

# UT3 – Acceso a Bases de Datos

# Lenguaje PHP y MySQL



w3schools.com

[http://www.w3schools.com/php/php\\_mysql\\_intro.asp](http://www.w3schools.com/php/php_mysql_intro.asp)

[https://www.tutorialspoint.com/php/php\\_and\\_mysql.htm](https://www.tutorialspoint.com/php/php_and_mysql.htm)

**CFGS- DAW**  
*Juan Espino*

# Bases de Datos

- Las Bases de Datos son **útiles para almacenar y recuperar información** de manera categorizada.
- Facilitan la **inserción, modificación, eliminación y búsqueda** de la información.
- Los datos de una BBDD se almacenan en **tablas**:
  - Una tabla es un conjunto de **datos relacionados**.
  - Los datos se almacenan en **registros** (filas) de la tabla.
  - Cada registro se compone de varias columnas que describen sus **campos** (atributos o propiedades).
  - Cada registro debe ser único en la tabla → debe haber al menos un campo que sea **clave: valor único y no nulo**.



Diagram illustrating the structure of a database table:

The table is labeled "tabla".

It consists of 5 rows (registros) and 5 columns (campos).

The columns are labeled: Cve. cliente, Nombre, Direccion, Ciudad, and Estado.

The data in the table is as follows:

Cve. cliente	Nombre	Direccion	Ciudad	Estado
1	Alfredo Godinez	Fresnillo #47	Veracruz	Veracruz
2	Gabriela Mora	El crospio #81	Guadalajara	Jalisco
3	Alejandra Avalos	Casa Mata #1	Morelia	Michoacan
4	Jaime Quintana	Miraflores #23	Uruapan	Michoacan
5	Carlos Miranda	Bio Bravo #95	Matamoros	Tamaulipas

Annotations:

- A red bracket on the left side of the first column is labeled "registro".
- A red bracket at the bottom of the first column is labeled "campo clave".
- A blue bracket spanning all columns is labeled "campo".
- A red bracket at the top of the table is labeled "tabla".

# Lenguaje PHP: acceso a Bases de Datos

- El lenguaje PHP permite crear aplicaciones web que **accedan a Bases de Datos** alojadas en el propio servidor web o en otro host accesible.

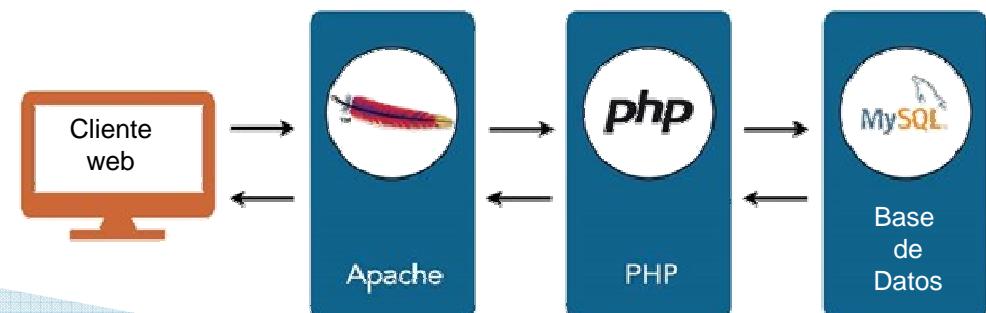


- **MySQL** es el SGDB más usado para proyectos web de envergadura mediana y pequeña:

- Funciona en modo servidor, pudiendo ejecutarse en la misma máquina que el servidor Apache o en otra distinta a la que se tenga acceso vía red.
- Suele integrarse en los entornos LAMP.
- Es fácil, fiable, rápido, gratuito.
- Compatible con SQL standard (Structured Query Language).



Solicitud a una Web que usa una BBDD:

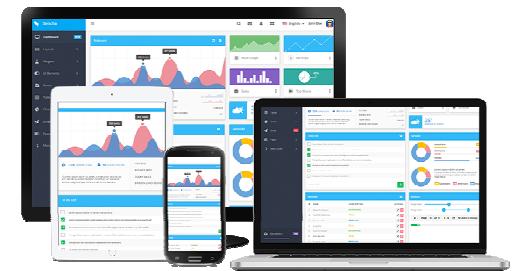
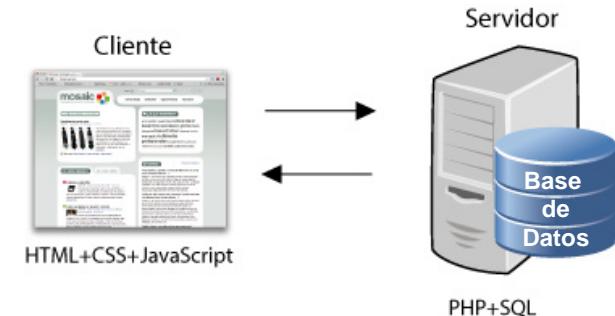


# Lenguaje PHP y MySQL

- PHP y MySQL permiten crear aplicaciones web dinámicas y que son multiplataforma (cross-platform):

Pueden usarse sin ningún inconveniente en distintas plataformas de hardware y SSOO. P.e. podríamos desarrollar en Windows y ejecutarlo en Linux/debian.

- PHP dispone de un driver (conector o librería) que permite acceder a BBDDs almacenadas en un servidor SGBD MySQL (puerto 3306/TCP). El servidor MySQL puede ejecutarse sobre el mismo equipo que el servidor web (localhost) o en cualquier otro host que esté accesible por medio de la red.

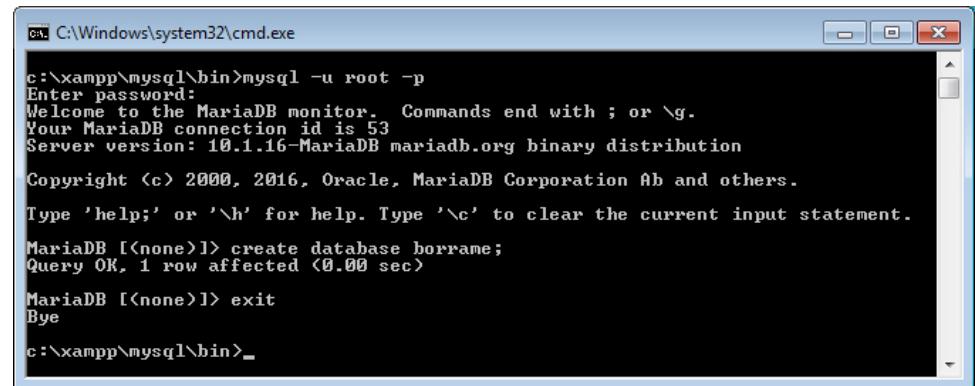


# MySQL: creación de y uso de BBDD

- Existen varias formas para crear y modificar una BBDD en MySQL:

- Mediante sentencias SQL escritas como comandos en el terminal/consola:

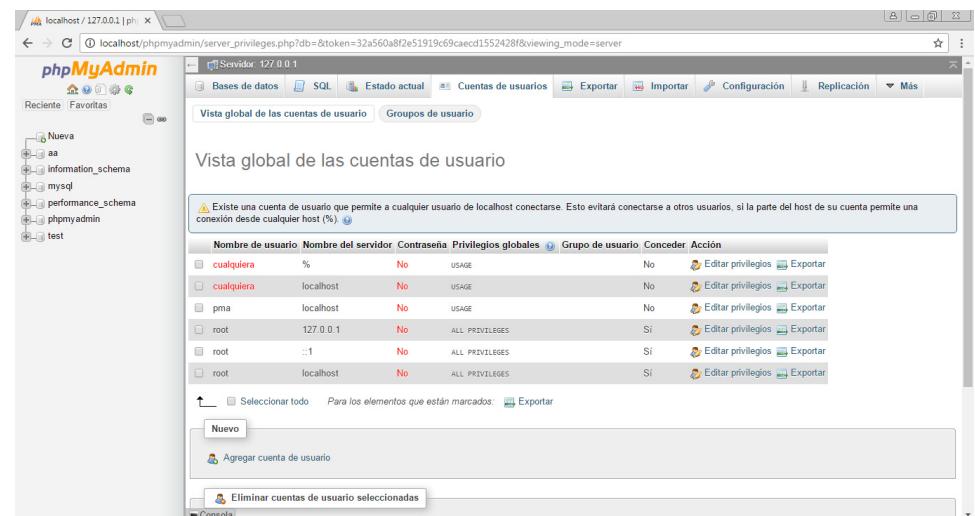
```
c:\>cd C:\xampp\mysql\bin  
c:\>mysql -u root -p  
      > Sentencias_SQL;
```



```
C:\Windows\system32\cmd.exe  
c:\>xampp\mysql\bin>mysql -u root -p  
Enter password:  
Welcome to the MariaDB monitor. Commands end with ; or \g.  
Your MariaDB connection id is 53  
Server version: 10.1.16-MariaDB mariadb.org binary distribution  
Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
MariaDB [(none)]> create database borrame;  
Query OK, 1 row affected (0.00 sec)  
MariaDB [(none)]> exit  
Bye  
c:\>xampp\mysql\bin>_
```

- Mediante la herramienta web phpMyAdmin:

<http://localhost/phpmyadmin>

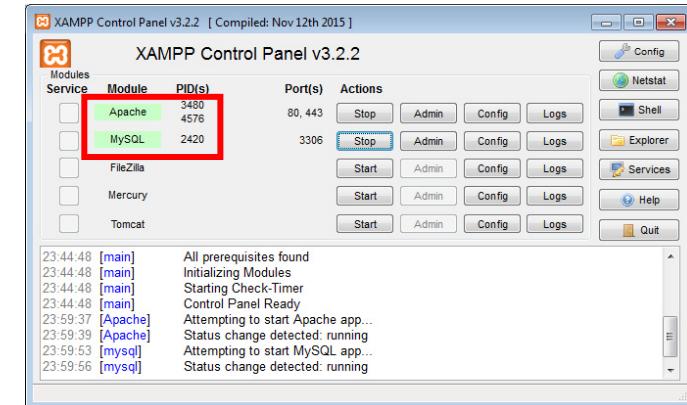


Nombre de usuario	Nombre del servidor	Contraseña	Privilegios globales	Grupo de usuario	Conceder	Acción
cuálquiera	%	No	USAGE	No		
cuálquiera	localhost	No	USAGE	No		
pma	localhost	No	USAGE	No		
root	127.0.0.1	No	ALL PRIVILEGES	Si		
root	::1	No	ALL PRIVILEGES	Si		
root	localhost	No	ALL PRIVILEGES	Si		

La instalación de MySQL que viene con XAMPP no cuenta con una **password para el usuario root** del servidor MySQL.  
Es conveniente asignarle una con phpMyAdmin

# MySQL: creación de y uso de BBDD

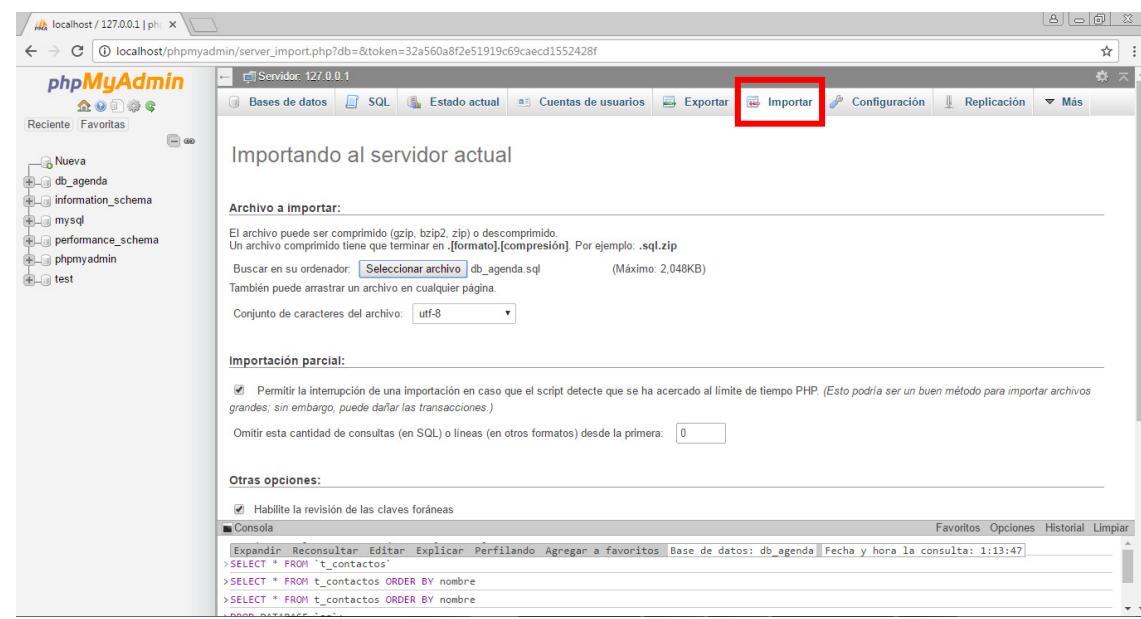
- Previamente se debe **arrancar el servidor MySQL** con el panel de control de XAMPP:



- Podemos crear la BBDD editando un archivo de texto **.sql** que contenga las sentencias SQL a modo de *script SQL*. Luego se debe **importar** el archivo desde:

- la consola mediante la orden de **mysql**:

```
source ruta_archivo.sql
```



- el **phpMyAdmin**: seleccionando el archivo **.sql** a subir:

# MySQL: creación de BBDD

- **Ejemplo:** crear una BD simple en MySQL que cuente con una tabla para almacenar contactos telefónicos: db\_agenda y t\_contactos



```
DROP DATABASE IF EXISTS db_agenda;
CREATE DATABASE db_agenda;
USE db_agenda;

CREATE TABLE t_contactos (
    nombre VARCHAR(20) NOT NULL,
    telefono VARCHAR(9) NOT NULL,
    PRIMARY KEY (nombre)
);

INSERT INTO t_contactos (nombre, telefono) VALUES ('Luisa', 547813477);
INSERT INTO t_contactos (nombre, telefono) VALUES ('Antonio', 628447634);
INSERT INTO t_contactos (nombre, telefono) VALUES ('Ana', 873652114);
INSERT INTO t_contactos (nombre, telefono) VALUES ('Jose', 784589647);
INSERT INTO t_contactos (nombre, telefono) VALUES ('Daniel', 678147994);

SELECT * FROM t_contactos ORDER BY nombre;
```

Contenido del archivo:  
C:\xampp\htdocs\db\_agenda.sql



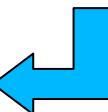
```
C:\Windows\system32\cmd.exe - mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 120
Server version: 10.1.16-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2016, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
MariaDB [(none)]> source c:\xampp\htdocs\db_agenda.sql
Query OK, 1 row affected <0.05 sec>

Query OK, 1 row affected <0.00 sec>
Database changed
Query OK, 0 rows affected <0.06 sec>
Query OK, 1 row affected <0.00 sec>
5 rows in set <0.00 sec>

MariaDB [db_agenda]> exit
```



The screenshot shows the phpMyAdmin interface for the 'db\_agenda' database. The left sidebar lists databases like 'information\_schema', 'mysql', 'performance\_schema', 'phpmyadmin', and 'test'. The 't\_contactos' table is selected under the 'db\_agenda' database. The main area displays the table structure with columns 'nombre' and 'telefono'. Below the structure, a list of 5 rows is shown, each with a 'Copiar' (Copy) link. The data is as follows:

	nombre	telefono
1	Ana	873652114
2	Antonio	628447634
3	Daniel	678147994
4	Jose	784589647
5	Luisa	547813477

## MySQL: acceso a BBDD con PHP

- En el entorno XAMPP se ejecutan juntos el servidor de BBDDs MySQL y el servidor web Apache (capas de Negocio y Datos juntas).

- Las BBDDs creadas/importadas en MySQL se guardan como subdirectorios de:

**C:\xampp\mysql\data**

- Cada subdirectorio contiene los archivos de la BBDD:

- .frm definición de las tablas tanto MyISAM como InnoDB
- .MYD datos de las tablas MyISAM (mecanismo de almacenamiento antiguo)
- .MYI índices de las tablas MyISAM (mecanismo de almacenamiento antiguo)
- .idb espacio de tablas InnoDB con los datos (mecanismo de almacenamiento actual)
- .trn tablas con disparadores



# MySQL: acceso a BBDD con PHP

- **Esquema de uso de BBDDs mediante el lenguaje PHP:**

- 1º. **Conectar** con PHP con una Base de Datos existente:

```
$conexion=new mysqli($URLservidor,$usuarioMySQL,$password,$nombreBD);
```

- 2º. **Operaciones SQL:** crear, insertar, modificar, borrar, buscar

```
$conexion->query($consultasSQL);
```

- 3º. **Desconectar** con la BBDD:

```
$conexion->close();
```

---

```
1º. <?php $servidor = "localhost"; $usuario = "root"; $clave; $nombreBD;
      $conexion = new mysqli($servidor, $usuario, $clave, $nombreBD);
      if ($conexion->connect_error)
          die("Error en la conexión: ".$conexion->connect_error);
      $sql = "CONSULTA EN LENGUAJE SQL";
      if ($conexion->query($sql) === TRUE)
          echo "<span style=\"color:green;\">Operación realizada con éxito</span>";
      else
          echo "<span style=\"color:red;\">Error: ".$conexion->error."</span>";
  2º. $conexion->close();
  3º. ?>
```

# Lenguaje PHP: operaciones con BBDD MySQL

- Consultas SQL para **operaciones con Bases de Datos** (conjunto de tablas):

```
CREATE DATABASE [ IF NOT EXISTS ] base_de_datos;
```

```
DROP DATABASE [ IF EXISTS ] base_de_datos;
```

```
USE base_de_datos;
```



- Consultas SQL para **operaciones con Tablas completas**:

```
CREATE TABLE [ IF NOT EXISTS ] tabla (campo1, campo2, ... );
```

Siendo campo = *Nombre Tipo Modificadores*

Tipos de datos más comunes: INT, VARCHAR(n), TIMESTAMP, ENUM('a','b','c')

Modificadores posibles: NOT NULL, AUTO\_INCREMENT, PRIMARY KEY, DEFAULT valor

```
DROP TABLE [ IF EXISTS ] tabla;
```

```
ALTER TABLE tabla ...;
```

---

```
$consultasSQL = "CREATE | DROP | INSERT | DELETE | UPDATE ...";  
if ($conexion->query($consultasSQL) === TRUE) {  
    echo "OK";  
} else {  
    echo "Error: ".$conexion->error;  
}
```



Tutorial del lenguaje SQL:

<http://www.w3schools.com/sql/default.asp>  
<https://mariadb.com/kb/es/basic-sql-statements>

# Lenguaje PHP: operaciones con BBDD MySQL

- Consultas SQL para **operaciones con Registros** (filas de una tabla):



**INSERT INTO** tabla (campo1, campo2,...) **VALUES** (valor1, valor2,...);

\$conexion->insert\_id; indica el valor del último campo auto incremental que se haya insertado.



**DELETE FROM** tabla **WHERE** campo1=valor [AND|OR campo2=...];



**UPDATE** tabla **SET** campo1=valor[,campo2=...] **WHERE** campo1=valor [AND|OR campo2=...];



**SELECT** campo1,campo2,... | \* **FROM** tabla/s **WHERE** campo1=valor [AND|OR campo2=...];

```
$consultasQL = "SELECT ...";  
  
if ($resultado = $conexion->query($consultasQL)) {  
    if ($resultado->num_rows > 0) {  
        while ($registro = $resultado->fetch_assoc()) {  
            echo "campo1:".$registro["campo1"].",campo2:".$registro["campo2"]."<br>";  
        }  
    }  
    $resultado->close();  
}
```

Las strings en SQL van entre comillas simples;  
mientras que los números y el valor NULL no.

	Nombre	Naturaleza	Centro
5			
6	Adrastea	Satélite	Júpiter
7	Amaltea	Satélite	Júpiter
8	Ananke	Satélite	Júpiter
9	Ariel	Satélite	Urano
10	Atlas	Satélite	Saturno

# MySQL: acceso a BBDD con PHP

- **Ejemplo:** aplicación web (sin AJAX) que gestiona una **agenda de contactos** telefónicos online que se almacena en la BD "db\_agenda" existente en el server web/MySQL:



```
<html>
<head><title>Agenda de contactos</title>
    <style>
        .titulo {color:#000088; font-family:verdana; font-size:18px; font-weight:bold;}
        .texto {color:#000088; font-family:verdana; font-size:14px;}
    </style>
</head>
<body bgcolor="#11AAEE">
    <p class="titulo">Base de Datos de contactos:</p>
    <form method="post" action="" class="texto">
        <table>
            <tr> <td>Nombre:</td>
            <td colspan="2"><input type="text" name="nombre" size="20" maxlength="20"></td>
        </tr>
            <tr> <td>Teléfono:</td>
            <td colspan="2"><input type="number" name="telefono" size="20"></td>
        </tr>
            <tr height="10"></tr>
            <tr> <td><input type="submit" name="buscar" value="BUSCAR"></td>
                <td><input type="submit" name="anadir" value="AÑADIR"></td>
                <td><input type="submit" name="eliminar" value="ELIMINAR"></td>
            </tr>
        </table>
    </form>
    <br><hr align="left" width="270">
```

archivo →  
agenda.php

La captura de pantalla muestra una interfaz web para gestionar una agenda de contactos. En la parte superior, hay un formulario para buscar, añadir o eliminar contactos. La tabla principal muestra los siguientes datos:

Nombre	Teléfono
Ana	873652114
Antonio	628447631
Daniel	678147994
Jose	784589647
Luisa	547813477

A continuación del formulario, se indica que se han encontrado 5 contactos.

La web deberá mostrar un error si:

- El nombre a buscar no se encuentra
- El nombre a añadir ya existe
- El nombre a eliminar no existe

# MySQL: acceso a BBDD con PHP

```
<?php
    function existeContacto ($nombre) {
        global $conexion;
        $consultaSQL = "SELECT * FROM t_contactos WHERE nombre = '". $nombre . "' ";
        $resultado = $conexion->query($consultaSQL);
        if ($resultado->num_rows > 0)    return true;
        else    return false;
    }

    function buscarContacto ($nombre) { // muestra una tabla con los contactos encontrados
        global $conexion;
        if ($nombre == "") {
            $consultaSQL = "SELECT * FROM t_contactos";
        }
        else {
            $consultaSQL = "SELECT * FROM t_contactos WHERE nombre LIKE '". $nombre . "%'";
        }
        $encontrados = 0;
        if ($resultado = $conexion->query($consultaSQL)) {
            if ($resultado->num_rows > 0) {
                echo "<table class=\"texto\" cellpadding=\"10\" border=\"1\""
                    style="border-collapse:collapse;" "><tr><th>Nombre</th><th>Teléfono</th></tr>";
                while ($registro = $resultado->fetch_assoc()) {
                    echo "<tr><td width=\"100\">". $registro["nombre"] . "</td><td"
                        width="100">". $registro["telefono"] . "</td></tr>";
                }
                echo "</table>";
                echo "<p class=\"texto\">". $resultado->num_rows . " contactos encontrados en la agenda</p>";
                $encontrados = $resultado->num_rows;
            }
            $resultado->close();
        }
        return $encontrados;
    }
}
```

↑ continuación del archivo **agenda.php**

# MySQL: acceso a BBDD con PHP

```
if ($_SERVER['REQUEST_METHOD'] == "POST") {
    $servidor = "localhost";    $usuario = "root";      $clave = "";      $baseDatos = "db_agenda";
    $conexion = new mysqli($servidor, $usuario, $clave, $baseDatos);
    if ($conexion->connect_error)
        die("Conexión fallida: ".$conexion->connect_error);

    if (isset($_POST['buscar'])) {
        $nombre = $_POST['nombre'];
        if (buscarContacto($nombre) == 0)
            echo "<p style=\"color:red;\">No se encuentra en la agenda el nombre indicado</p>";
    }

    if (isset($_POST['anadir'])) {
        $nombre = $_POST['nombre'];      $telefono = $_POST['telefono'];
        if ($nombre == "" || $telefono == "") {
            echo "<p style=\"color:red;\">Indique el nombre y el teléfono del contacto a añadir</p>";
            buscarContacto("");
        } else {
            if (existeContacto($nombre)) {
                echo "<p style=\"color:red;\">Ya existe en la agenda el nombre indicado</p>";
            } else {
                $consultaSQL = "INSERT INTO t_contactos (nombre, telefono) VALUES
                    ('".$nombre."','".$.$telefono."');";
                if ($conexion->query($consultaSQL) === TRUE)
                    echo "<p style=\"color:green;\">Contacto añadido correctamente a la agenda</p>";
                else
                    echo "<p style=\"color:red;\">".$conexion->error."</p>";
            }
            buscarContacto("");
        }
    }
}
```



↑ continuación del archivo agenda.php

# MySQL: acceso a BBDD con PHP



```
if (isset($_POST['eliminar'])) {
    $nombre = $_POST['nombre'];
    if ($nombre == "") {
        echo "<p style=\"color:red;\">Debe indicar el nombre del contacto a eliminar</p>";
        buscarContacto("");
    } else {
        $consultaSQL = "DELETE FROM t_contactos WHERE nombre='".$nombre."'";
        if (existeContacto($nombre)) {
            if ($conexion->query($consultaSQL) === TRUE) {
                echo "<p style=\"color:green;\">Contacto eliminado de la agenda</p>";
            } else {
                echo "<p style=\"color:red;\">".$conexion->error."</p>";
            }
        } else {
            echo "<p style=\"color:red;\">No existe en la agenda el nombre indicado</p>";
        }
        buscarContacto("");
    }
}
$conexion->close();
}

?>
</body>
</html>
```

↑ continuación del archivo **agenda.php**



# MySQL: importando datos a una BBDD

- **Actividad:** añadir a la aplicación web de la agenda de contactos telefónicos las siguientes funcionalidades:

- botón para **actualizar** un contacto.
- botón para **importar** contactos a partir de un archivo y que *se añadan* a los que ya existen en la BD. Opcionalmente, el usuario podrá elegir si desea que los contactos del archivo sobreesciban a los contactos existentes (borrándolos todos y para añadir solo los nuevos).

Añade **AJAX** para refrescar de la web solo la tabla de resultados tras cada operación:



The screenshot shows a web browser window titled "Agenda de contactos" with the URL "localhost/agenda/agenda.php". The page has a blue header with the title "Base de Datos de Contactos:". Below the header is a form with fields for "Nombre" and "Teléfono", each accompanied by a placeholder icon of a person with a list. Underneath the form are four buttons: "BUSCAR" (magnifying glass), "AÑADIR" (green plus), "ELIMINAR" (red minus), and "ACTUALIZAR" (refresh). To the right of these buttons are two checkboxes: "Seleccionar archivo" (with an "IMPORTAR" button) and "Sobreescibir". Below the form is a table with columns "Nombre" and "Teléfono", containing five rows of data. At the bottom of the table, a message says "5 contactos encontrados en la agenda".

Nombre	Teléfono
Ana	873652114
Antonio	628457631
Daniel	678147994
Jose	784589647
Luisa	547813477

5 contactos encontrados en la agenda

# MySQL: importando datos a una BBDD

- Para importar los contactos, el usuario debe subir al server un fichero de texto CSV (Comma Separated Values) generado a partir de una hoja Excel que contenga los contactos.
- Al importar, el script PHP accederá a la tabla de la BD y le irá insertando los contactos que encuentre en el fichero CSV subido:



Si existen contactos repetidos no serán insertados ni dará error; ya que el campo Nombre es clave.

Microsoft Excel - agenda

	A	B	C	D	E	F	G	H	I	J	K
1	Carla	487549127									
2	Jorge	665148965									
3	Marcos	820469177									
4	Rosa	662547730									
5	Fran	758796364									
6	Marta	667544179									
7	Alberto	627841794									
8	Margarita	840517384									
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											
20											
21											
22											

archivo  
**contactos.xls**

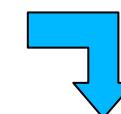
Guardar como

Guardar en: agenda

Nombre: agenda

Guardar como tipo: CSV (delimitado por comas)

Menú Archivo → Guardar como:  
tipo CSV (delimitado por comas)



agenda: Bloc de notas

Archivo Edición Formato Ver Ayuda

Carla;487549127
Jorge;665148965
Marcos;820469177
Rosa;662547730
Fran;758796364
Marta;667544179
Alberto;627841794
Margarita;840517384

archivo  
**contactos.csv**

# MySQL: acceso a BBDD con PHP

```
<!DOCTYPE html> <html>
<head>
    <title>Agenda de contactos</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <style>
        .titulo { color:#000088; font-family:verdana; font-size:18px; font-weight:bold; }
        .texto { color:#000055; font-family:verdana; font-size:16px; }
        .boton { height:36px; width:110px; font-family:Arial; font-size:10px; line-height:36px; }
        .caja { font-family:Arial; font-size:14px; height:24px; width:200px; }
        .error { color:red; font-family:verdana; font-size:15px; }
        .ok { color:green; font-family:verdana; font-size:15px; }
    </style>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"> </script>
    <script>
        function ajax (accion, nombre, telefono) {
            var xhr = new XMLHttpRequest();
            xhr.onreadystatechange = function() {
                if (this.readyState == 4 && this.status == 200) {
                    document.getElementById('resultado').innerHTML = this.responseText;
                }
            };
            xhr.open("POST", "agenda.php", true);
            xhr.setRequestHeader("Content-type", "application/x-www-form-urlencoded");
            var parametros;
            if (accion == "importar")
                parametros = "accion=" + accion + "&sobreescribir=" + nombre;
            else
                parametros = "accion=" + accion + "&nombre=" + nombre + "&telefono=" + telefono;
            xhr.send(parametros);
        }

        function buscarContacto (nombre) { ajax("buscar", nombre, ""); }
    </script>
</head>
<body>
    <div>
        <h1>Agenda de contactos</h1>
        <div>
            <input type="text" value="Nombre" />
            <input type="button" value="Buscar" />
        </div>
        <table border="1">
            <thead>
                <tr>
                    <th>Nombre</th>
                    <th>Teléfono</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td>Juan Pérez</td>
                    <td>987654321</td>
                </tr>
                <tr>
                    <td>Ana Gómez</td>
                    <td>987654320</td>
                </tr>
                <tr>
                    <td>Pedro Martínez</td>
                    <td>987654322</td>
                </tr>
            </tbody>
        </table>
    </div>
</body>
</html>
```

## • Solución:

archivo →  
agenda.html

# MySQL: acceso a BBDD con PHP

```
function anadirContacto (nombre, telefono) { ajax("anadir", nombre, telefono); }
function eliminarContacto (nombre) { ajax("eliminar", nombre, ""); }
function actualizarContacto (nombre, telefono) { ajax("actualizar", nombre, telefono); }
function importarContactos (fichero, sobreescribir) {
    var data = new FormData();
    data.append('accion', 'importar');
    data.append('fichero', fichero.files[0]);
    data.append('sobreescribir', sobreescribir);
    $.ajax({
        url:"agenda.php", type:"POST", contentType:false, data:data, processData:false,
        cache:true, success: function(data){
            $("#resultado").html(data);
        }
    });
}
</script>
</head>
<body bgcolor="#11AAEE">
<p class="titulo">Base de Datos de Contactos:</p>
<table class="texto" width="520" border="0">
    <tr>
        <td rowspan="2" width="140"></td>
        <td>Nombre:</td>
        <td><input type="text" name="nombre" id="nombre" size="22" maxlength="20" class="caja"></td>
    </tr>
    <tr>
        <td>Teléfono:</td>
        <td><input type="number" name="telefono" id="telefono" size="22" class="caja" max="9999999999"></td>
    </tr>
</table> <br>
```

↑ continuación del archivo  
agenda.html

# MySQL: acceso a BBDD con PHP

```
<table width="450" border="0">
<tr>
    <td><button type="submit" name="buscar" class="boton" value="BUSCAR"
onClick="buscarContacto(document.getElementById('nombre').value);">
        &nbsp;BUSCAR</button></td>
    <td><button type="submit" name="anadir" class="boton" value="ANADIR"
onClick="anadirContacto(document.getElementById('nombre').value, document.getElementById('telefono').value);">
        &nbsp;AÑADIR</button></td>
    <td><button type="submit" name="eliminar" class="boton" value="ELIMINAR"
onClick="eliminarContacto(document.getElementById('nombre').value);">
        &nbsp;ELIMINAR</button></td>
    <td><button type="submit" name="actualizar" class="boton" value="ACTUALIZAR"
onClick="actualizarContacto(document.getElementById('nombre').value, document.getElementById('telefono').value);">
        &nbsp;ACTUALIZAR</button></td>
</tr>
<tr>
    <td colspan="2" rowspan="2"></td>
    <td rowspan="2">
        <button type="submit" name="importar" class="boton" value="IMPORTAR"
onClick="importarContactos(document.getElementById('fichero'), document.getElementById('sobreescibir').checked);">
            &nbsp;IMPORTAR</button>
        <td><input type="file" name="fichero" id="fichero" class="boton" value="IMPORTAR"></td>
    </tr>
    <tr><td><input type="checkbox" name="sobreescibir" id="sobreescibir" value="sobreescibir">
        <span class="boton">Sobreescibir</span></td></tr>
</table>
<hr width="455" align="left"> <br>
<div id="resultado" style="max-width:450px;"> </div>
</body> </html>
```

↑ continuación del archivo  
agenda.html

# MySQL: acceso a BBDD con PHP

```
<?php
    require("funcionesBD.php");
    if (isset($_REQUEST["accion"])) {
        $accion = $_REQUEST["accion"];
        $servidor = "localhost"; $usuario = "root"; $clave = ""; $baseDatos = "db_agenda";
        $conexion = new mysqli($servidor, $usuario, $clave, $baseDatos);
        if ($conexion->connect_error)
            die("Conexión fallida: ".$conexion->connect_error);

        if ($accion == "buscar") {
            $nombre = $_REQUEST['nombre'];
            $contactosEncontrados = buscarContactos($nombre);
            if (count($contactosEncontrados) == 0) {
                echo "<p class=\"error\">No se encuentra en la agenda el nombre indicado</p>";
            } else {
                mostrarContactos($contactosEncontrados);
            }
        }

        if ($accion == "anadir") {
            $nombre = $_REQUEST['nombre']; $telefono = $_REQUEST['telefono'];
            if ($nombre == "" || $telefono == "") {
                echo "<p class=\"error\">Indique el nombre y teléfono del contacto a añadir</p>";
            } else {
                if (!is_numeric($telefono))
                    echo "<p class=\"error\">Indique un teléfono correcto para el contacto a añadir</p>";
                else
                    anadirContacto($nombre, $telefono);
            }
            $contactosEncontrados = buscarContactos("TODOS");
            mostrarContactos($contactosEncontrados);
        }
    }
}
```



↑ archivo  
agenda.php

# MySQL: acceso a BBDD con PHP

```
if ($accion == "eliminar") {  
    $nombre = $_REQUEST['nombre'];  
    if ($nombre == "") { echo "<p class=\"error\">Indique el nombre del contacto a eliminar</p>";  
    } else { eliminarContacto($nombre); }  
    $contactosEncontrados=buscarContactos("TODOS"); mostrarContactos($contactosEncontrados);  
}  
  
if ($accion == "actualizar") {  
    $nombre = $_REQUEST['nombre']; $telefono = $_REQUEST['telefono'];  
    if ($nombre == "" || $telefono == "") {  
        echo "<p class=\"error\">Debe indicar el nombre y el teléfono del contacto a actualizar</p>";  
    } else {  
        if (!is_numeric($telefono)) {  
            echo "<p class=\"error\">Indique un teléfono correcto para el contacto</p>";  
        } else { actualizarContacto($nombre, $telefono); }  
    }  
    $contactosEncontrados=buscarContactos("TODOS"); mostrarContactos($contactosEncontrados);  
}  
  
if ($accion == "importar") {  
    if (isset($_FILES['fichero'])) {  
        $fichero = $_FILES['fichero']; $nombre=$fichero['name']; $tipo=$fichero['type'];  
        $sobreescribir = $_REQUEST['sobreescribir'];  
        /* if ($tipo != 'application/vnd.ms-excel') {  
            echo "<p class=\"error\">Debe indicar un archivo CSV de contactos</p>";  
        } else { */ importarContactos($fichero, $sobreescribir); /* } */  
    } else { echo "<p class=\"error\">Debe indicar un archivo CSV de contactos</p>"; }  
    $contactosEncontrados=buscarContactos("TODOS"); mostrarContactos($contactosEncontrados);  
}  
$conexion->close();  
}  
?>
```



↑ continuación del archivo  
agenda.php

# MySQL: acceso a BBDD con PHP

```
<?php

function mostrarContactos ($arrayContactos) {    // crea una tabla con los contactos recibidos
    echo "<table class=\"texto\" cellpadding=\"10\" border=\"1\" style=\"border-collapse:collapse; width=\"450\">";
    echo "<tr><th>Nombre</th><th>Teléfono</th></tr>";
    foreach ($arrayContactos as $nombre=>$telefono) {
        echo "<tr><td width=\"100\">".$nombre."</td><td width=\"100\">".$telefono."</td></tr>";
    }
    echo "</table>";
    echo "<p class=\"texto\">".count($arrayContactos)." contactos encontrados en la agenda</p>";
}

function existeContacto ($nombre) {
    global $conexion;
    $consultaSQL = "SELECT * FROM t_contactos WHERE nombre = '".$nombre."'";
    $resultado = $conexion->query($consultaSQL);
    if ($resultado->num_rows > 0) return true;
    else return false;
}

function buscarContactos ($nombre) {    // devuelve un array asociativo con los contactos encontrados
    global $conexion;    $contactosEncontrados = array();
    if ($nombre == "" || $nombre == "TODOS") $consultaSQL = "SELECT * FROM t_contactos";
    else $consultaSQL = "SELECT * FROM t_contactos WHERE nombre LIKE '".$nombre."%'";
    if ($resultado = $conexion->query($consultaSQL)) {
        if ($resultado->num_rows > 0) {
            while ($registro = $resultado->fetch_assoc())
                $contactosEncontrados[$registro["nombre"]] = $registro["telefono"];
        }
        $resultado->close();
    }
    return $contactosEncontrados;
}
```



↑ archivo  
funcionesBD.php

# MySQL: acceso a BBDD con PHP

```
function anadirContacto ($nombre, $telefono) {  
    global $conexion;  
    if (existeContacto($nombre)) {  
        echo "<p class=\"error\">Ya existe en la agenda el nombre indicado</p>";  
    } else {  
        $consultaSQL =  
            "INSERT INTO t_contactos (nombre, telefono) VALUES ('".$nombre."','".$telefono."');"  
        if ($conexion->query($consultaSQL) === TRUE) {  
            echo "<p class=\"ok\">Contacto añadido correctamente a la agenda</p>";  
        } else {  
            echo "<p class=\"error\">".$conexion->error."</p>";  
        }  
    }  
}  
  
function eliminarContacto ($nombre) {  
    global $conexion;  
    $consultaSQL = "DELETE FROM t_contactos WHERE nombre='".$nombre."'";  
    if (existeContacto($nombre)) {  
        if ($conexion->query($consultaSQL) === TRUE) {  
            echo "<p class=\"ok\">Contacto eliminado correctamente de la agenda</p>";  
        } else {  
            echo "<p class=\"error\">".$conexion->error."</p>";  
        }  
    } else {  
        echo "<p class=\"error\">No existe en la agenda el nombre indicado</p>";  
    }  
}
```



↑ continuación del archivo  
[funcionesBD.php](#)

# MySQL: acceso a BBDD con PHP

```
function actualizarContacto ($nombre, $telefono) {    global $conexion;
    if (!existeContacto($nombre)) {
        echo "<p class=\"error\">No existe el nombre del contacto indicado</p>";
    } else {
        $consultaSQL =
            "UPDATE t_contactos SET telefono=". $telefono ." WHERE nombre='". $nombre . "' ";
        if ($conexion->query($consultaSQL) === TRUE) {
            echo "<p class=\"ok\">Contacto actualizado correctamente en la agenda</p>";
        } else { echo "<p class=\"error\">". $conexion->error. "</p>"; }
    }
}

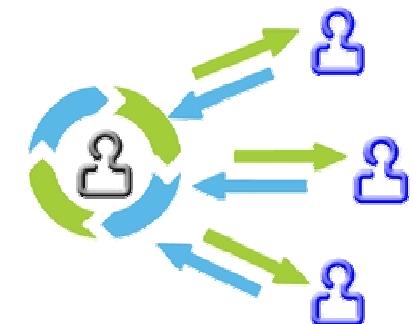
function importarContactos ($fichero, $sobreescibir) {    global $conexion;
    $nombreTmp = $fichero['tmp_name'];    $nombreDef = "./". $fichero['name'];
    $resultado = copy($nombreTmp, $nombreDef);    if ($resultado == 0) unlink($nombreTmp);
    if ($sobreescibir == "true") {    // Borra todos los registros de la tabla:
        $consultaSQL = "DELETE FROM t_contactos";    $conexion->query($consultaSQL);
    }
    $ficheroCSV = fopen($nombreDef, "r");
    while (!feof($ficheroCSV)) {
        $contacto = fgets($ficheroCSV);
        if ($contacto != "") {
            $valores=explode(";", $contacto); $nombre=$valores[0]; $telefono=$valores[1];
            $consultaSQL =
                "INSERT INTO t_contactos (nombre, telefono) VALUES ('". $nombre . "','" . $telefono . "' )";
            $conexion->query($consultaSQL);
        }
    }
    fclose($ficheroCSV);
    echo "<p class=\"ok\">Contactos importados correctamente a la agenda</p>";
}
```

?>

↑ continuación del archivo  
**funcionesBD.php**

# MySQL: acceso a BBDD con PHP

- **Ejemplo:** aplicación web para la gestión y publicación de las **notas de una única asignatura**. Podrán iniciar sesión dos tipos de usuarios:
  - Los usuarios de tipo **Alumno** podrán:
    - Consultar la nota que tienen en la asignatura (sólo verán la suya).
  - Los usuarios de tipo **Profesor** podrán:
    - Añadir y eliminar otros usuarios (de tipo Alumno y de tipo Profesor) de la asignatura (de la aplicación).
    - Consultar y modificar las notas de la asignatura de todos los Alumnos.
- La capa de **datos** se almacena en una BBDD MySQL.
- La web usa **AJAX** para el acceso a la BBDD → mejorar las prestaciones y la amigabilidad de la aplicación.
- Cada módulo de la aplicación (usuarios y notas) se implementa mediante dos archivos: un documento HTML (**presentación**) y uno PHP (**lógica**).



# MySQL: acceso a BBDD con PHP

## archivo a importar db\_notas.sql

```
CREATE TABLE t_usuarios (
    usuario VARCHAR(10) NOT NULL,
    clave VARCHAR(10) NOT NULL,
    tipo enum('Profesor', 'Alumno'),
    PRIMARY KEY (usuario)          /* no puede haber dos usuarios con el mismo nombre */
);

CREATE TABLE t_notas (
    alumno VARCHAR(10) NOT NULL,    /* es el campo usuario de la tabla t_usuarios */
    nota INT(2) UNSIGNED,
    PRIMARY KEY (alumno),           /* un alumno no puede tener más de una nota */
    FOREIGN KEY (alumno) REFERENCES t_usuarios(usuario) /* obliga a insertar solo
                                                           nombres de alumno que sean
                                                           usuarios existentes en
                                                           t_usuarios */
        ON DELETE CASCADE /* si se borra un usuario de
                           t_usuarios se borran sus notas
                           como alumno */
        ON UPDATE CASCADE /* si se actualiza el nombre
                           de un usuario de t_usuarios se
                           actualiza su nombre de alumno */
);
```



# MySQL: acceso a BBDD con PHP

## archivo login.php



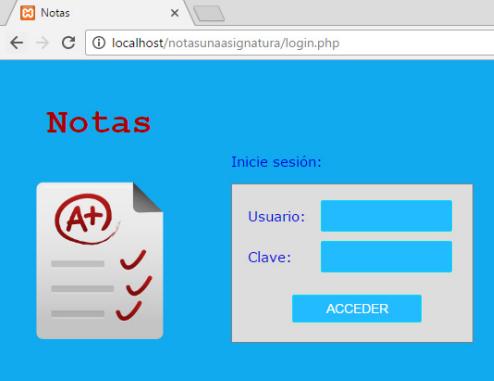
Notas

Inicie sesión:

Usuario:

Clave:

ACCEDER



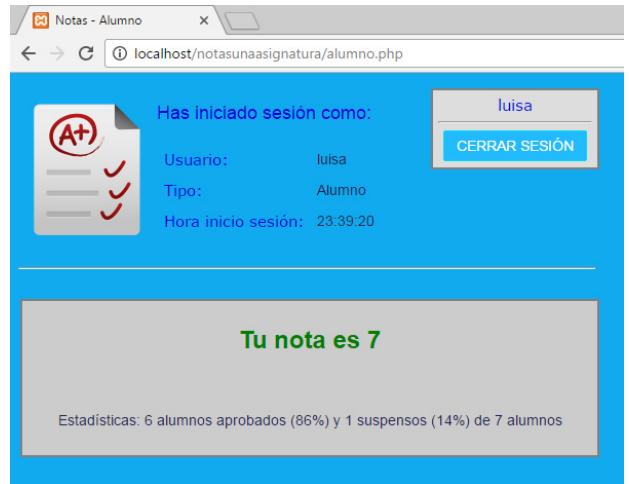
Has iniciado sesión como: luisa

Usuario: luisa  
Tipo: Alumno  
Hora inicio sesión: 23:39:20

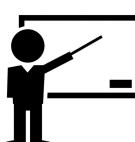
CERRAR SESIÓN

Tu nota es 7

Estadísticas: 6 alumnos aprobados (66%) y 1 suspensos (14%) de 7 alumnos



## archivo profesor.php



Has iniciado sesión como: pro

Usuario: pro  
Tipo: Profesor  
Hora inicio sesión: 23:49:21

CERRAR SESIÓN

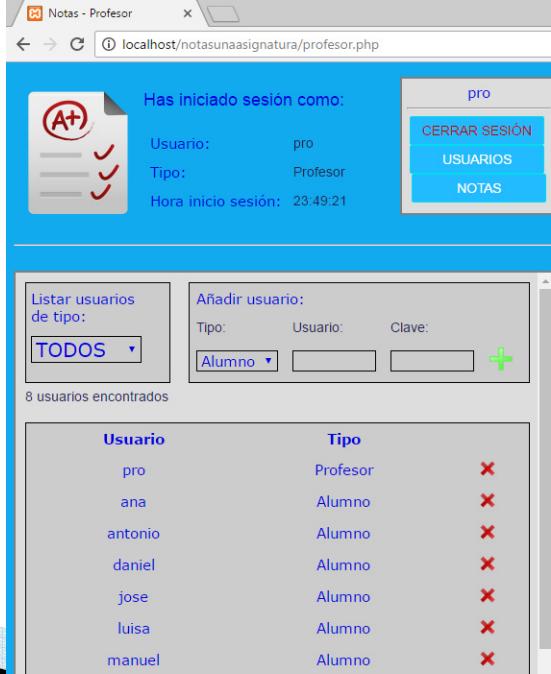
USUARIOS NOTAS

Listar usuarios de tipo: TODOS

Añadir usuario: Tipo: Usuario: Clave: +

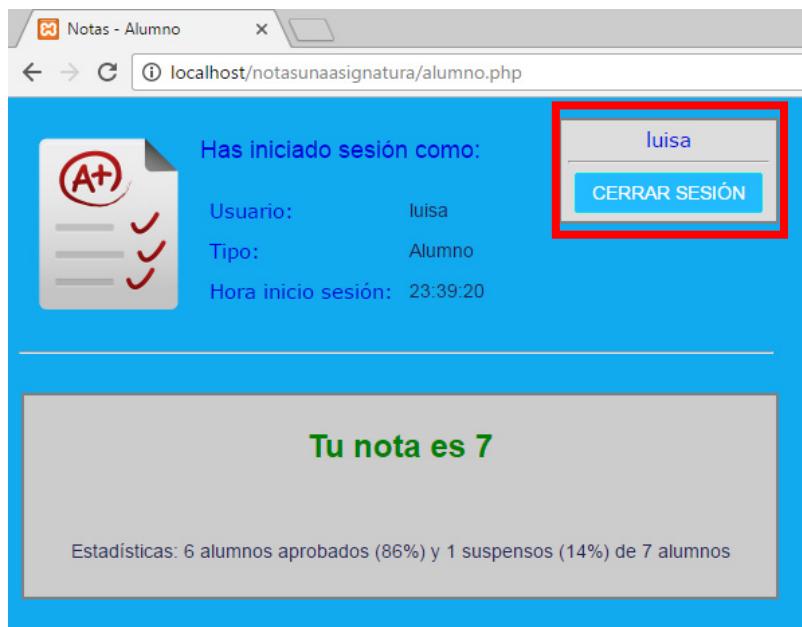
8 usuarios encontrados

Usuario	Tipo	
pro	Profesor	X
ana	Alumno	X
antonio	Alumno	X
daniel	Alumno	X
jose	Alumno	X
luisa	Alumno	X
manuel	Alumno	X



## archivo alumno.php

# MySQL: acceso a BBDD con PHP



archivo [alumno.php](#)

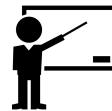


Esta interfaz permite al usuario de tipo Alumno:

- Consultar la nota de la asignatura (sólo la suya).
- Consultar las estadísticas de la asignatura: porcentaje de aprobados, de suspensos y total de alumnos.
- Cerrar su sesión.

# MySQL: acceso a BBDD con PHP

## archivo profesor.php



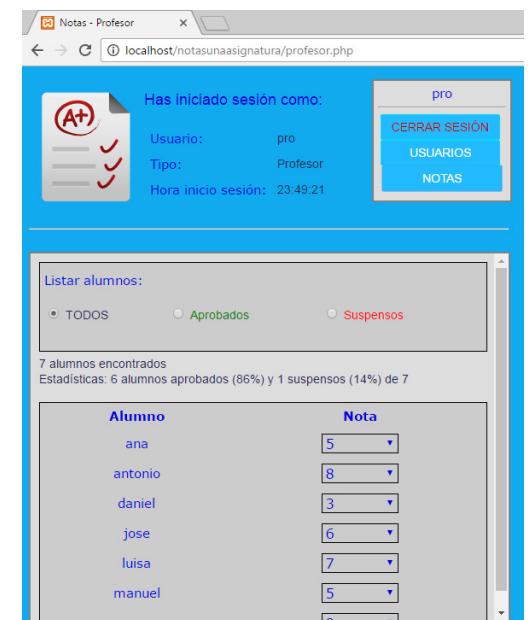
The screenshot shows a web interface for a professor. At the top, it displays session information: 'Has iniciado sesión como: Usuario: pro Tipo: Profesor Hora inicio sesión: 23:49:21'. Below this, there are two main sections: 'USUARIOS' and 'NOTAS'. The 'USUARIOS' section contains a table with columns 'Usuario' and 'Tipo', listing users like pro, ana, antonio, daniel, jose, luisa, and manuel, all categorized as 'Alumno'. The 'NOTAS' section is partially visible. A red box highlights the 'NOTAS' button.

archivos  
usuarios.html  
y usuarios.php



This screenshot shows the 'usuarios.php' page. It displays session info and a sidebar with 'CERRAR SESIÓN', 'USUARIOS', and 'NOTAS'. The main area shows a table of users with columns 'Usuario' and 'Tipo', listing the same users as the first screenshot. A green arrow points from the 'USUARIOS' section of the first screenshot to this table.

archivos  
notas.html  
y notas.php



This screenshot shows the 'notas.php' page. It displays session info and a sidebar with 'CERRAR SESIÓN', 'USUARIOS', and 'NOTAS'. The main area shows a table of student grades with columns 'Alumno' and 'Nota', listing students ana, antonio, daniel, jose, luisa, and manuel with their respective grades.

Esta interfaz permite al usuario de tipo Profesor:

- **USUARIOS:** Consultar/Editar los usuarios.
- **NOTAS:** Consultar/Editar las notas de la asignatura.
- Cerrar su sesión.

# MySQL: acceso a BBDD con PHP

archivos **usuarios.html** y **usuarios.php** (AJAX)

Esta interfaz permite al usuario de tipo Profesor:

- Listar/buscar usuarios por tipo: TODOS, sólo Alumnos, o sólo Profesores.
- Añadir un usuario (de tipo Alumno o Profesor) en caso de que no exista ya.
- Eliminar usuarios (de cualquier tipo). Si se elimina un usuario de tipo Alumno se deberá eliminar su nota.

# MySQL: acceso a BBDD con PHP

The screenshot shows a web application window titled "Notas - Profesor". The URL in the address bar is "localhost/notasunaasignatura/profesor.php". The page displays session information: "Has iniciado sesión como:", "Usuario: pro", "Tipo: Profesor", and "Hora inicio sesión: 23:49:21". A sidebar on the right contains links: "CERRAR SESIÓN", "USUARIOS", and "NOTAS", with "NOTAS" highlighted by a red box. Below this, a section titled "Listar alumnos:" offers filtering options: "TODOS" (selected), "Aprobados", and "Suspensos". It shows "7 alumnos encontrados" with a breakdown: "Estadísticas: 6 alumnos aprobados (86%) y 1 suspensos (14%) de 7". A table lists student names and their corresponding grades:

Alumno	Nota
ana	5
antonio	8
daniel	3
jose	6
luisa	7
manuel	5

archivos [notas.html](#)  
y [notas.php](#) (AJAX)

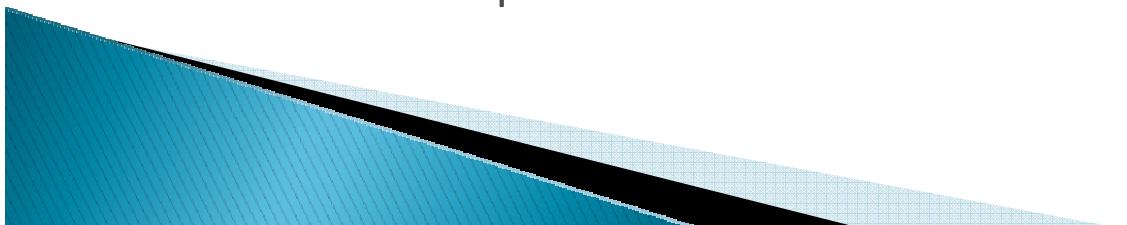
Esta interfaz permite al usuario de tipo Profesor:

- Listar/buscar los Alumnos: Aprobados en la asignatura, los suspensos, o TODOS.
- Consultar las estadísticas de la asignatura: aprobados, suspensos y alumnos totales.
- Insertar/modificar la nota de cualquier alumno.

## MySQL: acceso a BBDD con PHP

• **Actividad:** modificar la aplicación web de notas para que se **pueda crear varias asignaturas** de manera que:

- Cada Alumno podrá tener nota en varias asignaturas distintas. Cuando inicie sesión se le mostrará la nota de todas sus asignaturas. Cuando se crea un usuario Alumno no se le pone nota en ninguna asignatura: primero debe agregarlo (matricularlo) en su/s asignaturas algún profesor.
- Un profesor podrá impartir más de una asignatura; pero cada asignatura sólo será impartida por un único profesor.
- Además de poder gestionar las notas de los alumnos de sus asignaturas, el profesor podrá crear y eliminar asignaturas. Cuando cree una asignatura, inicialmente no contará con ningún alumno matriculado. Cuando elimine una asignatura se borrarán todas las notas de sus alumnos.
- Si se borra un alumno se borrarán todas sus notas (de todas sus asignaturas).
- Si se borra un profesor se borrarán las notas de los alumnos de sus asignaturas.



# MySQL: acceso a BBDD con PHP

```
CREATE TABLE t_usuarios (
    usuario VARCHAR(10) NOT NULL,
    clave VARCHAR(10) NOT NULL,
    tipo enum('Profesor', 'Alumno'),
    PRIMARY KEY (usuario) /* no puede haber dos usuarios con el mismo nombre */
);

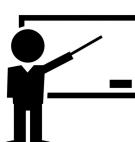
CREATE TABLE t_asignaturas (
    asignatura VARCHAR(10) NOT NULL,
    profesor VARCHAR(10) NOT NULL,
    PRIMARY KEY (asignatura), /* no puede haber dos asignaturas con el mismo nombre */
    FOREIGN KEY (profesor) REFERENCES t_usuarios(usuario) /* obliga a insertar solo nombres de
    profesores que sean usuarios existentes en t_usuarios */
        ON DELETE CASCADE /* si se borra un usuario de t_usuario se borran sus
        asignaturas como profesor */
        ON UPDATE CASCADE /* si se actualiza el nombre de un usuario de t_usuario
        se actualiza su nombre como profesor */
);

CREATE TABLE t_notas (
    alumno VARCHAR(10) NOT NULL,
    asignatura VARCHAR(10) NOT NULL,
    nota INT(2) UNSIGNED,
    PRIMARY KEY (alumno, asignatura), /* un alumno puede tener varias notas si son de asignaturas distintas */
    FOREIGN KEY (alumno) REFERENCES t_usuarios(usuario) /* obliga a insertar solo nombres de alumno
    que sean usuarios existentes en t_usuarios */
        ON DELETE CASCADE /* si se borra un usuario de t_usuarios se borran sus notas */
        ON UPDATE CASCADE, /* si se actualiza el nombre de un usuario de
        t_usuarios se actualiza su nombre como alumno */
    FOREIGN KEY (asignatura) REFERENCES t_asignaturas(asignatura) /* obliga a insertar solo
    nombres de asignaturas que existan en t_asignaturas */
        ON DELETE CASCADE /* si se borra una asignatura en t_asignaturas se borran sus notas */
        ON UPDATE CASCADE /* si se actualiza el nombre de una asignatura de
        t_asignaturas se actualiza su nombre como asignatura */
);
```

Para soportar estas nuevas funcionalidades, se ha modificado la BBDD añadiendo una nueva tabla (**t\_asignaturas**) y modificando la tabla **t\_notas**.

# MySQL: acceso a BBDD con PHP

## archivo login.php



Notas

Inicie sesión:

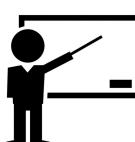
Usuario:

Clave:

ACCEDER



## archivo profesor.php



Has iniciado sesión como:

Usuario: pro  
Tipo: Profesor  
Hora inicio sesión: 01:05:21

CERRAR SESIÓN

USUARIOS

ASIGNATURAS

NOTAS

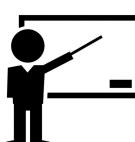
Listar usuarios de tipo: TODOS

Añadir usuario:

Tipo: Usuario: Clave:

Alumno ▾   

Usuario	Tipo	
fol	Profesor	✗
pro	Profesor	✗
ana	Alumno	✗
antonio	Alumno	✗
carlos	Alumno	✗
daniel	Alumno	✗



Has iniciado sesión como:

Usuario: daniel  
Tipo: Alumno  
Hora inicio sesión: 01:03:17

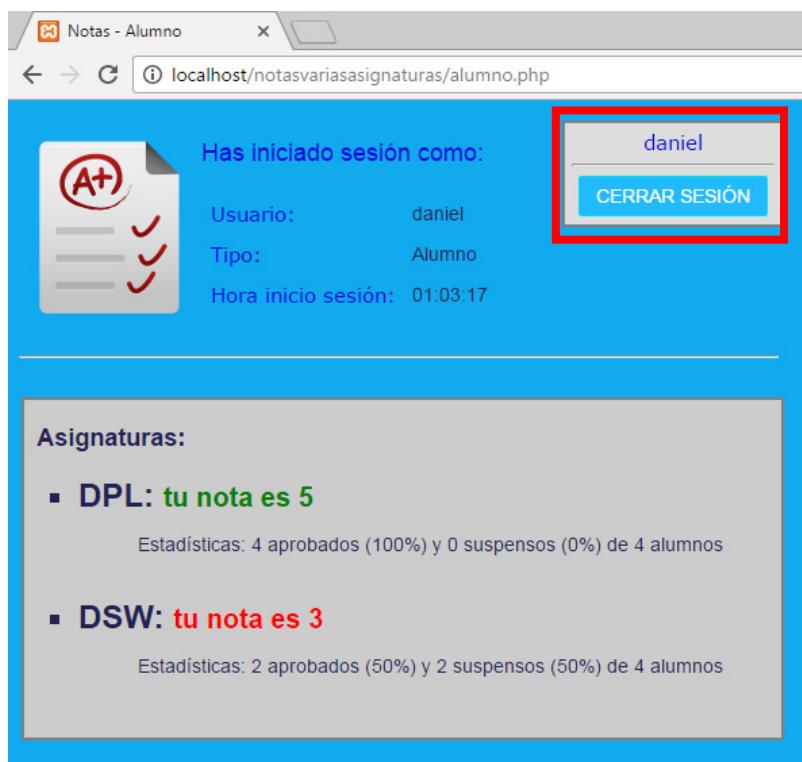
CERRAR SESIÓN

Asignaturas:

- DPL: tu nota es 5  
Estadísticas: 4 aprobados (100%) y 0 suspensos (0%) de 4 alumnos
- DSW: tu nota es 3  
Estadísticas: 2 aprobados (50%) y 2 suspensos (50%) de 4 alumnos

## archivo alumno.php

# MySQL: acceso a BBDD con PHP



archivo [alumno.php](#)

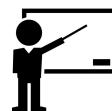


Esta interfaz permite al usuario de tipo Alumno:

- Consultar la nota de las asignaturas en las que figure (sólo la suya).
- Consultar las estadísticas de cada una de sus asignaturas: porcentaje de aprobados, de suspensos y total de alumnos.
- Cerrar su sesión.

# MySQL: acceso a BBDD con PHP

## archivo profesor.php



localhost/notasvariasasignaturas/profesor.php

Has iniciado sesión como:  
Usuario: pro  
Tipo: Profesor  
Hora inicio sesión: 01:05:21

CERRAR SESIÓN

**USUARIOS**

ASIGNATURAS

NOTAS

Listar usuarios de tipo: TODOS

Añadir usuario: Tipo: Usuario: Clave: Alumno +

Usuario	Tipo
fol	Profesor
pro	Profesor
ana	Alumno
antonio	Alumno
carlos	Alumno
daniel	Alumno

archivos  
**usuarios.html**  
y **usuarios.php**

Has iniciado sesión como:  
Usuario: pro  
Tipo: Profesor  
Hora inicio sesión: 01:05:21

CERRAR SESIÓN

USUARIOS

ASIGNATURAS

NOTAS

Listar usuarios de tipo: TODOS

Añadir usuario: Tipo: Usuario: Clave: +

Usuario	Tipo
fol	Profesor
pro	Profesor
ana	Alumno
antonio	Alumno
carlos	Alumno
daniel	Alumno

archivos  
**asignaturas.html**  
y **asignaturas.php**

archivos  
**notas.html**  
y **notas.php**



localhost/notasvariasasignaturas/profesor.php

Has iniciado sesión como:  
Usuario: pro  
Tipo: Profesor  
Hora inicio sesión: 01:05:21

CERRAR SESIÓN

USUARIOS

ASIGNATURAS

NOTAS

Listar notas de la asignatura: DPL

Listar alumnos: • TODOS • Aprobados • Suspensos

4 alumnos encontrados Estadísticas: 4 alumnos aprobados (100%) y 0 suspensos (0%) de 4

Alumno	Nota
antonio	8
daniel	5
luisa	7
manuel	5

Has iniciado sesión como:  
Usuario: pro  
Tipo: Profesor  
Hora inicio sesión: 01:05:21

CERRAR SESIÓN

USUARIOS

ASIGNATURAS

NOTAS

Listar alumnos de asignatura: DPL

Eliminar asignatura: DPL - X

Añadir asignatura: +

7 alumnos encontrados

Alumno	Matriculado
ana	
antonio	<input checked="" type="checkbox"/>
carlos	
daniel	<input checked="" type="checkbox"/>
jose	
luisa	<input checked="" type="checkbox"/>

Esta interfaz permite al usuario Profesor:

- USUARIOS:** Consultar/Editar los usuarios.
- ASIGNATURAS:** Editar las asignaturas.
- NOTAS:** Consultar/Editar las notas.
- Cerrar su sesión.

# MySQL: acceso a BBDD con PHP

The screenshot shows a web application for managing users. At the top, it displays session information: "Has iniciado sesión como: pro", "Tipo: Profesor", and "Hora inicio sesión: 01:05:21". A sidebar on the left has a red checkmark icon and a dropdown menu for "Listar usuarios de tipo: TODOS". The main area shows a table of users:

Usuario	Tipo	
fol	Profesor	X
pro	Profesor	X
ana	Alumno	X
antonio	Alumno	X
carlos	Alumno	X
daniel	Alumno	X

To the right, there is an "Añadir usuario:" form with fields for "Tipo: Alumno", "Usuario:", and "Clave:", and a green "+" button. A red box highlights the "USUARIOS" button in the sidebar menu.

archivos **usuarios.html**  
y **usuarios.php** (AJAX)

Esta interfaz permite al usuario de tipo Profesor:

- Listar/buscar usuarios por tipo: TODOS, sólo Alumnos, o sólo Profesores.
- Añadir un usuario (de tipo Alumno o Profesor) en caso de que no exista ya.
- Eliminar usuarios (de cualquier tipo). Si se elimina un usuario de tipo Alumno se deberá eliminar su/s nota/s. Si se elimina un profesor se debe eliminar todas sus asignaturas y sus notas.

# MySQL: acceso a BBDD con PHP

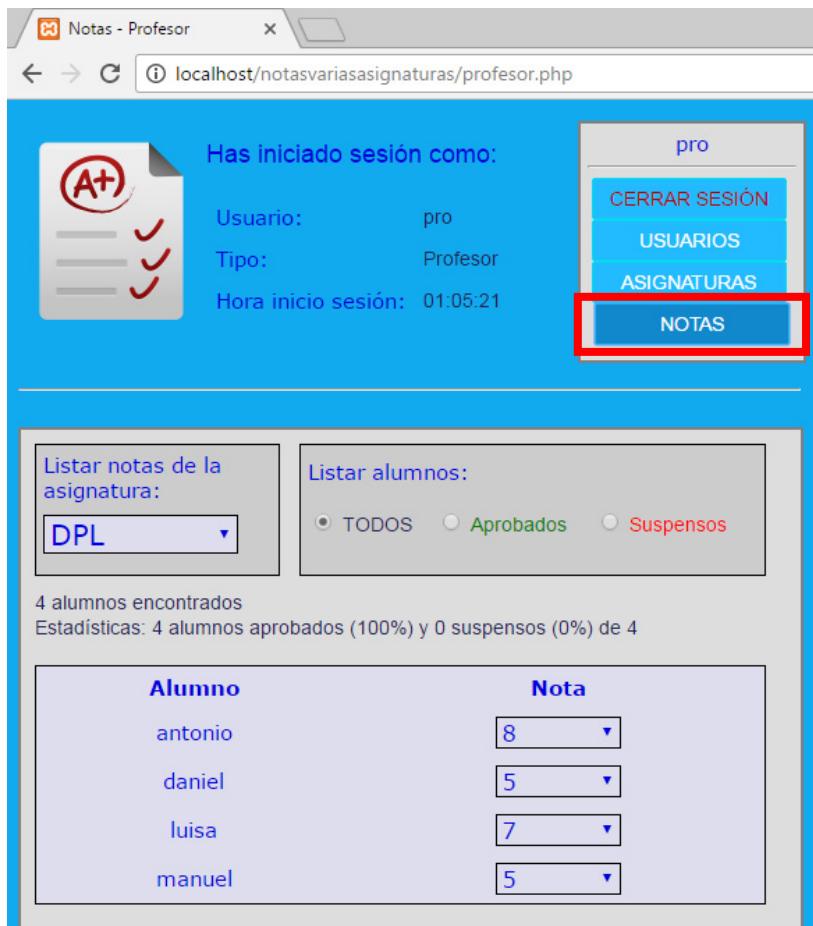


archivos [asignaturas.html](#) y [asignaturas.php](#) (AJAX)

Esta interfaz permite al usuario de tipo Profesor:

- Listar/buscar usuarios de tipo Alumno que hay en sus asignaturas. Puede agregarlos (matricularlos) o quitarlos de la asignatura elegida.
- Añadir una asignatura que se asociará a dicho profesor.
- Eliminar alguna de las asignaturas del profesor usuarios. Si se elimina una asignatura se deberá eliminar todas las notas.

# MySQL: acceso a BBDD con PHP



archivos [notas.html](#)  
y [notas.php](#) (AJAX)

Esta interfaz permite al usuario de tipo Profesor:

- Listar/buscar los Alumnos de una de sus asignaturas (sólo las suyas): solo los aprobados, los suspensos, o TODOS.
- Consultar las estadísticas de la asignatura elegida.
- Insertar/modificar la nota de cualquier alumno para la asignatura elegida.

# PDO: Acceso a otros tipos de BBDD

- **Ejemplo:** acceso a BBDD con la **librería PDO** (PHP Data Objects) en lugar de la librería MySQLi:

```
<!DOCTYPE html>
<html> <head> <title> Acceso a BBDD con la librería PDO </title> </head>
<body bgcolor="#E6E600">

<?php
    $servidor = "localhost";           $usuario = "root";           $clave = "";
    try {
        $conexion = new PDO("mysql:host=$servidor;dbname=", $usuario, $clave);
        $conexion->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
        echo "<br>Conectado con el servidor MySQL";
        $sql = "DROP DATABASE IF EXISTS db_usuarios;";
        $conexion->exec($sql);
        $sql = "CREATE DATABASE db_usuarios;";
        $conexion->exec($sql);
        $sql = "USE db_usuarios;";
        $conexion->exec($sql);
        echo "<br><br>Base de datos db_usuarios creada";
        $sql = "CREATE TABLE t_usuarios (
                    id INT AUTO_INCREMENT NOT NULL,
                    usuario VARCHAR(15) NOT NULL,
                    clave VARCHAR(10) NOT NULL,
                    PRIMARY KEY (id)
                );";
        $conexion->exec($sql);
        echo "<br><br>Tabla t_usuarios creada vacía";
```

## PDO resulta útil para:

- Conectar con distintos tipos de BD (hasta 12, incluido MySQL).
- Desarrollar código fácil de migrar a otra BD.
- Desarrollar scripts para importar/exportar desde un SGBD a otro SGBD.

# PHP: librería PDO

```
$sql = "INSERT INTO t_usuarios (usuario, clave) VALUES ('manuel', 'manuel2');";
$connexion->exec($sql);
echo "<br><br>Insertado el usuario 'manuel' con clave 'manuel2' en la tabla t_usuarios";

$sql = "INSERT INTO t_usuarios (usuario, clave) VALUES ('laura', 'laura_4');";
$connexion->exec($sql);
echo "<br><br>Insertado el usuario 'laura' con clave 'laura_4' en la tabla t_usuarios";

$sql = "INSERT INTO t_usuarios (usuario, clave) VALUES ('jose', 'jose');";
$connexion->exec($sql);
echo "<br><br>Insertado el usuario 'jose' con clave 'jose' en la tabla t_usuarios";

$sql = "INSERT INTO t_usuarios (usuario, clave) VALUES ('administrador', 'admin50');";
$connexion->exec($sql);
echo "<br><br>Insertado el usuario 'administrador' con clave 'admin50' en la tabla t_usuarios";

$sql = "UPDATE t_usuarios SET clave='jose45' WHERE usuario='jose'";
$stmt = $connexion->prepare($sql);
$stmt->execute();
echo "<br><br>Actualizado el valor de la clave del usuario 'jose' en la tabla t_usuarios";
echo "<br>".$stmt->rowCount()." registros actualizados";

$sql = "DELETE FROM t_usuarios WHERE usuario='manuel';";
$connexion->exec($sql);
echo "<br><br>Borrado el usuario 'manuel' de la tabla t_usuarios";
```

# PHP: librería PDO



```
$sql = "SELECT * FROM t_usuarios ORDER BY usuario;"  
$sentencia = $conexion->prepare($sql);  
$sentencia->execute();  
  
$resultado = $sentencia->setFetchMode(PDO::FETCH_ASSOC);  
$registros = $sentencia->fetchAl()  
//print_r($registros);  
echo "<p>Listado del contenido actual de la tabla t_usuarios:</p>";  
echo "<table border=\"1\" width=\"350\">";  
echo "<tr> <th>id</th> <th>usuario</th> <th>clave</th> </tr>";  
foreach ($registros as $registro) {  
    echo "<tr>";  
    echo "<td>".$registro['id']."</td>";  
    echo "<td>".$registro['usuario']."</td>";  
    echo "<td>".$registro['clave']."</td>";  
    echo "</tr>";  
}  
echo "</table>";  
  
} catch (PDOException $e) {  
    echo "Connection failed: ". $e->getMessage();  
}  
// Desconectar:  
$conexion = null;  
echo "<br>Desconectado del servidor MySQL";  
?  
</body>  
</html>
```

The screenshot shows a browser window titled 'Acceso a BBDD con la librería PDO'. The address bar says 'localhost/pdo.php'. The page content is a log of database operations:

- Conectado con el servidor MySQL
- Base de datos db\_usuarios creada
- Tabla t\_usuarios creada vacía
- Insertado el usuario 'manuel' con clave 'manuel2' en la tabla t\_usuarios
- Insertado el usuario 'laura' con clave 'laura\_4' en la tabla t\_usuarios
- Insertado el usuario 'jose' con clave 'jose' en la tabla t\_usuarios
- Insertado el usuario 'administrador' con clave 'admin50' en la tabla t\_usuarios
- Actualizado el valor de la clave del usuario 'jose' en la tabla t\_usuarios  
1 registros actualizados
- Borrado el usuario 'manuel' de la tabla t\_usuarios
- Listado del contenido actual de la tabla t\_usuarios:

id	usuario	clave
4	administrador	admin50
3	jose	jose45
2	laura	laura_4

Desconectado del servidor MySQL

# BBDD: Prevenir ataques de tipo SQL Injection

- Ataque que consiste en ejecutar en la aplicación web **código SQL malicioso**.



- **Funcionamiento:** el atacante inserta (inyecta) su código SQL malicioso en los campos de entrada de los formularios de la aplicación; ya que estos valores se usan para construir las sentencias SQL de la aplicación.

- Con su SQL, el atacante puede modificar las consultas originales y ejecutar otras para:

- acceder a la web
- obtener información de alguna tabla
- borrar datos

A screenshot of a web-based administrator login interface. The 'Username' field contains the value 'hi' or 1=1--'. The 'Password' field contains a series of asterisks. Below the fields is a 'login' button. The URL bar above the form shows a long string of characters, likely a URL-encoded payload.

- Es un ataque muy sencillo y peligroso

# BBDD: ataques de tipo SQL Injection

- **Ejemplo:** cuando nos autentificamos en una aplicación, ésta toma el nombre de usuario y la clave que hemos introducido en el formulario y los añade a una sentencia SQL para buscarnos en la base de datos :

Username:	<input type="text"/>
Password:	<input type="password"/>
<input type="button" value="Submit"/>	

```
SELECT * FROM User WHERE username='jose' AND password='ejemplo';
```

Si un usuario malintencionado escribe esto como sus credenciales:

**Username:** jose

**Password:** ' ; DELETE FROM User WHERE '1'='1'



Entonces el query resultante sería:

```
SELECT * FROM User WHERE  
username = 'jose' AND password = '' ; DELETE FROM User WHERE '1'='1'
```

¡¡Ahora un query se ha convertido en 2 y ambos se ejecutan, con lo que se borra toda la tabla de usuarios!!

# MySQL: SQL Injection

- **Ejemplo:** dada una aplicación web que usa la siguiente BBDD para el login de sus usuarios:

- El atacante puede acceder a la web introduciendo estas credenciales:

```
DROP DATABASE IF EXISTS db_usuarios;
CREATE DATABASE db_usuarios;  USE db_usuarios;
CREATE TABLE t_usuarios (
    id INT AUTO_INCREMENT NOT NULL,
    usuario VARCHAR(15) NOT NULL,
    clave VARCHAR(10) NOT NULL,
    PRIMARY KEY (id)
);
INSERT INTO t_usuarios (usuario, clave) VALUES ('manuel', 'manuel2');
INSERT INTO t_usuarios (usuario, clave) VALUES ('jose', 'jose');
INSERT INTO t_usuarios (usuario, clave) VALUES ('administrador', 'adm50');
```

Usuario: cualquier valor

Clave:

```
' OR '1' = '1      /* siempre es cierto */

' UNION SELECT * FROM t_usuarios WHERE id = '1

' UNION SELECT * FROM t_usuarios WHERE usuario = 'administrador

' UNION SELECT * FROM t_usuarios WHERE usuario LIKE '%admin%
```



A screenshot of a web browser window titled "Acceso". The URL is "localhost/sqlinjection/login.php". The form has two fields: "Usuario" containing "aaaaaaaaaa" and "Clave" containing "' OR '1' = '1". A blue arrow points from this screen to the next one.

A screenshot of a web browser window titled "Principal". The URL is "localhost/sqlinjection/principal.php". The page displays a message "Has iniciado sesión como:" followed by "USUARIO: manuel" and "ID: 1". There is a "CERRAR SESIÓN" button. A blue arrow points from the previous screen to this one.

# MySQL: Prevenir ataques SQL Injection

- Si el método de envío usado por el formulario fuese GET, también, se podría injectar el SQL en la propia URL de la petición enviada por la página de login:

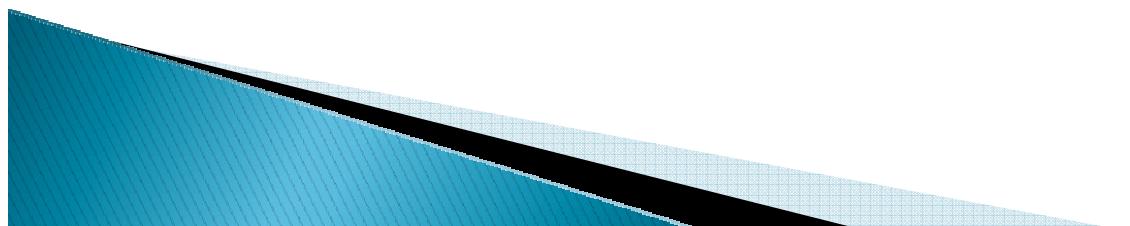


http://localhost/login.php?usuario=a&clave=' OR '1' = '1

- **Solución:** usar la función de la clase mysqli:

```
string real_escape_string (string $valor)
```

para limpiar los valores leídos (escapar comillas) del formulario antes de construir las sentencias SQL correctas con ellos.



# MySQL: Prevenir ataques SQL Injection

```
<?php
    session_start();
    if (isset($_SESSION['usuario'])) {
        $_SESSION = array();
        setcookie(session_name(), '', time() - 100, '/');
        session_destroy();
        header("Location: login.php");
    }
    $error = "";
    if ($_SERVER['REQUEST_METHOD'] == "POST" || $_SERVER["REQUEST_METHOD"] == "GET") {
        if (empty($_REQUEST['usuario']) || empty($_REQUEST['clave'])) {
            $error = "Debes llenar todos los datos";
        } else {
            $conexion = new mysqli("localhost", "root", "", "db_usuarios");
            if ($conexion->connect_error) die("Conexión fallida: " . $conexion->connect_error);
            $usuario = $_REQUEST['usuario'];
            $clave = $_REQUEST['clave'];
            // $usuario = $conexion->real_escape_string($usuario);
            // $clave = $conexion->real_escape_string($clave);           // echo $clave
            $consultasQL = "SELECT * FROM t_usuarios WHERE usuario = '" . $usuario . "' AND clave = '" . $clave . "' ";
            $resultado = $conexion->query($consultasQL);
            if ($resultado->num_rows > 0) {
                $registro = $resultado->fetch_assoc();
                $_SESSION['usuario'] = $registro["usuario"];   $_SESSION['id'] = $registro["id"];
                $resultado->close();
                $conexion->close();
                header("Location: principal.php");
            } else {
                $error = "Fallo en la autentificación";
            }
            $conexion->close();
        }
    }
?>
```

↑ archivo  
**login.php**

# MySQL: Prevenir ataques de tipo SQL Injection

```
<!DOCTYPE html>
<html>
<head>
    <title>Acceso</title>
</head>
<body bgcolor="#5588BB">

    <br>
    <h3 style="color:white;">Inicie sesión:</h3>

    <form action="" method="post">
        <table style="color:white;">
            <tr>
                <td>Usuario:</td> <td><input type="text" name="usuario" size="45"></td>
            </tr>
            <tr>
                <td>Clave:</td> <td><input type="text" name="clave" size="45"></td>
            </tr>
            <tr height="50">
                <td colspan="2" align="center"><input type="submit" value="ACCEDER"></td>
            </tr>
        </table>
    </form>

    <span style="color:red; font-size:18px;">
        <?php echo $error; ?>
    </span>

</body>
</html>
```

↑ continuación del archivo  
**login.php**

# MySQL: Prevenir ataques de tipo SQL Injection

```
<?php
    session_start();
    if (isset($_SESSION['usuario'])) {
        $usuario = $_SESSION['usuario'];
        $id = $_SESSION['id'];
    } else
        header("Location: login.php");
?>

<!DOCTYPE html>
<html> <head> <title>Principal</title> </head>
<body bgcolor="#5588BB">
    <br>
    <h3 style="color:white;">Has iniciado sesión como:</h3>
    <table style="color:white;" cellpadding="5">
        <tr><td>USUARIO:</td> <td> <?php echo $usuario; ?> </td></tr>
        <tr><td>ID:</td> <td> <?php echo $id; ?> </td></tr>
    </table>
    <br> <hr align="left" width="450"> <br>
    <div style="position:absolute; left:300px; top:15px; width:150px; height:65px;
        background:#DDDDDD; border:2px solid gray; padding:5px; text-align:center;
        font-family:verdana; color:blue; font-size:14px;">
        <span class="titulo"> <?php echo $usuario; ?> </span>
        <hr>
        <button type="button" onClick="window.location.assign('logout.php');">
            CERRAR SESIÓN</button>
    </div>
</body>
</html>
```

archivo →  
logout.php

← archivo  
principal.php

```
<?php
    session_start();
    if (isset($_SESSION['usuario'])) {
        $_SESSION = array();
        setcookie(session_name(), '', time() - 100, '/');
        session_destroy();
    }
    header("Location: login.php");
?>
```