

## 1 Scrierea instructiunilor switch, break si default

În cazul în care vrem să comparăm valoarea unei singure variabile cu o succesiune de valori, în locul instrucțiunii "if" putem folosi instrucțiunea switch:

De exemplu, să presupunem că valoarea variabilei \$numar este 1, 2 sau 3, reprezentând respectiv dimensiunile "mica", "medie" și "mare". Iată un mic script care afișează dimensiunile asociate valorilor variabilei \$numar, folosind funcția switch alături de break și default care vor fi explicate mai jos:

```
<?php
switch($numar) {
    case(1):
        echo "mic";
        break;
    case(2):
        echo "mediu";
        break;
    case(3):
        echo "mare";
        break;
    default:
        echo "Acesta nu este un cod valabil";
}
?>
```

Acțiunea unei instrucțiuni switch este determinată de valoarea unei expresii întregi, nu de valoarea unei expresii conditionale. Numele variabilei este dat între parantezele care urmează după cuvântul cheie switch. Acoladele delimitează o serie de instrucțiuni case și o instrucțiune default opțională, fiecare dintre instrucțiunile cuprinse între acolade putând avea instrucțiuni asociate.

Când este executată, instrucțiunea **switch** încearcă să stabilească o identitate între valoarea variabilei sale și valoarea asociată unei instrucțiuni case. Se vor executa instrucțiunile asociate primei instrucțiuni "case" pentru care identitatea respectivă este valabilă.

Dacă valoarea variabilei din instrucțiunea **switch** nu corespunde nici uneia din valorile asociate instrucțiunilor case, se vor executa instrucțiunile asociate instrucțiunii default, dacă există o asemenea instrucțiune (nu este obligatorie).

Un procedeu de programare indicat constă în aceea că fiecare instrucțiune case din cadrul unei instrucțiuni switch să se încheie cu o instrucțiune break.

Instrucțiunea break determină încheierea executiei instrucțiunii switch, sare peste "case-urile" rămase și se execută codurile care mai sunt (dacă există) după "switch". În absența instrucțiunii "break", executia trece la următoarea instrucțiune "case" sau "default", fapt nedorit în majoritatea cazurilor.

## 2 Instructiuni repetitive for() si while()

Scrierea instructiunilor "for"

Instructiunea for() este o instructiune bucla sau o instructiune iterativa; cu alte cuvinte, o instructiune care executa în mod repetat instructiunile asociate.

Sintaxa generala a instructiunii for este urmatoarea:

```
for(expresie1; conditie; expresie2)
{
    Instructiune(1);
    Instructiune(2);
    Instructiune(3);
    .....
}
```

expresie1 se executa o singura data si anume la inceputul instructiunii; conditie este o expresie logica ; instructiunea for() se executa atita timp cat valoarea expresiei : conditie are valoarea TRUE.

De fiecare data dupa executia instructiunilor dintre acolade se executa : expresie2.

Iata un exemplu practic de utilizare a unei instructiuni "for":

```
<?php
$suma = 0;
for ($n=1; $n<=3; $n++) {
    $suma += $n;
    echo "<br /> Suma intregilor de la 1 la $n este $suma.";
}
?>
```

Rezultatul afisat va fi:

Suma intregilor de la 1 la 1 este 1

Suma intregilor de la 1 la 2 este 3

Suma intregilor de la 1 la 3 este 6

Instructiunea for() este utila pentru numarare si executarea în mod repetat a unor actiuni.

**foreach** este o constructie care functioneaza doar cu matrice generand erori cand sunt folosite variabile cu tipuri de date diferite sau variabile neinitializate. Sintaxa folosita este:

```
foreach ($matrice as $cheie=>$valoare) {  
    instructiuni de executat;  
}
```

Exemplu:

```
<?php  
$personal = array(  
    "Florin" => director,  
    "Catalin" => inginer,  
    "Mihai" => economist,  
    "Sorin" => sofer,  
    "Diana" => secretara  
);  
  
foreach ($personal as $nume => $meserie) {  
    echo "$nume => $meserie<br>";  
}  
?>
```

### Scrierea instructiuni while

O instructiune while() sau do while are o singura conditie. Fara o expresie de initializare sau o expresie pas; opereaza similar ca o instructiune "for()". În consecinta, urmatoarele doua instructiuni sunt echivalente:

```
for ( ; $i<= 3; ) { $suma += $i; }
```

```
while ($i <= 3) { $suma ++ $i; }
```

Instructiunea while() executa un set de instructiuni atita timp cit valoarea expresiei "conditie" este adevarata:

Forma generala a instructiuni while() este urmatoarea :

```
while(conditie)  
{  
    Instructiune(1);  
    Instructiune(2);  
    Instructiune(3);  
    .....  
}
```

Setul de instructiuni Instructiune(1), Instructiune(2), etc... se vor executa atita timp cit valoarea expresiei logice "conditie" are valoarea TRUE.

Sa studiem un exemplu practic:

```
<?php
$n = 1;
$suma = 0;
while ($n <= 3)
{
    $suma = $suma + $n;
    $n++;
}

echo "Suma este $suma";
?>
```

In browser va aparea urmatorul rezultat:

Suma este 6

Scrierea instructiuni do while

Diferenta dintre instructiunea while() si instructiunea do while este urmatoarea: daca la instructiunea while() se verifica intai conditia si numai daca rezultatul este TRUE se trece la executarea corpului functiei, in cazul functiei do while intai se executa corpul functiei si dupa aceea se verifica conditia. De la do while corpul functiei va fi executat cel putin o singura data, indiferent de rezultatul conditiei.

Forma generala a instructiuni do while este urmatoarea :

```
do {
    Instructiune(1);
    Instructiune(2);
    Instructiune(3);
    .....
} while(conditie);
```

Iata exemplul prezentat la functia while(), de data aceasta folosind functia do while :

```
<?php
```

```
$n = 1;
$suma = 0;
do {
    $suma = $suma + $n;
    $n++;
} while ($n <= 3);

echo "Suma este $suma";
?>
```

Rezultatul va fi acelasi: "Suma este 6".