

Corso base PHP

Powered by www.dcopelli.it

" Trucchi e consigli di programmazione Web "



>> Le basi del linguaggio PHP: le istruzioni, la sintassi e il concetto di variabile

Ciao, spero tu abbia installato l'editor di testo preferito o quello che ti ho consigliato e che tu sia riuscito a creare il tuo "server Web" casalingo, perché da adesso, inizieremo il viaggio nella programmazione vera e propria.

Abbiamo ora tutti gli strumenti per scrivere la nostra prima (semplice) applicazione in PHP.

In questa lezione, imparerai a mostrare dei messaggi personalizzati ai navigatori del tuo ipotetico sito web. Partiamo nel creare un'applicazione molto semplice ma fondamentale: mostrerai ai tuoi navigatori il messaggio "Benvenuto nel mio sito".

Non è un'applicazione entusiasmante, ma giusto per capire come si inizia e sgomberare il campo dalle domande iniziali che potrebbero sorgere nella tua mente.

Come primo basilare esercizio, crea un file scritto pari pari in questo modo:

```
<?php  
echo "Benvenuto nel mio sito";  
?>
```

e salvarlo con il nome "**test.php**"

NB: Attenzione all'estensione: ho scritto .php !!!

NB: Attenzione a rispettare le maiuscole e minuscole, perché PHP interpreta test.php diversamente da Test.php. In gergo tecnico si dice che il linguaggio PHP è "case sensitive" ossia **sensibile alle maiuscole e minuscole**.

Semplice non è vero? Dopo averlo scritto e salvato, lo dovrai caricare nella cartella che ti ho indicato nel video dove ho parlato di web server locale.

Analizziamo allora in dettaglio il tuo primo "programma", anche se in realtà è solo una riga di istruzioni.

PUNTO 1.

Dovrai sempre iniziare i pezzi di programmazione in PHP con il simbolo "<?php" e terminarli con il simbolo "?>". Questi sono chiamati rispettivamente "TAG di apertura" e "TAG di chiusura".

NB: Un errore classico è quello di dimenticarsi di chiudere la parte scritta in PHP con il simbolo "?>". Attenzione quindi soprattutto all'inizio!!!

NB: Non occorre iniziare a scrivere il codice PHP partendo dal primo carattere sulla sinistra. Puoi spostarti a destra come vuoi.

Esempio:

```
<?php  
echo "Benvenuto nel mio sito";  
?>
```

oppure

```
<?php  
    echo "Benvenuto nel mio sito";  
?>
```

etc.

PUNTO 2.

Ho parlato di pezzi di programmazione perché all'interno di una pagina del tuo sito puoi mescolare pezzi di codice HTML e pezzi di codice PHP. Questa è una caratteristica di molti linguaggi di programmazione compreso PHP. Pertanto fra poche lezioni e' probabile che la tua homepage sarà costituita da più pezzi, come nell'esempio qui sotto:

```
<html>  
<body>  
  
<?php  
primo pezzo di codice in PHP  
?>  
...  
corpo della tua homepage con il testo e le immagini  
...  
<?php  
secondo pezzo di codice in PHP  
?>  
...  
altre righe della tua homepage  
...  
<?php  
terzo pezzo di codice in PHP  
?>  
....  
</body>  
</html>
```

Come vedi ho sempre aperto e chiuso i pezzi di codice PHP con i due tag "<?php" e "?>".

NB: Puoi inserire quanti pezzi di codice vuoi, non c'è un limite.

NB: Non occorre andare su una nuova riga per scrivere i due tag di apertura e chiusura. Potresti benissimo scrivere:

`<?php pezzo di codice in PHP ?>`

A volte conviene usare un metodo, a volte l'altro. L'importante è che il tag `?>` sia **sempre scritto alla fine di tutte le righe in PHP**.

PUNTO 3.

All'interno dei 2 tag ho scritto:

`echo "Benvenuto nel mio sito";`

Ebbene, attraverso questa scritta "magica", puoi far "stampare" a video quello che vuoi.

In questo caso non serve a molto, nel senso che potevi scriverlo direttamente senza usare "echo", ma vedrai che questa parolina sarà parte indispensabile di molti tuoi futuri e più complessi programmi.

Pertanto prima cosa da annotare nel tuo quaderno:

Per scrivere e far visualizzare sul monitor del visitatore del tuo sito, una frase o un gruppo di frasi, bisogna usare:

`echo "Testo da visualizzare a video";`

NB: Attenzione a non dimenticare le **doppie virgolette**. Queste sono indispensabili e ti consiglio di metterle sempre.

NB: Attenzione a non dimenticare il **punto e virgola** alla fine. Questo non è sempre indispensabile ma ti consiglio di metterlo sempre.

NB: E se volessi scrivere una frase molto lunga? A volte conviene spezzarla, in modo che se devi fare delle modifiche, queste siano più rapide.

Come spezzarla? Devi semplicemente andare a capo riga.

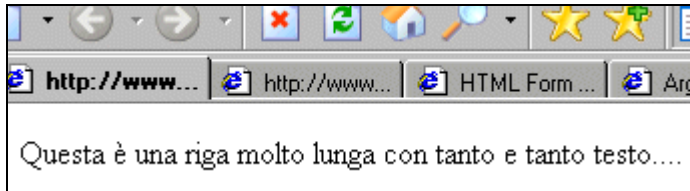
Vediamo allora un esempio: invece di scrivere:

`echo "Questa è una riga molto lunga con tanto e tanto testo...."`

potresti scrivere:

`echo "Questa è una riga
molto lunga con tanto
e tanto testo....";`

Il risultato è lo stesso come puoi vedere qui sotto:



ma la leggibilità del codice PHP è migliore se la riga è molto lunga.

NB: Occhio a terminare l'ultima riga con le **doppie virgolette e il punto e virgola!**

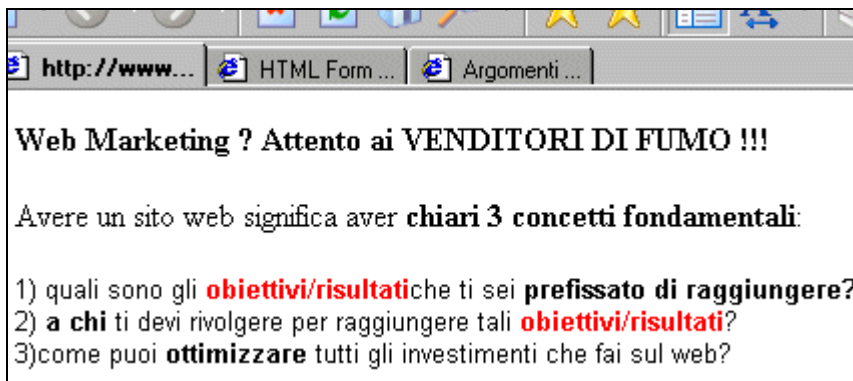
Un'alternativa era scriverla così:

```
echo "Questa è una riga"  
echo " molto lunga con tanto"  
echo " e tanto testo...."
```

Vedi tu quale è più veloce da scrivere.

Bene, prima di terminare questa breve lezione introduttiva, volevo solo ricordarti che il testo che inserisci dopo "echo", può essere anche del codice HTML, anzi a dire il vero, nella maggior parte delle tue applicazioni future, sarà proprio del codice HTML.

Vuoi un esempio ? Eccolo preso dal mio sito:



Questa pagina è stata ottenuta con questa istruzione:

```
<?php  
echo "<b>Web Marketing ? Attento ai VENDITORI DI FUMO !!!</b><br><br>Avere un sito web  
significa aver <b>chiari 3 concetti fondamentali</b>:<p><font face=arial size=2> 1) quali sono gli  
<b><font color=#FF0000>obiettivi/risultati</font></b>che ti sei <b>prefissato di raggiungere?  
</b><br>2) <b>a chi</b> ti devi rivolgere per raggiungere tali <b><font  
color=#FF0000>obiettivi/risultati</font></b>?<br>3)<font face=arial size=2>come puoi  
<b>ottimizzare</b> tutti gli investimenti che fai sul web?</font>"  
?>
```

Troppo complesso?

Niente paura, tutto questo testo HTML lo puoi scrivere con un normale editor di pagine web come Dreamweaver o Visual Studio e poi, copiarlo e incollarlo all'interno della scritta "echo" con le regole che ti ho insegnato prima.

Dopotutto, spero tu abbia già delle basi di HTML. Se non è così, ti consiglio di leggere qualche guida per farti un'idea di quali sono i tag principali! In libreria ne puoi trovare di molto semplici e con illustrazioni.

NB: Avrai notato che **NON** ho inserito le normali doppie virgolette tipiche di molti tag HTML come per esempio `<p style="color: #FF0000">`.

Secondo te perché ?

Ebbene questo è un altro tipico problema di quando si scrive del codice HTML e poi lo si integra con il programmino che si sta scrivendo in PHP.

Se avessi scritto il testo di prima, mantenendo il corretto stile dei tag HTML, ossia scrivendo anche le virgolette ove necessario, allora il programmino **sarebbe andato in errore**.

Per ovviare al problema esistono **3 soluzioni**:

- la prima consiste nel togliere le doppie virgolette come ho fatto io;
- la seconda è quella di scrivere al posto di `<p style="color: #FF0000">` - questo - `<p style=\'color: #FF0000\'>` - cioè far precedere le doppie virgolette dal carattere \
- la terza è quella di inserire invece delle doppie virgolette, le singole virgolette così: `<p style='color: #FF0000'>`

Troppo semplice ? Allora complichiamo un pò di più le cose.

Le Variabili

Una caratteristica fondamentale di tutti i linguaggi di programmazione e anche della tecnologia PHP, è la possibilità di utilizzare quelle che vengono definite con il nome di **VARIABILI**.

Cosa sono le VARIABILI ?

In modo molto semplice, puoi pensare ad una variabile come ad un **contenitore**. Così come l'olio lo si conserva in bottiglie di vetro, così una stringa (es. "Ciao a tutti", "hello", "benvenuto" etc) o un numero (es. 35,45,12 etc), potresti "conservarla" in una variabile.

Oltre alla bottiglia di vetro potresti usarne una di plastica oppure di ceramica a seconda di che liquido devi conservare. La stessa cosa si fa in PHP, ossia a seconda di cosa si mette dentro si può usare una variabile di un tipo oppure di un altro.

Ma a cosa servono allora le VARIABILE ?

Proseguendo con l'analogia dimmi, secondo te, a cosa serve la bottiglia ? Oltre alla funzione principale di **contenere** qualcosa, serve per **trasportare** questo qualcosa dove voglio io. Nel caso della programmazione, questo qualcosa tipicamente è una stringa, un numero prelevato da un database o da un form, e che si deve avere la necessità di usare in più punti del programma.

Vediamo allora subito un esempio pratico di utilizzo di una variabile all'interno di un programma.

Esempio: definisco una variabile con il nome **bottiglia** al cui interno metto la stringa **"olio"**

```
<?php
$bottiglia = 'olio';
?>
```

Attraverso questa scrittura **\$nomevariabile = 'Contenuto da inserire'**, non fai altro che inserire all'interno della **variabile "\$bottiglia"** il **valore "olio"**. In gergo tecnico si dice che hai **assegnato** alla variabile \$bottiglia il valore "olio".

Quest'ultima, come abbiamo visto in precedenza, in gergo tecnico si dice **stringa** ossia è un pezzo di testo, ma potresti anche usare dei numeri o delle frasi o qualcosa di più complesso come vedremo dopo.

NB: Errore tipico e' dimenticarsi di inserire il **simbolo del dollaro \$** prima del nome della variabile. Quest'ultima non deve contenere spazi (NO! \$la bottiglia = ""), e caratteri particolari come &,/,) etc.

NB: Altro **errore tipico** è inserire delle frasi che hanno all'interno delle virgolette singole.

Esempio:

```
<?php
$domanda = 'Qual è il migliore amico dell'uomo?';
?>
```

In questo caso si verifica un errore, perché l'apostrofo di "dell'uomo", viene considerato di chiusura della frase, anche se questa non è ancora terminata. Bisogna allora usare il simbolo \ prima dell'apostrofo, come si era visto per le doppie virgolette dei tag HTML.

```
<?php
$domanda = 'Qual è il migliore amico dell\'uomo?';
?>
```

Quindi altra cosa da annotare nel tuo quaderno:

Per **assegnare** un valore ad una variabile, devi usare la seguente sintassi:

\$nomevariabile = 'Stringa da inserire'; (nel caso di stringhe di testo es. 'Ciao a tutti', 'Benvenuto')
\$nomevariabile = Numero; (nel caso di numeri es. 12,34,45)
\$nomevariabile = "Stringa con variabili"; (vedi dopo)

NB: Attenzione a **non dimenticare il punto e virgola alla fine**. Questa è una caratteristica del linguaggio PHP e molto spesso vedrai sarà causa di errori da parte tua. Quindi OCCHIO!!!

Esempio: definisco una variabile con il nome **eta** al cui interno metto il numero 54.

```
<?php
$eta = 54;
?>
```

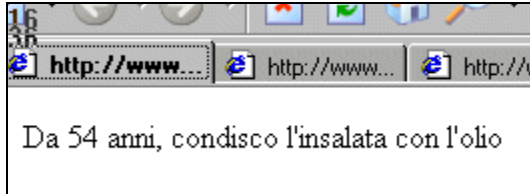
NB: ho volutamente scritto \$eta **senza accento**, perché una regola fondamentale è che **non puoi usare all'interno dei nomi di variabile gli accenti, gli spazi, e altre simboli dell'alfabeto come &,%,\$**

etc.

Mettendo insieme le due variabili potresti scrivere:

```
<?php
$eta = 54;
$bottiglia = 'olio';
echo "Da $eta anni, condisco l'insalata con l'$bottiglia";
?>
```

A leggerla così, non ha molto senso, ma se la fai elaborare dal server, questo è il risultato:



Tornando allora ad uno dei primi esercizi di questa lezione, vediamo come possiamo iniziare a scrivere le nostre applicazioni in modo **ELEGANTE**.

Invece di scrivere:

```
<?php
echo "<b>Web Marketing ? Attento ai VENDITORI DI FUMO !!!</b><br>Avere un
sito web significa aver <b>chiari 3 concetti fondamentali</b>:<p><font face=arial size=2> 1) quali
sono gli <b><font color=#FF0000>obiettivi/risultati</font></b>che ti sei <b>prefissato di
raggiungere? </b><br>2) <b>a chi</b> ti devi rivolgere per raggiungere tali <b><font
color=#FF0000>obiettivi/risultati</font></b>?<br>3)<font face=arial size=2>come puoi
<b>ottimizzare</b> tutti gli investimenti che fai sul web?</font>";
?>
```

potrei sfruttare un contenitore che ho chiamato HTML, all'interno del quale vado ad inserire tutto il codice che compare dopo echo, in questo modo:

```
<?php
$HTML = '<b>Web Marketing ? Attento ai VENDITORI DI FUMO !!!</b><br>Avere un sito web
significa aver <b>chiari 3 concetti fondamentali</b>:<p><font face=arial size=2> 1) quali sono gli
<b><font color=#FF0000>obiettivi/risultati</font></b>che ti sei <b>prefissato di raggiungere?
</b><br>2) <b>a chi</b> ti devi rivolgere per raggiungere tali <b><font
color=#FF0000>obiettivi/risultati</font></b>?<br>3)<font face=arial size=2>come
puoi<b>ottimizzare</b> tutti gli investimenti che fai sul web?</font>';
echo "$HTML";
?>
```

Forse stai intuendo dove voglio arrivare. Infatti cosa ti viene in mente se ti dicessi che all'interno del testo della variabile \$HTML potresti inserire ulteriori variabili in questo modo:

```
<?php
$nome = 'Davide';
$HTML = "Ciao $nome, sei esperto di <b>Web Marketing ? Attento ...</b>..";
echo "$HTML";
?>
```

Se pertanto potessi riconoscere il navigatore che accede al mio sito e recuperare il nome da un archivio in cui ho inserito i suoi dati, allora potrei ogni volta visualizzare un messaggio del tipo:

Ciao **Davide**, sei esperto di **Web Marketing ? Attento ...**

Oppure :

Ciao **Luca**, sei esperto di **Web Marketing ? Attento ...**

Potente non è vero?

NB: Avrai notato che alla variabile \$HTML ho assegnato una stringa, utilizzando le **doppie virgolette**. Chiediti allora cosa succede se invece usi le singole virgolette, come mostrato qui sotto:

```
<?php
$nome = 'Davide';
$HTML = 'Ciao $nome, sei esperto di <b>Web Marketing ? Attento ...</b>..';
echo "$HTML";
?>
```

Ebbene quello che vedrai a video sarà:

Ciao \$nome, sei esperto di **Web Marketing ? Attento ...**

Ossia la variabile \$nome non è stata cambiata con il suo valore. Questo perché **tutto quello che tu inserisci all'interno delle singole virgolette, viene visualizzato così com'è, senza che il server lo elabori per vedere se ci sono all'interno delle variabili da sostituire con il relativo valore.**

In gergo tecnico si dice che tutto quello all'interno delle doppie virgolette viene **"interpretato"** dal server e se all'interno trova delle variabili (es. \$nome, \$HTML etc), queste sono sostituite con il relativo valore, prima di mostrare a video l'intera frase.

Vedrai in seguito come ti diventeranno familiari le variabili, soprattutto quando vedremo insieme come recuperare i valori che un tuo potenziale cliente inserisce all'interno di un modulo web presente nel tuo sito, per creare un tuo database marketing personale.

Ho analizzato un pò tutte le problematiche a cui potresti andare incontro all'inizio della scrittura del tuo primo programma in PHP.

L'importante è chiarire ogni dubbio subito e per questo **ti invito a fare delle prove**, per verificare se hai appreso tutto quello spiegato in questa lezione.

Se ritieni che questo corso possa essere utile ad un tuo amico allora comunicagli questo link:

<http://www.dcopelli.it/corsi/corso-php.htm?imk=amico>

in modo che possa anche lui ricevere gratuitamente queste lezioni. Un tuo prezioso aiuto per mantenere gratis il corso base.

Alla prossima lezione, in cui inizieremo a rendere dinamiche le pagine web del tuo sito!

Buona programmazione in PHP

Davide Copelli {ing}