

Laboratorio Progettazione Web

Le funzioni in PHP – Lezione 11

Andrea Marchetti – IIT-CNR
andrea.marchetti@iit.cnr.it
2011/2012

Funzioni

- Una funzione è un'insieme di istruzioni che svolgono una particolare "funzione" ed eventualmente restituiscono un valore
 - `somma(2,3)` // esegue la somma tra 2 e 3 e restituisce il risultato
- Nella stesura di un programma si può trovare più volte la stessa sequenza di codice.
- **Conviene** estrarre questa sequenza e metterlo in una funzione e quando serve invocare quella funzione.
 - meno codice (meno errori, performance migliori)
 - codice più chiaro (manutenzione migliore)
 - riuso

Funzioni interne o Built-In

- PHP ha centinaia di funzioni built-in pronte all'uso.

- Alcune delle quali abbiamo già incontrato

```
print("Hello World");
```

```
rand(1,100) // genera un numero casuale da 1 a 100
```

```
date("Y/m/d"); // fornisce la data odierna
```

- Le funzioni possono prendere qualsiasi numero di argomenti, anche 0

```
phpinfo()
```

Definire una funzione

- La sintassi generale per definire una funzione è:

```
function nome_funzione ($arg_1,$arg_2,...,$arg_n)
{
    // istruzioni con $arg_1,$arg_2,...,$arg_n
    return valore;
}
```

- I nomi di funzioni sono **case-insensitive**
 - print, Print, PrInT // sono la stessa funzione
- Una funzione può contenere uno o più "**return**"
- L'istruzione **return** cessa l'esecuzione della funzione e restituisce al chiamante un eventuale valore

Invocare una funzione

- La sintassi per invocare una funzione è *nome_funzione (\$arg_1,\$arg_2,...,\$arg_n);*
- L'invocazione della funzione può avvenire anche prima di averla definita

Esempio con restituzione di valore

```
<?php // somma.php?val1=2&val2=3

$val1 = $_GET['val1'];
$val2 = $_GET['val2'];

echo "<h1>$val1+$val2=" . somma($val1,$val2) . "</h1>";

function somma($add1, $add2){
    $addizione = $add1 + $add2;
    return $addizione;
}
?>
```

Esempio senza restituzione di valore

```
<?php // somma1.php?val1=2&val2=3
$val1 = $_GET['val1'];
$val2 = $_GET['val2'];

echo htmlTop();
echo "<h1>$val1+$val2=".somma($val1,$val2)."</h1>";
echo htmlBottom();

function somma($add1, $add2){
    $addizione = $add1 + $add2;
    return $addizione;
}

function htmlTop(){
    echo <<<_HTML
<html>
    <head><title>Somma</title></head>
    <body>
_HTML;
}

function htmlBottom(){
    echo <<<_HTML
    </body>
</html>
_HTML;
}
?>
```

Restituzione di più di un valore

- Se voglio restituire più di un valore uso un array
- Supponiamo che oltre alla somma voglio restituire il massimo tra i due valori usando sempre la stessa funzione
 - `SommaMax(&num1; $num2);`

Restituzione di più di un valore

```
<?php // sommaMax.php?val1=2&val2=3

$val1 = $_GET['val1'];
$val2 = $_GET['val2'];

$elaborazione = sommaMax($val1,$val2);
echo "<h1>Elaborazione dei numeri $val1 e $val2</h1>";
echo "<h2>Somma = $elaborazione[somma]</h2>";
echo "<h2>Max    = $elaborazione[max]</h2>";

function sommaMax($num1, $num2){
    $result['somma'] = $num1 + $num2;
    $result['max']   = ($num1>$num2)?$num1:$num2;
    return $result;
}
?>
```

Esempio con restituzione di valore

```
<?php // somma.php?val1=2&val2=3

$val1 = $_GET['val1'];
$val2 = $_GET['val2'];

echo "<h1>$val1+$val2=" . somma($val1,$val2) . "</h1>";

function somma($add1, $add2){
    $addizione = $add1 + $add2;
    return $addizione;
}
?>
```

Uso di variabili globali

- Posso evitare di passare dei parametri ad una funzione e farla operare su variabili globali

```
<?php // somma.php?val1=2&val2=3
```

```
$val1 = $_GET['val1'];
```

```
$val2 = $_GET['val2'];
```

```
echo "<h1>$val1+$val2=" . somma() . "</h1>";
```

```
function somma(){  
    global $val1; global $val2;  
    $addizione = $val1 + $val2;  
    return $addizione;  
}  
?>
```

Visibilità di una variabile

- Variabili locali
 - sono le variabili definite internamente ad una funzione e sono visibili solo all'interno della funzione stessa
- Variabili globali
 - sono accessibili da tutte le parti del codice
- Variabili statiche
 - come le variabili locali sono definite internamente ad una funzione e sono visibili solo all'interno della funzione stessa, ma mantengono il loro valore a fronte di molteplici chiamate della funzione

Librerie di funzioni

- Nel programmare in PHP conviene pensare di crearsi delle librerie di funzioni di utilità che potrà riutilizzare in più di un programma
- Si potranno utilizzare nel proprio codice librerie sviluppate da altri programmatori
- Non c'è bisogno di ricopiare ogni volta queste funzioni nel codice che si sta sviluppando
- Si possono salvare in un file separato e usare delle istruzioni PHP per includerle

Librerie di funzioni

- Ad esempio uno potrebbe crearsi una libreria
 - per la connessione ad un DB
 - per gestire il codice HTML
 - in generale per raggruppare funzioni dello stesso tipo e riutilizzabili in script differenti

Esempio

```
<?php // somma1.php?val1=2&val2=3
$val1 = $_GET['val1'];
$val2 = $_GET['val2'];

echo htmlTop();
echo "<h1>$val1+$val2=".somma($val1,$val2)."</h1>";
echo htmlBottom();

function somma($add1, $add2){
    $addizione = $add1 + $add2;
    return $addizione;
}

function htmlTop(){
    echo <<<_HTML
<html>
    <head><title>Somma</title></head>
    <body>
_HTML;
}

function htmlBottom(){
    echo <<<_HTML
    </body>
</html>
_HTML;
}
?>
```

Posso crearmi una libreria html.php

```
<?php
/*****
*** LIBRERIA HTML          ***
***                        ***
*** Contiene le funzioni   ***
*** htmlTop() - stampa l'inizio ***
*** htmlBottom() - stampa la fine ***
*****/

function htmlTop(){
    echo <<<_HTML
<html>
    <head><title>Somma</title></head>
    <body>
_HTML;
}

function htmlBottom(){
    echo <<<_HTML
    </body>
</html>
_HTML;
}
?>
```

Inclusione di un file

```
<?php // somma.php?val1=2&val2=3

include "html.php"; // ATTENZIONE l'inclusione della libreria
                        va fatta prima di usare
                        le funzioni o le variabili in essa contenute

$val1 = $_GET['val1'];
$val2 = $_GET['val2'];

echo htmlTop();
echo "<h1>$val1+$val2=".somma($val1,$val2)."</h1>";
echo htmlBottom();

function somma($add1, $add2){
    $addizione = $add1 + $add2;
    return $addizione;
}

?>
```

Esercizio

- Riscrivere il test di geografia utilizzando la tabella appena creata