



PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

FORMATO GUÍA DE APRENDIZAJE

1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: Tecnólogo en Análisis y Desarrollo de Software
- Código del Programa de Formación: 2560979
- Nombre del Proyecto (si es formación Titulada): Construcción de Software Orientado a Diferentes Sectores Productivos del Área Metropolitana de Cúcuta, Norte de Santander
- Fase del Proyecto (si es formación Titulada): CONSTRUCCIÓN
- Actividad de Proyecto(si es formación Titulada)
- Competencia: 220501096 - DESARROLLAR LA SOLUCIÓN DE SOFTWARE DE ACUERDO CON EL DISEÑO Y METODOLOGÍAS DE DESARROLLO.
- Resultados de Aprendizaje Alcanzar:
CREAR COMPONENTES FRONT-END DEL SOFTWARE DE ACUERDO CON EL DISEÑO.
- Duración de la Guía

2. PRESENTACIÓN

- Esta guía de aprendizaje se presenta: Creación de una base de datos.

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

- GENERAR PLANTILLAS Y ESTILOS.
- CREAR INTERFACES GRÁFICAS DE USUARIO EN APLICACIONES DE ESCRITORIO, WEB Y MÓVILES.

TEMAS A DESARROLLAR

3.1. INTRODUCCIÓN

Una de las funciones principales de PHP es conectarnos con alguna base de datos en el servidor. Las operaciones básicas se hacen empleando como lenguaje el SQL. El más empleado en la actualidad en la web es el gestor de base de datos MySQL.



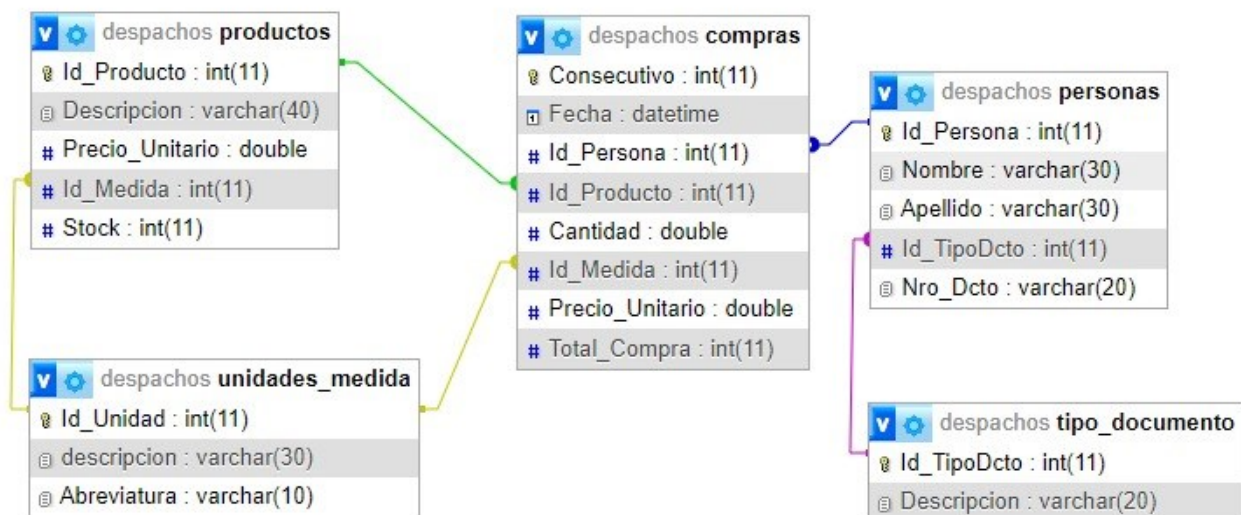
Para crear una base de datos el XAMPP instala también un programa codificado en PHP que nos permite interactuar con el MySQL. Este programa se llama PHPMyAdmin (como hemos visto nos permite crear las bases de datos, tablas, índices, usuarios, etc.).

Conexión a una Base de datos MySQL con PHP

Se presentan 3 formas para conectarse a MySQL desde PHP, junto con sus ejemplos. Para aprovechar al máximo a MySQL, es muy importante entender cómo conectarse desde un programa php a la base de datos MySQL.

- Conexión usando PHP MySQL PDO (recomendada)
- Desde PHP usando la extensión "mysqli" (recomendada)
- Conéctese en la forma antigua MySQL (No recomendada)

Para poder realizar estas conexiones debemos de tener un servidor WEB y debe de contar con PHP y MySQL. En los ejemplos utilizaremos la base de datos Despachos:



3.2. CONEXIÓN A MYSQL DESDE PHP USANDO PHP MySQL PDO (Recomendado)

La extensión **Objetos de Datos de PHP** ([PDO por sus siglās en inglés](#)) define una interfaz ligera para poder acceder a bases de datos en PHP. **PDO** son las siglas de **PHP Data Object** y esta orientado exclusivamente al paradigma de la programación orientada a objetos. Una de las grandes ventajas es que admite diferentes tipos de bases de datos que se usan en PHP, como MySQL, MariaDB, PostgreSQL, MSSQL, Informix etc.

Uno de los puntos más importantes que soporta PDO son las **declaraciones preparadas**. Utilizando estas declaraciones, se aceleran las consultas de MySQL y se previenen ataques de inyección de SQL.

Cuando creamos una conexión a una base de datos MySQL utilizando PDO, necesitamos crear un nuevo objeto PDO con un: (DSN – Data Source Name), nombre de usuario y contraseña.



Deberemos definir un DSN que es el tipo de base de datos, un \$host (nombre o ip del servidor), un \$dbname (nombre de la base de datos), un \$usuario (nombre de usuario) y un \$password (contraseña). Estas variables las declaramos antes de realizar la conexión.

En este ejemplo, utilizaremos un try/catch que básicamente significa que ejecutará lo que hay dentro del “try” y si hay un error ejecutará la parte del “catch”. Si la conexión es correcta, se mostrará el mensaje “Conexión satisfactoria”. Si la conexión resulta fallida, se mostrará un mensaje de error y se detendrá la ejecución.

Crear el siguiente archivo .php en la carpeta que esté realizando la conexión con el servidor:

```
<?php
/* Conectar a una base de datos invocando al controlador */
$hostname = 'mysql:dbname=Despachos;host=localhost';
$usuario = 'root';
$clave = '';

try {
    $conn = new PDO($hostname, $usuario, $clave);

    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    echo 'Se ha conectado a la Base de Datos.<br>';

    $sql = 'SELECT * FROM Unidades_Medida';

    print "<br>Unidades de Medida:<br><br>";
    foreach ($conn->query($sql) as $row) {
        // Imprime datos de MySQL
        Print                                     "<b>". $row['Descripcion']. "</b>"
        "<b>". $row['Abreviatura']. "</b><br>";
    }
    $conn = null;
}
catch(PDOException $err) {
    // Imprime error de conexión
    echo "ERROR: No se pudo conectar a la base de datos: " . $err->getMessage();
}
?>
```

3.3. CONEXIÓN A MYSQL DESDE PHP USANDO LA EXTENSIÓN “mysqli” (Recomendada)

MySQLi es una extensión de MySQL que agrega ciertas funciones a la interfaz de conexión con una base de datos MySQL. Las siglas se corresponden con **MySQL Improved** (MySQL Mejorado). MySQLi soporta tanto el paradigma procedimental como el paradigma de la programación orientada a objetos. El paradigma procedimental se hereda de la antigua extensión (y que no debemos usar) mysql.

En las primeras versiones de MySQL, se dividen las tareas en **procedimientos lineales**, lo que dificulta la modificación porque se tiene que editar el código desde la parte superior. Sin embargo, en la nueva versión



mejorada de MySQLi, se ven los datos como un conjunto de objetos, lo que nos permite agregar o eliminar datos de una forma sumamente sencilla.

Un dato a tener en cuenta, en la actualidad la mayoría de servidores ya posee la [extensión MySQLi](#). Por lo tanto, no tenemos que estar haciendo malabares para encontrar dicha extensión.

PROCEDIMENTAL

En el siguiente script se incluye todos los pasos detallados anteriormente. Tan solo debemos guardarlo en un archivo.php y ejecutarlo desde un navegador o desde la consola.

```
<?php
$conn = mysqli_connect("localhost", "root", "", "Despachos");
if (!$conn) {
    echo "Error al conectar a la base de datos: " . mysqli_connect_error() .
"\n";
}
echo 'Conectado a la base de datos.<br><br>';
$result = $conn->query("SELECT * FROM Unidades_Medida");
echo "Numero de resultados: $result->num_rows";
$result->close();
$conn->close();
?>
```

- Se requiere conocer cuatro parámetros que tenemos que pasar a la función: \$servername (dirección o nombre del servidor), \$database (base de datos), \$usuario (usuario), \$clave (contraseña).
- Cuando se realiza la conexión la BD, si no es correcta, se ejecuta la función die(). Lo que hace esta función es detener la ejecución por completo y mostrar un mensaje de error y/o advertencia seguido de un mensaje de error exacto que describe el problema.
- Por último, se cerrará la conexión con la base de datos utilizando la funciónmysql_close(), a la cual le pasaremos como variable la conexión creada previamente. Si no especificamos el cierre de la conexión, MySQL cerrará por sí solo la conexión una vez finalice el script.

ORIENTADO A OBJETOS

Cree un archivo .php en la raíz de nuestro servidor o dentro de una determinada carpeta para que se ajuste a sus necesidades:

```
<?php
$conn = new mysqli("localhost", "root", "", "Despachos");

if ($conn->connect_error) {
    die("ERROR: No se puede conectar al servidor: " . $conn->connect_error);
}

echo 'Conectado a la base de datos.<br>';

$result = $conn->query("SELECT * FROM Unidades_Medida");

echo "Numero de resultados: $result->num_rows";
```



```
$result->close();  
$conn->close();  
?>
```

- MySQLi, iniciará una nueva conexión usando la extensión mysqli. Esta función está siendo muy utilizada y tendrá estos cuatro criterios para su funcionamiento.
- Hostname, donde se ejecuta la base de datos MySQL
- Nombre de usuario, para MySQL
- Clave, para acceder a MySQL
- Nombre Base de datos MySQL, para poder interactuar con la información.
- La función de consulta, en la cual mostrara datos existentes en la tabla de la base de datos.

3.4. CONEXIÓN A MYSQL DESDE PHP USANDO FORMA ANTIGUA MySQL (No recomienda)

Conectarse mediante métodos tradicionales de MySQL (no recomendado). Hay que tener en cuenta que si tenemos activado la [versión del PHP v7.0 y/o superiores](#), este tipo de conexión no funcionara, porque esta versión ya desconoce las funciones de mysql_connect.

Esta extensión de conexión fue obsoleta a partir de la versión de PHP 5.5. Sin embargo, a partir de la versión de php 7.0 hacia adelante, esto ni siquiera se reconocerá, ya que fue eliminado por completo.

Cree el siguiente archivo .php en su directorio de trabajo:

```
<?php  
// Datos para acceder a MySQL  
$conn = mysql_connect('localhost', 'root', '');  
mysql_select_db("Despachos");  
  
//Muestra error si no son las credenciales correctas  
if (!$conn) {  
die('ERROR: No se pudo conectar con el servidor: ' . mysql_error());  
}  
echo 'Se ha conectado con la database.<br>';  
  
$result = mysql_query('SELECT * FROM Unidades_Medida');  
// Devuelve resultados de la base de datos  
$row = mysql_fetch_row($result);  
// Imprime resultados con echo  
echo "Unidad de Medida 1: ". $row[0]. " ".$row[1]. " <br>";  
  
mysql_close($conn);  
?>
```

3.5. CONCLUSION Y RECOMENDACIÓN



Se han presentado tres tipos de conexión, la tercera conexión no está disponible si tienen la versión del PHP actualizado. Se recomienda utilizar el método de MySQLi y el método de PDO.

4. OTRAS FUENTES

<https://www.baulphp.com/3-formas-para-conectarse-a-mysql-desde-php/>

5. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)				

6. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					