

Laboratorio Progettazione Web

Array in PHP

Andrea Marchetti – IIT-CNR
andrea.marchetti@iit.cnr.it
2012/2013

Overview

- Introduzione
- Array Numerici
- Array Associativi
- Array Multidimensionali
- Funzioni per manipolare gli array
- Esempio completo

Introduzione

- Gli array sono un tipo di dato speciale che consente di memorizzare più di un valore
- PHP ha una gestione molto efficace degli array che consente di scrivere codice snello che elimina parti noiose
- Supponiamo di voler memorizzare i nomi delle squadre di calcio
 - Inter
 - Juventus
 - Milan
 - Napoli
 - Roma

Introduzione

- Se uso le normali variabili me ne servono 5
 - \$team1="Inter";
 - \$team2="Juventus";
 - \$team3="Milan";
 - \$team4="Napoli";
 - \$team5="Roma";
- Per memorizzare liste di valori **meglio** usare gli array
- Per **accedere** agli elementi uso il nome dell'array e un suo indice

Introduzione

- In PHP esistono 2 tipi di Array
 - **Numerici**
 - **Associativi**
- Abbiamo inoltre array
 - Monodimensionali
 - Muldimensionali

Tipi di array

Numerici

1	Inter
2	Juventus
3	Napoli
4	Roma
5	Milan
6	

Associativi

nome	Inter
città	Milano
scudetti	18
coppe	

Operazioni

1. Creazione
2. Accesso lettura, scrittura
3. Inserimento

Operazioni su array

- Creazione
- Accesso in lettura e scrittura agli elementi
- Inserimento di nuovi elementi
- Cancellazione di elementi
 - Si usa una funzione: `unset()`

Array Numerici

- Gli array numerici immagazzinano ciascun elemento usando un indice numerico che **inizia da 0**
- [Per creare il nostro array numerico ho tre modi](#)

Array Numerici - Creazione

- Usando la parola chiave **array**
 - `$team = array ("Inter", "Juventus", "Milan", "Napoli", "Roma");`
- Inserendo i valori ad uno a uno specificando l'indice
 - `$team[0]="Inter";`
 - `$team[1]="Juventus";`
 - `$team[2]="Milan";`
 - `$team[3]="Napoli";`
 - `$team[4]="Roma";`
- Inserendo i valori ad uno a uno senza specificare l'indice
 - `$team[]="Inter";`
 - `$team[]="Juventus";`
 - `$team[]="Milan";`
 - `$team[]="Napoli";`
 - `$team[]="Roma";`

Nota sugli array in Php

- A differenza di altri linguaggi di programmazione, in Php gli elementi possono essere di tipo diverso
- `$a = array (11, "abcd", 2.34);`
- Per [visualizzare](#) un array utilizzare
 - **`print_r ($a);`**
 - `// Array ([0] => 11 [1] => abcd [2] => 2.34)`
 - **`var_dump($a);`**
 - `// array(3) { [0]=> int(11) [1]=> string(4) "abcd" [2]=> float(2.34) }`

Array Numerici - Accesso

- Per accedere ai singoli valori si usa
 - Il nome dell'array
 - L'indice dell'elemento
- echo "A Milano abbiamo due squadre:
`$team[0]` e `$team[2]`";

Array Numerici - Accesso

- Per accedere a tutti gli elementi di un array
`$teams = array ("Inter", "Juventus", "Milan", ...);`
- Costrutto **for** con la funzione **sizeof()** o **count()**

```
for ($i=0;$i<sizeof($teams);$i++)
```

```
    echo "<li>$teams[$i]</li>";
```

- Costrutto **foreach ... as**

```
foreach($teams as $team)
```

```
    echo "<li>$team</li>";
```

Nome
array

Variabile dove il parser PHP ad ogni
ciclo inserisce un elemento
dell'array

Array Numerici - Inserimento

- Consideriamo il nostro array di partenza
 - `$team = array ("Inter", "Juventus", "Milan", "Napoli", "Roma");`
- Per **aggiungere** nuovi elementi si hanno due modi
 - Specificando l'indice
 - `$team[5] = "Lazio" ;`
 - Senza specificare l'indice
 - `$team[] = "Lazio" ;`
- `echo "A Roma abbiamo due squadre: $team[4] e $team[5]";`

Array Associativi

- Non sempre un indice è il modo migliore (per il programmatore) per riferire gli elementi di un array
- Consideriamo il caso in cui voglia memorizzare le caratteristiche di una squadra: ad ex il Milan
 - Il **nome** della squadra è **MILAN**
 - La **città** dove gioca è **Milano**
 - Ha vinto **18 scudetti**

Array Associativi

- Con un array numerico

- `$milan = array ("Milan", "Milano", 18);`
- `echo "<p>$milan[0] gioca a $milan[1] e ha vinto $milan[2] scudetti.</p>"; // Non è molto leggibile`

- Con un array associativo

- `$milan = array ("nome" => "Milan", "citta" => "Milano", "scudetti" => 18);`
- `echo "<p>".$milan['nome']." gioca a ".$milan['citta']." e ha vinto ".$milan['scudetti']." scudetti.</p>";`

Array Associativi - Creazione

- Abbiamo due modi

- Con la parola chiave array

- `$milan = array ("nome" => "Milan", "citta" => "Milano", "scudetti" => 18);`

- Inserendo ad uno ad uno la coppia **chiave=valore**

- `$milan['nome'] = "Milan";`

- `$milan['citta'] = "Milano";`

- `$milan['scudetti'] = 18;`

Array Associativi - Accesso

- Per accedere ai singoli valori si usa
 - Il **nome** dell'array
 - la **chiave** dell'elemento

echo `.$milan['nome']`." gioca a ".

`.$milan['citta']`." e ha vinto ".

`.$milan['scudetti']`." scudetti";

Array Associativi - Accesso

- Per accedere a tutti gli elementi di un array associativo

– `$milan = array ("nome" => "Milan", "citta" => "Milano", "scudetti" => 18);`

- **foreach(nome-array AS chiave=>valore)**

```
foreach($milan as $key => $value)  
    echo "<li>$key = $value</li>";
```

Nome
array

Variabile `$key` dove il parser PHP ad ogni ciclo inserisce la chiave di un elemento dell'array

Variabile `$value` dove il parser PHP ad ogni ciclo inserisce un elemento dell'array

Array Associativi - Inserimento

- Per aggiungere un elemento
 - `$milan = array ("nome" => "Milan", "citta" => "Milano", "scudetti" => 18);`
 - `$milan["web"] = "www.acmilan.com";`

Array Multidimensionali

- Un elemento di un array può contenere a sua volta un array, e così via creando array multidimensionali
- Proviamo a inserire in unico array le descrizioni di tutte le squadre

Array Associativo Multidimensionale

```
$teams = array( "milan" => array ( "nome"=>"Milan",  
                                     "citta" => "Milano",  
                                     "scudetti" => 18),  
              "inter" => array ( "nome" => "Inter",  
                                   "citta" => "Milano",  
                                   "scudetti" => 18),  
              "juve"  => array ( "nome" => "Juventus",  
                                   "citta" => "Torino",  
                                   "scudetti" => 26),  
              );
```

Array Associativo Multidimensionale

Accesso

- Si usa il costrutto foreach annidato

```
foreach($teams as $key => $value){  
    echo "Team=$key";  
    foreach($value as $chiave => $valore)  
        echo "$chiave = $valore";  
}
```

Array Associativo Multidimensionale

Inserimento

```
print_r($teams);
```

```
Array([milan] => Array ( [nome] => Milan    [citta] => Milano [scudetti] => 18 )  
      [inter] => Array ( [nome] => Inter    [citta] => Milano [scudetti] => 17 )  
      [juve]  => Array ( [nome] => Juventus [citta] => Torino [scudetti] => 26 ))
```

```
$teams["inter"]["coppeltalia"]=7;
```

```
print_r($teams);
```

```
Array([milan] => Array ( [nome] => Milan    [citta] => Milano [scudetti] => 18 )  
      [inter] => Array ( [nome] => Inter    [citta] => Milano [scudetti] => 17 [coppeltalia] => 7 )  
      [juve]  => Array ( [nome] => Juventus [citta] => Torino [scudetti] => 26 ))
```

Array Numerico Multidimensionale

```
$tabPitagoriga = array ( array ( 1, 2, 3, 4, 5, 6, 7, 8, 9,10 ),  
                          array ( 2, 4, 6, 8,10,12,14,16,18,20 ),  
                          array ( 3, 6, 8,12,15,18,21,24,27,30 )  
                        );
```

```
echo "2x3 = ". $tabellaPitagoriga[2-1][3-1];
```


Funzioni per manipolare gli Array

- <http://php.net/manual/en/ref.array.php>
- `count()`, `sizeof()` = dimensione
- `is_array()` = controlla se la variabile è un array
- `shuffle()` = riordina in modo casuale un array
- `sort()`, `rsort()` = riordina un array numerico
- `asort()`, `arsort()` = riordina un array associativo
- **`explode()` = converte una stringa in array**
- `extract()` = Per ogni coppia chiave/valore crea una variabile con nome=chiave e valore associato
- `unset()` = Utile per rimuovere elementi da un array

Funzioni per manipolare gli Array

```
$temp = explode(' ', "Il campionato più bello del mondo !");
```

Array

```
(  
    [0] => Il  
    [1] => campionato  
    [2] => più  
    [3] => bello  
    [4] => del  
    [5] => mondo  
    [6] => !  
)
```

\$_GET, \$_POST

- \$_GET, \$_POST sono due ARRAY associativi creati dal Sistema per contenere le coppie chiave, valore rispettivamente di una GET e di una POST
- Ex `http://localhost/capitale.php?paese=Italia`
- `extract($_GET); // $paese=$_GET["paese"];`

Esercizio

- Creare uno script PHP per fare domande di geografia sulle capitali europee
 - Crea una pagina HTML dove sono elencati alcuni paesi europei di cui uno selezionabile (ex SELECT)



The image shows a web form interface. On the left, there is a dropdown menu with a blue border. The menu is currently open, showing a list of country names: 'ITALIA' (highlighted in a light blue bar), 'FRANCIA', 'SPAGNA', and '...'. To the right of the dropdown menu is a blue button with the text 'Query' in white.

- Quando l'utente preme un bottone di query restituisce la capitale corrispondente
La capitale di Italia è Roma