

1 Exemplo empregando as clase mysqli e mysqli stmt

• tabla.sql

```
CREATE TABLE IF NOT EXISTS `pruebas` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `nombre` varchar(15) NOT NULL,  
  `apellidos` varchar(25) DEFAULT NULL,  
  `edad` tinyint(4) NOT NULL,  
  `fechanac` date NOT NULL,  
  `password` varchar(50) NOT NULL,  
  PRIMARY KEY (`id`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
```

• conexion.php

```
<?php  
/////////////////////////////////  
// conexion.php  
/////////////////////////////////  
  
// Datos configuración de la conexión al Servidor MySQL.  
  
$nombrebase="basedatos";  
$usuario="usuario";  
$servidor="localhost";  
  
$password="password";  
/////////////////////////////////  
  
// Hacemos la conexión al servidor de MySQL utilizando la clase mysqli  
// Utilizamos la @ antes de crear el objeto, ya que si da error,  
// queremos mostrarlo con un mensaje personalizado.  
  
$mysqli=@new mysqli($servidor,$usuario,$password,$nombrebase);  
  
if ($mysqli->connect_error)  
{  
  die ("Error en conexión base datos: ".$mysqli->connect_error);  
}  
?>
```

• altas.php

```
<?php  
// Crear el formulario para recoger datos de los usuarios.  
  
// Si estamos recibiendo datos del formulario...  
// Insertamos esos datos en la tabla.  
  
if ( isset($_POST['nombre']) && $_POST['nombre']!="")  
{  
  // Nos conectamos a la base de datos.  
  require("conexion.php");  
  
  // Preparamos la instrucción SQL. Prepare nos devuelve un objeto de la clase mysqli_stmt  
  // o false si hay un error.  
  $stmt = $mysqli->prepare("insert into pruebas (nombre,apellidos,edad,fechanac,password)  
  values(?,?,?,?,?)") or die($mysqli->error);  
  
  // Enlazamos los parámetros (5 cadenas de texto 's')  
  $stmt->bind_param("sssss",$_POST['nombre'],$_POST['apellidos'],$_POST['edad'],$_POST['fechanac'],  
  $_POST['password']);
```

```

// Ejecutamos la sentencia preparada.
$stmt->execute() or die($mysqli->error);

// Mostramos el mensaje x filas insertadas correctamente.
printf("%d fila insertada correctamente.\n", $stmt->affected_rows);

// Cerramos la sentencia preparada.
$stmt -> close();

// Cerramos la conexión.
$mysqli->close();
}
else
{
// En otro caso mostramos el formulario de altas.
?>

    <h2> Alta de Usuarios </h2>

    <form name="formulario" method="post">
    Nombre: <input name="nombre" type="text"/><br/>
    Apellidos: <input name="apellidos" type="text"/><br/>
    Edad: <input name="edad" type="text" size="2" maxlength="2" /><br/>
    Fecha Nacimiento: <input name="fechanac" type="text"/><br/>
    Password: <input name="password" type="password"/><br/>
    <input type="reset" value="Limpiar"/>
    <input type="submit" value="Alta Usuario"/><br/>
    </form>

<?php
}
?>

```

• bajas.php

```

<?php
// Conectamos a la base de datos
require("conexion.php");

if (isset($_POST['id']) && $_POST['id']!="-1")
{
// Construimos la sentencia de borrado SQL.
$stmt = $mysqli->prepare("delete from pruebas where id=?") or die($mysqli->error);

// Enlazamos los parámetros (1 entero 'i')
// http://es2.php.net/manual/es/mysqli-stmt.bind-param.php

$stmt->bind_param("i", $_POST['id']);

// Ejecutamos la sentencia preparada.
$stmt->execute() or die($mysqli->error);

// Mostramos mensaje
echo "Registro borrado correctamente";

// Cerramos la sentencia preparada.
$stmt->close();
}
?>

<h2> Borrado de registros </h2>
<form name="formulario" method="post">
<select name="id">
<option value="-1">Selecione registro</option>
<?php
// Cubrimos dinámicamente el formulario.

// Construimos la sentencia de select SQL

```

```

$stmt = $mysqli->prepare("select id,nombre,apellidos from pruebas") or die($mysqli->error);

// No tenemos ningún parámetro a enlazar, así que..
// ejecutamos la sentencia preparada directamente
$stmt->execute() or die($mysqli->error);

// Si es una consulta de select almacenamos el resultado con el método store_result() del objeto $stmt.
$stmt->store_result();

// Número de filas obtenidas, si fuera necesario.
// $numfilas=$stmt->num_rows;

// Vinculamos unas variables a los resultados con el método bind_result() de stmt,
// antes de llamar al método fetch() de stmt
$stmt->bind_result($campoid, $camponombre, $campoapellidos);

// Recorremos el recordset:
while ($stmt->fetch())
{
    $cadena=sprintf("<option value='%s'>%s %s</option>", $campoid, $camponombre, $campoapellidos);
    echo $cadena;
}

// Liberamos el resultado.
$stmt-> free_result();

// Cerramos la sentencia preparada.
$stmt->close();

// Cerramos conexión
$mysqli->close();
?>
</select>
<input type="submit" value="Borrar Registro"/>
</form>

```

• modificaciones.php

```

<?php
// Conectamos a la base de datos.
require("conexion.php");

// Si recibimos datos formulario de selección.
if (isset($_POST['selectid']) && $_POST['selectid']!=-1)
{
    // Preparamos la consulta SQL
    $stmt= $mysqli->prepare("select id,nombre,apellidos,edad from pruebas where id=?")
    or die($mysqli->error());

    // Enlazamos los parámetros (1 entero 'i')
    // http://es2.php.net/manual/es/mysqli-stmt.bind-param.php
    // Vinculamos los parámetros a la consulta
    $stmt->bind_param("i", $_POST['selectid']);

    // Ejecutamos la consulta preparada.
    $stmt->execute();

    // Si es una consulta de select almacenamos el resultado.
    $stmt->store_result();

    // Vinculamos unas variables a los resultados con el método bind_result() de stmt,
    // antes de llamar al método fetch() de stmt
    $stmt->bind_result($elid, $elnombre, $losapellidos, $laedad);

    // Leemos la primera fila de ese recordset
    // Los datos que hemos leído estará en las variables que hemos vinculado antes.
    $stmt->fetch();
}

```

```

// Construimos el formulario y ponemos los datos
// de ese usuario en los campos.
?>

<h2>Edición de Usuario</h2>

<form name="formulario" method="post">
Nombre: <input name="nombre" type="text" value="$nombre" /><br/>
Apellidos: <input name="apellidos" type="text" value="$apellidos" /><br/>
Edad: <input name="edad" type="text" size="2" maxlength="2" value="$edad" /><br/>
<input type="hidden" name="id" value="$id" />

<input type="reset" value="resetear" />
<input type="submit" value="Actualizar" /><br/>
</form>

<?php
}
// Si recibimos datos del formulario de edición...
if (isset($_POST['id']))
{
// Preparamos la consulta SQL
$stmt = $mysqli->prepare("update pruebas set nombre=?,apellidos=?,edad=? where id=?");

// Enlazamos los parámetros (3 string:'s' y 1 entero:'i')
// http://es2.php.net/manual/es/mysqli-stmt.bind-param.php
$stmt->bind_param("sssi",$_POST['nombre'],$_POST['apellidos'],$_POST['edad'],$_POST['id']);

// Ejecutamos consulta de update.
$stmt->execute() or die($mysqli->error());

// Cerramos la sentencia preparada.
$stmt -> close();

// Mostramos mensaje de OK.
echo "Actualización correcta de los datos del usuario: {$_POST['nombre']} ";
}
?>
<h2> Actualización de Registros</h2>
<form name="formulario" method="post">
<select name="selectid">
<option value="-1">Selecione registro</option>
<?php
// Cubrimos dinámicamente el formulario.

// Construimos la sentencia de select SQL
$stmt = $mysqli->prepare("select id,nombre,apellidos from pruebas") or die($mysqli->error);

// No tenemos ningún parámetro a enlazar, así que..
// ejecutamos la sentencia preparada directamente
$stmt->execute() or die($mysqli->error);

// Si es una consulta de select almacenamos el resultado con el método store_result() del objeto $stmt.
$stmt->store_result();

// Número de filas obtenidas, si fuera necesario.
$numfilas=$stmt->num_rows;

// Vinculamos unas variables a los resultados con el método bind_result() de stmt,
// antes de llamar al método fetch() de stmt
$stmt->bind_result($campoid, $camponombre, $campoapellidos);

// Recorremos el recordset:
while ($stmt->fetch())
{
$cadena=sprintf("<option value='%s'%s %s</option>", $campoid, $camponombre, $campoapellidos);
echo $cadena;
}

// Liberamos el resultado.
$stmt-> free_result();

// Cerramos la conexión.

```

```
$mysqli->close();
?>
</select>
<input type="submit" value="Seleccionar"/>
</form>
```

• consultas.php

```
<?php
// Conectarse a la base de datos.
require("conexion.php");

// Preparamos la consulta.
$stmt = $mysqli->prepare("SELECT nombre,apellidos,edad from pruebas") or die($mysqli->error);

// Ejecutamos la consulta preparada.
$stmt->execute();

// Si es una consulta de select almacenamos el resultado.
$stmt->store_result();

// Número de filas obtenidas en $stmt->num_rows
echo "Se han encontrado $stmt->num_rows registros.<br/><br/>";

// Vinculamos unas variables a los resultados con el método bind_result() de stmt,
// antes de llamar al método fetch() de stmt
$stmt->bind_result($nombre, $apellidos, $edad);

// Recorremos el recordset.
echo "<table border='1'>";
echo "<tr><th>Nombre</th><th>Apellidos</th><th>Edad</th></tr>";

while ($stmt->fetch())
{
echo "<tr><td>".$nombre."</td><td>".$apellidos."</td><td>".$edad."</td></tr>";
}

echo "</table>";

// Liberamos el espacio que ocupa ese resultado en memoria.
$stmt -> free_result();

// Cerramos la conexión con el servidor mysql.
$mysqli->close();
?>
```

--Veiga 17:39 8 ene 2013 (CET)