

Unidad 1

Nivelación PHP y MySQL

PHP Y MYSQL

Desde PHP es posible invocar funciones y ejecutar sentencias específicas de MySQL.

Lo primero que tenemos que hacer es conectar con la base de datos, para esto usaremos la función `mysqli_connect()`

```
$conexion=mysqli_connect("servidor ", "usuario_mysql ", "contraseña_mysql ",  
"nombre_base_datos ")
```

Para entender, la interconexión entre ambos se realiza mediante una serie de instrucciones.


`mysqli_query ($conexion, consulta)` envía una consulta a la base activa asociada al identificador de enlace `$conexion`.

Devuelve TRUE si la sentencia se ha ejecutado correctamente y FALSE en caso contrario.

La ejecución de esta instrucción requiere, como vimos, que previamente hayamos creado una conexión a un servidor MySQL con la función `mysqli_connect()`, estableciendo en los tres primeros parámetros de la función el host, usuario y password de conexión, lo cual nos permitirá luego interactuar con una base de datos y las tablas que la compongan, y que se haya asociado justamente, una base de datos al identificador de enlace mediante la misma instrucción asignando como cuarto parámetro el nombre de la base.

Vamos a plantear una sencilla base de datos.

Allí crearemos una tabla llamada "clientes" con sólo cuatro campos, id, nombre, actividad y web.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra
1	id 	int(5)			No	Ninguna		AUTO_INCREMENT
2	nombre	varchar(20)	utf8_unicode_ci		No	Ninguna		
3	actividad	varchar(50)	utf8_unicode_ci		No	Ninguna		
4	web	varchar(30)	utf8_unicode_ci		No	Ninguna		

Selección y lectura de registros con PHP

Una vez creada la tabla clientes en nuestra base de datos ejemplo, haremos la consulta de los datos que se encuentran en la tabla.

Utilizamos el comando SELECT de SQL para crear una selección de nuestra tabla y mostrar todos los datos en pantalla por medio de un bucle. Con PHP.

Dentro de una base de datos, organizada por tablas, la selección de una tabla entera o de un cierto número de registros resulta una operación rutinaria. La siguiente es una forma bastante clásica de mostrar en pantalla a partir de un bucle los registros seleccionados por una sentencia SQL:

```
<!doctype html>
<html>
<head>
<title>Lectura de datos</title>
</head>
<body>
<h1>Lectura de la tabla</h1>
<?php
//conexion con la base y seleccion de la base de datos
$conexion = mysqli_connect("localhost","root","","unidad2_php");
//ejecutamos la sentencia sql
$result=mysqli_query($conexion, "SELECT * from clientes");
?>
<table align="center" border="1">
<tr>
<th>Nombre</th>
<th>Actividad</th>
<th>Web</th>
</tr>
<?php
//mostramos los registros
while ($row=mysqli_fetch_array($result))
{
echo '<tr><td>'.$row['nombre'].'</td>';
echo '<td>'.$row['actividad'].'</td>';
echo '<td>'.$row['web'].'</td></tr>';
}
mysqli_free_result($result)
?>
```

```
</table>
<nav>
<ul>
<li><a href="insertar.html">añadir un nuevo registro</a></li>
<li><a href="actualizar.php">actualizar un registro existente</a></li>
<li><a href="borrar.php">borrar un registro</a></li>
</ul>
</nav>
</body>
</html>
```

Los pasos a realizar son, en un principio: Conexión a la base y ejecución de la sentencia.

La información de dicha ejecución será almacenada en una variable (\$result).

El siguiente paso será plasmar en pantalla la información recogida en \$result.

Esto lo haremos mediante la función `mysqli_fetch_array` que devuelve una variable array con los contenidos de un registro a la vez que se posiciona sobre el siguiente.

El bucle `while` nos permite leer e imprimir secuencialmente cada uno de los registros.

La función `mysqli_free_result` se encarga de liberar la memoria utilizada para llevar a cabo la consulta.

Aunque no es necesaria su utilización, resulta altamente aconsejable.

Introducción de nuevos registros con PHP

Para ello usaremos un formulario y en el ACTION del FORM indicaremos que debe ser procesado una página PHP, llamada "insertar.php". Esta página lo que hará será introducir los datos del formulario en la base de datos.

```
<form action="insertar_registro.php">
```

Los datos del registro son recogidos a partir de un formulario que obtiene los datos y los envía a una página PHP que se encarga de procesarlos.

```
<!doctype html>
<html>
<head>
<title>Insertar datos</title>
<meta charset="utf-8">
</head>
<body>

<h1>Insertar un registro</h1>
<form method="post" action="insertar_registro.php">
<input type="text" name="nombre" placeholder="Nombre">
<input type="text" name="actividad" placeholder="Actividad">
<input type="text" name="web" placeholder="Sitio Web">
<input type="submit" value="Insertar">
</form>
<ul>
<li><a href="mostrar_datos.php">Volver al listado de Clientes</a></li>
</ul>
</body>
</html>
```

Llegados a la página destino del formulario (insertar_registro.php), lo primero que haremos es establecer un vínculo entre el programa y la base de datos.

Esta conexión se lleva a cabo con la función `mysqli_connect`.

Luego de seleccionar la base a utilizar debemos generar una orden de inserción del registro en lenguaje SQL. Esta orden será ejecutada por medio de la función `mysqli_query`, luego redireccionamos al archivo `mostrar_registro.php`:

```
<?php
//conexion con la base y seleccion de la base de datos
$conexion = mysqli_connect("localhost","root","","unidad2_php");

//traigo los datos del formulario
$nombre = $_POST['nombre'];
$actividad = $_POST['actividad'];
$web = $_POST["web"];

//ejecución de la sentencia sql
mysqli_query($conexion, "INSERT INTO clientes (nombre, actividad, web) VALUES
('$nombre','$actividad','$web')");<

header("Location: mostrar_datos.php");
?>
```


Borrado de un registro con PHP

Otra de las operaciones elementales que se pueden realizar sobre una base de datos es borrar un registro. Para hacerlo, SQL nos propone sentencias del tipo DELETE.

Cabe señalar que primero debemos seleccionar el registro que se desea borrar y luego realizar el borrado propiamente dicho. Para ello crearemos un menú desplegable dinámico, donde se podrá seleccionar el elemento que se desea borrar.

Luego se pasará a una página PHP una referencia al elemento seleccionado, para borrarlo de la base de datos.

```
<!doctype html>
<html>
<head>
<title>Borrar Datos</title>
<meta charset="utf-8">
</head>
<body>
<h1>Borrar un registro</h1>

<?php
//conexion con la base y seleccion de la basede datos
$conexion = mysqli_connect("localhost","root","","unidad2_php");
?>

<form method="post" action="borrar_registro.php">
<?php
//creamos la sentencia sql y la ejecutamos

$ssql="SELECT id, nombre FROM clientes ORDER BY nombre";
$result=mysqli_query($conexion, $ssql);
?>
<select name="id">
<?php
//mostramos los registros en forma de menú desplegable
while ($row=mysqli_fetch_array($result)){
echo '<option value="'. $row['id']. "'>'. $row['nombre']. '</option>';
}
mysqli_free_result($result);
?>
```

```
</select>
<input type="submit" value="borrar">
</form>

<ul>
    <li><a href="mostrar_datos.php">Volver al listado de Clientes</a></li>
</ul>

</body>
</html>
```

El siguiente paso es hacer efectiva la operación a partir de la ejecución de la sentencia SQL que construimos a partir de los datos del formulario:

```
<?php
//pasamos los datos del formulario
$id = $_POST["id"];
//conexion con la base y seleccion de la basede datos
$conexion = mysqli_connect("localhost","root","","unidad2_php");
//creamos la sentencia sql y la ejecutamos
$sql="DELETE FROM clientes WHERE id='$id'";
mysqli_query($conexion,$sql);
?>
```


Actualización de un registro de base de datos con PHP

Para mostrar cómo se actualiza un registro presente en nuestra base de datos, vamos a hacerlo también a partir de un formulario a través del cual establecemos el registro que vamos a modificar y luego el script definido en el action del formulario se encargará de ejecutar la sentencia sql encargada de hacer las modificaciones en la tabla.

El archivo del formulario va a ser esta vez un script PHP en el que efectuaremos una llamada a nuestra base de datos para construir un menú desplegable donde aparezcan todos los nombres.

Quedaría así:

```
<!doctype html>
<html>
<head>
<title>Actualizar datos</title>
<meta charset="utf-8">
</head>
<body>
<h1>Actualizar un registro</h1>

<?php
//conexion con la base y seleccion de la basede datos
$conexion = mysqli_connect("localhost","root","","unidad2_php");
?>
<form method="post" action="actualizar_registro.php">
<?php
//creamos la sentencia sql y la ejecutamos
$sql="SELECT id, nombre FROM clientes ORDER BY nombre";
$result=mysqli_query($conexion,$sql);
?>
<select name="id">
<?php
//generamos el menu desplegable
while ($row=mysqli_fetch_array($result)){
echo '<option value="'. $row['id']. "'>'. $row['nombre']. '</option>';
}
?>
</select>
<input type="text" name="actividad" placeholder="Actividad">
<input type="text" name="web" placeholder="Web">
```

```
<input type="submit" value="Actualizar">
</form>
<ul>
<li><a href="mostrar_datos.php">Volver al listado de Clientes</a></li>
</ul>
</body>
</html>
```

La manera de operar para construir el menú desplegable es la misma que para visualizar la tabla.

De nuevo empleamos un bucle while en combinación con la función `mysqli_fetch_array` lo que nos permite mostrar cada una de las opciones.

El script de actualización será muy parecido al de inserción:

```
<?php
//pasamos los datos del formulario
$id = $_POST['id'];
$actividad = $_POST['actividad'];
$web = $_POST['web'];
//conexion con la base y seleccion de la base de datos
$conexion = mysqli_connect("localhost","root","","unidad2_php");
//creamos la sentencia sql y la ejecutamos
$sql="UPDATE clientes SET actividad='$actividad', web='$web' WHERE id=$id";
mysqli_query($conexion, $sql);
header("Location: mostrar_datos.php");
?>
```