

LENGUAJES DE PROGRAMACIÓN DE INTERNET

TRABAJO DE CLSI

David Biayna y Antonio Sánchez

ÍNDICE DEL TRABAJO

- INTRODUCCIÓN 3
- TIPOS DE LENGUAJE DE PROGRAMACIÓN 4-5
- LENGUAJES DE INTERNET: CLASIFICACIÓN Y CARACTERÍSTICAS 6-7
- ESTUDIO DETALLADO DE UN LENGUAJE 8-11
- EJEMPLO DE APLICACIÓN 12
- HERRAMIENTAS DE AYUDA O CÓDIGO ABIERTOS 13
- CONCLUSIÓN 14
- BIBLIOGRAFÍA 14

1. INTRODUCCIÓN

- En este trabajo, hablaremos de los lenguajes de programación. Pero nos fijaremos sobretodo en los que se utilizan en aplicaciones de Internet.
- Un lenguaje de programación es una técnica estándar de comunicación que permite expresar las instrucciones que han de ser ejecutadas en un ordenador. Un lenguaje de programación permite a un programador especificar de manera precisa: sobre qué datos un ordenador debe operar, cómo deben ser estos almacenados y transmitidos y qué acciones debe tomar. Un programa escrito en un lenguaje de programación necesita pasar por un proceso de compilación, es decir, ser traducido al lenguaje de máquina, o ser interpretado para que pueda ser ejecutado por el ordenador.
- También existen los lenguajes que se usan generalmente para la creación de contenido para sitios Web, y eso es lo que veremos en los siguientes apartados.

2.TIPOS DE LENGUAJE DE PROGRAMACIÓN

- Existen dos tipos de lenguajes claramente diferenciados; los lenguajes de bajo nivel y los de alto nivel.
El ordenador sólo entiende un lenguaje conocido como código binario o código máquina, consistente en ceros y unos. Es decir, sólo utiliza 0 y 1 para codificar cualquier acción.
Los lenguajes más próximos a la arquitectura hardware se denominan lenguajes de bajo nivel y los que se encuentran más cercanos a los programadores y usuarios se denominan lenguajes de alto nivel.

LENGUAJES DE BAJO NIVEL

- Son lenguajes totalmente dependientes de la máquina, es decir que el programa que se realiza con este tipo de lenguajes no se pueden migrar o utilizar en otras máquinas. Al estar prácticamente diseñados a medida del hardware, aprovechan al máximo las características del mismo. Dentro de este grupo se encuentran:
 - El lenguaje máquina: este lenguaje ordena a la máquina las operaciones fundamentales para su funcionamiento. Consiste en la combinación de 0's y 1's para formar las ordenes entendibles por el hardware de la maquina. Este lenguaje es mucho más rápido que los lenguajes de alto nivel. La desventaja es que son bastantes difíciles de manejar y usar, además de tener códigos fuente enormes donde encontrar un fallo es casi imposible.
 - El lenguaje ensamblador es un derivado del lenguaje maquina y esta formado por abreviaturas de letras y números llamadas nemotécnicos. Con la aparición de este lenguaje se crearon los programas traductores para poder pasar los programas escritos en lenguaje ensamblador a lenguaje máquina. Como ventaja con respecto al código máquina es que los códigos fuentes eran más cortos y los programas creados ocupaban menos memoria. Las desventajas de este lenguaje siguen siendo prácticamente las mismas que las del lenguaje ensamblador, añadiendo la dificultad de tener que aprender un nuevo lenguaje difícil de probar y mantener.

LENGUAJES DE ALTO NIVEL

- Son aquellos que se encuentran más cercanos al lenguaje natural que al lenguaje máquina.
Están dirigidos a solucionar problemas mediante el uso de EDD's (abreviaturas). Se tratan de lenguajes independientes de la arquitectura del ordenador. Por lo que, en principio, un programa escrito en un lenguaje de alto nivel, lo puedes migrar de una máquina a otra sin ningún tipo de problema.
Estos lenguajes permiten al programador olvidarse por completo del funcionamiento interno de la máquina/s para la que están diseñando el programa. Tan solo necesitan un traductor que entienda el código fuente como las características de la máquina.
Suelen usar tipos de datos para la programación y hay lenguajes de propósito general (cualquier tipo de aplicación) y de propósito específico (como FORTRAN para trabajos científicos).

LENGUAJES DE PROGRAMACIÓN DE INTERNET

- A diferencia de los otros lenguajes, estos sirven sobretodo para crear páginas Web y su contenido.

Estos son algunos ejemplos de los lenguajes de Internet que hay:

- PHP, ASP, JavaScript, etc....
- Hay más pero los mencionaremos a todos estos y más en el siguiente punto.

3. CLASIFICACIÓN Y CARACTERÍSTICAS

- **ASP(Active Server pages)**
 - Ambiente de aplicación abierto y gratuito que se puede combinar con código HTML ,scripts y componentes del Activex del servidor para crear soluciones dinámicas y poderosas para la Web.
 - Funciona al lado del servidor, lo que significa que cuando el usuario pida un documento ASP, las instrucciones de programación dentro del script son ejecutadas para enviar únicamente el código HTML restante.
- La seguridad que tiene el programador sobre su código ya que sólo se encuentra en los archivos del servidor que al ser enviado a través de la Web se ejecuta, lo cual hace que el usuario sólo tenga acceso a la página restante de su navegador

HTML (LENGUAJE PARA MARCADO DE HIPERTEXTO)

- Se trata de un lenguaje para estructurar documentos a partir de texto en World Wide Web.
- Este lenguaje se basa en tags (instrucciones que le dicen al texto como deben mostrarse) y atributos (parámetros que dan valor al tag).
- HTML tiene dos ventajas que lo hacen prácticamente imprescindibles a la hora de diseñar una presentación web: Su compatibilidad y su facilidad de aprendizaje debido al reducido número de tags que usa.
- Los documentos escritos en HTML constan del texto del documento y los tags que pueden llevar atributos. Esto llevado a la práctica, vendría a ser:
 - **<tag>** texto afectado **</tag>**

JAVASCRIPT

- Es un lenguaje interpretado que permite incluir macros en páginas Web. Estas macros se ejecutan en el ordenador del visitante de nuestras páginas, y no en el servidor.
- Controlar las ventanas del navegador y el contenido que muestran.
- Programar páginas dinámicas simples
- Evitar depender del servidor Web para cálculos sencillos
- Capturar los eventos generados por el usuario y responder a ellos sin salir a Internet.
- Simular el comportamiento de las macros *CGI* cuando no es posible usarlas
- Comprobar los datos que el usuario introduce en un formulario antes de enviarlos.
- Comunicarse con el usuario mediante diversos métodos.
- Soporta cuatro tipos de datos, no es necesario declarar el tipo de variables, argumentos de funciones ni valores de retorno de las funciones
- El tipo de las variables cambia implícitamente cuando es necesario, lo que dificulta el desarrollo de programas complejos, pero ayuda a programar con rapidez macros sencillas
- El lenguaje JavaScript se inserta en documentos **HTML**(*HyperText Markup Languaje*), de forma que su código queda reflejado en la propia página y no es llamado o cargado de ninguna fuente externa
- Sensible a las Mayúsculas

PHP (HYPERTEXT PRE PROCESSOR)

- Es un [lenguaje de programación](#) usado generalmente para la creación de contenido para sitios web
- Se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios web
- PHP permite a la mayoría de los [programadores](#) experimentados crear aplicaciones complejas con una curva de aprendizaje muy suave
- 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
- Es posible utilizar PHP para generar archivos [PDF](#), [Flash](#), así como imágenes en diferentes formatos, entre otras cosas.

4. ESTUDIO DETALLADO DE PHP

- **Historia**
- PHP es un lenguaje creado por una gran comunidad de personas. El sistema fue desarrollado originalmente en el año 1994 por Rasmus Lerdorf como un CGI escrito en C que permitía la interpretación de un número limitado de comandos. El sistema fue denominado Personal Home Page Tools y adquirió relativo éxito gracias a que otras personas pidieron a Rasmus que les permitiese utilizar sus programas en sus propias páginas. Dada la aceptación del primer PHP y de manera adicional, su creador diseñó un sistema para procesar formularios al que le atribuyó el nombre de FI (Form Interpreter) y el conjunto de estas dos herramientas, sería la primera versión compacta del lenguaje: PHP/FI.
- La siguiente gran contribución al lenguaje se realizó a mediados del 97 cuando se volvió a programar el analizador sintáctico, se incluyeron nuevas funcionalidades como el soporte a nuevos protocolos de Internet y el soporte a la gran mayoría de las bases de datos comerciales. Todas estas mejoras sentaron las bases de PHP versión 3. Actualmente PHP se encuentra en su versión 4, que utiliza el motor Zend, desarrollado con mayor meditación para cubrir las necesidades actuales y solucionar algunos inconvenientes de la anterior versión. Algunas mejoras de esta nueva versión son su rapidez -gracias a que primero se compila y luego se ejecuta, mientras que antes se ejecutaba mientras se interpretaba el código-, su mayor independencia del servidor web -creando versiones de PHP nativas para más plataformas- y un API más elaborado y con más funciones.

FUNCIONES

- Poco a poco el PHP se va convirtiendo en un lenguaje que nos permite hacer de todo. En un principio diseñado para realizar poco más que un contador y un libro de visitas, en estos momentos se pueden realizar una multitud de tareas útiles para el desarrollo del web:
- **Funciones de correo electrónico**
Podemos con una facilidad asombrosa enviar un e-mail a una persona o lista parametrizando toda una serie de aspectos tales como el e-mail de procedencia, asunto, persona a responder...
- **Gestión de bases de datos**
- El lenguaje PHP ofrece interfaces para el acceso a la mayoría de las bases de datos comerciales y por ODBC a todas las bases de datos posibles en sistemas Microsoft, a partir de las cuales podremos editar el contenido de nuestro sitio con absoluta sencillez.
- **Tratamiento de imágenes**
- Puede resultar muy tedioso uniformar en tamaño y formato miles de imágenes recibidas día tras día. Todo esto puede ser también automatizado eficazmente mediante PHP. También puede parecer útil el crear botones dinámicos, es decir, botones en los que utilizamos el mismo diseño y solo cambiamos el texto.
- **Para Internet** (tratamiento de cookies, accesos restringidos, comercio electrónico...) o para **propósito general** (funciones matemáticas, explotación de cadenas, de fechas, corrección ortográfica, compresión de archivos...) son realizadas por este lenguaje.

INSTALACIÓN

- PHP, requiere de la instalación de un servidor en nuestro PC para poder trabajar en local. Este modo de trabajo resulta más práctico que colgar los archivos por FTP en el servidor y ejecutarlos desde Internet.
- Antes comenzar a crear nuestros programas en PHP, es necesario:
 - Convertir nuestro ordenador en un servidor. Esto se hace instalando uno de los varios servidores disponibles para el sistema operativo de nuestra máquina.
 - Introducir en nuestro servidor los archivos que le permitirán la comprensión del PHP.

SINTAXIS

- PHP se escribe dentro de la propia página web, junto con el código HTML y, como para cualquier otro tipo de lenguaje incluido en un código HTML, en PHP necesitamos especificar cuáles son las partes constitutivas del código escritas en este lenguaje.
- Esto se hace, como en otros casos, delimitando nuestro código por etiquetas.
- ```
<? y ?>
<% y %>
<?php y ?>
<script lenguaje="php">
```
- Otra característica general de los scripts en PHP es la forma de separar las distintas instrucciones. Para hacerlo, hay que acabar cada instrucción con un punto y coma ";". Para la última expresión, la que va antes del cierre de etiqueta, este formalismo no es necesario.
- Los comentarios tienen una gran utilidad ya que es muy fácil olvidarse del funcionamiento de un script programado un tiempo atrás y resulta muy útil si queremos hacer rápidamente comprensible nuestro código a otra persona.
- Ejemplo de script:
- ```
<?
$mensaje="Tengo hambre!!"; //Comentario de una línea
echo $mensaje; #Este comentario también es de una línea
/*En este caso
mi comentario ocupa
varias líneas, lo ves? */
?>
```
- Si usamos doble barra (//) o el símbolo # podemos introducir comentarios de una línea. Mediante /* y */ creamos comentarios multilínea.

VARIABLES

- **Variables superglobales**

A partir de PHP 4.1.0, se dispone de un conjunto de variables de tipo array que mantienen información del sistema, llamadas superglobales porque se definen automáticamente en un ámbito global.

- **`$GLOBALS`, `$_SERVER`, `$_GET`, `$_POST`, `$_COOKIE`, `$_FILES`, `$_ENV`, `$_REQUEST`, `$_SESSION`.**

BUCLAS WHILE Y DO

- **Bucle while**

Sin duda el bucle más utilizado y el más sencillo. Lo usamos para ejecutar las instrucciones contenidas en su interior siempre y cuando la condición definida sea verdadera. La estructura sintáctica es la siguiente.

- while (condición)
{
 instruccion1;
 instruccion2;
 ...
}

- **Bucle do/while**

Este tipo de bucle no difiere en exceso del anterior. La sintaxis es la siguiente:

- do
{
 instruccion1;
 instruccion2;
 ...
}
while (condición)

- La diferencia con respecto a los bucles *while* es que este tipo de bucle evalúa la condición al final con lo que, incluso siendo falsa desde el principio, éste se ejecuta al menos una vez.

BUCLAS FOR Y FOREACH

- **Bucle for**

PHP está provisto de otros tipos de bucle que también resultan muy prácticos en determinadas situaciones

- `<?`

```
For ($size=1;$size<=6;$size++)  
{  
    echo"<font size=$size>Tamaño $size</font><br>\n";  
}  
?>
```

- **Bucle foreach**

Este bucle, implementado en las versiones de PHP4, nos ayuda a recorrer los valores de un array lo cual puede resultar muy útil por ejemplo para efectuar una lectura rápida del mismo

- `Foreach ($array as $clave=>$valor)`

```
{  
    instruccion1;  
    instruccion2;  
    ...;  
}
```

CONDICIONES

- **Las condiciones if**

Cuando queremos que el programa, llegado a un cierto punto, tome un camino concreto en determinados casos y otro diferente si las condiciones de ejecución difieren, nos servimos del conjunto de instrucciones *if*, *else* y *elseif*.

- if (condición1)

```
{
  Instrucción 1;
  Instrucción 2;
  ...
}
else
{
  if (condición2)
  {
    Instrucción A;
    Instrucción B;
    ...
  }
  else
  {
    Instrucción X
    ...
  }
}
```

5. EJEMPLO DE APLICACIÓN DE 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?
```

CONTINUACIÓN

```
<select name="comida">
  <option>Spaghetti</option>
  <option>Asado</option>
  <option>Pizza</option>
</select>
<input type="submit" name="muestra" value="Seguir">
</form>
<?php
}
?>
</body>
</html>
```

CARACTERÍSTICAS DEL CÓDIGO

- Las variables enviadas por un formulario utilizando el método POST, son recibidas en el lenguaje dentro de la matriz `$_POST`, lo cual facilita la obtención de este tipo de datos. Este mismo método es utilizado por el lenguaje para todas las fuentes de información en una aplicación web, tales como cookies en la matriz `$_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 de la matriz `$_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 `htmlspecialchars()`, que convierte los caracteres que tienen algún significado especial en el código HTML o que podrían desplegarse erróneamente en el navegador como acentos o diéresis, en sus equivalentes en formato HTML.

6.HERRAMIENTAS DE AYUDA O CÓDIGOS ABIERTOS

- Muchos blogger han optado en este ultimo tiempo en abandonar sus blog alojados en servidores como Blogger y crear uno en un servidor propio. Si quisiéramos hacer algo así, lo más probable es que recurramos a un programador para que nos diseñe el software que necesitamos para este fin, pero hace ya algún tiempo organizaciones como Wordpress o Drupal, por mencionar algunas, ofrecen desde su sitio la posibilidad de bajar gratuitamente los archivos necesarios para poder instalar un blog en casi cualquier servidor. Si bien los conocimientos necesarios para este tipo de acción no son muchos, hay que dedicarle poco más que los cinco minutos que necesitábamos con los Blogs predefinidos ya que tenemos que crear una base de datos dentro del servidor (la mayoría de los servidores ofrecen este servicio de manera muy simple), seguir los pasos del instructivo para instalar el software también dentro del servidor y una vez que lo logramos, completar los datos que nos vaya pidiendo.
- Si bien puede parecer en la primera impresión un tanto complejo para quien se está iniciando en el tema, la realidad es que quien haya adquirido un poco de práctica con Blogs predefinidos, subir a este nivel de Blogs puede ser mucho mas sencillo de lo que se cree. En nuestro caso particular, nos tomo poco más de dos horas dejar el software funcionando en el servidor. De ahí en más sólo queda hacerle las modificaciones que creamos necesarias en cuanto a la estética y a las funciones, que en nuestro caso tomo varias semanas de trabajo.

CONTINUACIÓN

- Una de las principales ventajas de este tipo de Software, es que al tener los archivos que se necesitan para crear un blog podemos, si tenemos algunos conocimientos básicos sobre lenguaje PHP, modificarlos para agregar dentro de la estructura cualquier cosa que se nos ocurra, desde banners publicitarios hasta animaciones promocionando distintos informes dentro de nuestro sitio. Además, existe una gran cantidad de usuarios desarrolladores que constantemente van creando modificadores para el sistema y ofreciéndolos gratuitamente en la Web. Si uno dedica tiempo a buscar elementos para incorporar al sitio seguramente los terminará encontrando y de esta manera sumará servicios y aplicaciones.
- En nuestro caso, sobre la base obtenida de Wordpress incorporamos en primer lugar un diseño nuevo basado en un modificador que había sido creado por un desarrollador. Agregamos los titulares de distintos sitios dentro de las columnas laterales, un calendario para poder acceder a las publicaciones de cada día y una barra en la columna lateral izquierda donde poder incorporar animaciones y banners. Sumamos también una función que muestra en el inicio de la página sólo un avance de las notas para poder mostrar así una vista previa de diversos temas y dar al visitante la opción de profundizar el tema que elija. Además, incorporamos otra función que calcula el tiempo estimado de lectura, los títulos de las últimas notas publicadas y los últimos comentarios hechos por los usuarios. El sitio, como cualquiera de los de este tipo, se encuentra así en constante evolución gracias al gran número de programadores compitiendo por ver quien crea el mejor Plug-in para Wordpress, lo que permite permanentemente agregar nuevos servicios para hacer más atractivo el sitio.

7. CONCLUSIÓN

- Al final del día es muy importante para un programador saber los tipos de lenguajes que hay, y también saber cuáles son sus preferencias. Por lo tanto hay que saber de cada lenguaje las características y los inconvenientes.

8. BIBLIOGRAFÍA

- http://www.mundostutoriales.com/tutoriales_programacion_web-mdtema261.htm
- http://directorio.adfound.com/programacion/lenguajes_de_programacion/internet/index.html
- <http://www.lenguajes-de-programacion.com/lenguajes-de-programacion.shtml>
- www.php.net
- www.desarrolloweb.com/articulos/330.php
- <http://es.wikipedia.org/wiki/PHP>
- <http://www.idefix.com.ar/degalope/index.php/2006/11/20/2422-codigos-abiertos-para-la-creacion-de-weblogs/>

FIN