

# Laboratorio Progettazione Web

## **Il linguaggio PHP – Array**

Andrea Marchetti– IIT-CNR

[andrea.marchetti@iit.cnr.it](mailto:andrea.marchetti@iit.cnr.it)

AA 2014/2015

# Overview

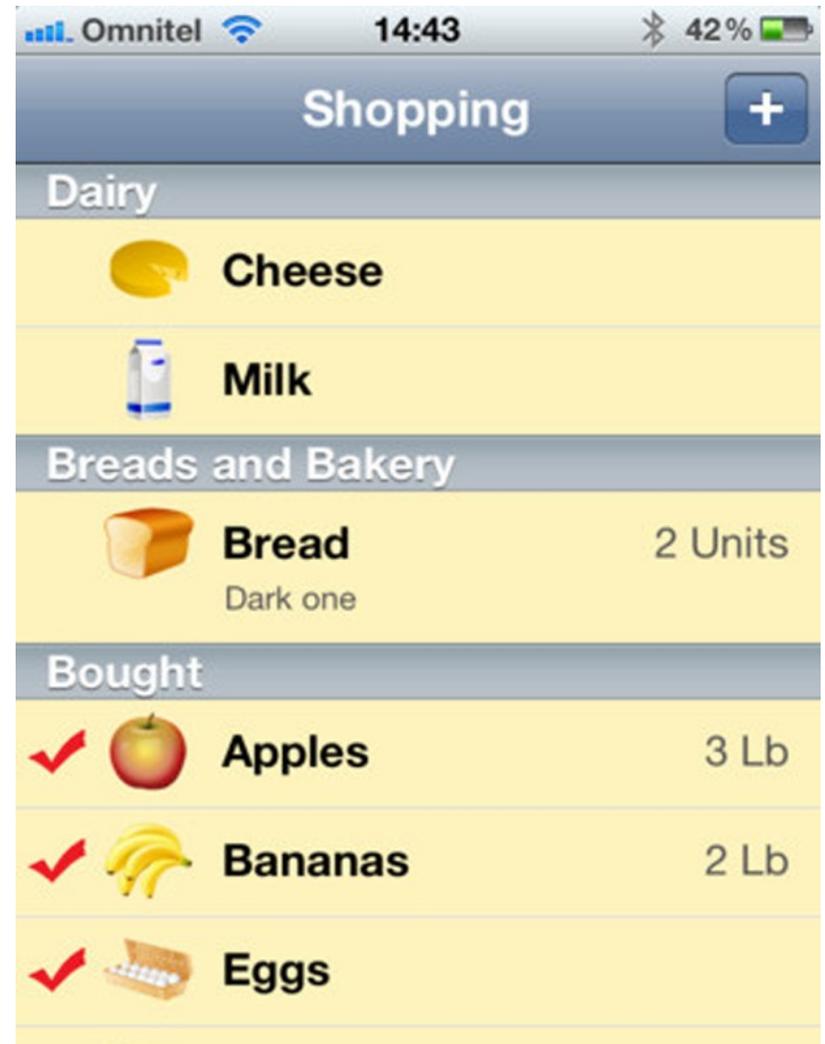
- Introduzione sugli array
- Array Numerici
- Array Associativi
- Array Multidimensionali
- Funzioni per manipolare gli array
- Esempio completo

# Array

- Un array è una variabile per memorizzare con un solo nome una **lista di cose** come nomi o numeri tra di loro attinenti
- Il PHP ha una gestione molto efficiente degli array

# Esempio

Esempio di una **lista di cose** che voglio trattare nello **stesso modo** ad esempio voglio visualizzarli in una lista



# Esempio

- Se uso le normali variabili me ne servono 6

```
$shoppingList1="Cheese";
```

```
$shoppingList2="Milk";
```

```
$shoppingList3="Bread";
```

```
$shoppingList4="Apples";
```

```
$shoppingList5="Bananas";
```

```
$shoppingList6="Eggs";
```

# Esempio

- Se voglio visualizzarli in un array

```
print ( "<table>" );  
print ( "<tr><td>$shoppingList1</td></tr>" );  
print ( "<tr><td>$shoppingList2</td></tr>" );  
print ( "<tr><td>$shoppingList3</td></tr>" );  
print ( "<tr><td>$shoppingList4</td></tr>" );  
print ( "<tr><td>$shoppingList5</td></tr>" );  
print ( "<tr><td>$shoppingList6</td></tr>" );  
print ( "</table>" );
```

# Esempio

```
$shoppingList = array("Cheese", "Milk", "Bread",  
                      "Apples", "Bananas", "Eggs");  
print("<table>");  
for($i=0; $i<sizeof($shoppingList); $i++){  
    print("<tr><td>$shoppingList[$i]</td></tr>");  
}  
print("</table>");
```

# Tipi di array

- In PHP esistono 2 tipi di Array
  - **Numerici**
  - **Associativi**
- Inoltre gli array si distinguono in
  - Monodimensionali
  - Multidimensionali

# Tipi di array

## Numerici

1	Cheese
2	Milk
3	Bread
4	Apples
5	Bananas
6	Eggs

## Associativi

nome	Cheese
categoria	Latticini
quantità	1kg

# 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

- Gli array numerici memorizzano ogni elemento usando un **indice** numerico che **inizia da 0**
- Per creare un **array numerico** ho tre modi

# Array Numerici - Creazione

- Usando la funzione **array()**  

```
$shoppingList = array( "Cheese" , "Milk" , "Bread" ,  
                      "Apples" , "Bananas" , "Eggs" ) ;
```
- Inserendo i valori ad uno ad uno specificando l'indice  

```
$shoppingList[0] = "Cheese" ;  
$shoppingList[1] = "Milk" ;  
$shoppingList[2] = "Bread" ;  
...
```
- Inserendo i valori ad uno ad uno omettendo l'indice  

```
$shoppingList[] = "Cheese" ;  
$shoppingList[] = "Milk" ;  
$shoppingList[] = "Bread" ;  
...
```

# Nota sugli Array in Php

- A differenza di altri linguaggi di programmazione in PHP gli elementi di un array possono essere di tipo diverso

```
$albergo = array("Hotel Roma", 56124, "Pisa");
```

- Per visualizzare un array utilizzare

```
print_r($albergo);
```

```
Array
(
    [0] => Hotel Roma
    [1] => 56124
    [2] => Pisa
)
```

```
var_dump($albergo);
```

```
array(3)
{
    [0]=> string(10) "Hotel Roma"
    [1]=> int(56124)
    [2]=> string(4) "Pisa"
}
```

# Array Numerici - Accesso

- Per accedere ai **singoli** elementi di un array numerico si usa la classica notazione: **nome dell'array più indice elemento** racchiuso da **parentesi quadre**

```
$shoppingList = array("Cheese", "Milk", "Bread",  
                    "Apples", "Bananas", "Eggs");  
print("Ricordati di comprare $shoppingList[0]");
```

# Array Numerici - Accesso

- Per accedere a tutti gli elementi di un array numerico abbiamo due possibilità

```
$shoppingList = array("Cheese", "Milk", "Bread",  
    "Apples", "Bananas", "Eggs");
```

- Costrutto for con la funzione count() o sizeof()

```
for($i=0; $i<sizeof($shoppingList); $i++){  
    print("<tr><td>$shoppingList[$i]</td></tr>");  
}
```

- Costrutto foreach ... as

```
foreach ($shoppingList as $item){  
    print("<tr><td>$item</td></tr>");  
}
```

nome  
dell'array

Variabile dove il processore PHP ad ogni  
iterazione inserisce un elemento dell'array

# Array Numerici - Inserimento

- Consideriamo il nostro array

```
$shoppingList =  
array( "Cheese" , "Milk" , "Bread" ,  
      "Apples" , "Bananas" , "Eggs" ) ;
```

- Per inserire nuovi elementi abbiamo 2 modi

- Specificando l'indice

```
$shoppingList[10] = "Wine" ;  
print_r( $shoppingList ) ;
```

- Senza specificare l'indice

```
$shoppingList[ ] = "Wine" ;  
print_r( $shoppingList ) ;
```

# Esercizio

- Scrivere un programma php che stampa gli elementi della mia lista di spesa che hanno un nome lungo 4;

```
$shoppingList =  
array( "Cheese" , "Milk" , "Bread" ,  
      "Apples" , "Bananas" , "Eggs" );  
strlen(); //funzione che calcola la  
lunghezza di una stringa
```

# Soluzione

```
$shoppingList = array  
( "Cheese" , "Milk" , "Bread" ,  
  "Apples" , "Bananas" , "Eggs" ) ;  
  
foreach ( $shoppingList as $item )  
    if ( strlen( $item ) == 4 )  
        print( "$item\n" ) ;
```

# 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

- Non sempre un indice è il modo migliore (per il programmatore) per riferire gli elementi di un array
- Consideriamo un esempio dal calcio
  - Il **nome** della squadra è **Milan**
  - la **città** dove gioca è **Milano**
  - Ha vinto **18** **scudetti**

Milan



Full name	Associazione Calcio Milan S.p.A. <sup>[1]</sup>
Nickname(s)	<i>i Rossoneri</i> (The Red and Blacks) <i>il Diavolo</i> (The Devil) Casciavit (Lombard for: Screwdrivers)
Founded	16 December 1899; 115 years ago <sup>[2]</sup>
Ground	San Siro, Milan
Capacity	80,018
Owner	Silvio Berlusconi
Honorary President	Silvio Berlusconi <sup>[3]</sup>
Head coach	Filippo Inzaghi
League	Serie A
2013–14	Serie A, 8th
Website	Club home page <a href="#">↗</a>

# 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($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" => 30),  
              );
```

# 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 e \$\_POST sono due array associativi creati dal sistema per contenere i parametri (nome, valore) rispettivamente di una GET e di una POST.
- Supponiamo di invocare un programma php sul server con la seguente URL  
<http://localhost/capitale.php?paese=Italia>

```
<?php
$paese = $_GET[ 'paese' ];
echo "Paese=$paese" ;
?>
```
- La funzione `extract($_GET)` crea in modo automatico tante variabili per ogni parametro passato

# 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 European countries: ITALIA, FRANCIA, SPAGNA, and three dots indicating more options. On the right, there 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