

Tablouri în PHP

Tipul tablou se referă la colecții de date indexate, numite **elements**, care pot fi utilizate împreună. Acest tip este foarte puternic și des folosit în PHP. Astfel, tablourile se dovedesc a fi cele mai potrivite containere pentru stocarea datelor în scopul manipulării acestora (sortare).

Tablourile sunt tipuri compuse care, spre deosebire de cele din C++, pot memora date de diverse tipuri. Tablourile, ca și variabilele simple, nu se declară. Iată un prim exemplu:

```
<?php
// initializare
$t[0] = 45;
$t[1] = 3;
$t[2] = 12;
$t[3] = 5;
// afisare
for ($i = 0; $i < 4; $i++)
    echo $t[$i]."<br>";

// tablou initializat cu ajutorul functiei array
$v = array(34, 55, "doi", "7", true, 66);
for ($i = 0; $i < 6; $i++)
    echo $v[$i]." ";
// s-au afisat valorile 34 55 doi 7 1 66

// o alta modalitate de utilizare a functiei array;
// indexarea vectorului incepe de la 1:
$a = array(
    1 => "Popescu",
    2 => "Ionescu",
    3 => "Stoica",
    4 => "Vasilache",
    5 => "Barbu"
);
echo $a[0]."<br>";// element inexistent!
for ($i = 1; $i <= 5; $i++)
    echo $a[$i]." ";
?>
```

Un element al unui tablou este precizat indicând numele tabloului și cheia (numit și indexul elementului).

Cheile pot fi:

- **numere întregi** (tabloul corespunzător este de tip numeric)
- **șiruri de caractere** (tabloul corespunzător este de tip asociativ)

Cheile având ultimul tip precizat trebuie incluse între ghilimele sau apostrofuri.

Exemple:

```
$masina["ford"] = "albastra";
$masina["ford"] = "albastra";
```

Crearea tablourilor

Exista urmatoarele 3 modalitati de creare a unui tablou:

a) Prin atribuire directa

Utilizarea acestei modalitati presupune precizarea numelui tabloului, precum si a cheilor si valorilor fiecarui element. In cazul in care se doreste utilizarea unei chei de tip sir de caractere, se procedeaza ca in exemplul urmator:

```
$elev["liceu"]="Procopiu";  
$elev["clasa"]=12;  
$elev["varsta"]=19;
```

Pentru crearea unui tablou prin atribuire directa, pot fi precizate si chei numerice, ca in exemplul urmator:

```
$orase[0]="Bucuresti";  
$orase[1]="Iasi";  
$orase[2]="Vaslui";
```

Este posibil sa nu se precizeze nici o cheie, ca in exemplul urmator:

```
$tariUE[]="Italia";  
$tariUE[]="Germania";  
$tariUE[]="Romania";
```

in acest exemplu, PHP creeaza automat tabloul **\$tariUE**, atribuind elementelor acestuia chei numerice succesive, incepand cu valoarea 0 (adica 0, 1 si 2).

b) Folosind functia array()

Pentru a crea un tablou nevid, valorile elementelor acestuia trebuie precizate ca argumente, sub forma unei liste ale carei componente sunt separate prin caracterul virgula, ca in exemplul prezentat in continuare:

```
$orase=array("Bucuresti", "Iasi", "Vaslui");
```

Valorile precizate ca argumente ale functiei vor fi stocate ca elemente in tabloul **\$orase**, in ordinea precizarii lor, cheile acestora avand valori intregi succesive incepand cu 0.

c) precizand cheile odata cu utilizarea functiei array()

PHP ofera posibilitatea de a preciza cheile elementelor odata cu crearea unui tablou prin intermediul functiei array().
Exemplu in care cheile sunt numerice:

```
$orase=array(0=>"Bucuresti", 1=>"Iasi", 2=>"Vaslui");
```

Daca se doreste precizarea unor chei de tip sir de caractere, se procedeaza astfel:

```
$orase2=array (  
"Capitala"=>"Bucuresti",  
"Moldova"=>"Iasi",  
"Moldovaa"=>"Vaslui");
```

Parcurgerea tablourilor (Array)

Exista diverse modalitati pentru a parcurge elementele unui tablou (operatie cunoscuta si sub denumirea de traversarea tabloului), cea mai eleganta dintre acestea constand in utilizarea instructiunii foreach, cu urmatoarea sintaxa:

```
foreach($tablou as [$scheie=>] [&] $valoare) {  
  grup de instructiuni  
}
```

Variabila **\$scheie** este optionala si, atunci cand este specificata, contine cheia elementului curent.

De asemenea, caracterul „&” este optional si, atunci cand este prezent, permite modificarea explicita a valorii elementului **\$valoare** prin intermediul instructiunilor incluse in grupul grup) si transmiterea automata a valorii catre tablou (\$tablou).

În cele mai multe cazuri, nu se doreste modificarea valorii elementelor, astfel incat caracterul „&” nu se foloseste.

Exemplu

```
<?php  
$orase=array("Bucuresti", "Iasi", "Vaslui");  
echo "Orase romanesti:<br />";  
foreach($orase as $scheie=>$valoare) {  
  echo "$scheie=$valoare<br />";  
}  
/* Se va afisa:  
Orase romanesti:  
0=Bucuresti  
1=Iasi  
2=Vaslui */  
?>
```

O alta modalitate de parcurgere a elementelor unui tablou consta in utilizarea functiilor list() si each(), ca in exemplul urmatoar:

```
<?php  
$orase=array("Bucuresti", "Iasi", "Vaslui");  
echo "Orase romanesti:<br />";  
reset($orase);  
while (list($scheie, $valoare) = each($orase)) {  
  echo "$scheie=$valoare<br />";  
}  
?>
```

! Functia list() permite atribuirea simultana de valori mai multor variabile utilizand elementele unui tablou: list(\$variabila1, \$variabila2, ...) = \$tablou;

! Functia each() intoarce perechea cheievaloare curenta a tabloului primit ca argument si deplaseaza pointerul intern la urmatorul element.

! Daca se foloseste functia each() pentru parcurgerea unui tablou, spre deosebire de instructiunea foreach, este necesar ca – in prealabil – sa se apeleze functia reset(), care va muta pointerul intern la inceputul tabloului.

Există funcția **count** cu ajutorul căreia se poate determina lungimea unui tablou. Mai jos este un exemplu de utilizare a acestei funcții:

```
<?php
$v = array(34, 55, 5, 6, 1, 75, 9, 7);
echo "Lungimea vectorului este ".count($v)."<br>";
echo "Elementele tabloului : ";
for ($i = 0; $i < count($v); $i++)
    echo $v[$i]." ";
?>
```

Sortarea tablourilor (Array)

1. sort() – sorteaza ascendent un tablou in functie de valoarea elementelor
2. rsort() – sorteaza descendent un tablou in functie de valoarea elementelor
3. asort() – sorteaza ascendent un tablou asociativ in functie de valoarea elementelor
4. arsort() – sorteaza descendent un tablou asociativ in functie de valoarea elementelor
5. ksort() - sorteaza ascendent un tablou in functie de cheie
6. krsort() - sorteaza descendent un tablou in functie de cheie

Matricele sunt tablouri bidimensionale. Inițializarea lor se face după același principii ca și tablourile unidimensionale. Iată mai jos două exemple de matrice:

```
<?php
// matrice cu 3 linii si 7 coloane:
for ($i = 0; $i < 3; $i++)
    for ($j = 0; $j < 7; $j++)
        $mat[$i][$j] = ($i + 1) * ($j + 1);
for ($i = 0; $i < 7; $i++)
{
    for ($j = 0; $j < 7; $j++)
        echo $mat[$i][$j]." ";
    echo "<br>";
}
// initializare cu functia array:
$a = array(
    1 => array(1, 2, 3, 4, 5, 6),
    2 => array(11, 12, 13, 14, 15, 16),
    3 => array(21, 22, 23, 24, 25, 26),
    4 => array(31, 32, 33, 34, 35, 36),
);

for ($i = 1; $i <= 4; $i++)
{
    for ($j = 1; $j <= 4; $j++)
        echo $a[$i][$j]." ";
    echo "<br>";
}
?>
```