



Festival Latinoamericano de Instalación
de Software Libre



¿Qué es PHP?

PHP son las siglas "Personal Home Page". Es un lenguaje de programación pensado en el web de forma que es ideal para la creación de páginas dinámicas. PHP es la versión libre del sistema equivalente de Microsoft ASP.

PHP es un lenguaje encapsulado dentro de los documentos html (aunque los CGI hechos con PHP terminan en extensión .php y no .html). De forma que se pueden introducir instrucciones php dentro de las páginas. Gracias a esto el diseñador gráfico del web puede trabajar de forma independiente al programador. PHP es interpretado por el servidor (apache) generando un HTML con el resultado de substituir las secuencias de instrucciones PHP por su salida.

Por lo tanto una web dinámica con PHP contiene una serie de documentos php que el servidor apache interpreta proporcionando al cliente documentos html con el resultado de las ordenes php.

PHP

<u>Desarrollador</u>	<u>PHP Group</u>
<u>Última versión</u>	4.4.2 5.1.2 / <u>13 de Enero, 2006</u>
<u>S.O.</u>	<u>Multiplataforma</u>
<u>Género</u>	<u>Lenguaje interpretado</u>
<u>Licencia</u>	<u>PHP License 3.01</u>
<u>En castellano</u>	-
<u>Sitio Web</u>	<u>www.php.net</u>

PHP es un lenguaje de programación usado generalmente para la creación de contenido para sitios web. PHP es el (acrónimo recursivo de "**PHP: Hypertext Preprocessor**", inicialmente PHP Tools, o, *Personal Home Page Tools*) es un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios web, y últimamente también para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica usando la librería GTK+.

Visión general

El fácil uso y la similaridad con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores experimentados crear aplicaciones complejas con una curva de aprendizaje muy suave. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones y prácticas.

Debido al diseño de PHP, también es posible crear aplicaciones con una interfaz gráfica para el usuario (también llamada GUI), utilizando la extensión PHP-GTK. También puede ser usado desde la Línea de comandos, de la misma manera como Perl o Python pueden hacerlo, esta versión de PHP se llama PHP CLI (*Command Line Interface*).

Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible

utilizar PHP para generar archivos [PDF](#), [Flash](#), así como imágenes en diferentes formatos, entre otras cosas.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como [MySQL](#), [Postgres](#), [Oracle](#), [ODBC](#), [IBM DB2](#), [Microsoft SQL Server](#) y [SQLite](#); lo cual permite la creación de [Aplicaciones web](#) muy robustas.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los [sistemas operativos](#) tales como [UNIX](#) (y de ese tipo, como [Linux](#)), [Windows](#) y [Mac OS X](#), y puede interactuar con los [servidores de web](#) más populares ya que existe en versión [CGI](#), módulo para [Apache](#), e [ISAPI](#).

El modelo PHP puede ser visto como una alternativa al sistema de [Microsoft](#) que utiliza [ASP.NET/C#/VB.NET](#), a [ColdFusion](#) de la compañía [Macromedia](#), a [JSP/Java](#) de [Sun Microsystems](#), y al famoso [CGI/Perl](#). Aunque su creación y desarrollo se da en el ámbito de los sistemas libres, bajo la licencia [GNU](#), existe además un compilador comercial llamado [Zend Optimizer](#).

Historia

PHP fue originalmente diseñado en Perl, seguidos por la escritura de un grupo de [CGI](#) binarios escritos en el lenguaje [C](#) por el [programador Danés-Canadiense Rasmus Lerdorf](#) en el año [1994](#) para mostrar su [currículum vitae](#) y guardar ciertos datos, como la cantidad de tráfico que su página web recibía. El [8 de junio](#) del [1995](#) fue publicado "Personal Home Page Tools" luego de que Lerdorf lo combinara con su propio *Form Interpreter* para crear [PHP/FI](#).

Dos programadores israelíes de [Technion](#), [Zeev Suraski](#) y [Andi Gutmans](#), reescribieron el analizador gramatical (*parser* en inglés) en el año [1997](#) y crearon la base del PHP 3, cambiando el nombre del lenguaje a la forma actual. Experimentaciones públicas de PHP 3 comenzaron inmediatamente y fue lanzado oficialmente en junio del [1998](#).

Para [1999](#), Suraski y Gutmans reescribieron el código de PHP, produciendo lo que hoy se conoce como [Zend Engine](#) o motor Zend. También conformaron a [Zend Technologies](#) en [Ratmat Gan, Israel](#). En mayo de [2000](#) PHP 4 fue lanzado bajo el poder del motor Zend Engine 1.0. El [13 de julio](#) de [2004](#), PHP 5 fue lanzado, utilizando el

motor Zend Engine II (o Zend Engine 2). La versión más reciente de PHP es la 5.1, que incluye el novedoso PDO (Objetos de Información de PHP o PHP Data Objects) y mejoras utilizando las ventajas que provee el nuevo Zend Engine 2.

Ventajas y Desventajas de PHP

Entre las ventajas podemos citar:

- La principal ventaja se basa en ser un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad.
- Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una muy buena documentación en su página oficial .
- Es Libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de [Programación Orientada a Objetos](#). (Clases y herencia.).
- Muy sencillo de aprender.
- Similar en sintaxis a C y a PERL

- El análisis léxico para recoger las variables que se pasan en la dirección lo hace PHP de forma automática. Librándose el usuario de tener que separar las variables y sus valores.
- Se puede incrustar código PHP con etiquetas HTML.
- Excelente soporte de acceso a base de datos.
- La comprobación de que los parámetros son validos se hace en el servidor y no en el cliente (como se hace con javascript) de forma que se puede evitar que chequear que no se reciban solicitudes adulteradas. Además PHP viene equipado con un conjunto de funciones de seguridad que previenen la insercción de ordenes dentro de una solicitud de datos.
- Se puede hacer de todo lo que se pueda transmitir por vía HTTP.

Y las desventajas:

- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.
- La legibilidad del código puede verse afectada al mezclar sentencias HTML y php.
- La orientación a objetos es aún muy deficiente para aplicaciones grandes.

Por que utilizar PHP y no otras opciones:

- PHP no soporta directamente punteros, como el C, de forma que no existen los problemas de depuración provocados por estos.

- Se pueden hacer grandes cosas con pocas líneas de código. Lo que hace que merezca la pena aprenderlo.
- El código PHP es mucho más legible que el de PERL , todo el que haya programado PERL podrá corroborar esta afirmación.
- Viene acompañado por una excelente biblioteca de funciones que permite realizar cualquier labor (acceso a base de datos, encriptación, envío de correo, gestión de un e-commerce, xml, creación de PDF ...)
- Al poderse encapsular dentro de código html se puede recoger el trabajo del diseñador gráfico e incrustar el código php posteriormente.
- Esta siendo utilizado con éxito en varios millones de sitios web.
- Hay multitud de aplicaciones php para resolver problemas concretos (weblogs, tiendas virtuales , periódicos , ...) listas para usar.
- Es multiplataforma, funciona en todas las plataformas que soporten apache.
- Es software libre. Se puede obtener en la web y su código esta disponible bajo la licencia GPL.

Anteriormente hemos visto la evolución de la creación de paginas HTML dinámicas a lo largo de la historia. Inicialmente se uso programas C o shell-scripts que devolvían información en hipertexto por su salida estándar. Posteriormente se descubrió que el lenguaje interpretado PERL era excelente para la labor de interpretación necesaria para la programación dinámica. Perl sigue siendo el "pegamento de la red" , pero la aparición de soluciones más adecuadas y sencillas hacen que PHP se convierta en la mejor opción actual para multitud de necesidades. Perl, desde mi punto de vista, ha llegado a su máximo y entra en fase retroceso. ¿ Cuanto durará PHP en ser una buena opción ? Nadie lo sabe, pero actualmente es uno de los paquetes para programación de internet más utilizados.

Que se necesita.

Se precisa que todo lo siguiente este instalado y configurado correctamente. En este curso no se va a explicar eso. Si se tienen dudas de como configurar apache recurase a la documentación que lo acompaña.

1 Sistema operativo

Recomiendo encarecidamente el uso de Linux. Aunque no es imprescindible este curso se tomará como base un sistema Linux instalado y configurado.

2 Hardware

La ejecución de una web dinámica no tiene por que precisar de una máquina muy potente. Un 486 con 16 Mb de RAM y con 500Mb de disco puede ser suficiente. Por supuesto disponer de una máquina más potente permitirá una ejecución más eficiente y una programación más cómoda. Estos ejemplos se van ejecutar en la máquina local por tanto no será preciso que el ordenador este conectado a internet.

3 Servidor Web

Php es un lenguaje integrado en apache. Por tanto es categórico la necesidad de instalar apache. Además se deben instalar los módulos de soporte de php de apache y configurarlo correctamente.

4 Gestor de base de datos

Postgres , es un gestor potente y software libre. Ha aumentado su eficiencia y velocidad en sus últimas versiones (usaremos la version 7.2). Generalmente se usa Mysql que es más ligero y veloz pero este carece de integridad referencial lo que hace que resulte dudoso definirlo como relacional. Este problema hace que Mysql a dia de hoy no resulte un sistema serio para aplicaciones grandes. Pero en cambio es útil para aplicaciones sencillas que no sea fundamental la integridad referencial además de ser muy sencillo de utilizar. Los ejemplos utilizados pueden modificarse fácilmente utilizando las funciones referidas a las base de datos que se utilice.

5 Interprete PHP

Se debe instalar el interprete PHP y soporte para acceso a base de datos Postgress.

6 Cliente web

Se puede usar cualquier cliente Web que soporte HTML 3.0 o superior, como el netscape. Si no se dispone de una terminal gráfica se pueden usar clientes en modo texto como links o como lynx.

Usos de PHP

Los principales usos del PHP son los siguientes:

- Programación de [páginas web](#) dinámicas, habitualmente en combinación con el motor de base datos [MySQL](#), aunque cuenta con soporte nativo para otros motores, incluyendo el estándar [ODBC](#), lo que amplía en gran medida sus posibilidades de conexión.
- Programación en consola, al estilo de [Perl](#), en [Linux](#), [Windows](#) y [Macintosh](#).
- Creación de aplicaciones gráficas independientes del navegador, por medio de la combinación de PHP y [GTK](#) ([GIMP](#) Tool Kit), que permite desarrollar aplicaciones de escritorio tanto para los sistemas operativos basados en [Unix](#), como para [Windows](#) y [Mac OS X](#).

Ejemplo de Código PHP

A continuación un ejemplo de una página web sencilla desarrollada utilizando el lenguaje PHP:

```
<html>
<head>
  <title>Ejemplo</title>
</head>
<body>
<?php
if (isset($_POST['muestra'])) {
    echo 'Hola, '.htmlentities($_POST['nombre'])
    .', tu comida favorita es:'. htmlentities($_POST['comida']);
```

```

} else {
?>
<form method="POST" action="?">
  ¿Cuál es tu nombre?
  <input type="text" name="nombre"/>
  ¿Cuál es tu comida favorita?
  <select name="comida">
    <option>Spaghetti</option>
    <option>Asado</option>
    <option>Pizza</option>
  </select>
  <input type="submit" name="muestra" value="Seguir">
</form>
<?php
}
?>
</body>
</html>

```

En este código es posible observar las siguientes características:

- Las variables enviadas por un formulario utilizando el método POST, son recibidas en el lenguaje dentro del arreglo \$_POST, lo cual facilita la obtención de este tipo de datos. Este mismo método es utilizado por el lenguaje para todas las fuentes de información en una aplicación web, tales como [cookies](#) en el arreglo \$_COOKIES, variables de URL en \$_GET (que en formularios puede servir para guardar los datos), variables de sesión utilizando \$_SESSION, y variables del servidor y del cliente por medio del arreglo \$_SERVER.
- El código PHP está incrustado dentro del [HTML](#) e interactúa con el mismo, lo que permite diseñar la página Web en un editor común de [HTML](#) y añadir el código dinámico dentro de las etiquetas <?php ?>.
- El resultado muestra y oculta ciertas porciones del código [HTML](#) en forma condicional.
- Es posible utilizar funciones propias del lenguaje para aplicaciones Web como htmlentities(), que convierte los caracteres que tienen algún significado especial en el marcado [HTML](#) o que podrían desplegarse erróneamente en el navegador como acentos o diéresis, en sus equivalentes en formato [HTML](#).

Aplicaciones desarrolladas con PHP

- [BerriBlog](#)
- [Blasten blt-Search](#)
- [Burning Board](#)
- [CMSformE](#)
- [Copermine](#)
- [Drupal](#)
- [eZ publish](#)
- [Gallery Project](#)
- [Geeklog](#)
- [Joomla!](#)
- [Mambo Open Source MOS](#)
- [MediaWiki](#) (desarrollado para [Wikipedia](#))
- [Moodle](#)
- [Phorum](#)
- [phpBB](#)

- [phpMyAdmin](#)
- [NukeET](#)
- [PHP-Nuke](#)
- [phpPgAdmin](#)
- [PhpWiki](#)
- [PmWiki](#)
- [PostNuke](#)
- [Smarty](#)
- [SPIP](#)
- [vBulletin](#)
- [WordPress](#)
- [Xaraya](#)
- [Xoops](#)
- [MiniBB](#)

Bibliografía

- [Página oficial de PHP](#) (en inglés)
- [Manual oficial de PHP en español](#)
- [PHP-Hispano](#) Comunidad hispana de programadores en php. Scripts, Foro, y canal de [IRC](#)

Ing. Gerardo Herrera - gherrera2k1@yahoo.com -

FLISOL 2006

Gracias por su atención