



Informatica **U**manistica

Funzioni

Laboratorio di Progettazione Web

AA 2010/2011

Claudio Lucchese / Chiara Renso

ISTI- CNR – claudio.lucchese@isti.cnr.it



UNIVERSITÀ DI PISA

Funzioni

Le funzioni sono blocchi di codice che possono essere chiamati da più punti dello script php

```
function nome_funzione (arg_1,arg_2,...,arg_n)
```

```
{
```

```
.... Script....
```

```
return valore;
```

```
}
```

Le funzioni possono avere argomenti in input (*arg1,arg_2, ... arg_n*) e possono restituire un valore

Le variabili definite dentro una funzione hanno come ambito la funzione stessa. Quindi al di fuori della funzione non esistono.

Funzioni

Creazione e chiamata di funzioni **con restituzione di valori**

```
<?php function add()
```

```
{   $somma=3+5;
```

```
return $somma;   }
```

```
.....
```

```
echo add();
```

```
?>
```

Funzioni

Chiamata di funzione **senza restituzione di valori**

```
<? function stampa()
```

```
{
```

```
echo "Funzione!";
```

```
}
```

```
stampa();
```

```
?>
```

Funzioni con argomenti

```
<? function add1($a,$b)
```

```
{
```

```
$somma=$a+$b;
```

```
return $somma;
```

```
}
```

```
echo add1(3,5);
```

```
?>
```

Funzioni

Le funzioni possono ritornare più valori, sotto forma di array

```
<? function tornaarray()
```

```
{   $myarray=array("pippo","pluto","paperino");
```

```
return $myarray;
```

```
}
```

```
.....
```

```
$arraytornato=tornaarray();
```

```
echo $arraytornato[1];
```

```
?>
```

“Valori” e “riferimenti”

- ◆ Se non specificato il passaggio dei parametri è “per valore”, viene creata una copia delle variabili

```
function fun($par) {  
    $par += 1;  
}  
$a = 5;  
fun($a);  
echo $a. "<br />";
```

- ◆ Il risultato è:

5

“Valori” e “riferimenti”

- ◆ Usando il carattere speciale **&** il passaggio dei parametri è “per riferimento”, viene modificata la variabile originale

```
function fun(&$par) {  
    $par += 1;  
}  
$a = 5;  
fun($a);  
echo $a. "<br />";
```

- ◆ Il risultato è:

Inclusione di file

Per condividere e riutilizzare porzioni di script e funzioni si usa frammentare il codice in più file in modo da poterlo includere quando serve. E' utile per modularizzare il codice in modo da non dover ripetere stessi pezzi di script in piu' parti (ad esempio connessione al database, header o footer di una pagina, form di login etc)

include (*nomefile*);

L'istruzione include va posizionata all'inizio dello script php o comunque prima dei riferimenti a variabili o funzioni definiti nel file da includere. L'interprete PHP segnala errore se il file non viene trovato, proseguendo comunque l'esecuzione

require (*nomefile*) è analogo tranne che genera un errore irreversibile qualora il file non venga trovato