

Nama : M.Thoriq Al Farqo

NIM : 08018122

Kelas A

Tugas P.web : Menerjemahkan E-book HTML (hal 505 - 506)

Menggunakan Logika Operator untuk mengetes dua atau lebih kondisi

Sebagai script yang kadang menjadi lebih kompleks, maka anda kadang-kadang akan ingin prosesor menguji lebih dari satu kondisi pada suatu waktu. Misalnya, script Anda mungkin menguji apakah pengunjung pendapatan tahunan lebih besar dari \$ 25.000 dan kurang dari \$ 75.000. Demikian juga, Anda mungkin menguji apakah pengunjung memiliki sejarah kredit yang baik atau jika pengunjung adalah homebuyer pertama kali. Untuk melakukan operasi seperti itu, Anda akan menggunakan PHP logical : AND operator (& & atau "and") dan operator logika OR (| | atau "or"). Bila script anda menggunakan AND logika atau operator logika OR untuk menguji lebih dari satu syarat, Anda akan menempatkan masing-masing kondisi dalam tanda kurung, seperti yang ditunjukkan dengan mengikuti pernyataan if, yang menggunakan operator logika OR untuk menguji apakah pengunjung yang memiliki kredit yang baik atau merupakan homebuyer pertama kali:

```
if (($hasGoodCredit) || ($firstTimeHomeBuyer))  
statement;
```

Dalam cara yang sama, pernyataan berikut menggunakan operator AND Logic untuk menentukan apakah pengunjung pendapatan di kisaran \$ 25.000 hingga \$ 75.000:

```
if (($income >= 25000) && ($income <= 75000))  
statement;
```

Bila script anda menggunakan logika ANDoperator, semua kondisi dalam pernyataan harus benar untuk kondisi keseluruhan untuk mengevaluasi sebagai benar. Jika kondisi apapun palsu, kondisi keseluruhan menjadipalsu. Sebagai contoh, jika pendapatan pengunjung tidak lebih besar dari atau sama dengan \$ 25.000, kondisi sebelumnya adalah palsu. Demikian juga, jika pendapatan pengunjung lebih besar dari \$ 75.000, kondisi salah. Agarkondisi benar,

pendapatan pengunjung harus lebih besar dari atau sama dengan \$ 25.000 dan kurang dari atau sama dengan \$ 75.000. Untuk pernyataan yang menggunakan operator logika OR untuk mengevaluasi sebagai benar, hanya satu syarat harus benar. Misalnya, jika pengunjung memiliki kredit yang telah ditetapkan baik, kondisi sebelumnya adalah benar. Jika pengunjung membeli rumah pertama atau dia, kondisi benar. Apalagi, jika pengunjung memiliki kredit yang baik dan adalah membeli rumah pertama atau dia, kondisi benar. Satu-satunya waktu kondisi akan palsu adalah jika pengunjung tidak memiliki kredit yang baik dan tidak membeli rumah pertama, dalam hal kedua kondisi adalah palsu, dan kondisi secara keseluruhan mengevaluasi yang salah.

Menggunakan Pernyataan *FOR* untuk mengulangi Pernyataan yang spesifik

Salah satu operasi yang paling umum dalam script anda adalah untuk menginstruksikan prosesor PHP mengulang satu atau lebih pernyataan sejumlah waktu yang tertentu, dalam hal ini Anda akan menggunakan pernyataan *For*. Statement pada PHP memiliki tiga ekspresi dan pernyataan sederhana dengan sintaks berikut:

```
for (expression1, expression2, expression3) statement;
```

PHP mengevaluasi ekspresi pertama (*expression1*) sekali di awal loop. Pada awal setiap iterasi, prosesor mengevaluasi ekspresi kedua (*expression2*). Jika ekspresi kedua mengevaluasi sebagai benar, prosesor PHP mengeksekusi pernyataan atau kelompok laporan dalam blok mulai akhir (*{}*) seperti ini:

```
for (expression1, expression2, expression3)  
{  
statement;  
statement;  
...  
statement;  
}
```

Sebaliknya, jika ekspresi kedua mengevaluasi salah, prosesor berhenti menjalankan PHP untuk loop. Pada akhir setiap iterasi, prosesor PHP mengevaluasi *expression3*. Anda dapat menggunakan PHP ekspresi hukum yang Anda inginkan dalam untuk loop dan salah satu atau semua tiga dari ekspresi boleh kosong. Namun, ekspresi kedua kosong (*expression2* di

sebelumnya

sintaks) berarti prosesor PHP akan mengeksekusi untuk loop tanpa batas. Jadi, berikut ini untuk loop dengan ekspresi kedua dan ketiga kosong akan berakhir hanya ketika istirahat bersyarat pernyataan di dalam loop (jika `$count == 10 break;`) menyebabkan prosesor PHP untuk melanjutkan eksekusi dengan pernyataan pertama yang mengikuti untuk loop:

```
for ($count = 1;;)
{
echo "Count = $count<br>";
if ($count == 10) break;
$count = ++$count;
}
```

Sebagai contoh lain, pernyataan berikut (dengan tiga ekspresi non-kosong) menampilkan sepuluh pertama nilai dalam array `stateNames` \$:

```
for ($stateIndex = 0; $stateIndex <= 9; ++$stateIndex)
echo "\$stateNames[$stateIndex] = $stateNames[$stateIndex]<br>";
```

increment ekspresi ketiga variabel `$stateIndex` setelah setiap iterasi dari untuk loop. Sementara ekspresi kedua (`$stateIndex <= 9`) adalah benar, prosesor PHP akan mengeksekusi untuk loop'sblok pernyataan-pernyataan satu dalam contoh ini.

Menggunakan *While* Loop Untuk mengulangi Statement ketika kondisi benar

Pada bagian sebelumnya, Anda belajar untuk menggunakan PHP loop untuk menginstruksikan prosesor PHP untuk mengulang satuatau lebih pernyataan sejumlah tertentu kali. Untuk situasi di mana Anda ingin prosesor PHP loop selama suatu kondisi tertentu tetap benar, tetapi belum tentu jumlah tertentu kali, Andadapat menggunakan PHP *while loop*. Format umum dari while loop adalah sebagai berikut:

```
while ($conditionIsTrue)
{
statement;
statement;
}
```

Ketika bertemu dengan PHP while loop, prosesor pertama tes dengan kondisi yang ditetapkan(dalam kurung setelah kata kunci sementara). Jika kondisi benar, prosesor akan mengeksekusi lingkaran sedangkan pernyataan blok (yaitu, laporan dalam kurung [{}]) yang mengikuti pernyataan sementara). Setelah menjalankan laporan terakhir di loop, prosesor lagi tes kondisi. Jika Kondisi masih benar, prosesor mengeksekusi loop sementara laporan lagi, kemudian memeriksa kondisilagi, dan seterusnya. Ketika kondisi tersebut akhirnya menjadi palsu, prosesor terus eksekusi padapernyataan pertama yang mengikuti pernyataan terakhir di while loop.

PHP do-while loop mirip dengan while loop dalam do-while loop juga menginstruksikan browser untuk mengulang satu set pernyataan sementara kondisi benar. Namun, daripada memeriksa kondisi

sebelum menjalankan laporan dalam loop (seperti halnya dengan while loop), loop do-while memeriksa kondisi setelah menjalankan laporan lingkaran itu. Bentuk umum loop do-while adalah sebagai berikut:

```
do
{
statement;
statement;
}
while ($conditionIsTrue)
```

Karena prosesor mengetes kondisi setelah melaksanakan do-while loop pernyataan, prosesor akan mengeksekusi laporan dalam loop setidaknya sekali prosesor-bahkan jika kondisi yang tes adalah false sebelum prosesor menjalankan laporan loop pertama kalinya.