bila Anda ingin mengubah layout ketiga halaman web maka Anda cukup mengubah modul **config.php**

Sekarang kita lanjutkan, bagaimana bila kita mengintegrasikan konsep modularitas yang terkait dengan pengaturan layout di atas dengan form processing?

OK... untuk contoh sederhananya, saya ambil contoh script menjumlahkan dua buah bilangan dengan inputnya melalui form.

Misalkan kita ambil desain layout halaman webnya seperti contoh sebelumnya, yaitu terdiri dari modul header.php dan footer.php nya sebagai berikut:

Sinan

config.php

```
<?php
$warnaBackground = "yellow";
$jenisFontHeading = "arial";
$jenisFontParagraf = "verdana";
?>
```

header.php

```
<?php
include "config.php";
?>
```

<html>

```
<head>
```

<title>Penjumlahan Bilangan</title>

```
</head>
```

<body bgcolor="<?php echo \$warnaBackground; ?>">

footer.php

</body> </html>

Selanjutnya kita buat form yang terletak dalam halaman yang menggunakan layout tersebut

form.php

<?php include "header.php"; ?>

<h1><font face="<?php echo \$jenisFontHeading; ?>">Input Bilangan</h1><font face="<?php echo \$jenisFontParagraf; ?>">Masukkan dua buah bilangan pada form berikut ini

<form method="post" action="proses.php"> Bilangan 1 <input type="text" name="bil1" />
 Bilangan 2 <input type="text" name="bil2" />
 <input type="submit" name="submit" value="Jumlahkan" /> </form>

<?php include "footer.php"; ?>

proses.php

<?php include "header.php";

\$bil1 = \$_POST['bil1']; \$bil2 = \$_POST['bil2']; \$hasil = \$bil1 + \$bil2; ?>

<h1><font face="<?php echo \$jenisFontHeading; ?>">Output</h1><font face="<?php echo \$jenisFontParagraf; ?>">Hasil penjumlahannya adalah <?php echo \$hasil; ?>

<?php include "footer.php"; ?>

Tugas

1. Coba buatlah desain layout sebuah halaman web (tanpa menggunakan tool apapun selain Notepad atau Notepad++) dengan tampilan berikut ini:

Nama Situs Anda	
Link Link Link	Heading(Judul Konten) Konten/isi
copyright	

2. Dengan menggunakan teknik modularitas, coba buat modul config.php, header.php dan footer.php dari hasil soal nomor 1. Modul config.php digunakan untuk mengatur konfigurasi seperti jenis font, ukuran font, warna background dll.

Keterangan:

Untuk bagian "Nama Situs Anda", "Link" dan "Copyright" bersifat fixed, artinya kelima halaman tersebut memiliki tampilan yang sama. Sehingga dalam hal ini bagian yang berubah-ubah untuk setiap halaman hanyalah bagian "Heading dan Konten" saja.

- 3. Dari hasil soal nomor 2, buat 5 halaman web berisi konten yang berbeda. Kelima halaman web tersebut harus terintegrasi dengan link yang ada pada bagian link (lihat desain layout dari soal no. 1)
- 4. Coba cari dan download lah sebuah layout instant dari internet, disarankan berasal dari <u>http://freecsstemplates.org</u>, buatlah modul header.php dan footer.php. Lalu buatlah 5 halaman web menggunakan layout tersebut dengan teknik modularitas.

Keterangan:

Biasanya desain layout template terdiri dari 4 bagian, yaitu: bagian *header* (bagian yang terletak di atas konten), *footer* (bagian yang terletak di bawah konten), *sidebar* (bagian yang terletak di samping kiri atau kanan konten), dan bagian konten itu sendiri. Nah.. dari keempat bagian, hanya bagian konten saja yang berubah-ubah, sedangkan bagian yang lain adalah fixed.

- 5. Kerjakan kembali tugas No. 2 bab 9 dengan mengintegrasikannya bersama layout yang Anda buat pada soal no. 1 di atas.
- 6. Kerjakan kembali tugas No. 8 bab 10 dengan mengintegrasikannya bersama layout yang Anda buat pada soal no. 1 di atas.

with an a start of the start of