

Capítulo 1: Generalidades.

Introducción:

PHP es un lenguaje interpretado diseñado para favorecer el desarrollo de web-sites dinámicos y aplicaciones para web-sites. La distribución más popular de PHP es como módulo para el web-server Apache, aunque puede funcionar con las limitaciones que ya conocemos, como un interprete para ejecutar aplicaciones Cgi en aquellos web-servers que no lo soporten como módulo.

PHP se distribuye en formato open-source y es gratuito, una instalación habitual de PHP consiste en compilar el módulo PHP y luego recompilar el Apache para que utilice el módulo recientemente compilado.

Generalidades:

La característica más importante de PHP es que permite combinar código html y código php en una misma página (de extensión php), por ejemplo:

```
<HTML>
<HEAD><TITLE>Hola</TITLE></HEAD>
<BODY>
Hola esta es una prueba. <BR />
<?php
    print("Hola soy una línea generada en php <BR />");
?>
</BODY>
</HTML>
```

Este ejemplo al guardarse en un archivo de extensión .php es automáticamente parseado por el interprete de php cuando el browser envía un pedido. El ciclo es el siguiente:

- El browser envía un pedido de un archivo con extensión php.
- El server analiza que la extensión del request es .php, obtiene el archivo y lo envía al interprete php.
- El interprete php del web-server parsea el archivo en busca de tags <? ?> y procesa todo lo que se encuentre entre dichos tags (puede haber varias apariciones de los tags en un mismo archivo), todo aquello que esta fuera de los tags se envía al browser sin interpretar.
- El resultado combinado de aquello que no debe interpretarse y el resultado del código interpretado se envía al browser.

En nuestro ejemplo el browser recibiría:

```
<HTML>
<HEAD><TITLE>Hola</TITLE></HEAD>
<BODY>
Hola esta es una prueba. <BR />
Hola soy una línea generada en php <BR />
</BODY>
</HTML>
```

Como podemos ver, es muy sencillo combinar código html y php. Para generar html desde php tenemos las siguientes opciones:

- Usar la función “print de php”
- Usar la función “echo de php”
- Cerrar el tag ?> escribir el código html deseado y volver a abrir el tag <?>

Generación de web sites dinámicos usando PHP

La tercera opción es la más eficiente en velocidad cuando el código html que debemos generar es fijo, cuando el código html es dinámico podemos usar una mezcla de print y tags que abren y cierran que suele ser lo mas eficiente, por ejemplo:

```
<form name="<? print("$nombre_form")?>">  
etc...
```

Conceptos básicos en la generación de sites dinámicos con PHP.

Una característica interesante de php es que permite realizar “includes” dentro de un script php, de esta forma se puede modularizar una página o el layout de una página en varios módulos php que se desarrollan en forma independiente, además pueden desarrollarse en php componentes reusables que luego se utilizan en otras páginas o incluso en otros sites.

Una forma común de trabajo usando php para generar sitios dinámicos es definir “templates” o “layouts” en los cuales se divide la página en “zonas” o “módulos” que serán desarrollados en php, el layout de la página con la forma y tamaño de cada zona se puede definir sin problemas usando por ejemplo tablas de html.

A lo largo de este curso desarrollaremos a modo de ejemplo un mini-portal de noticias dinámico al cual le agregaremos servicios o aplicaciones a medida que se estudian distintas características de php. Supongamos que tenemos por el momento un único “template” o “layout” para nuestro sitio que determina la forma en la cual se vera la “home page” del mismo, el equipo de diseño nos entrega el siguiente layout:

LOGO	BANNER		
BUSCADOR			
BOTON S1	BOTON S2	BOTON S3	BOTON S4
Barra de links y Aplicaciones	CONTENIDOS		
Información de copyrigh, y pie de página			

Como podemos ver el site cuenta con 4 secciones que se acceden desde una barra navegadora ubicada debajo del search box, además existe una barra de links y servicios a la izquierda y una zona de contenidos que es la zona principal de la página.

Aun sin saber que funcionalidad tiene o de que forma se debe implementar cada parte podemos esquematizar el layout de la página usando php y html de la siguiente forma:

```
<HTML>  
<HEAD>  
<TITLE>Layout</TITLE>
```

Generación de web sites dinámicos usando PHP

```
</HEAD>
<BODY>
<table width="640" border="1">
<tr><td width="20%">Logo</td><td colspan="3">Banner</td></tr>
<tr><td colspan="4">buscador</td></tr>
<tr><td>s1</td><td>s2</td><td>s3</td><td>s4</td></tr>
<tr><td width="20%">ColIzq</td><td colspan="3">Contenidos</td></tr>
</table>
</BODY>
</HTML>
```

Luego podemos reemplazar cada “zona” de la home page por un include en php que generara dinámicamente la parte de la página en cuestión:

```
<HTML>
<HEAD>
<TITLE>Layout</TITLE>
</HEAD>
<BODY>
<table width="640" border="1">
<tr><td width="20%"><? Include("logo.php");?></td><td
colspan="3"><?include("banner.php");?></td></tr>
<tr><td colspan="4"><?include("buscador.php");?></td></tr>
<tr><?include("botonera.php");?></tr>
<tr><td width="20%"><?include("izq.php");?></td><td
colspan="3"><?include("contenidos_home.php");?></td></tr>
</table>
</BODY>
</HTML>
```

De esta forma hemos modularizado el layout de la página y tenemos como resultado que deben desarrollarse los siguientes módulos:

- logo.php
- banner.php
- buscador.php
- botonera.php
- izq.php
- contenidos_home.php

Generación de web sites dinámicos usando PHP

Funcionalidad de PHP:

- Funciones de calendario y manipulación de calendarios usando MCAL
- Programación orientada a objetos
- Funciones para creación de archivos PDF
- Funciones de manejo de cajeros cybercash
- Parser de documentos XML
- WDDX
- Funciones de compresión de datos
- Manejo de archivos DBM
- Funciones para manipulación de fechas
- Funciones para manejo de directorios
- Funciones de encriptación de datos
- Funciones de acceso al filesystem
- Funciones para manejo de FTP
- Funciones de hashing
- Generación dinámica de imágenes
- Manejo de cuentas de mail IMAP y POP3
- Funciones para envío de mail
- Funciones de networking usando sockets
- Funciones matemáticas
- Serialización de estructuras de datos
- Acceso a bases de datos (Mysql, Oracle, Postgress, Sybase, etc)
- Manejo de expresiones regulares.
- Manejo de sesiones.