

# Introduzione alle funzioni PHP

Slide tratte da:

**W3CSCHOOLS.COM**

([https://www.w3schools.com/php/php\\_functions.asp](https://www.w3schools.com/php/php_functions.asp))

## Introduzione alle Funzioni PHP

- Il vero potere di PHP deriva dalle sue funzioni.
- PHP offre più di 1000 funzioni integrate
- Naturalmente è possibile creare funzioni personalizzate.
- Consulta questa [lista](#) per dare un'occhiata alle funzioni predefinite.

## Funzioni Utente Definite in PHP

- Oltre alle funzioni integrate di PHP, è possibile creare le tue funzioni.
- Una funzione è un *blocco di istruzioni* che può essere utilizzato ripetutamente in un programma.
- Una funzione non verrà eseguita automaticamente quando si carica una pagina.
- Una funzione verrà eseguita solo quando viene chiamata.

## Creare una Funzione Utente in PHP

- La dichiarazione di una funzione utente inizia con la parola chiave function.
- Sintassi:

PHP

```
function nomeFunzione() {  
    istruzioni da eseguire;  
}
```

- Nota: Il nome di una funzione deve iniziare con una lettera o un underscore.
- I nomi delle funzioni NON sono sensibili alle maiuscole/minuscole.
- Suggerimento: Assegna alla funzione un nome che rifletta cosa faccia!

## Esempio di Funzione Utente

PHP

```
// Definizione della funzione  
<?php  
function scriviMessaggio() {  
    echo "Ciao, mondo!";  
}  
  
scriviMessaggio(); // chiama la funzione  
?>
```

## Argomenti delle Funzioni PHP

- Le informazioni possono essere passate alle funzioni tramite argomenti.
- Gli argomenti sono specificati dopo il nome della funzione, all'interno delle parentesi.
- Puoi aggiungere tanti argomenti quanti ne vuoi, separandoli con una virgola.

## Esempio di Funzione con Argomento

PHP

```
<?php  
function cognome($nome) {  
    echo "$nome Rossi."<br>;  
}  
  
cognome("Andreagiancarlo");  
cognome("Guendalina");  
cognome("Asdrubale");  
cognome("Melchiorre");  
cognome("Dorotea");  
?>
```

## Funzioni PHP e Tipi di Dati

- In PHP, non è necessario dichiarare esplicitamente il tipo di dati di una variabile.
- PHP associa automaticamente un tipo di dati alla variabile in base al suo valore.
- Tuttavia, in PHP 7, è possibile utilizzare le dichiarazioni di tipo (usando la direttiva `strict_types`).
- Se abilitate, verrà generato un "Fatal Error" se il tipo di dati non corrisponde.

## Esempio con Dichiarazioni di Tipo (Senza Strict)

PHP

```
<?php
function sommaNumeri(int $a, int $b) {
    return $a + $b;
}
echo sommaNumeri(5, "5 giorni");
/* poiché Strict NON è abilitato, "5 giorni" viene
convertito in int(5), e restituirà 10 */
?>
```

## Esempio con Dichiarazioni di Tipo (Con Strict)

PHP

```
<?php
declare(strict_types=1);

function sommaNumeri(int $a, int $b) {
    return $a + $b;
}
echo sommaNumeri(5, "5 giorni");
/* poiché Strict è abilitato e "5 giorni" non è un
intero, verrà generato un errore */
?>
```

## Valore Predefinito degli Argomenti delle Funzioni

PHP

```
<?php
declare(strict_types=1);

function impostaAltezza(int $altezzaMinima = 50) {
    echo "L'altezza è: $altezzaMinima <br>";
}

impostaAltezza(350);
impostaAltezza(); // utilizzerà il valore predefinito di 50
impostaAltezza(135);
impostaAltezza(80);
?>
```

## Funzioni PHP - Restituzione dei Valori

- Le funzioni possono restituire un valore
- Usa l'istruzione `return`.

## Esempio di Funzione con Restituzione di Valore

PHP

```
<?php
declare(strict_types=1);
function somma(int $x, int $y) {
    $z = $x + $y;
    return $z;
}

echo "5 + 10 = " . somma(5, 10) . "<br>";
echo "7 + 13 = " . somma(7, 13) . "<br>";
echo "2 + 4 = " . somma(2, 4);
?>
```

14

## Dichiarazioni di tipo per il ritorno delle funzioni

- PHP, dalla versione 7 e seguenti, supporta le dichiarazioni di tipo per l'istruzione di ritorno.
- Per dichiarare un tipo di ritorno, aggiungi

`: tipo`

prima della parentesi graffa di apertura per la definizione del body della funzione.

## Esempio con Dichiarazioni di Tipo per il Ritorno

PHP

```
<?php
declare(strict_types=1);

function sommaNumeri(float $a, float $b): float {
    return $a + $b;
}

echo sommaNumeri(1.2, 5.2);
?>
```

16

N.B. Puoi specificare un tipo di ritorno diverso dai tipi degli argomenti.

## Passare argomenti per riferimento

- In PHP, di solito, gli argomenti vengono passati per valore.
- Per passare gli argomenti per riferimento utilizza l'operatore &
- Questo ti consentirà di modificare la variabile originale all'interno di una funzione.

## Esempio di Passaggio per Riferimento

PHP

```
<?php
function aggiungiCinque(&$valore) {
    $valore += 5;
}

$numero = 2;
aggiungiCinque($numero);
echo $numero;
?>
```

18

## Abbiamo finito!

Ora è importante fare  
parecchi esercizi!

