

Videocorso PHP



Funzioni, require, include
facciamo interagire il codice

Alessandro Flora

Funzioni, require e include in PHP

Come sarà impostata questa lezione

- All'interno delle slide ci saranno alcuni riferimenti utili per il successivo ripasso
- In particolare nelle slide sono riportati i concetti di base e la sintassi
- La seconda parte sarà un esercizio svolto; è vivamente consigliato provare a ragionare insieme creando uno script per provare in prima persona



Le funzioni in PHP

Il concetto di funzione

- Blocco di codice che si può richiamare per essere eseguito più volte
- È bene usarle per tenere ordinato lo script

```
function esempio() {  
    // codice della funzione  
}
```



Definizione della
funzione esempio

```
esempio();
```



Richiamiamo la
funzione esempio

Le funzioni in PHP

Gli argomenti nelle funzioni

- Si indicano come delle variabili
- Nel resto dello script non saranno visibili
- Vengono passati per valore (anche i vettori ma non gli oggetti)

```
function esempio($numero) {  
    echo $numero;  
}
```



Definizione della funzione esempio
con un argomento **\$numero**

```
$a = 5;  
esempio($a)
```



Passiamo alla funzione il valore
contenuto in **\$a** per valore (viene
copiato in **\$numero**)

Le funzioni in PHP

Gli argomenti nelle funzioni (passaggio per riferimento)

- Si usa una forma analoga a quella per valore
- Si usa il simbolo & posto prima del nome dell'argomento

```
function esempio(&$a) {  
    $a = 4;  
}
```



Definizione della funzione esempio con un argomento **\$a** passato per riferimento

```
$numero = 5;  
esempio($numero);
```



Passiamo alla funzione il valore contenuto in **\$numero** per riferimento. Dopo la chiamata **\$numero** avrà valore pari a 4

Le funzioni in PHP

Passaggio per valore e per riferimento

- Quando si passa un parametro per valore si sta effettuando una copia del valore contenuto dalla variabile; possiamo dunque modificarlo a nostro piacimento nella funzione senza ripercussioni sulla variabile originaria
- Quando si passa un parametro per riferimento stiamo passando alla funzione la sua collocazione nella memoria e dunque stiamo modificando la variabile originaria

Attenzione: non possiamo passare per riferimento qualunque cosa che non sia una variabile (per esempio numeri, stringhe o altro definito alla chiamata)

Le funzioni in PHP

La restituzione di valori nelle funzioni

- Utilizziamo la keyword **return** seguita dal nome della variabile o dal valore restituito

```
function prova($a) {  
    $b = $a*2;  
  
    return $b;  
}
```



La funzione restituisce il valore di **\$b** ovvero il doppio di **\$a**

```
$c = prova(3);
```



\$c assume il valore restituito da **prova(3)** ovvero 6

Le funzioni in PHP

Il controllo del tipo di dato in ingresso e in uscita

- Possiamo specificare dei tipi per i parametri e per la funzione; se i dati passati non fossero di quel tipo (o riconducibile tipo stringhe con numeri) otterremo errore

```
function prova(int $a): int {  
    $b = $a*2;  
  
    return $b;  
}
```



La funzione riceverà
esclusivamente valori interi e
restituirà esclusivamente un intero

```
$c = prova(3);
```




Non avremo ricevuto errore
neppure se avessimo passato a
prova '3' perché riconducibile
all'intero 3

require e include in PHP

Collegare gli script

- Le istruzioni `require` e `include` hanno significato identico a meno della gestione degli errori (`require` dà errore se non trova lo script indicato, `include` no)
- Lo scopo è quello di eseguire interamente uno script PHP (eventualmente integrando classi e funzioni definite in tale file)

// codice precedente

`require 'script.php';` 

// codice successivo

Dopo il codice precedente ma prima di quello successivo verrà interamente eseguito il codice presente all'interno dello script denominato **script.php**

require e include in PHP

Collegare gli script: i percorsi

- Se inseriamo lo script nella stessa cartella del file da cui è chiamato possiamo indicare solamente il nome dello script

✓ SERVER

🐘 index.php

🐘 script.php



```
// codice precedente
```

```
require 'script.php';
```

```
// codice successivo
```

require e include in PHP

Collegare gli script: i percorsi

- Se inseriamo lo script in una cartella diversa dal file da cui è chiamato dobbiamo anche indicare il/i nome/i delle cartelle in cui è posto script

```
✓ SERVER
  ✓ prova
    🐘 script.php
    🐘 index.php
```



```
// codice precedente
require 'prova/script.php';
// codice successivo
```

header in PHP

La funzione header

- Serve a specificare le intestazioni della pagina (formato, chi può effettuare collegarsi,...)
- La useremo per reindirizzare mediante la seguente sintassi
- A differenza di **require** e **include** non si rimane nello script ma si esce dallo stesso

```
header("location: /pagina.html");
```



Analogamente a prima dobbiamo indicare un percorso relativo. In questo caso saremo portati alla pagina chiamata **pagina.html** collocata nella cartella principale del progetto

Richiami di HTML e CSS

HTML e CSS

- Sono linguaggi di formattazione e non di programmazione
- Possiamo usarli per descrivere il contenuto grafico di una pagina web
- Li useremo per dare un aspetto invitante alle applicazioni web



contenuto grezzo



personalizzazione grafica

Richiami di HTML e CSS

Struttura HTML di una pagina

```
<!DOCTYPE html>

<html lang="it-IT">
  <head>
    <title>Titolo della pagina</title>
    <meta charset="UTF-8">
    <link rel="stylesheet" href="style.css">
  </head>

  <body>
  </body>
</html>
```



head: parte relativa alle informazioni sulla pagina e eventuali link a file css, favicon, meta, ...



body: parte della pagina visibile dall'utente con all'interno i vari tag

Richiami di HTML e CSS

Tag HTML più utili

```
<section></section> <!-- sezione della pagina -->
<article></article> <!-- articolo -->
<p></p> <!-- paragrafo -->
<aside></aside> <!-- spazio laterale -->
<header></header> <!-- intestazione -->
<nav></nav> <!-- barra o menù di navigazione -->
<footer></footer> <!-- piè di pagina -->
<label></label> <!-- legenda (moduli) -->
<a href=""></a> <!-- ipertesto -->
<img src=""> <!-- immagine -->
<ul></ul> <!-- elenco puntato -->
<ol></ol> <!-- elenco ordinato -->
<li></li> <!-- elemento dell'elenco ordinato o puntato-->
```

Richiami di HTML e CSS

Struttura di un file CSS

selettore



```
body {  
    font-family: 'Liberation Sans', sans-serif;  
    font-size: 20px;  
}
```

```
.classeDiElementi {  
    width: 20%;  
}
```



valore

```
#elementoDaID {  
    border: 1px solid black;  
}
```



proprietà

Richiami di HTML e CSS

Inserire gli ID e le classi in HTML

Utilizziamo la sintassi riportata in seguito. Gli ID tendenzialmente identificano un solo elemento mentre le classi identificano un gruppo di elementi con proprietà analoghe. Questa distinzione è utile per gli script JavaScript mentre è meno trasparente in HTML e CSS.

```
<div id="inseriamoID">  
  <p class="inseriamoClasse"></p>  
</div>
```