



## Capitolo 1

Iniziamo con un po' di storia del PHP, questo comodo, facile e rapido linguaggio venne creato nel 1994 da Rasmus Lerdorf, come alternativa al piu' famoso Server Side Language del periodo, il perl.

### **Cosa vuol dire Server Side Language?**

Un SSL e' un linguaggio (Es. Perl, PHP, ASP) che viene eseguito sul server e che restituisce ed invia all'utente delle pagine html.

### **Quali sono i vantaggi dei SSL?**

Come prima cosa l'enorme flessibilita' e comodita' d'uso, ad esempio noi possiamo interagire facilmente con il server, per ottenere informazioni banali (es. data e ora del server) o piu' complesse (es. contenuto di cartelle), cosa che non potremmo fare con linguaggi interpretati in locale, come Java.

Visto che il sorgente e' ospitato sul server, si possono realizzare script piu' sicuri e funzionali. Ad esempio se noi vogliamo proteggere l'accesso ad una pagina con una password, se usiamo java, l'utente potra' visionare il sorgente dello script e quindi anche la password, mentre in un SSL trovandosi il sorgente sul server, nessuno potra' visionarlo (legalmente) e quindi non sara' tanto facile trovare la password.

Infine l'enorme interfacciabilita' con database quale MySQL rende i SSL, ed in particolare il php, enormemente pratico, rapido e funzionale.

### **Concetti fondamentali sulla programmazione in PHP:**

Iniziamo a vedere come inserire nelle pagine web i nostri script. Il codice del nostro script dovra' essere inserito nei tag "<?php //codice ?>", come nel seguente esempio:

```
<?php
//Qui andra' inserito il nostro codice
?>
```

Come in tutti i linguaggi la funzione principale e' quella di visualizzare sul monitor dei messaggi, vediamo come fare con il php:

```
<?php
echo 'Ciao!!';
echo "Ciao!!";
```

```
<?php
```

```
?>
```

```
<?php
```

```
print 'Ciao!!';
```

```
print "Ciao!!";
```

```
?>
```

Prima di tutto osserviamo come ogni riga di codice deve essere chiusa con il solito ";".

Studiamo ora le funzioni, i due script visualizzeranno esattamente lo stesso output, infatti le due funzioni "print" e "echo" sono equivalenti, scrivono sulla pagina il testo chiuso tra apici ( ' ) o doppi apici ( " ), le funzioni di questi due segni di punteggiatura le studieremo piu' avanti quando parleremo di variabili, infatti al contrario di echo/print la loro funzione non e' uguale.

Guardando ora l'output della pagina nel nostro browser preferito, noteremo che la pagina e' un tantino brutta, e' allineata a sinistra, le parole sono tutte attaccate, per cercare di formattare un po' il testo dobbiamo usare ora l'html, che e' necessario per "l'interfaccia grafica del php".

Risistemiamo il codice di conseguenza:

```
<?php
```

```
echo '<center>Ciao!!</center><br>';
```

```
echo "<center>Ciao!!</center><br>";
```

```
?>
```

```
<?php
```

```
print '<center>Ciao!!</center><br>';
```

```
print "<center>Ciao!!</center><br>";
```

```
?>
```

Ora l'output e' decisamente cambiato; cosi' come abbiamo utilizzato i tag "<br>" e "<center>" possiamo utilizzare tutti gli altri tag allo stesso modo.

**Next Part: A grande richiesta inserimento del codice php in pagine HTML.**

**Lord\_Dex**

[kapo.web@gmail.com](mailto:kapo.web@gmail.com)