

# CURSO ADMINISTRACIÓN APACHE WEB SERVER

## PRÁCTICA 5: Instalación del gestor de bases de datos MySQL. Programación de aplicaciones web con PHP y MySQL

### **Objetivos**

- Instalación y configuración del servidor MySQL sobre la plataforma Win32.
- Arranque y parada del servidor MySQL (comandos en línea, front-ends de gestión del servidor)
- Creación de una base de datos y configuración básica.
- Librería PHP para acceso a bases de datos MySQL
- Programación de una aplicación PHP para gestión de una base de datos de alumnos.
- Pruebas de funcionamiento.

### **Materiales**

- PC con S.O Windows
- Conexión a Internet (Ethernet)
- Software Apache v1.3 con módulo SSL, PHPv4.3.
- Servidor MySQL v 4.0.16

### **Desarrollo de la práctica**

- 1) Verificar la instalación realizada en prácticas anteriores del servidor web Apache con el módulo PHP 4.3. (práctica 1)
- 2) Descarga del software e instalación de MySQL ( opcional: front-ends de gestión de MySQL)
- 3) Administración de ficheros de configuración
- 4) Creación de la base de datos
- 5) Programación en PHP de un script de gestión de la Base de datos de alumnos

## 1. Verificar el servidor Web Apache

Comprobaremos que el servidor web apache está instalado y configurado correctamente incluyendo los módulos PHP, etc., tal como se describe en las sesiones anteriores. En esta práctica utilizaremos las versiones 1.3.28 de Apache y 4.3.3 de PHP.

Con `phpinfo()` se debe obtener la siguiente salida:

### mysql

MySQL Support		enabled
Active Persistent Links		0
Active Links		0
Client API version		3.23.49

Directive	Local Value	Master Value
mysql.allow_persistent	On	On
mysql.connect_timeout	60	60
mysql.default_host	<i>no value</i>	<i>no value</i>
mysql.default_password	<i>no value</i>	<i>no value</i>
mysql.default_port	<i>no value</i>	<i>no value</i>
mysql.default_socket	<i>no value</i>	<i>no value</i>
mysql.default_user	<i>no value</i>	<i>no value</i>
mysql.max_links	Unlimited	Unlimited
mysql.max_persistent	Unlimited	Unlimited
mysql.trace_mode	Off	Off

Estos valores de configuración de MySQL se pueden modificar en el fichero `php.ini`.

```
[MySQL]
; Allow or prevent persistent links.
mysql.allow_persistent = On

; Maximum number of persistent links. -1 means no limit.
mysql.max_persistent = -1

; Maximum number of links (persistent + non-persistent). -1 means no limit.
mysql.max_links = -1

; Default port number for mysql_connect(). If unset, mysql_connect() will use
; the $MYSQL_TCP_PORT or the mysql-tcp entry in /etc/services or the
; compile-time value defined MYSQL_PORT (in that order). Win32 will only look
; at MYSQL_PORT.
mysql.default_port =

; Default socket name for local MySQL connects. If empty, uses the built-in
; MySQL defaults.
```

```
mysql.default_socket =  
  
; Default host for mysql_connect() (doesn't apply in safe mode).  
mysql.default_host =  
  
; Default user for mysql_connect() (doesn't apply in safe mode).  
mysql.default_user =  
  
; Default password for mysql_connect() (doesn't apply in safe mode).  
; Note that this is generally a *bad* idea to store passwords in this file.  
; *Any* user with PHP access can run 'echo get_cfg_var("mysql.default_password")  
; and reveal this password! And of course, any users with read access to this  
; file will be able to reveal the password as well.  
mysql.default_password =  
  
; Maximum time (in seconds) for connect timeout. -1 means no limit  
mysql.connect_timeout = 60  
  
; Trace mode. When trace_mode is active (=On), warnings for table/index scans and  
; SQL-Errors will be displayed.  
mysql.trace_mode = Off
```

## 2. Instalación de MySQL

Para ello vamos a utilizar la versión v4.0.16 de MySQL para win32 que se puede obtener en la página web del curso o en la página de MySQL ([www.mysql.com](http://www.mysql.com)). Descomprimos el fichero y ejecutamos la aplicación setup.

El directorio por defecto que utilizaremos es `c:/mysql`

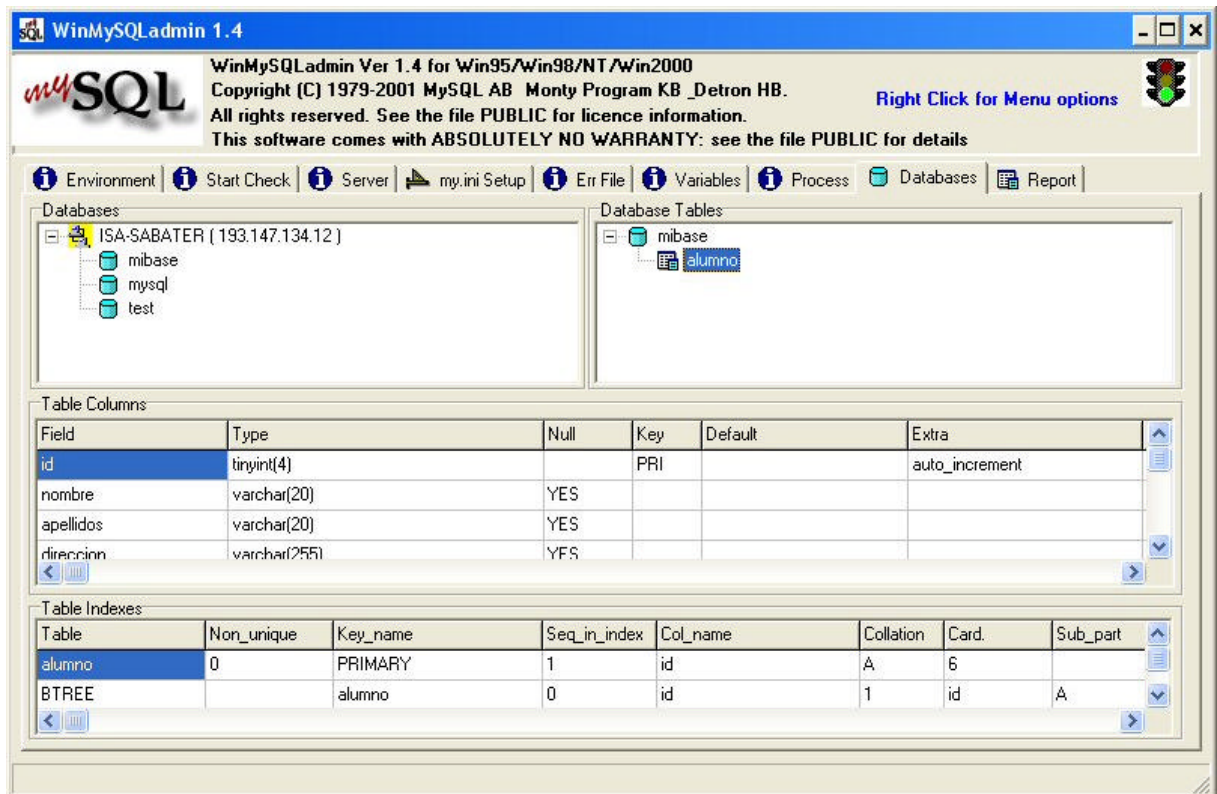
en la documentación de MySQL se nos indica que se debe modificar `php.ini` añadiendo la extensión `.dll` necesaria, pero en el `php.ini` encontramos:

```
;Windows Extensions  
;Note that MySQL and ODBC support is now built in, so no dll is needed for it.  
;
```

luego no debemos modificar nada, dado que el soporte de MySQL ya está incluido.

Para la gestión del servidor MySQL instalado podemos optar por dos opciones:

- Ø Utilizar los comandos en línea dentro del directorio `c:\mysql\bin>`, tal y como vamos a hacer en esta práctica. En este caso utilizaremos el ejecutable `winmysqladmin.exe` incluido en la instalación para visualizar la correcta instalación. Esta aplicación no permite crear ni modificar bases de datos, pero si permite modificar el `my.ini` en caso de que fuera necesario. Se puede utilizar esta aplicación para visualizar las bases de datos y tablas que vamos añadiendo a lo largo de la práctica.



- Ø Utilizar cualquiera de los múltiples front-end existentes en la red para la gestión de MySQL tanto como servicio de NT o como aplicación de windows. Se recomienda utilizar el MySQL Control Center ([www.mysql.com](http://www.mysql.com)) ; el SQLyog! (<http://www.phpfreaks.com/mysqlmanual.php>) o cualquier otro front-end. Estos programas evitan tener que aprender los comandos de gestión de MySQL.

### 3. Creación de una base de datos

Creamos la base “mibase” con el comando:

```
mysqladmin -u root create mipase
```

y visualizamos que esta base se ha creado correctamente con la aplicación winmysqladmin.exe tal y como se muestra en la imagen anterior.

Luego para introducir una tabla en “mibase” creo en el directorio c:\mysql\bin el fichero de texto mipase.dump con el siguiente contenido:

```
CREATE TABLE alumno ( id tinyint(4) DEFAULT '0' NOT NULL
AUTO_INCREMENT, nombre varchar(20), apellidos varchar(20), direccion
varchar(255), comentarios varchar(50), PRIMARY KEY (id), UNIQUE id (id));

INSERT INTO alumno VALUES (1,'Jose','Martinez Beltra','C/ calle,
elche','sobresaliente');
INSERT INTO alumno VALUES (2,'Pepe','Juan Roig','Avda. Ferrocarril s/n 03202
Elche','aprobado');
INSERT INTO alumno VALUES (3,'Juan','Azorin Tarí','Partida del castillo S/N
Alicante','aprobado');
```

E introducimos la tabla creada en “mibase” con el comando:

```
mysql -u root mipase < mipase.dump
```

## 4. Primer script. Consulta de la base

el primer php-script nos muestra como realizar una conexión con la base de datos y una consulta de la misma.

```
<html>
<body>
<?php
$db = mysql_connect("localhost", "root");
mysql_select_db("mibase",$db);
$result = mysql_query("SELECT * FROM alumno",$db);
printf("Nombre: %s<br>\n", mysql_result($result,0,"nombre"));
printf("Apellidos: %s<br>\n", mysql_result($result,0,"apellidos"));
printf("Dirección: %s<br>\n", mysql_result($result,0,"direccion"));
printf("Comentario: %s<br>\n", mysql_result($result,0,"comentarios"));
?>
</body>
</html>
```

la función `mysql_connect( )` abre la comunicación entre el servidor MySQL en el host especificado. `mysql_select_db( )` permite seleccionar la base de datos a utilizar. Los resultados de la consulta SQL realizada por la función `mysql_query( )` son almacenados en la variable `$result`.

La salida generada por el parseador de PHP es:

```
<html>
<body>
Nombre: Jose<br>
```

```
Apellidos: Martinez<br>
Dirección: su calle, elche<br>
Comentario: sobresaliente<br>
</body>
</html>
```

El siguiente ejemplo muestra la utilización del `while( )` para recorrer la base de datos:

Ø Usando índices numéricos con `mysql_fetch_row( )`:

```
<html>
<body>
<?php
$db = mysql_connect("localhost", "root");
mysql_select_db("mibase",$db);
$result = mysql_query("SELECT * FROM alumno",$db);
echo "<table border=1>\n";
echo "<tr><td>Nombre</td><td>Comentarios</td>\n";
while ($myrow = mysql_fetch_row($result)) {
    printf("<tr><td>%s %s</td><td>%s</td></tr>\n",
        $myrow[1], $myrow[2], $myrow[3]);
}
echo "</table>\n";
?>
</body>
</html>
```

Ø Usando los nombres de los campos con `mysql_fetch_array( )`:

```
<html>
<body>
<?php
$db = mysql_connect("localhost", "root");
mysql_select_db("mibase",$db);
$result = mysql_query("SELECT * FROM alumno",$db);
if ($myrow = mysql_fetch_array($result)) {
    echo "<table border=1>\n";
    echo "<tr><td>Nombre</td><td>Comentario</td></tr>\n";
    do {
        printf("<tr><td>%s %s</td><td>%s</td>\n", $myrow["nombre"],
            $myrow["apellidos"], $myrow["direccion"]);
    } while ($myrow = mysql_fetch_array($result));
    echo "</table>\n";
} else {
    echo "Lo siento, no hay resultados!";
}
?>
</body>
</html>
```

## 5. Jugando con la base de datos

Una potente característica de PHP/MySQL es la posibilidad de obtener información utilizando 'querystring'. Existen varias posibilidades de realizar estas operaciones, como utilizar el método GET de los formularios HTML, o aprovechando la característica de PHP name=value, de forma que se utilice la sintaxis de

`<a href=http://host/pagina.php?name=value>`

El siguiente ejemplo muestra como se puede añadir links a las entradas de la base de datos

```
<html>
<body>
<?php
$db = mysql_connect("localhost", "root");
mysql_select_db("mibase",$db);
$result = mysql_query("SELECT * FROM alumno",$db);
if ($myrow = mysql_fetch_array($result)) {
    do {
        printf("<a href=\""%s?id=%s\"">%s %s</a><br>\n", $PHP_SELF, $myrow["id"],
        $myrow["nombre"], $myrow["apellidos"]);
    } while ($myrow = mysql_fetch_array($result));
} else {
    echo "Lo siento, no hay resultados!";
}
?>
</body>
</html>
```

En este ejemplo se ha utilizado la variable `$PHP_SELF`. Esta variable almacena el nombre del script php y su localización, de forma que puede utilizarse para completar la url del hiperlink.

**NOTA:** para que esta variable funcione, la opción **register\_globals** del fichero **php.ini** debe activarse. En caso contrario el parseador de PHP obvia esta variable y no completa la dirección. Otra solución que puede emplearse es utilizar **\$\_SERVER['PHP\_SELF']** en lugar de **\$PHP\_SELF**.

Completando el ejemplo anterior podemos añadir que la consulta a la base de datos se realice tomando en consideración el registro que se quiere consultar.

```
<html>
<body>
<?php
$db = mysql_connect("localhost", "root");
mysql_select_db("mibase",$db);

// muestra las entradas individualmente
if ($id) {
    $result = mysql_query("SELECT * FROM alumno WHERE id=$id",$db);
    $myrow = mysql_fetch_array($result);
    printf("Nombre: %s\n<br>", $myrow["nombre"]);
    printf("Apellidos: %s\n<br>", $myrow["apellidos"]);
    printf("Dirección: %s\n<br>", $myrow["direccion"]);
    printf("Comentarios: %s\n<br>", $myrow["comentarios"]);
} else {
    // muestra la lista de alumnos
    $result = mysql_query("SELECT * FROM alumno",$db);
    if ($myrow = mysql_fetch_array($result)) {
        // si hay resultados
        do {
            printf("<a href=\"%s?id=%s\">%s %s</a><br>\n", $_SERVER['PHP_SELF'],
$myrow["id"], $myrow["nombre"], $myrow["apellidos"]);
        } while ($myrow = mysql_fetch_array($result));
    } else {
        // si no hay resultados
        echo "Lo siento, no hay resultados!";
    }
}
?>
</body>
</html>
```

En este ejemplo se puede comprobar que en la segunda llamada a la página .php se añade un valor a la variable id, realizando la consulta de un registro concreto.



## 6. Uso de formularios. Introducción y modificación de datos.

Utilizando la variable `$PHP_SELF` y la comando SQL `INSERT INTO` podemos introducir datos en la base de datos:

```
<html>
<body>
<?php
if ($submit) {
    // process form
    $db = mysql_connect("localhost", "root");
    mysql_select_db("mibase",$db);
    $sql = "INSERT INTO alumno (nombre,apellidos,direccion,comentarios) VALUES
('$nombre','$apellidos','$direccion','$comentarios)";
    $result = mysql_query($sql);
    echo "Gracias por introducir la información.\n";
} else{
    // display form
    ?>
    <form method="post" action="<?php echo $PHP_SELF?>">
    Nombre:<input type="Text" name="nombre"><br>
    Apellidos:<input type="Text" name="apellidos"><br>
    Direccion:<input type="Text" name="direccion"><br>
    Comentarios:<input type="Text" name="comentarios"><br>
    <input type="Submit" name="submit" value="Enter informacion">
    </form>
    <?php
    } // end if
    ?>
</body>
</html>
```

```
<html>
<body>
<?php
$db = mysql_connect("localhost", "root");
mysql_select_db("mibase",$db);
if ($id) {
    // consultad de la DB
    $sql = "SELECT * FROM alumno WHERE id=$id";
    $result = mysql_query($sql);
    $myrow = mysql_fetch_array($result);
    ?>
    <form method="post" action="<?php echo $PHP_SELF?>">
    <input type="hidden" name="id" value="<?php echo $myrow["id"] ?>">
    Nombre:<input type="Text" name="nombre" value="<?php echo $myrow["nombre"]
    ?>"><br>
    Apellidos:<input type="Text" name="apellidos" value="<?php echo
    $myrow["apellidos"] ?>"><br>
    Dirección:<input type="Text" name="direccion" value="<?php echo
    $myrow["direccion"] ?>"><br>
    Comentarios:<input type="Text" name="comentarios" value="<?php echo
    $myrow["comentarios"] ?>"><br>
    <input type="Submit" name="submit" value="Enter informacion">
    </form>
    <?php
    } else {
    //lista de alumnos
    $result = mysql_query("SELECT * FROM alumno",$db);
    while ($myrow = mysql_fetch_array($result)) {
        printf("<a href='\"%s?id=%s\"'>%s %s</a><br>\n", $PHP_SELF, $myrow["id"],
        $myrow["nombre"], $myrow["apellidos"]);
    }
    }
    ?>
</body>
</html>
```

También se puede utilizar el método UPDATE para actualizar la información de la DB

```
<html>
<body>
<?php
$db = mysql_connect("localhost", "root");
mysql_select_db("mibase",$db);
if ($id) {
    if ($submit) {
        $sql = "UPDATE alumno SET
nombre='$nombre',apellidos='$apellidos',direccion='$direccion',comentarios='$comentarios' WHERE id=$id";
        $result = mysql_query($sql);
        echo "Gracias! informacion actualizada.\n";
    } else {
        // consulta de la DB
        $sql = "SELECT * FROM alumno WHERE id=$id";
        $result = mysql_query($sql);
        $myrow = mysql_fetch_array($result);
        ?>
        <form method="post" action="<?php echo $PHP_SELF?>">
        <input type="hidden" name="id" value="<?php echo $myrow["id"] ?>">
        Nombre:<input type="Text" name="nombre" value="<?php echo $myrow["nombre"]
?>"><br>
        Apellidos:<input type="Text" name="apellidos" value="<?php echo
$myrow["apellidos"] ?>"><br>
        Dirección:<input type="Text" name="direccion" value="<?php echo
$myrow["direccion"] ?>"><br>
        Comentarios:<input type="Text" name="comentarios" value="<?php echo
$myrow["comentarios"] ?>"><br>
        <input type="Submit" name="submit" value="Enter informacion">
        </form>
        <?php
        }
    } else {
        //lista de alumnos
        $result = mysql_query("SELECT * FROM alumno",$db);
        while ($myrow = mysql_fetch_array($result)) {
            printf("<a href=\"%s?id=%s\">%s %s</a><br>\n", $PHP_SELF, $myrow["id"],
$myrow["nombre"], $myrow["apellidos"]);
        }
    }
?>
</body>
</html>
```

## 7. Gestión de una base de alumnos.

```
<html>
<body>
<?php
$db = mysql_connect("localhost", "root");
mysql_select_db("mibase",$db);
if ($submit) {
    // si no hay ID estamos añadiendo, si no es edicion
    if ($id) {
        $sql = "UPDATE alumno SET
nombre='$nombre',apellidos='$apellidos',direccion='$direccion',comentarios='$comentarios' WHERE id=$id";
    } else {
        $sql = "INSERT INTO alumno (nombre,apellidos,direccion,comentarios) VALUES
('$nombre','$apellidos','$direccion','$comentarios')";
    }
    $result = mysql_query($sql);

    echo "Registro actualizado o editado<p>";

} elseif ($delete) {
    // borra un registro
    $sql = "DELETE FROM alumno WHERE id=$id";
    $result = mysql_query($sql);
    echo "$sql Registro borrado!<p>";
} else {
    // esta parte del codigo es cuando no presionamos submit
    if (!$id) {
        // imprime la lista si no edito
        $result = mysql_query("SELECT * FROM alumno",$db);
        while ($myrow = mysql_fetch_array($result)) {
            printf("<a href='%s?id=%s'>%s %s</a> \n", $PHP_SELF, $myrow["id"],
$myrow["nombre"], $myrow["apellidos"]);
            printf("<a href='%s?id=%s&delete=yes'>(BORRAR)</a><br>", $PHP_SELF,
$myrow["id"]);
        }
    }
?>
<P>
<a href="<?php echo $PHP_SELF?>">AÑADIR UN REGISTRO</a>
<P>
<form method="post" action="<?php echo $PHP_SELF?>">
<?php
if ($id) {
```

```
// para editar debo seleccionar un registro
$sql = "SELECT * FROM alumno WHERE id=$id";
$result = mysql_query($sql);
$myrow = mysql_fetch_array($result);
$id = $myrow["id"];
$nombre = $myrow["nombre"];
$apellidos = $myrow["apellidos"];
$direccion = $myrow["direccion"];
$comentarios = $myrow["comentarios"];

// imprime la ID
?>
<input type=hidden name="id" value="<?php echo $id ?>">
<?php
}
?>
nombre:<input type="Text" name="nombre" value="<?php echo $nombre ?>"><br>
apellidos:<input type="Text" name="apellidos" value="<?php echo $apellidos ?>"><br>
direccion:<input type="Text" name="direccion" value="<?php echo $direccion
?>"><br>
comentarios:<input type="Text" name="comentarios" value="<?php echo $comentarios
?>"><br>
<input type="Submit" name="submit" value="Enter information">
</form>
<?php
}
?>
</body>
</html>
```

