

IL LINGUAGGIO PHP

PHP è un linguaggio di scripting per server e un potente strumento per creare pagine Web dinamiche e interattive.

Che cos'è PHP?

- PHP è l'acronimo di "PHP: Hypertext Preprocessor"
- PHP è un linguaggio di scripting open source ampiamente utilizzato
- Gli script PHP vengono eseguiti sul server

Che cos'è un file PHP?

- I file PHP possono contenere testo, HTML, CSS, JavaScript e codice PHP
- Il codice PHP viene eseguito sul server e il risultato viene restituito al browser come semplice HTML
- I file PHP hanno estensione ".php"

Cosa può fare PHP?

- PHP può generare contenuti di pagine dinamiche
- PHP può creare, aprire, leggere, scrivere, eliminare e chiudere file sul server
- PHP può raccogliere i dati dei moduli
- PHP può inviare e ricevere cookie
- PHP può aggiungere, eliminare, modificare i dati nel tuo database
- PHP può essere utilizzato per controllare l'accesso dell'utente
- PHP può crittografare i dati

Sintassi PHP

Uno script PHP viene eseguito sul server e il risultato HTML semplice viene restituito al browser.

Uno script PHP può essere posizionato in qualsiasi punto del documento.

Uno script PHP inizia con `<?php` e termina con `?>`:

```
<?php  
// PHP code goes here  
?>
```

L'estensione di file predefinita per i file PHP è ".php".

Un file PHP normalmente contiene tag HTML e alcuni codici di scripting PHP.

Tutte le istruzioni PHP terminano con ";".

Case sensitive

In PHP le parole chiave (es: if, else, while, echo, ecc), classi, funzioni e funzioni definite dall'utente **NON SONO** case sensitive.

I nomi delle variabili **SONO** case sensitive.

Commenti

I commenti sono fondamentali per stesura e leggibilità del codice perché consentono di facilitare la comprensione di scelte progettuali e passaggi.

I commenti possono essere inseriti in due modi:

- commento su singola riga
`// Questo è un commento`
- commento su righe multiple
`/*
Questo
è un
commento
*/`

Variabili in PHP

A differenza di altri linguaggi di programmazione, PHP non ha alcun comando per dichiarare una variabile. Viene creato nel momento in cui gli si assegna per la prima volta un valore.

```
<?php  
$txt = "Hello world!"; // se il valore è una stringa va racchiuso tra ""  
$x = 5;  
$y = 10.5;  
?>
```

Regole per le variabili PHP:

- Una variabile inizia con il segno \$, seguito dal nome della variabile
- Un nome di variabile deve iniziare con una lettera o il carattere di sottolineatura
- Un nome di variabile non può iniziare con un numero
- Un nome di variabile può contenere solo caratteri alfanumerici e caratteri di sottolineatura (Az, 0-9 e _)
- I nomi delle variabili fanno distinzione tra maiuscole e minuscole (\$age e \$AGE sono due variabili diverse)

Strutture condizionali

- **if-esle**

```
<? if (condizione)  
  {  
      istruzioni se condizione vera;  
  }  
  else  
  {  
      istruzioni se condizione falsa;  
  }  
?>
```

- **switch**

```
<?
    switch (variabile)
    {
        case valore1:
            istruzioni se variabile=valore1;
            break;
        case valore2:
            istruzioni se variabile=valore2;
            break;
        .....
        default:
            istruzioni default;
    }
?>
```

Strutture di iterazione

- **ciclo while**

```
<?
    while (condizione)
    {
        istruzioni;
    }
?>
```

- **ciclo do-while**

```
<?
    do{
        istruzioni;
    }while (condizione)
?>
```

- **ciclo for**

```
<?
    for (valore iniziale contatore; condizione; incremento)
    {
        istruzioni;
    }
?>
```

Array

Per dichiarare un array in PHP si utilizza la funzione **array()**:

```
$nomi = array('luca', 'anna', 'luisa', 'marco', 'giulia');
```

L'indice dell'array parte da 0, quindi l'array \$nomi avrà 5 elementi indicizzati da 0 a 4.

L'accesso all'elemento dell'array viene effettuato indicando tra parentesi quadre l'indice dell'elemento desiderato:

```
$nomi[2]
```

Per aggiungere un elemento in coda all'array:

```
$nomi[]='lara';
```

Oltre agli array scalari, dove gli elementi sono individuati da un indice numerico, in PHP è possibile definire **array associativi**, dove ogni elemento è individuato da una coppia **chiave-valore**. L'indice numerico è sostituito da una chiave di identificazione che rappresenta la chiave di accesso all'elemento stesso.

```
$eta = array("luca"=>18, "anna"=>19, "luisa"=>14);
```

Se volessimo stampare l'età di luisa:

```
echo $eta[luisa];
```

Per ciclare sugli elementi di un array associativo si utilizza il costrutto **foreach**.

```
foreach ($nome_array as $chiave => $valore){  
    istruzioni da eseguire su ogni element dell'array;  
}
```

Dove la variabile **\$chiave** conterrà il valore della chiave dell'i-esimo elemento e la variabile **\$valore** conterrà il valore dell'i-esimo elemento.