

MYSQLI

PHP y MySQLi



¿QUÉ ES MYSQLI?

Introducción

- MySQLi es el acrónimo de MySQL **improved** (MySQL mejorado)
- Las funciones de MySQL están en **desuso** desde la versión 4.1.3
- MySQLi se desarrolló para aprovechar las nuevas **funcionalidades** encontradas en los sistemas MySQL desde esa versión 4.1.3

Introducción

- La extensión MySQLi contiene muchos beneficios, entre los que se encuentran:
 - ✓ Interfaz orientada a objetos
 - ✓ Soporte para Declaraciones Preparadas
 - ✓ Soporte para Múltiples Declaraciones
 - ✓ Soporte para Transacciones
 - ✓ Mejoradas las opciones de depuración
 - ✓ Soporte para servidor empotrado

CONEXIÓN CON EL SERVIDOR

Conexión con el servidor

- Lo primero que hay que hacer es **crear** una conexión desde el documento PHP hasta el Sistema Gestor de Bases de Datos
- Al crear la conexión se genera un **canal** a través del cual se van a enviar todas las consultas y todas las peticiones a la Base de Datos

Conexión con el servidor

- **mysqli_connect(servidor, usuario, pass, bd)**
 - **Servidor**: Nombre del servidor al que nos queremos conectar (localhost)
 - **Usuario**: Nombre del usuario con el que nos vamos a conectar (root)
 - **Pass**: Contraseña del usuario con el que nos conectamos al servidor (vacía)
 - **BD**: Base de datos a la que queremos conectarnos de todas las que haya en el servidor
- Devuelve un conector (canal) que debemos recoger en una variable.

REALIZACIÓN DE CONSULTAS

Realización de consultas

- La consulta que queramos enviar a la base de datos será una **cadena de texto**.
- Puede estar almacenada en una variable o no
- Podemos realizar **cualquier operación** de las que hemos visto en SQL
- Para ejecutar una consulta debemos utilizar la función:

Realización de consultas

- **mysqli_query (conector, consulta)**
 - **conector**: canal creado al hacer mysqli_connect
 - **consulta**: cadena de texto con la operación que queremos ejecutar.
- Dependiendo del tipo de operación que se ejecute el resultado de mysqli_query va a ser diferente

```
$datos = mysqli_query($conector, $consulta)
```

Resultado de *mysqli_query*

- Si la consulta es un **SELECT**:
 - ✓ FALSE si ocurre un error.
 - ✓ Conjunto de datos si todo va bien.
- Si la consulta era **distinta de SELECT** (insert, update, delete):
 - ✓ FALSE si ocurre un error.
 - ✓ TRUE si todo va bien.

MANIPULACIÓN DE LOS RESULTADOS

Manipulación de los resultados

- Una vez ejecutada la consulta, podemos saber a cuántas filas a afectado dicha consulta:
- **mysqli_affected_rows (conector)**
 - Devuelve el número de filas afectadas por la última consulta realizada de tipo **insert, update o delete**
 - Devuelve el número de filas devueltas por la última consulta realizada de tipo **select**

Manipulación de los resultados

- Normalmente una sentencia Select devuelve un **conjunto de valores**
- Este conjunto de valores **no** es accesible directamente
- Hay que ir recogiendo fila a fila y trabajando con ellas de forma independiente

Manipulación de los resultados

➤ `mysqli_fetch_array` (datos)

- Devuelve una **array asociativo** en el que encontraremos una de las filas del conjunto de datos.
- El array tiene por cada campo devuelto por el SELECT **dos** posiciones.
 - ✓ *Númerica*: 0, 1, 2...
 - ✓ *Asociativa*: Nombre del campo devuelto
- Devuelve **NULL** cuando no hay más datos para mostrar

Ejemplo

```
$consulta = "select nombre, edad from alumnos";  
$resul = mysqli_query($conexion, $consulta);  
  
$num_campos = mysqli_num_rows($resul);  
$fila=mysqli_fetch_array($resul)
```

- En este caso el array **\$fila** tendrá la siguiente estructura:

0	Nombre	1	Edad
Juan	Juan	24	24

Manipulación de los resultados

➤ `mysqli_fetch_assoc` (datos)

- Devuelve un **array asociativo** en el que encontraremos una de las filas del conjunto de datos.
- En este caso el array **sólo tiene una posición** por cada campo devuelto por el select
- El nombre de la posición será el nombre del campo devuelto por select
- Devuelve **Null** cuando no hay más datos

Manipulación de los resultados

➤ `mysqli_fetch_all` (datos)

- Devuelve en este caso una **matriz posicional** que tendrá:
 - ✓ Tantas filas como filas devuelva la consulta select
 - ✓ Tantas columnas como campos le pidamos a la consulta select.

Manipulación de los resultados

```
$consulta = "select nombre, edad from alumnos";  
$datos = mysqli_query($conector, $consulta);  
$matriz = mysqli_fetch_all($datos);
```

	0	1
0	Ramón Torres	19
1	María López	21
2	Paloma Ruiz	24
3	Isabel Perea	25
4

COMPROBACIÓN DE ERRORES

Comprobación de errores

➤ **mysqli_error (conector)**

- Devuelve el mensaje de error para la última acción que se haya hecho por el conector
- Lo que devuelve es una cadena de texto

Comprobación de errores

➤ Ejemplo

```
$consulta = "select nommbre from alumnos";  
  
$datos = mysqli_query($conexion, $consulta);  
  
echo mysqli_error($conexion);
```

Unknown column 'nommbre' in 'field list'

Comprobación de errores

➤ `mysqli_error_list` (conector)

- Devuelve en este caso una **matriz** en la que se almacenarán una lista con los errores que se hayan producido.
- Cada error será una fila de la matriz que tendrá:
- **Errno**: número del error ocurrido
- **Error**: cadena de texto asociada al error
- **Sqlstate**: error con la nomenclatura SQLSTATE

Comprobación de errores

➤ Ejemplo

```
$conexion = mysqli_connect("localhost", "root", "", "cenro");  
  
$consulta = "select nommbre from alumnos";  
$datos = mysqli_query($conexion, $consulta);  
  
print_r(mysqli_error_list($enlace));
```

```
Array  
(  
    [0] => Array  
        (  
            [errno] => 1054  
            [sqlstate] => 42S22  
            [error] => Unknown column 'nommbre' in 'field list'  
        )  
)
```


CERRAR LA CONEXIÓN

Cerrar la conexión

- Una vez hecho todo lo necesario es muy importante cerrar la conexión con la Base de Datos
- Mientras la conexión esté abierta nadie más podrá usar la Base de Datos
- **mysqli_close(conector)**
 - Cierra la conexión abierta

EJEMPLO

Ejemplo

- Vamos a conectarnos a la base de datos CENTRO para mostrar el nombre de todos sus alumnos
- Para ello:

Ejemplo

- 1º Abrimos la conexión y elegimos la Base de datos a la que nos queremos conectar:

```
$conexion = mysqli_connect ('localhost', 'root', '', 'centro');  
if (!$conexion)  
{  
    echo "<h3> ~~~~~ERROR AL CONECTAR CO EL SGBD~~~~~</h3>";  
}
```

Ejemplo

- 2º Escribimos la consulta que queremos ejecutar:

```
$consulta = "select nom_alum from alumnos";
```

Ejemplo

- 3º Ejecutamos la consulta y comprobamos el número de registros que ha devuelto

```
$resul = mysqli_query($conexion, $consulta);  
$num_registros = mysqli_affected_rows ($conexion);
```

Ejemplo

- 4º Mostramos por pantalla los datos que hemos obtenido

```
echo "Hay un total de $num_registros alumnos en el centro<br>";  
echo "Nombre de los alumnos: <br>";  
  
while ($fila = mysqli_fetch_array($resul))  
{  
    echo "<br>".$fila['nom_alum'];  
}
```


Ejemplo

- 5º y último: cerramos la conexión con la base de datos

```
mysqli_close($conexion);
```

Ejemplo

Hay un total de 5 alumnos en el centro
Nombre de los alumnos:

Lucia

Luis

Monica

Cesar

Roberto

CONSEJOS

Consejos

- Puesto que ciertas acciones las realizaremos siempre igual:
 - ✓ Abrir conexión
 - ✓ Comprobar que todo ha ido bien
 - ✓ Seleccionar la base de datos
 - ✓ Etc
- Lo mejor es colocarlo en funciones a la que iremos llamando cuando sea necesario

MYSQLI

PHP y MySQLi

