



Sede: _____ Jornada: _____ Curso: _____ Asignatura: _____ Nombre del estudiante: _____

GUÍA 2: SOFTWARE

CURSO	DOCENTE	CORREO
1104	JUAN CARLOS TORRES ARDILA	jucatoceid@gmail.com
Meta de Comprensión:	El estudiante comprende los conceptos básicos de las bases de datos en la modelamiento de software para el diseño de aplicaciones web	

FUNDAMENTACIÓN TEÓRICA

VARIABLES EN PHP

En PHP las variables se representan con un signo de dólar seguido por el nombre de la variable. El nombre de la variable es sensible a minúsculas y mayúsculas.

Los nombres de variables siguen las mismas reglas que otras etiquetas en PHP. Un nombre de variable válido tiene que empezar con una letra o un carácter de subrayado (underscore), seguido de cualquier número de letras, números y caracteres de subrayado. Como expresión regular se podría expresar como: '[a-zA-Z_\x7f-\xff][a-zA-Z0-9_\x7f-\xff]*'

```

<?php
$var = 'Roberto';
$Var = 'Juan';
echo "$var, $Var";      // imprime "Roberto, Juan"

$4site = 'aun no';     // inválido; comienza con un número
$_4site = 'aun no';   // válido; comienza con un carácter de subrayado
$täyte = 'mansikka';  // válido; 'ä' es ASCII (Extendido) 228
?>

```

De forma predeterminada, las variables siempre se asignan por valor. Esto significa que cuando se asigna una expresión a una variable, el valor completo de la expresión original se copia en la variable de destino. Esto quiere decir que, por ejemplo, después de asignar el valor de una variable a otra, los cambios que se efectúen a una de esas variables no afectarán a la otra. Para más información sobre este tipo de asignación, vea Expresiones.

PHP también ofrece otra forma de asignar valores a las variables: asignar por referencia. Esto significa que la nueva variable simplemente referencia (en otras palabras, "se convierte en un alias de" ó "apunta a") la variable original. Los cambios a la nueva variable afectan a la original, y viceversa.

TIPOS DE VARIABLES, CARACTERÍSTICAS GENERALES Y ASPECTOS ESPECÍFICOS DE PHP DE ESTOS ELEMENTOS BÁSICOS DE LA PROGRAMACIÓN.

El **concepto de variable**. No obstante podemos entender una variable como un dato almacenado en una referencia. Técnicamente una variable apunta a una posición de la memoria, donde se almacena un dato. Las variables se utilizan en los lenguajes de programación para darle un nombre a ese dato, a esa



Sede: _____ Jornada: _____ Curso: _____ Asignatura: _____ Nombre del estudiante: _____

posición de la memoria, de manera que se pueda entender o saber lo que contiene. Al final, esos datos almacenados son los que se utilizan para conseguir los resultados de los programas.

Por su parte, un tipo de datos es la característica de un dato almacenado. Es decir, si es de una forma concreta, numérica, alfanumérica, etc. Todos los lenguajes tipan de alguna manera los datos, aunque algunos son más permisivos que otros a la hora de realizar operaciones con variables de distintos tipos.

\$ en el nombre de las variables

Para PHP, las variables eran definidas comenzando siempre por el símbolo dólar (\$). Es quizás una de las características más fuertes del lenguaje. Al ver el dólar al principio del nombre de cualquier variable podrás rápidamente deducir que tal código está escrito en PHP.

Las variables siempre deberían tener un nombre descriptivo sobre lo que ellas van a almacenar. Por tanto, al nombre de una variable en PHP le colocaremos el símbolo \$.

`<?php $total = 300 ?>`

Tipos de datos en PHP

Dependiendo de la información que contenga, una variable puede ser considerada de uno u otro tipo:

Variables numéricas: Este tipo de variables almacena cifras, números, que pueden tener dos clasificaciones distintas:

- Enteros \$entero=2002; Números sin decimales
- Reales \$real=3.14159; Números con o sin decimal

Variables alfanuméricas: Este tipo de datos almacena textos compuestos, cadenas de caracteres, que pueden contener letras, símbolos y números o cifras.

- Cadenas Almacenan variables alfanuméricas \$cadena="Hola amigo";

Boleanas: Este tipo de variables almacena un valor lógico, que puede valer verdadero o falso. Es muy común en la programación este tipo de variables boleanas.

- Boleano verdadero \$verdadero = true;
- Boleano falso \$falso = false;

Matrices, tablas o arrays: Es un tipo de datos en el que, en lugar de tener un dato, podemos almacenar un conjunto de ellos, a los que accedemos a través de índices. Cada una de las casillas de un array o los datos de nuestra matriz a su vez almacenan informaciones numéricas y/o alfanuméricas, pudiendo mezclar tipos de variables de manera arbitraria entre sus distintas casillas.

- Arrays Son las variables que guardan las tablas
\$sentido[1]="ver";
\$sentido[2]="tocar";
\$sentido[3]="oir";



Sede: _____ Jornada: _____ Curso: _____ Asignatura: _____ Nombre del estudiante: _____

```
$sentido[4]="gusto";  
$sentido[5]="oler";
```

Objetos: Se trata de conjuntos de variables y funciones asociadas. Presentan una complejidad mayor que las variables vistas hasta ahora pero su utilidad es más que interesante. Entraremos con detalle en los objetos más adelante, ya que su complejidad hace difícil explicarlas ahora.

PHP tiene tipado dinámico: A diferencia de otros lenguajes, PHP posee una gran flexibilidad a la hora de operar con variables. En efecto, cuando definimos una variable asignándole un valor, el ordenador le atribuye un tipo. Si por ejemplo definimos una variable entre comillas, la variable será considerada de tipo cadena:

```
$variable="5"; //esto es una cadena
```

Sin embargo, si pedimos en nuestro script realizar una operación matemática con esta variable, no obtendremos un mensaje de error sino que la variable cadena será asimilada a numérica (PHP hará todo lo posible por interpretar nuestra operación, aunque técnicamente no tenga mucho sentido hacer determinadas operaciones):

```
<?
```

```
$cadena="5"; //esto es una cadena
```

```
$entero=3; //esto es un entero
```

```
echo $cadena+$entero
```

```
?>
```

Este script dará como resultado "8". La variable cadena ha sido asimilada en entero (aunque su tipo sigue siendo cadena) para poder realizar la operación matemática. Del mismo modo, podemos operar entre variables tipo entero y real. No debemos preocuparnos de nada, PHP se encarga durante la ejecución de interpretar el tipo de variable necesario para el buen funcionamiento del programa.

PHP es sensible a las mayúsculas y minúsculas

PHP entiende de manera distinta las mayúsculas y minúsculas. En el caso del nombre que le damos a una variable, no es lo mismo escribirla con mayúscula o minúscula, o mezclando mayúsculas y minúsculas de distinta manera. Por tanto, hay que tener mucho cuidado a la hora de escribir los nombres de variables, y no cambiar mayúsculas por minúsculas, ya que PHP entenderá dos variables distintas aunque nosotros podamos intentar referirnos a la misma. Cuando estamos empezando quizás sea un buen consejo trabajar asignando nombres a las variables siempre en minúsculas, para evitar este tipo de malentendidos a veces muy difíciles de localizar.

En el caso que tengamos una variable con un nombre compuesto de varias palabras, en PHP es una práctica común colocar la variable toda en minúscula y separar las palabras por guiones bajos.

```
<?php $mi_variable_bonita = "me gusta PHP" ?>
```



Sede: _____ Jornada: _____ Curso: _____ Asignatura: _____ Nombre del estudiante: _____

Variables asignadas por referencia

En PHP también podemos asignar variables por referencia, aunque a decir verdad no es una característica que se use mucho. En ese caso no se les asigna un valor, sino otra variable, de tal modo que las dos variables comparten espacio en memoria para el mismo dato.

La notación para asignar por referencia es colocar un "&" antes del nombre de la variable.

<?php

```
$foo = 'Bob'; // Asigna el valor 'Bob' a $foo
```

```
$bar = &$foo; // Referencia $foo vía $bar.
```

```
$bar = "Mi nombre es $bar"; // Modifica $bar...
```

```
echo $foo; // $foo también se modifica.
```

```
echo $bar;
```

?>

Esto dará como resultado la visualización dos veces del string "Mi nombre es Bob". Algo como:

Mi nombre es Bob Mi nombre es Bob

VARIABLES DE CONEXIÓN DE LA BASE DE DATOS DEL SERVIDOR

Este es un proceso muy fácil pero de cuidado para conectar cualquier documento de php a la base de datos, recordemos el código para tal fin

```
<?php
    // Ejemplo de conexión a base de datos MySQL con PHP.
    //
    // Ejemplo realizado por Oscar Abad Folgueira: http://www.oscarabadfolgueira.com y
https://www.dinapyme.com

    // Datos de la base de datos
    $usuario = "root";
    $password = "";
    $servidor = "localhost";
    $basededatos = "alumnos";

    // creación de la conexión a la base de datos con mysql_connect()
    $conexion = mysql_connect( $servidor, $usuario, "" ) or die ("No se ha podido conectar
al servidor de Base de datos");

    // Selección del a base de datos a utilizar
    $db = mysqli_select_db( $conexion, $basededatos ) or die (no se ha podido conectar a la
base de datos" );
```



Sede: _____ Jornada: _____ Curso: _____ Asignatura: _____ Nombre del estudiante: _____

// Establecer y realizar cualquier sentencia a la base de datos por ejemplo select.
guardamos en variable.

```
$consulta = "SELECT * FROM alumnos";
```

```
$resultado = mysqli_query( $conexion, $consulta ) or die ( "Algo ha ido mal en la consulta a la base de datos");
```

?>

APROPIACIÓN CONCEPTUAL

¿Variables en php?

¿Cuáles son los tipos de Variables en php?

PRODUCTO FINAL

Formulario de enlace software Estudiantes

Nota: Este es un formulario de ejemplo en el que los datos aquí escritos se trasladan a otra página.

Nombre: . contraseña: e-mail:

Para la Guía No 2 elabore un formulario de consulta y modificación de datos (total 2) en HTML acompañado del archivo de PHP requerido de acuerdo a la sentencia además debe realizar la operación en su respectiva tabla en Mysql que tenga campos de nombres, email, contraseña, sexo, nivel de estudios, temas de interés (música, deportes, cine, libros, ciencias), una opinión, para aplicar las sentencias de Mysql utilizando el método post.

AJUSTES RAZONABLES

Para los estudiantes de proceso de Inclusión se ajustara su presentación de su trabajo de acuerdo a las siguientes condiciones

Para la Guía No 2 elabore un formulario de consulta y modificación de datos (total 1) en HTML acompañado del archivo de PHP (diseño acompañado por el docente) requerido de acuerdo a la sentencia además debe realizar la operación en su respectiva tabla en Mysql que tenga campos de nombres, email, contraseña, sexo, nivel de estudios, temas de interés (música, deportes, cine, libros, ciencias), una opinión, para aplicar las sentencias de Mysql utilizando el método post.