

PHP – Bases de Datos I

Conexión con el servidor MySQL

Para empezar a trabajar con bases de datos es necesario que el servidor Apache y MySQL estén iniciados. Para conectar ambos es necesario utilizar una función PHP, **mysqli_connect ()**, que tiene la siguiente sintaxis:

```
$conexion = mysqli_connect ($host, $username, $passwd,  
$dbname) ;
```

donde:

- ☐ **\$conexion:** es la variable que almacena el identificador de la conexión.
- ☐ **\$host:** es el servidor de base de datos. si la base de datos es local el servidor es localhost
- ☐ **\$username:** El nombre de usuario de MySQL.
- ☐ **\$passwd:** es la clave del usuario.
- ☐ **\$dbname:** *[Opcional]* Si se proporciona, especificará la base de datos predeterminada a usar cuando se realizan consultas.

Estos tres valores tienen que ir encerrados entre comillas. Ejemplo:

```
$conexion = mysqli_connect ("localhost", "root",  
"","base_de_datos_1");
```

La conexión se cerrará tan pronto como la ejecución del script finalice, a menos que se cierre explícitamente con la función **mysqli_close ()**.

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

Un ejemplo de conexión al servidor podría ser éste:

```
<?php

$servidor = "localhost";
$usuario = "root";
$clave = "";
$dbname = "base_de_datos_1";

$conexion = mysqli_connect ($servidor,$usuario,$clave,
$dbname);

if ($conexion)
    echo "<h2>CONEXION REALIZADA CORRECTAMENTE</h2>";
else{
    echo"<h2>ERROR: no tienes permiso de conexión</h2>";
    exit();
}

$conexion = mysqli_close($conexion);

if ($conexion){
    echo "<h2>CONEXION CERRADA AUTOMATICAMENTE</h2>"
}

?>
```

Otra forma más corta de comprobar si la conexión se ha realizado correctamente es añadiendo a la función **mysqli_connect ()** la expresión **or die ('mensaje')**:

```
$conexion = mysqli_connect ($servidor,$usuario,$clave,
$dbname) or die ('No pudo conectar');
```

Es una opción alternativa a **exit()**. En el caso que se produzca un error en la conexión, se interrumpirá la ejecución del script y aparecerá en la ventana del navegador el texto incluido en 'mensaje'. Se puede aplicar también a la función **mysqli_close ()**:

```
mysqli_close($conexion) or die ('no se pudo cerrar
correctamente');
```

Para obtener una mayor información sobre los posibles errores de conexión se puede usar:

mysqli_connect_error: devuelve una cadena con la descripción del último error de conexión o null en caso de éxito.

```
$link = mysqli_connect('localhost', 'fake_user', 'my_password', 'my_db');
```

```
if (!$link) {
    die('Error de conexión: ' . mysqli_connect_error());
}
```

Conexión de una base de datos MySQL

Una vez conectados a MySQL si aún no hemos seleccionado a la base de datos deseada a través de `mysqli_connect`, podríamos seleccionar la base de datos con la que trabajar a partir de la función:

```
mysqli_select_db ($conexion, "nombrebd")
```

donde **nombrebd** es el nombre de la base de datos MySQL, con la que trabajaremos y **\$conexion** es el identificador de la conexión, es decir, la variable creada al establecer la conexión con MySQL. Para comprobar si la conexión a la base de datos se ha realizado correctamente usaremos la siguiente instrucción **if**:

Nota:

Esta función debería ser usada solo para cambiar la base de datos por defecto para la conexión. Se puede seleccionar la base de datos por defecto en el cuarto parámetro de la función **mysqli_connect()**.

```
if (mysqli_select_db ($conexion, "nombrebd")) {  
    //conexion realizada correctamente, no haría  
    //falta escribir instrucciones  
    //el programa continua después del if  
}  
else{  
    echo "Conexión NO REALIZADA";  
    exit ( );  
    //Si no conecta la conexión,  
    //finaliza la ejecución  
}
```

Otra forma de comprobar la conexión sería añadiendo **or die ()**:

```
mysqli_select_db ($conexion, "nombrebd") or die ('no  
    puedo conectar');  
//La conexión a la BD se ha realizado correctamente
```

El siguiente ejemplo intenta conectar a una bd Empresa:

```
<?
$servidor = "localhost";
$usuario = "root";
$clave = "";
$dbname= "Empresa";

$conexion = mysqli_connect($servidor,$usuario,$clave,
                           $dbname) or die ('NO PUDO CONECTARSE');

echo "<br><h2><b><center>CONEXIÓN REALIZADA
CORRECTAMENTE.</center></b></h2>";

//Seleccionamos otra base de datos
mysqli_select_db($conexion,"base_de_datos_1")
    or die("<br><h2><b>***CONEXIÓN a la base de datos NO
REALIZADA***</b></h2>");

echo "<br><h2><b>***CONEXIÓN a la base de datos
REALIZADA***</b></h2>";

mysqli_close($conexion) or die('NO SE PUDO CERRAR LA
CONEXIÓN');

echo "<br><h2><b><center>CONEXIÓN CERRADA CORRECTAMENTE.</
center></b></h2>";

?>
```

Consulta de registros

Para que PHP se comunice con MySQL utilizaremos la función PHP **mysqli_query ()**. El formato es el siguiente:

```
$resultado = mysqli_query ($conexion , "consulta_sql")
```

donde:

- ☐ **consulta_sql:** contiene la instrucción propia de MySQL, por ejemplo, una orden SELECT..
- ☐ **\$conexion:** contiene el identificador de la conexión.
- ☐ **\$resultado:** recoge el resultado devuelto por la ejecución de la orden SQL.

Por ejemplo, para consultar todos los datos de los departamentos de la tabla DEPARTAMENTOS escribimos lo siguiente:

```
$resultado = mysqli_query ($conexion, "SELECT * FROM  
Departamentos");
```

El resultado recogido en la variable **\$resultado** está estructurado en filas y la función **mysqli_fetch_array ()** recoge cada una de las filas. el formato es el siguiente:

```
$fila = mysqli_fetch_array ($resultado);
```

La función recoge en la variable **\$fila** la primera fila y coloca su puntero interno al comienzo de la fila siguiente. Por ello, para la lectura completa de todas las filas de la variable **\$resultado** se requieren llamadas sucesivas a **mysqli_fetch_array ()** (un **bucle**) hasta que haya sido leída la última fila del resultado.

```
while ($fila = mysqli_fetch_array ($resultado) {  
    //lo que sea  
}
```

Cuando el puntero interno alcance el final de la última fila del resultado, la función devolverá FALSE.

La variable **\$fila** tiene estructura de array en la que el **índice es el nombre de la columna de la tabla** de la que se ha extraído cada resultado; también **puede ser un nº** que empieza en 0 y que **indica el orden de las columnas recuperadas**; la primera columna tiene el índice 0 asociado, la segunda el índice 1, y así sucesivamente.

Ejercicio 1: A partir de la BD suministrada (Departamentos, Empleados, etc.) crear un script que conecte con dicha base de datos y devuelva este resultado:

Departamento: 10
Nombre: CONTABILIDAD
Localidad: SEVILLA

Departamento: 20
Nombre: INVESTIGACIÓN
Localidad: MADRID

Departamento: 30
Nombre: VENTAS
Localidad: BARCELONA

Departamento: 40
Nombre: PRODUCCIÓN
Localidad: BILBAO

PHP dispone de dos funciones que permiten conocer el nº de registros de la tabla afectados por una sentencia MySQL:

- ☐ **mysqli_num_rows (\$resultado):** devuelve el nº de registros que cumplen las condiciones establecidas en una consulta. Sólo es válido para sentencias tipo SELECT.
- ☐ **mysqli_affected_rows (\$resultado):** devuelve el nº de registros afectados por una orden INSERT, UPDATE o DELETE.

Ejercicio 2: Realiza un script PHP que consulte los datos: APELLIDO, OFICIO y SALARIO de los empleados de la tabla EMPLEADOS del departamento 20, visualizando al final el número de empleados recuperados.

Apellido: SANCHEZ
Oficio: EMPLEADO
Salario: 1040.00

Apellido: JIMENEZ
Oficio: DIRECTOR
Salario: 2900.00

Apellido: GIL
Oficio: ANALISTA
Salario: 3000.00

Apellido: ALONSO
Oficio: EMPLEADO
Salario: 1430.00

Apellido: FERNANDEZ
Oficio: ANALISTA
Salario: 3000.00

Número de empleados en el departamento 20 : 5

Ejercicio 3: Modifica el ejercicio 1 para devolver este resultado.

LISTADO DE LOS DEPARTAMENTOS

Numero	Nombre	Localidad
10	CONTABILIDAD	SEVILLA
20	INVESTIGACI?N	MADRID
30	VENTAS	BARCELONA
40	PRODUCCI?N	BILBAO

Numero de departamentos: 4

Ejercicio 4: Modifica el ejercicio 3 para devolver este resultado.

LISTADO DE LOS EMPLEADOS DEL DEPARTAMENTO 20

APELLIDO	OFICIO	SALARIO
SANCHEZ	EMPLEADO	1040.00
JIMENEZ	DIRECTOR	2900.00
GIL	ANALISTA	3000.00
ALONSO	EMPLEADO	1430.00
FERNANDEZ	ANALISTA	3000.00

Número de Empleados: 5