

# Curso teórico practico de PHP

El siguiente texto ilustra contenidos dados en clase **basados** en documentos consultados en la red y ejercicios puntuales que se hacían para resumir cada punto a estudiar. Algunos documentos consultados son los siguientes:

Tutorial de PHP, [On-Line] [Fecha de consulta el: 25 febrero 2013] Disponible en: <http://programacionphp.yaia.com/>

Wextensible, [On-Line] [Fecha de consulta el: 25 febrero 2013] Disponible en: <http://www.wextensible.com/temas/xhtml-css/tabla.html>

Css\_Texto, [On-Line] [Fecha de consulta el: 25 febrero 2013] Disponible en: [http://www.mclibre.org/consultar/amaya/css/css\\_texto.html](http://www.mclibre.org/consultar/amaya/css/css_texto.html)

Varios ejercicios fueron dados en clase de manera no documentada que simplemente eran para dar a entender el tema que se estudiaba y que eran codificados directamente en la pizarra. Codificar en la pizarra permitía realizar modificaciones y arreglos que se hacían en conjunto con los alumnos, de esta forma, los alumnos se sentían parte de cada uno de los objetivos dados.

# Conceptos básicos

**El lenguaje PHP es un lenguaje de programación de estilo clásico, con esto quiero decir que es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones. No es un lenguaje de marcas como podría ser HTML, XML o WML.**

Está más cercano a JavaScript o a C, para aquellos que conocen estos lenguajes. El código PHP no se mezcla con las etiquetas HTML, ya que está acotado siempre por los símbolos `<? y ?>`, de forma que para los programas compositores de páginas web son etiquetas que no soporta y las deja tal y como están. De esta forma, puede trabajarse a la vez en el diseño visual de la página y en la funcionalidad (programación) de la misma cómodamente.

Pero a diferencia de Java o JavaScript que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso nos permite acceder a los recursos que tenga el servidor como por ejemplo podría ser una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una página XML.



## Instalación de APACHE o XAMPP

En este punto la instalación de apache o xampp se hace a través de guías de internet. Los problemas encontrados se resolverán en el aula de clase.

## Nuestro primer PHP

La ventaja que tiene PHP sobre otros lenguajes de programación que se ejecutan en el servidor (como podrían ser los script CGI Perl), es que nos permite intercalar las sentencias PHP en las paginas HTML, es un concepto algo complicado de entender si no se ha visto nunca como funciona unas paginas PHP o ASP.

```
<!-- Ejemplo 1 -->

<html>

<head>

    <title>Ejemplo de PHP</title>

</head>

<body>

Parte de HTML normal.

<BR><BR>

<?php

    echo "Parte de PHP<br>";

    for($i=0;$i<10;$i++)

    {

        echo "Linea ".$i."<br>";

    }

?>

</body>

</html>
```

# Variables

Una variable es un contenedor de información, en el que podemos meter números enteros, números decimales, caracteres... el contenido de las variables se puede leer y se puede cambiar durante la ejecución de una página PHP.

En PHP todas las variables comienzan con el símbolo del dólar \$ y no es necesario definir una variable antes de usarla. Tampoco tienen tipos, es decir que una misma variable puede contener un número y luego puede contener caracteres.

```
$a = 1;  
$b = 3.34;  
$c = "Hola Mundo";
```

# Arrays

Los arrays en PHP son bastante potentes y flexibles. Son variables comunes anidadas bajo un mismo nombre que contienen distintos valores en su memoria dependiendo del índice donde apuntamos.

```
$arrValores[0] = 1;  
$arrValores[1] = "Una cadena";
```

# Condicionales

Las sentencias condicionales nos permiten ejecutar o no unas ciertas instrucciones dependiendo del resultado de evaluar una condición. Las más frecuentes son la instrucción `if` y la instrucción `switch`.

## Sentencia `if ... else`

La sentencia `if` ejecuta una serie de instrucciones u otras dependiendo de la condición que le pongamos. Probablemente sea la instrucción más importante en cualquier lenguaje de programación.

```
<?php  
    if (condición)  
    {  
        Sentencias a ejecutar cuando la  
        condición es cierta.  
    }  
    else  
    {  
        Sentencias a ejecutar cuando la  
        condición es falsa.  
    }  
?>  
?>
```

## Sentencia switch ... case

Con la sentencia switch podemos ejecutar unas u otras instrucciones dependiendo del valor de una variable, en el ejemplo anterior, dependiendo del valor de la variable \$posicion se ejecuta el bloque 1 cuando el valor es "arriba", el bloque 2 cuando el valor es "abajo" y el bloque 3 si no es ninguno de los valores anteriores.

# Bucles

Los bucles nos permiten iterar conjuntos de instrucciones, es decir repetir la ejecución de un conjunto de instrucciones mientras se cumpla una condición.

## Sentencia while

```
<?php
  while (condición)
  {
    instrucciones a ejecutar.
  }
?>
```

## Sentencia for

La instrucción for es la instrucción de bucles más completa. En una sola instrucción nos permite controlar todo el funcionamiento del bucle.

El primer parámetro del for, es ejecutado la primera vez y sirve para inicializar la variable del bucle, el segundo parámetro indica la condición que se debe cumplir para que el bucle siga ejecutándose y el tercer parámetro es una instrucción que se ejecuta al final de cada iteración y sirve para modificar el valor de la variable de iteración.

```
<?php
  for (inicial ; condición ; ejecutar en iteración)
  {
    instrucciones a ejecutar.
  }
?>
```

## Sentencia printf

```
<?php
  printf(cadena formato, variable1, variable2...);
?>
```

# Manejo de cadenas

Dado el uso del lenguaje PHP el tratamiento de cadenas es muy importante, existen bastantes funciones para el manejo de cadenas, a continuación explicaremos las más usadas.

1. **strlen(cadena)**. Nos devuelve el número de caracteres de una cadena.
2. **split(separador,cadena)**. Divide una cadena en varias usando un carácter separador.
3. **sprintf(cadena de formato, var1, var2...)**. Formatea una cadena de texto al igual que printf pero el resultado es devuelto como una cadena.
4. **substr(cadena, inicio, longitud)**. Devuelve una subcadena de otra, empezando por inicio y de longitud longitud.
5. **chop(cadena)**. Elimina los saltos de línea y los espacios finales de una cadena.
6. **strpos(cadena1, cadena2)**. Busca la cadena2 dentro de cadena1 indicándonos la posición en la que se encuentra.
7. **str\_replace(cadena1, cadena2, texto)**. Reemplaza la cadena1 por la cadena2 en el texto.

# Funciones

El uso de funciones nos da la capacidad de agrupar varias instrucciones bajo un solo nombre y poder llamarlas a estas varias veces desde diferentes sitios, ahorrándonos la necesidad de escribirlas de nuevo.

```
<?php
function Nombre(parametro1, parametro2...)
{
    instrucción1;
    instrucción2;
    instrucción3;
    instrucción4;

    return valor_de_retorno;
}
?>
```

## Las Funciones require() e include()

Las funciones require() e include() son bastante similares. Con las dos podemos incluir archivos, funciones, o partes de código (su uso más común, para agregar la cabecera y pie de una página). Primero hablaremos sobre require() y luego sobre include()

### Require()

La forma de uso es <? require("archivo") ?>

Con require() se llama a un archivo específico. Digamos... estático. Este archivo puede contener como dijimos antes funciones o partes de código HTML o lo que nosotros queramos.

Cuando es llamado, todo su contenido se agrega al código del archivo que lo llamó. (Diferencia con include() que veremos luego)

### Include()

La forma de uso es igual a la anterior. Así: <? include("archivo.php") ?> La utilidad es la misma, usar procedimientos, funciones o lo que sea que esté en otro archivo... pero con la diferencia de que el solo se incluye lo que se utiliza.

Pensemos en el mismo ejemplo de antes. Tenemos por un lado el archivo saludos.php, y por otro nuestro archivo que usamos para llamarlo. Pero supongamos que es el archivo de uso, sólo llamamos a la variable \$despedida. Solo esa variable será agregada.

Esto es posible, ya que la llamada al archivo con include() se realiza en tiempo de ejecución. De este modo solo se le pide lo que usamos, y no todo su contenido.

Otra cosa diferente es que `include()` se puede usar dinámicamente. Tal vez dependiendo de la elección de alguna cosa por el usuario, podríamos usar una variable para definir nuestro `include()`

# Usando las fechas y horas en PHP

En este capítulo veremos como usar, mostrar y utilizar las fechas desde PHP.

La función `Date()` es la que usaremos para mostrar las fechas u horas.

Antes de ver un ejemplo tengo que presentarles una tabla, que contiene los códigos de formato de fecha...

Código	Display
a	am o pm
A	AM o PM
h	Hora (de 1 a 12)
H	Hora (de 0 a 23)
i	Minutos
s	Segundos
j	Dia (sin los ceros)
d	Dia (con los ceros)
D	Abreviacion del dia (en ingles)
l	Nombre del dia (en ingles)
z	Numero de dia del año (desde 1 hasta 365)
m	Numero del Mes
M	Abreviacion del mes (en ingles)
F	Nombre del mes (en ingles)
y	Año (formato de 2 digitos)
Y	Año (formato de 4 digitos)

Para mostrar cualquiera de los datos anteriores usamos la función **Date()** de esta forma:

**Date("codigo")**

```
<?php
echo Date("d/m/y");
?>
```



## Acceso a variables globales

Este es uno de los puntos en los que PHP se diferencia de C y es un punto importante y causa de bastantes quebraderos de cabeza para el primerizo en PHP.

Para acceder a una variable global desde dentro de una función es imprescindible declararla dentro de la función como global \$variable, ya que de otro modo PHP pensará que se desea hacer referencia a una variable local:

# Algunos ejercicios prácticos

```
<!-- Ejemplo 1 -->
```

```
<html>
```

```
<head>
```

```
  <title>Ejemplo de PHP</title>
```

```
</head>
```

```
<body>
```

```
<H1>Ejemplo de burbuja</H1>
```

```
<?php
```

```
    $a=array(22,5,18,112,12,186,23,16,123,76,9);
```

```
    $n=sizeof($a);
```

```
    echo "<h3>Ascendente</h3>";
```

```
    echo "Recorremos un bucle para mostrar el array antes de ser  
ordenado<br/>";
```

```
    for($i=0;$i<$n;$i++){
```

```
        echo "a[".$i."] = ".$a[$i].";
```

```
    }
```

```
    echo "<br/><br/>";
```

```
    for($i=1;$i<$n;$i++){
```

```
        for($j=0;$j<($n-$i);$j++){
```

```
            if(($a[$j])>($a[$j+1])){
```

```
                $aux=$a[$j];
```

```
                $a[$j]=$a[$j+1];
```

```
                $a[$j+1]=$aux;
```

```
    }  
  }  
}
```

```
echo "Recorremos un bucle para mostrar el array una vez ordenado<br/>";
```

```
for($i=0;$i<$n;$i++){  
    echo "\$a[".$i."] = ".$a[$i]."<br/>";  
}
```

```
//ORDENAMIENTO DE MAYOR A MENOR
```

```
echo "<h3>Descendente</h3>";
```

```
echo "<br >";
```

```
$b=array(22,5,18,112,12,186,23,16,123,76,9);
```

```
for($i=1;$i<$n;$i++){  
    for($j=0;$j<($n-$i);$j++){  
        if(($b[$j])<($b[$j+1])){  
            $aux=$b[$j];  
            $b[$j]=$b[$j+1];  
            $b[$j+1]=$aux;  
        }  
    }  
}
```

```
echo "Recorremos un bucle para mostrar el array una vez ordenado<br/>";
```

```
for($i=0;$i<$n;$i++){  
    echo "a[".$i."] = ".$b[$i]."<br/>";  
}
```

```
?>
```

```
</body>
```

```
</html>
```

```
<!-- Ejemplo -->
<html>
<head>
  <title>Ejemplo de PHP</title>
<?php
  function TraductorMes($mes){
    $mes_letras = "";
    switch($mes){
      case 1:
        $mes_letras="enero";
        break;
      case 2:
        $mes_letras="febrero";
        break;
      case 3:
        $mes_letras="marzo";
        break;
      case 4:
        $mes_letras="abril";
        break;
      case 5:
        $mes_letras="mayo";
        break;
      case 6:
        $mes_letras="junio";
        break;
      case 7:
        $mes_letras="julio";
        break;
      case 8:
        $mes_letras="agosto";
        break;
      case 9:
        $mes_letras="septiembre";
        break;
      case 10:
        $mes_letras="octubre";
        break;
      case 11:
        $mes_letras="noviembre";
        break;
      case 12:
        $mes_letras="Diciembre";
        break;
      default:
        $mes_letras="Error";
        break;
    }
  }
  return $mes_letras;
```

```

    }
?>
</head>
<body>
<H1>Funcion de mes</H1>

<form method="get">
  <label>Mes: </label>
  <input type="text" name="lol">
  <input type="submit" value="Enviar" name="Enviar">
</form>
<?php
  $mescompleto=TraductorMes($_GET['lol']);
  echo $mescompleto."<br>";

  echo "El mes actual es ".TraductorMes(Date("m"));
?>
</body>
</html>

```

```

<!-- Formulario 1 -->
<html>
<head>
  <title>Ejemplo de PHP</title>
</head>
<body>
<H1>Ejemplo de procesado de mayor de edad y entrada al sistema</H1>
<p>Los campos con (*) son requeridos.</p>
  <form action="post.php" method="get">
    <!-- Este es el nombre-->
    <label>Nombres: (*) </label>
    <input type="text" name="nombre">
    <!-- Este es el apellido-->
    <label><br >Apellidos: (*) </label>
    <input type="text" name="apellido">
    <!-- Este es el cedula-->
    <label><br >Cedula: (*) </label>
    <input type="text" name="cedula">
    <!-- Este es el Email-->
    <label><br >Email: (*) </label>
    <input type="text" name="email">
    <!-- Este es el Asunto-->
    <label><br >Asunto de visita: (*) </label><br />
    <textarea name="cuestion" rows="5" cols="50"></textarea><br />
    <!-- Este es el Sexo-->
    <label>Seleccione su sexo </label>
    <select name="sexo">

```

```
        <option>Femenino</option>
        <option>Masculino</option>
        <option selected="selected">-</option>
    </select>

    <label><br >Ciudad de nacimiento: (*) </label>
    <input type="text" name="ciudad"><br />

    <label><br >Año de nacimiento: (*) </label>
    <input type="text" name="ano"><br />

    <input type="submit" value="Enviar" name="Enviar">
</form>
```

```
</body>
</html>
```

```
<!-- Ejemplo2.php -->
<html>
<head>
<title>Ejemplo de PHP</title>
</head>
<body>
<?php
$a = 1;
$b = 3.34;
$c = "Hola Mundo";
echo $a, "<br>", $b, "<br>", $c;
?>
</body>
</html>
```

```
<!-- Ejemplo3.php -->
<HTML>
<BODY>
<FORM ACTION="tratar_form.php">
<INPUT TYPE="TEXT" NAME="Nombre">
<INPUT TYPE="TEXT" NAME="Apellido[0]">
<INPUT TYPE="TEXT" NAME="Apellido[1]">
<INPUT TYPE="SUBMIT" NAME="btnAceptar" VALUE="Aceptar">
</FORM> </BODY> </HTML>
```

# A continuación el aporte dado por los instructores del curso de PHP:

<Curso teórico practico de PHP... El documento a continuación sujeto a: >  
Copyright (C) 2013 Ricardo Xavier Graterol y Jesús Manuel Salcedo

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

**Ejercicio 1** donde debemos crear un array y recorrerlo para obtener su contenido.

```
<html>
<head>
  <title>Ejemplo de PHP</title>
</head>
<body>
<H1>Examen de PHP</H1>
<?php
  $personas[0]=array("nombre"=>"Maduro","cedula"=>9145156);
  $personas[1]=array("nombre"=>"Elias","cedula"=>9268156);
  $personas[2]=array("nombre"=>"Jaua","cedula"=>3245156);
  $personas[3]=array("nombre"=>"Iris","cedula"=>6745156);
  $personas[4]=array("nombre"=>"Tibisay","cedula"=>9145156);
  $personas[5]=array("nombre"=>"Mario","cedula"=>9149845);
  $personas[6]=array("nombre"=>"Tarek","cedula"=>2345156);
  $personas[7]=array("nombre"=>"Diosdado","cedula"=>6777553);
  $personas[8]=array("nombre"=>"Hector","cedula"=>913556);
  echo "<h3>Administradores poco eficientes del foro</h3>";
  $n = sizeof($personas);
  echo "<font color=\"red\">";
  for($i=0;$i<$n;$i++){
    echo "Nombre del administrador <b>".$personas[$i]["nombre"];
    echo "</b> y la cedula del administrador es ".$personas[$i]["cedula"]."<br>";
  }
  echo "</font>";
  echo "<br/><br/>";
  $paises=array(3 ,6 , 1 , 1233 , 2 , 0 );
  sort($paises);
  while (list($key,$val)=each($paises)) {
  echo " Elemento $key igual a $val<br> ";
  }
  ?>
</body>
</html>
```

**De resultado se tiene lo siguiente: (las comillas delimitan el resultado)**

**“Examen de PHP**

### **Administardores poco eficientes del foro**

Nombre del administrador **Maduro** y la cedula del administardor es 9145156.

Nombre del administrador **Elias** y la cedula del administardor es 9268156.

Nombre del administrador **Jaua** y la cedula del administardor es 3245156.

Nombre del administrador **Iris** y la cedula del administardor es 6745156.

Nombre del administrador **Tibisay** y la cedula del administardor es 9145156.

Nombre del administrador **Mario** y la cedula del administardor es 9149845.

Nombre del administrador **Tarek** y la cedula del administardor es 2345156.

Nombre del administrador **Diosdado** y la cedula del administardor es 6777553.

Nombre del administrador **Hector** y la cedula del administardor es 913556.

Elemento 0 igual a 0

Elemento 1 igual a 1

Elemento 2 igual a 2

Elemento 3 igual a 3

Elemento 4 igual a 6

Elemento 5 igual a 1233?”



**Ejercicio 2** donde debemos utilizar el método de la burbuja para realizar operaciones de ordenado.

```
<html>
<head>
  <title>Ejemplo de PHP</title>
</head>
<body>
<H1>Ejemplo de burbuja</H1>
<?php

  $a=array(22,5,18,112,12,186,23,16,123,76,9);

  $n=sizeof($a);
  echo "<h3>Ascendente</h3>";
  echo "Recorremos un bucle para mostrar el array antes de ser ordenado<br/>";
  for($i=0;$i<$n;$i++){
    echo "a[".$i."] = ".$a[$i]." ";
  }
  echo "<br/><br/>";
  for($i=1;$i<$n;$i++){
    for($j=0;$j<($n-$i);$j++){
      if(($a[$j])>($a[$j+1])){
        $aux=$a[$j];
        $a[$j]=$a[$j+1];
        $a[$j+1]=$aux;
      }
    }
  }
  echo "Recorremos un bucle para mostrar el array una vez ordenado<br/>";
  for($i=0;$i<$n;$i++){
    echo "\$a[".$i."] = ".$a[$i]."<br/>";
  }
  //ORDENAMIENTO DE MAYOR A MENOR
  echo "<h3>Descendente</h3>";
  echo "<br >";
  $b=array(22,5,18,112,12,186,23,16,123,76,9);
  for($i=1;$i<$n;$i++){
    for($j=0;$j<($n-$i);$j++){
      if(($b[$j])<($b[$j+1])){
        $aux=$b[$j];
        $b[$j]=$b[$j+1];
        $b[$j+1]=$aux;
      }
    }
  }
}
```

```

    }
}

echo "Recorremos un bucle para mostrar el array una vez ordenado<br/>";
for($i=0;$i<$n;$i++){
    echo "a[".$i."] = ".$b[$i]."<br/>";
}
?>
</body>
</html>

```

**De resultado se tiene lo siguiente: (las comillas delimitan el resultado)**

### **“Ejemplo de burbuja Ascendente**

Recorremos un bucle para mostrar el array antes de ser ordenado

a[0] = 22 a[1] = 5 a[2] = 18 a[3] = 112 a[4] = 12 a[5] = 186 a[6] = 23 a[7] = 16 a[8] = 123 a[9] = 76  
a[10] = 9

Recorremos un bucle para mostrar el array una vez ordenado

\$a[0] = 5  
\$a[1] = 9  
\$a[2] = 12  
\$a[3] = 16  
\$a[4] = 18  
\$a[5] = 22  
\$a[6] = 23  
\$a[7] = 76  
\$a[8] = 112  
\$a[9] = 123  
\$a[10] = 186

### **Descendente**

Recorremos un bucle para mostrar el array una vez ordenado

a[0] = 186  
a[1] = 123  
a[2] = 112  
a[3] = 76  
a[4] = 23  
a[5] = 22  
a[6] = 18  
a[7] = 16  
a[8] = 12  
a[9] = 9  
a[10] = 5”

**Ejercicio 3** Crearemos una función que busque en los meses del año y al introducir un numero entre 0 y 12 la función muestra que numero corresponde con que mes.

```
<html>
<head>
  <title>Ejemplo de PHP</title>
<?php
  function TraductorMes($mes){
    $mes_letras = "";
    switch($mes){
      case 1:
        $mes_letras="enero";
        break;
      case 2:
        $mes_letras="febrero";
        break;
      case 3:
        $mes_letras="marzo";
        break;
      case 4:
        $mes_letras="abril";
        break;
      case 5:
        $mes_letras="mayo";
        break;
      case 6:
        $mes_letras="junio";
        break;
      case 7:
        $mes_letras="julio";
        break;
      case 8:
        $mes_letras="agosto";
        break;
      case 9:
        $mes_letras="septiembre";
        break;
      case 10:
        $mes_letras="octubre";
        break;
      case 11:
        $mes_letras="noviembre";
        break;
```

```

        case 12:
            $mes_letras="Diciembre";
            break;
        default:
            $mes_letras="Error";
            break;
    }
    return $mes_letras;
}
?>
</head>
<body>
<H1>Funcion de mes</H1>

<form method="get">
    <label>Mes: </label>
    <input type="text" name="lol">
    <input type="submit" value="Enviar" name="Enviar">
</form>
<?php
    $mescompleto=TraductorMes($_GET['lol']);
    echo $mescompleto."<br>";

    echo "El mes actual es ".TraductorMes(Date("m"));
?>
</body>
</html>

```

**De resultado se tiene lo siguiente: (las comillas delimitan el resultado)**

Al introducir el valor de 7 obtenemos lo siguiente“

## Funcion de mes

Mes:

julio  
El mes actual es enero

**Ejercicio 4** En este ejemplo se creará un formulario que no redireccione a ninguna otra pagina pero que se estructure de forma tal que prepare a los lay

```
<html>
<head>
  <title>Ejemplo de PHP</title>
</head>
<body>
<H1>Ejemplo de procesado de mayor de edad y entrada al sistema</H1>
<p>Los campos con (*) son requeridos.</p>
  <form action="post.php" method="get">
    <!-- Este es el nombre-->
    <label>Nombres: (*) </label>
    <input type="text" name="nombre">
    <!-- Este es el apellido-->
    <label><br >Apellidos: (*) </label>
    <input type="text" name="apellido">
    <!-- Este es el cedula-->
    <label><br >Cedula: (*) </label>
    <input type="text" name="cedula">
    <!-- Este es el Email-->
    <label><br >Email: (*) </label>
    <input type="text" name="email">
    <!-- Este es el Asunto-->
    <label><br >Asunto de visita: (*) </label><br />
    <textarea name="cuestion" rows="5" cols="50"></textarea><br />
    <!-- Este es el Sexo-->
    <label>Seleccione su sexo </label>
    <select name="sexo">
      <option>Femenino</option>
      <option>Masculino</option>
      <option selected="selected"></option>
    </select>

    <label><br >Ciudad de nacimiento: (*) </label>
    <input type="text" name="ciudad"><br />

    <label><br >Año de nacimiento: (*) </label>
    <input type="text" name="ano"><br />

    <input type="submit" value="Enviar" name="Enviar">
  </form>
```

```
</body>
</html>
```

**De resultado se tiene lo siguiente: (las comillas delimitan el resultado)**

“

## Ejemplo de procesamiento de mayor de edad y entrada al sistema

Los campos con (\*) son requeridos.

Nombres: (\*)

Apellidos: (\*)

Cedula: (\*)

Email: (\*)

Asunto de visita: (\*)

Seleccione su sexo

Ciudad de nacimiento: (\*)

Año de nacimiento: (\*)

“

**Ejercicio 5** En este ejemplo se usa el ejercicio anterior y se recibe los datos para ser procesado y mostrar en otra pagina los datos introducidos en el formulario. A continuación se muestra el código del archivo que recibe los datos del formulario.

```
<html>
<head>
  <title>Ejemplo de PHP</title>
</head>
<body>
<H1>Ejemplo de procesamiento de mayor de edad y entrada al sistema</H1>
<p>Los campos con (*) son requeridos.</p>
  <form action="post.php" method="get">
    <!-- Este es el nombre-->
    <label>Nombres: (*) </label>
    <input type="text" name="nombre">
    <!-- Este es el apellido-->
    <label><br >Apellidos: (*) </label>
    <input type="text" name="apellido">
    <!-- Este es el cedula-->
    <label><br >Cedula: (*) </label>
    <input type="text" name="cedula">
    <!-- Este es el Email-->
    <label><br >Email: (*) </label>
```

```
<input type="text" name="email">
<!-- Este es el Asunto-->
<label><br >Asunto de visita: (*) </label><br />
<textarea name="cuestion" rows="5" cols="50"></textarea><br />
<!-- Este es el Sexo-->
<label>Seleccione su sexo </label>
<select name="sexo">
  <option>Femenino</option>
  <option>Masculino</option>
  <option selected="selected"></option>
</select>
<label><br >Ciudad de nacimiento: (*) </label>
<input type="text" name="ciudad"><br />
<label><br >Año de nacimiento: (*) </label>
<input type="text" name="año"><br />
<input type="submit" value="Enviar" name="Enviar">
</form>
</body>
</html>
```

### **Al introducir lo siguiente:**

nombre: jose

apellido: perez

cedula: 5555555555

email: [jbadgffg@ameil.com](mailto:jbadgffg@ameil.com)

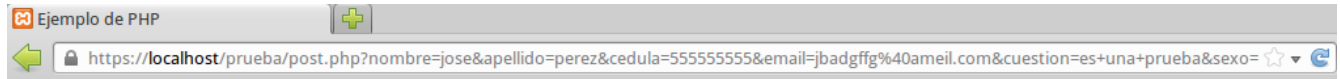
Asunto de visita: es una prueba

Sexo: masculino

Ciudad de nacimiento: merida

Año de nacimiento: 1996

### **El resultado que se tiene es el siguiente:**



### **Estos datos son los que se agregaran a la base de datos**

Su nombre es jose y su apellido es perez

Su cedula es 5555555555

La cuestion que quiere tratar es es una prueba

Su ciudad es mérida

Sexo: Masculino

### **Atencion, no eres mayor de edad**