

2 - PHPMYADMIN

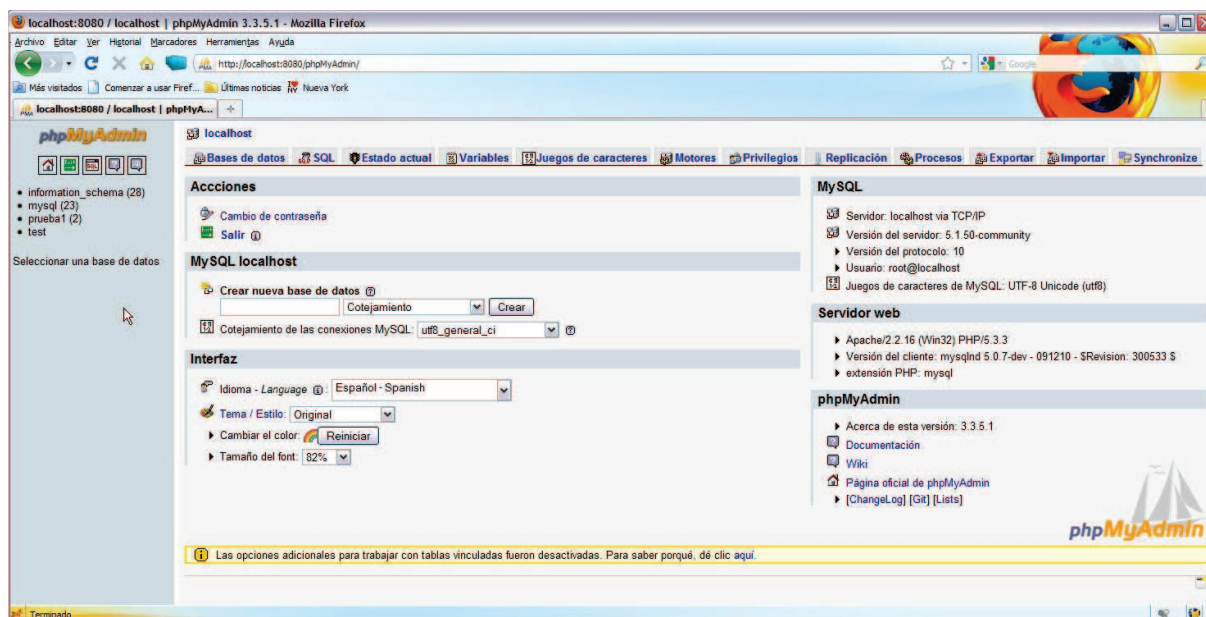
phpMyAdmin es una aplicación desarrollada por gente externa a MySQL, que podemos catalogar dentro de las aplicaciones “cliente” pero, a diferencia de **mysql**, es un cliente gráfico o, mejor dicho, un cliente web, en el que todas las acciones se pueden hacer de forma más intuitiva, aunque más elaborada, que desde línea de comandos.

phpMyAdmin nos va a permitir, de forma centralizada, consultar aspectos variados de la configuración del servidor (conjuntos de caracteres, colaciones, variables, estado), gestionar los usuarios con sus privilegios y realizar distintas operaciones con las bases de datos, tablas y registros (crear, examinar, insertar, modificar, borrar, importar, exportar, ejecutar comandos SQL, ...).

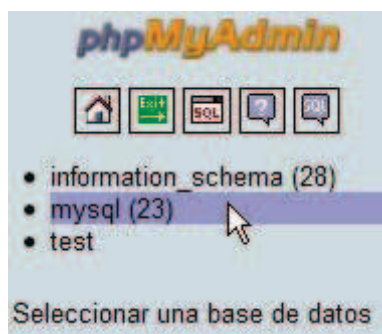


De ahora en adelante, para todas las acciones que se realicen sobre el SGBD MySQL, se proporcionará una descripción sobre cómo realizarlas desde **phpMyAdmin** y desde la aplicación **mysql** o cualquier otra desde línea de comandos.

Veamos el funcionamiento básico de la aplicación, de la que iremos entrando en más detalle en el próximo apartado.



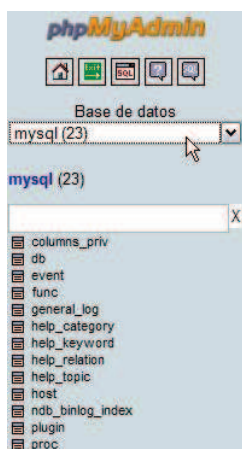
Ventana principal de phpMyAdmin



En la parte izquierda de la aplicación tenemos un conjunto de botones que proporcionan las siguientes funcionalidades (de izquierda a derecha):

- Mostrar la página principal de la aplicación
- Salir de la aplicación (log out)
- Abrir una ventana de comandos MySQL
- Ver la ayuda (local) de phpMyAdmin
- Ver la ayuda (on-line) de MySQL.

Además, tenemos una lista con las bases de datos del servidor a las que tenemos acceso (puede haber más). La selección de una base en la lista se puede considerar equivalente a la sentencia “use base_de_datos” en el cliente **mysql**.



Una vez seleccionada una base de datos, la parte izquierda de la aplicación cambia su aspecto para mostrarnos las tablas que contiene la base de datos seleccionada. Además, ahora en un desplegable, podemos seguir viendo las bases de datos a las que tenemos acceso y seleccionar una de ellas para cambiar la base de datos con la que vamos a trabajar.

En esta versión, también disponemos de un campo de texto que nos sirve de filtro a la hora de buscar una tabla o tablas concretas entre el conjunto de tablas de una base de datos.

En la zona central de la aplicación se nos presentan las acciones más comunes relacionadas con el usuario y con el SGBD, junto con las opciones de configuración del interfaz de phpMyAdmin.

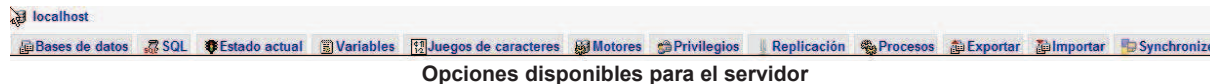


Finalmente, la parte derecha se utiliza para mostrar información sobre la sesión actual como usuario conectado, versión del servidor MySQL, versión del servidor Apache, etc.



2.1 - Navegación por la aplicación

En cada pantalla, tenemos una serie de botones y enlaces en la parte superior que nos permiten movernos por las opciones más comunes para el componente con el que estemos trabajando, así cómo visualizar y seleccionar el elemento con el que queremos trabajar.



Como se puede extraer de las imágenes anteriores, en cada momento podemos ver en la parte superior sobre qué elemento de la base de datos estamos trabajando, y acceder de forma rápida a las distintas acciones que podemos realizar sobre dicho elemento.

3 - OPERACIONES SOBRE MYSQL

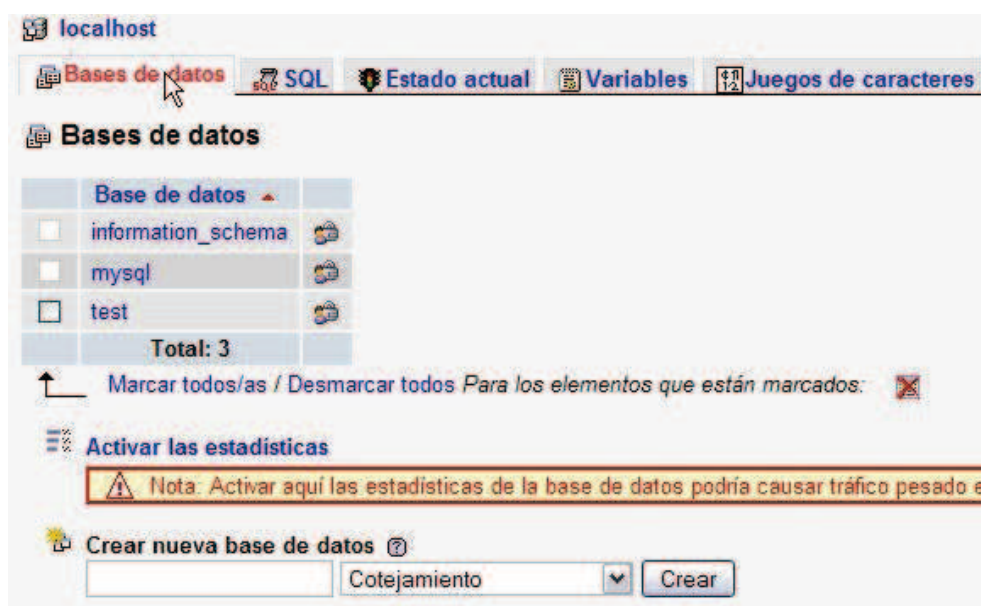
Llegados a este punto, vamos a detallar las acciones más comunes que suelen realizarse con un SGBD, de forma que tengamos un punto de partida a la hora de realizar dichas acciones sobre nuestras propias bases de datos.

3.1 - Gestión de Bases de Datos



La primera necesidad que nos va a surgir es la de crear una base de datos para empezar a trabajar, pudiendo así posteriormente crear tablas, insertar datos, realizar consultas sobre la misma y asignar privilegios a usuarios.

Las características más destacables de las bases de datos son su nombre, junto con el conjunto de caracteres y colación utilizados para la misma.

Desde **phpMyAdmin**, accedemos a la gestión de base de datos pulsando el enlace del mismo nombre en la página principal, o bien, desde la opción *Base de datos* a nivel de servidor.



Como podemos observar, la creación de una BD consiste simplemente en elegir un nombre y, si queremos, cambiar la colación (y por tanto el conjunto de caracteres) por defecto que se aplicará a todos los elementos de la base de datos.

Además, podemos eliminar  las bases de datos que previamente hayamos seleccionado (previa confirmación) o bien, acceder a la gestión de los privilegios que se aplican sobre una base de datos en concreto .

Como última acción dentro de esta sección, pulsando sobre el nombre de una base de datos, accedemos a la gestión de los elementos que la componen y las operaciones que sobre estos podemos realizar.



```
// Creación de una base de datos (parámetros por defecto)
mysql> CREATE DATABASE pruebas;
// Creación de una base de datos (especificando charset y collation)
mysql> CREATE DATABASE pruebas
    -> DEFAULT CHARACTER SET latin1 COLLATE latin1_spanish_ci;
// Eliminación de una base de datos
mysql> DROP DATABASE pruebas;
// Selección de una base de datos para trabajar con ella
mysql> USE pruebas;
```

3.2 - Gestión de tablas

La gestión de tablas, es uno de los pasos más complejos y delicados de la administración y uso de una base de datos. Es el momento de crear los contenedores de la información que queremos gestionar en nuestro sistema, las tablas, y para ello debemos seleccionar el tipo de columna adecuado, las claves principales, las claves ajenas que determinarán las relaciones entre tablas, etc.

Todos los puntos a tener en cuenta, estarán determinados por el esquema de la base de datos, que será nuestro punto de partida a la hora de crear un conjunto de tablas. Vamos a basarnos en el siguiente ejemplo

Esquema de la relación	
Profesor (dni cadena(9) vnn nombre cadena(25) vnn apellidos cadena (50) vnn cuerpo enum ('primaria', 'secundaria') vnn especialidad enum (informática, matemáticas, física y química, dibujo) PK (dni))	
Alumno (expediente entero(5) vnn nombre cadena(25) vnn apellidos cadena (50) vnn fechNacimiento fecha PK (expediente))	Dar_Clase_A (profesor cadena(9) vnn alumno entero (5) vnn asignatura cadena (20) vnn nota1 entero(2) nota2 entero(2) nota3 entero(2) PK (profesor, alumno) AK (profesor -> Profesor.dni) AK (alumno -> Alumno.expendiente))

A la gestión de tablas podemos acceder desde la gestión de base de datos (pulsando sobre el nombre de una base de datos) o bien seleccionando una base de datos en el desplegable de la parte izquierda de la aplicación.



Como todavía no existen tablas en la base de datos, lo único que podemos hacer es crear una nueva tabla. Fijándonos en el esquema anterior, vamos a crear la tabla Profesor (el número de campos se puede modificar posteriormente).

localhost prueba

Estructura SQL Buscar Generar una consulta Exportar Importar Operaciones

Privilegios Eliminar

No se han encontrado tablas en la base de datos.

Crear nueva tabla en la base de datos prueba

Nombre: Profesor Número de campos: 5

Continuar

A continuación debemos especificar las características de cada una de las columnas (nombre, tipo de datos, tamaño, nulos o no, valores por defecto, clave primaria), junto con las características específicas de la tabla (motor de almacenamiento, conjunto de caracteres y colación).

localhost prueba Profesor

Campo	Tipo	Longitud/Valores ¹	Predeterminado ²	Cotejamiento	Atributos	Nulo	Índice	A.I.	Comentarios
dni	CHAR	9	None			<input type="checkbox"/>	PRIMARY	<input type="checkbox"/>	
nombre	VARCHAR	25	None			<input type="checkbox"/>	---	<input type="checkbox"/>	
apellidos	VARCHAR	50	None			<input type="checkbox"/>	---	<input type="checkbox"/>	
cuerpo	ENUM	secundaria	None			<input type="checkbox"/>	---	<input type="checkbox"/>	
especialidad	ENUM	a'dibujo	None			<input checked="" type="checkbox"/>	---	<input type="checkbox"/>	

Comentarios de la tabla: Tabla de profesores

Motor de almacenamiento: InnoDB

Cotejamiento:

definición de la PARTICIÓN:

Guardar Añadir 1 campo(s) Continuar

Repetimos el proceso, primero para la tabla Alumno...

localhost prueba Alumno

Campo	Tipo	Longitud/Valores ¹	Predeterminado ²	Cotejamiento	Atributos	Nulo	Índice	A.I.	Comentarios
expediente	MEDIUMINT	5	None		UNSIGNED	<input type="checkbox"/>	PRIMARY	<input type="checkbox"/>	
nombre	VARCHAR	25	None			<input type="checkbox"/>	---	<input type="checkbox"/>	
apellidos	VARCHAR	50	None			<input type="checkbox"/>	---	<input type="checkbox"/>	
fechaNacimiento	DATE		None			<input checked="" type="checkbox"/>	---	<input type="checkbox"/>	

Comentarios de la tabla: Tabla de alumnos

Motor de almacenamiento: InnoDB

Cotejamiento:

definición de la PARTICIÓN:

Guardar Añadir 1 campo(s) Continuar



... y después para la tabla de “*unión*” Dar_clase_a.

Campo	Tipo	Longitud/Valores ¹	Predeterminado ²	Cotejamiento	Atributos	Nulo	Índice	A.L	Comentarios
profesor	CHAR	9	None			<input type="checkbox"/>	PRIMARY	<input type="checkbox"/>	
alumno	MEDIUMINT	5	None		UNSIGNED	<input type="checkbox"/>	PRIMARY	<input type="checkbox"/>	
asignatura	VARCHAR	20	None			<input type="checkbox"/>		<input type="checkbox"/>	
nota1	TINYINT		None		UNSIGNED	<input checked="" type="checkbox"/>		<input type="checkbox"/>	
nota2	TINYINT		None		UNSIGNED	<input checked="" type="checkbox"/>		<input type="checkbox"/>	
nota3	TINYINT		None		UNSIGNED	<input checked="" type="checkbox"/>		<input type="checkbox"/>	

Una vez creadas las tablas, a falta de establecer las relaciones entre ellas, vamos a ver qué acciones que podemos realizar sobre cada una (o sobre varias a la vez, desde el desplegable inferior) en la vista de gestión de base de datos.

- **Examinar:** Muestra los registros de la tabla, para edición o borrado.
- **Estructura:** Ver y modificar la estructura de la tabla y las relaciones entre tablas (claves ajenas).
- **Buscar:** Configurar una búsqueda personalizada en una tabla.
- **Insertar:** Insertar registros con ayuda para establecer los valores.
- **Vaciar:** Borrar todos los registros de una tabla.
- **Eliminar:** Eliminar la tabla de la base de datos.

Tabla	Acción	Registros ¹	Tipo	Cotejamiento	Tamaño	Residuo a depurar
<input type="checkbox"/> alumno		0	InnoDB	latin1_spanish_ci	16.0 KB	-
<input type="checkbox"/> dar_clase_a		0	InnoDB	latin1_spanish_ci	16.0 KB	-
<input type="checkbox"/> profesor		0	InnoDB	latin1_spanish_ci	16.0 KB	-
3 tabla(s)	Número de filas	0	InnoDB	latin1_spanish_ci	48.0 KB	0 Bytes

Para establecer la relación entre las tablas vamos a ver la estructura de la tabla *Dar_clase_a*, pasando a la vista de gestión de tablas.

Campo	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
<input type="checkbox"/> profesor	char(9)	latin1_spanish_ci		No	None		
<input type="checkbox"/> alumno	mediumint(5)		UNSIGNED	No	None		
<input type="checkbox"/> asignatura	varchar(20)	latin1_spanish_ci		No	None		
<input type="checkbox"/> nota1	tinyint(3)		UNSIGNED	Sí	NULL		
<input type="checkbox"/> nota2	tinyint(3)		UNSIGNED	Sí	NULL		
<input type="checkbox"/> nota3	tinyint(3)		UNSIGNED	Sí	NULL		

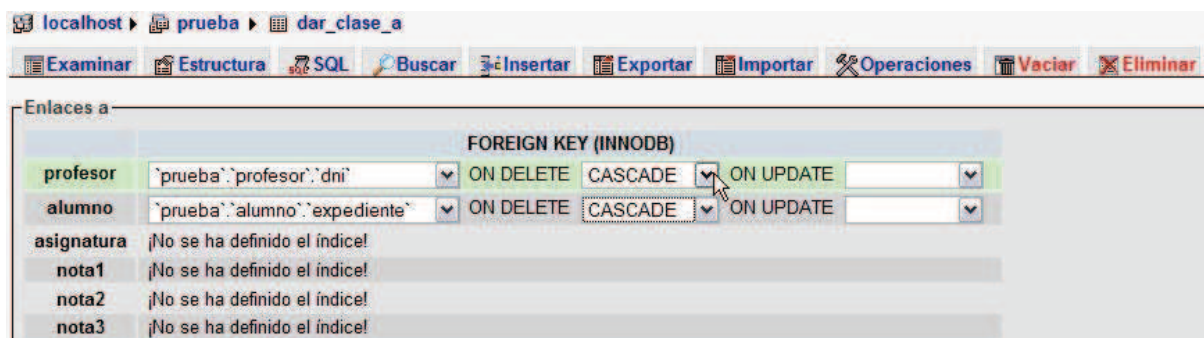
☐ Al final de la tabla
 ☐ Al comienzo de la tabla
 ☐ Después de profesor

Índices:

Acción	Nombre de la clave	Tipo	Único	Empacado	Campo	Cardinalidad	Cotejamiento	Nulo	Comentario
<input checked="" type="checkbox"/>	PRIMARY	BTREE	Sí	No	profesor	0	A		
<input checked="" type="checkbox"/>					alumno	0	A		



En la parte central tenemos el enlace a la **vista de relaciones**, donde vamos a poder indicar la relación de las columnas de esta tabla con las columnas de otras tablas, como podemos ver en la siguiente figura.



Para poder establecer una clave ajena, el campo en la tabla y el campo al cual hace referencia, deben ser exactamente del mismo tipo.
Por ejemplo en nuestro caso, el campo expediente de la tabla Alumno se ha definido como un mediumint(5) unsigned, por lo que el campo alumno de la tabla Dar_clase_a, debe tener la misma definición.

A continuación debemos especificar las características de cada una de las columnas (nombre, tipo de datos, tamaño, nulos o no, valores por defecto, clave primaria), junto con las características específicas de la tabla (motor de almacenamiento, conjunto de caracteres y colación).

```
// Creación de la tabla Profesor
mysql> CREATE TABLE Profesor (
  -> dni CHAR( 9 ) NOT NULL, nombre VARCHAR( 25 ) NOT NULL,
  -> apellidos VARCHAR( 50 ) NOT NULL,
  -> cuerpo ENUM( 'primaria', 'secundaria' ) NOT NULL,
  -> especialidad ENUM( 'informática', 'matemáticas',
  -> 'física y química', 'dibujo' ) NULL,
  -> PRIMARY KEY ( dni )
  -> ) ENGINE = innodb COMMENT = 'Tabla de profesores';

// Creación de las claves ajenas
mysql> ALTER TABLE dar_clase_a
  -> ADD FOREIGN KEY ( profesor )
  -> REFERENCES pruebas.profesor ( dni )
  -> ON DELETE CASCADE ;
mysql> ALTER TABLE dar_clase_a
  -> ADD FOREIGN KEY ( alumno )
  -> REFERENCES pruebas.alumno ( expediente )
  -> ON DELETE CASCADE ;

// Eliminación de una tabla
mysql> DROP TABLE dar_clase_a;
// si hemos escrito todos los comandos necesarios para crear
// las tablas y las claves ajenas en un archivo, podemos lanzar
// las acciones en modo batch
mysql> SOURCE fichero_con_sentencias.sql
```




3.3 - Inserción, Modificación y Borrado de Registros

Para insertar registros en una tabla, nos ubicamos en la vista de gestión de base de datos o bien en la vista de gestión de tablas, y con la opción **Insertar** accedemos a la pantalla desde la que podemos crear nuevos registros, con ayudas a la hora de introducir los datos en las distintas columnas.

Campo	Tipo	Función	Nulo	Valor
expediente	mediumint(5) unsigned			45
nombre	varchar(25)			Vicente
apellidos	varchar(50)			Martínez Martínez
fechaNacimiento	date		<input checked="" type="checkbox"/>	

La modificación y borrado de registros se realiza desde la opción **Examinar**, que nos muestra el detalle de todos los registros en una tabla. Podemos hacerlo registro a registro, o bien seleccionando aquellos sobre los que nos interese realizar alguna acción.

Mostrando registros 0 - 1 (~2¹ total. La consulta tardó 0.0005 seg)

SELECT * FROM 'alumno' LIMIT 0, 20

Mostrar: 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

Organizar según la clave: Ninguna

+ Opciones

	expediente	nombre	apellidos	fechaNacimiento
<input checked="" type="checkbox"/>	45	Vicente	Martínez Martínez	NULL
<input type="checkbox"/>	67	Pedro	Gómez Antón	1986-08-11

Marcar todos/as / Desmarcar todos Para los elementos que están marcados:

Mostrar: 30 filas empezando de 0

en modo horizontal y repetir los encabezados cada 100 celdas

```
// Insertar un registro completo en la tabla
mysql> INSERT INTO alumno VALUES
-> ( 75, UCASE( 'Vicente' ), UCASE( 'Martínez Martínez' ),
-> '1974-12-06' );
// Modificación de un registro
mysql> UPDATE alumno
-> SET apellidos = 'GONZALEZ HEREDIA'
-> WHERE expediente=43;
// Eliminación de un registro
mysql> DELETE FROM alumno where expediente=43;
// si hemos escrito todos los comandos necesarios para insertar
// los registros en las tablas, podemos lanzar las acciones en
// modo batch
mysql> SOURCE fichero_con_sentencias_sql
```

3.4 - Consultas

Las consultas de una tabla determinada las podemos realizar de dos formas diferentes.

La más sencilla es usando la opción de **Examinar**, donde se realiza una consulta de todos los registros de la tabla con todos sus campos.

La segunda forma es desde la opción **Buscar** (presente en diferentes vistas) donde podemos especificar tantos filtros como queramos para refinar nuestra búsqueda entre los datos de la tabla.

Para consultas más complejas, que involucren varias tablas, tendremos que utilizar la ventana de comandos SQL, accesible desde la parte izquierda de la aplicación.

```
// Consultar todos los datos del alumno con expediente 43
mysql> SELECT * FROM alumno WHERE expediente=43;
// Consultar sólo el nombre de los alumnos con expediente > 50
mysql> SELECT nombre FROM alumno WHERE expediente>50;
// Consultar las notas de informática del primer trimestre
mysql> SELECT nombre, apellidos, nota1 FROM
-> FROM alumno AS A, dar_clase_a AS D
-> WHERE A.expediente = D.alumno
-> AND asignatura = 'informática'
```



4 - GESTIÓN DE USUARIOS Y PERMISOS

Otra de las tareas que se suelen realizar con las bases de datos, además de operar con las tablas y la información que estas contienen, hace referencia a la gestión de los usuarios y los permisos que estos tienen sobre las bases de datos.

Ya hemos hablado de este punto a la hora de fijar un password para las cuentas, y también hemos descrito el sistema de privilegios que utiliza MySQL. Vamos a ver ahora como se lleva a la práctica.

Para acceder a la gestión de usuarios y permisos, tenemos el enlace **Privilegios** en la página principal de **phpMyAdmin**. En esta pantalla podemos crear y eliminar usuarios, además de modificar los privilegios de los usuarios existentes en el sistema.

Para agregar un usuario, debemos introducir los parámetros que identifican al usuario en MySQL, esto es, nombre de usuario, host desde el que se le permite el acceso y contraseña.

En ese primer momento, podemos especificar los privilegios globales del usuario en el servidor (aquellos que se guardan en la tabla *user*). Si concedemos un permiso global a un usuario, éste tendrá concedido ese permiso sobre todas las bases de datos, tablas y columnas de la base de datos.

Una vez creado el usuario y otorgados los privilegios globales oportunos, podemos editar los permisos de éste en el resto de componentes del servidor, comenzando por las bases de datos.

Vista global de usuarios

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y

Z

[Mostrar todo]

	Usuario	Servidor	Contraseña	Privilegios globales ¹	Conceder	Acción
<input type="checkbox"/>	cualquiera	%	--	USAGE	No	
<input type="checkbox"/>	root	127.0.0.1	Sí	ALL PRIVILEGES	Sí	
<input type="checkbox"/>	root	localhost	Sí	ALL PRIVILEGES	Sí	
<input type="checkbox"/>	vicente	%	Sí	USAGE	No	

↑

Marcar todos/as / Desmarcar todos



Editar los privilegios

En la pantalla de edición de permisos del usuario, podemos volver a modificar los permisos globales o bien escoger una base de datos sobre la que otorgarle permisos específicos.

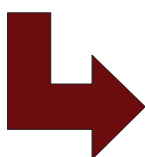
Privilegios específicos para la base de datos

Base de datos Privilegios Conceder Privilegios específicos para la tabla Acción

Ninguna

Añadir privilegios a esta base de datos: Use el campo de texto: 1

Continuar



Editar los privilegios: Usuario 'vicente'@% - Base de datos prueba

Privilegios específicos para la base de datos (Marcar todos/as / Desmarcar todos)

Nota: Los nombres de los privilegios de MySQL están expresados en inglés

Datos	Estructura	Administración
<input type="checkbox"/> SELECT	<input type="checkbox"/> CREATE	<input type="checkbox"/> GRANT
<input type="checkbox"/> INSERT	<input type="checkbox"/> ALTER	<input type="checkbox"/> LOCK TABLES
<input type="checkbox"/> UPDATE	<input type="checkbox"/> INDEX	<input type="checkbox"/> REFERENCES
<input type="checkbox"/> DELETE	<input type="checkbox"/> DROP	
<input type="checkbox"/> CREATE TEMPORARY TABLES		
<input type="checkbox"/> SHOW VIEW		
<input type="checkbox"/> CREATE ROUTINE		
<input type="checkbox"/> ALTER ROUTINE		
<input type="checkbox"/> EXECUTE		
<input type="checkbox"/> CREATE VIEW		
<input type="checkbox"/> EVENT		
<input type="checkbox"/> TRIGGER		

Permite borrar datos.

Continuar

La pantalla de asignación de privilegios sobre la base de datos es muy parecida a la de los privilegios globales. En ella seleccionamos que permisos queremos conceder al usuario sobre toda la base de datos. Además, podemos seleccionar una tabla a la que conceder privilegios específicos.

Privilegios específicos para la tabla

Tabla Privilegios Conceder Privilegios específicos para la columna Acción

Ninguna

Añadir privilegios a esta tabla: Use el campo de texto:

Continuar

La pantalla de asignación de privilegios específicos para una tabla permite conceder ciertos privilegios sobre toda una tabla, o sobre determinadas columnas de una tabla, además de privilegios específicos de las tablas (los que aparecen más a la



derecha).

Editar los privilegios: Usuario 'vicente@%' - Base de datos prueba - Tabla profesor

Privilegios específicos para la tabla¹

SELECT	INSERT	UPDATE	REFERENCES	
dni nombre apellidos cuerpo especialidad	dni nombre apellidos cuerpo especialidad	dni nombre apellidos cuerpo especialidad	dni nombre apellidos cuerpo especialidad	<input type="checkbox"/> DELETE <input type="checkbox"/> CREATE <input type="checkbox"/> DROP <input type="checkbox"/> GRANT <input type="checkbox"/> INDEX <input type="checkbox"/> ALTER <input type="checkbox"/> CREATE_VIEW <input type="checkbox"/> SHOW_VIEW <input type="checkbox"/> TRIGGER
<input type="radio"/> Ninguna	<input type="radio"/> Ninguna	<input type="radio"/> Ninguna	<input type="radio"/> Ninguna	

Permite la creación de tablas nuevas.



No es recomendable que los programadores sepan la clave del administrador del SGBD, y mucho menos que las aplicaciones accedan como root a la BD.

Imaginemos un DELETE * FROM tabla desde una aplicación por un bug en la misma.



Para las aplicaciones PHP que vamos a desarrollar, vamos a crear un usuario **cursophp@localhost** con password **cefire**.

A este usuario le concederemos TODOS los permisos sobre las diferentes bases de datos que creemos durante el curso.

```
// Creamos un usuario, y le denegamos cualquier permiso de acceso
mysql> CREATE USER vicente@localhost
-> IDENTIFIED BY cefireuser;
mysql> GRANT USAGE ON *.* TO vicente@localhost
-> IDENTIFIED BY cefireuser;

// Concesión de privilegios globales
mysql> GRANT ALL PRIVILEGES ON *.*
-> TO vicente@localhost WITH GRANT OPTION;
// Concesión de privilegios sobre una base de datos
mysql> GRANT ALL PRIVILEGES ON pruebas.*
-> TO vicente@localhost WITH GRANT OPTION;
// Concesión de privilegios sobre una tabla
mysql> GRANT ALL PRIVILEGES ON pruebas.alumno
-> TO vicente@localhost WITH GRANT OPTION;
// Concesión de privilegios sobre una columna
mysql> GRANT SELECT ( expediente ) ON pruebas.alumno
-> TO vicente@localhost;

// Eliminar el usuario
mysql> DROP USER vicente@localhost;
```

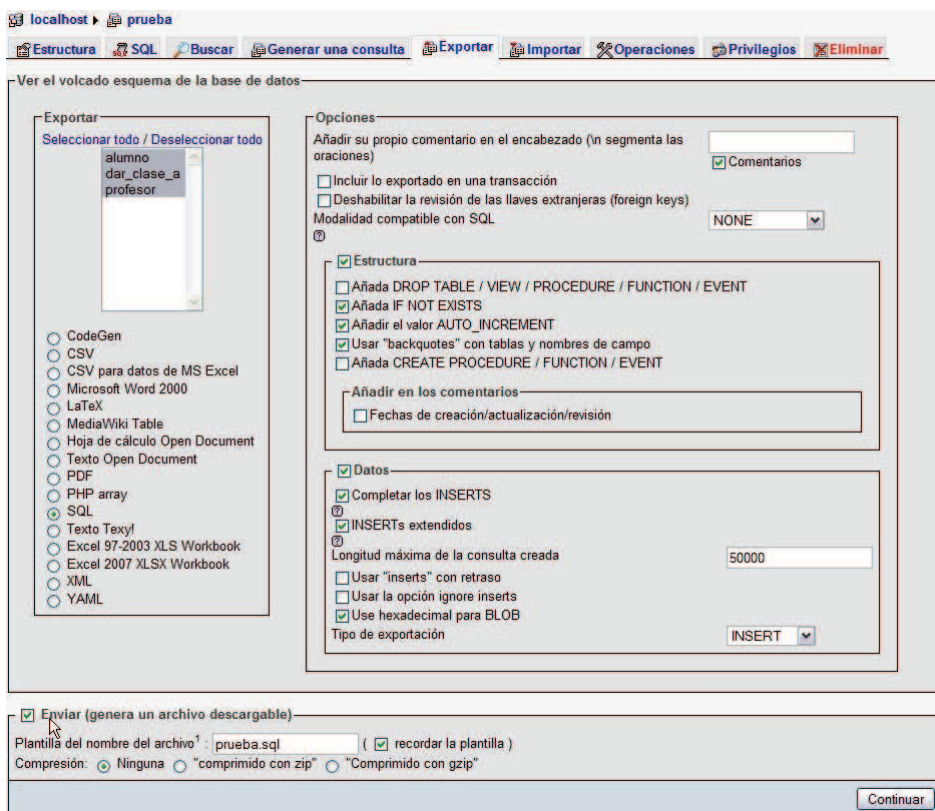

5 - COPIAS DE SEGURIDAD. IMPORTAR Y EXPORTAR.

Después de todo el trabajo que hemos realizado (con lo que nos ha costado), no sería deseable que se pierdan los esquemas de las bases de datos creadas ni la información introducida en las tablas.

Para evitar estas situaciones, tenemos la oportunidad de realizar una exportación del esquema y/o los datos de las tablas.

La opción **Exportar** funciona de forma diferente en función del contexto desde el que se invoque.

- Si lo hacemos desde la vista de gestión de base de datos, podemos volcar el contenido completo de una o más bases de datos,
- Si lo hacemos desde la vista de gestión de tablas, podemos seleccionar la tabla o tablas, de una base de datos, de las que realizar la exportación.
- Si estamos trabajando con una tabla concreta, podremos escoger si realizamos la exportación del esquema de la tabla y/o sus datos.



Las opciones, en cualquiera de sus variantes, son similares.

- En la parte derecha, podemos seleccionar qué elementos (bases de datos, tablas) queremos incluir en la exportación. Además, debemos seleccionar el formato de la exportación. Las opciones comentadas son para formato SQL.



- En la parte derecha, debemos seleccionar si queremos incluir en la exportación la estructura de los elementos seleccionados y si queremos que los datos existentes en las tablas formen parte de dicha exportación. Las distintas opciones en uno y otro apartado, tienen mucho que ver con la utilidad que le vamos a dar a esta exportación.
- Por último, en la parte inferior, podemos indicar que, en lugar de mostrar el resultado por pantalla, genere un archivo que descargamos a nuestro disco de forma automática.

El proceso inverso a la exportación, lo tenemos con la opción **Importar**. En este caso tenemos que proporcionar un fichero con comandos SQL (normalmente el resultado de una exportación) que el servidor ejecutará.

Archivo a importar

Localización del archivo de texto Examinar (Tamaño máximo: 4,096KB)

Juego de caracteres del archivo: utf8

La compresión escogida para el archivo a importar se detectará automáticamente de: Ninguna, gzip, zip

Importación parcial

☒ Permita la interrupción de la importación en el caso de que el script detecte que se ha acercado a su límite de tiempo. Esto podría ser un buen método para importar archivos grandes: sin embargo, puede dañar las transacciones.

Número de registros (consultas) a saltarse desde el inicio

Formato del archivo importado

☐ CSV
☐ Open Document Spreadsheet
☒ SQL
☐ Excel 97-2003 XLS Workbook
☐ Excel 2007 XLSX Workbook
☐ XML

Opciones

Modalidad compatible con SQL

☒ Do not use AUTO_INCREMENT for zero values

Continuar

```
// Realizamos la exportación de la base de datos pruebas
D:\wamp\Mysql5\bin>mysqldump -u root -p pruebas >bdpruebas.sql
// Realizamos la exportación de la tabla alumno
D:\wamp\Mysql5\bin>mysqldump -u root -p pruebas alumno >alumno.sql

// Realizamos la importación de la base de datos pruebas
D:\wamp\Mysql5\bin>mysql -u root -p pruebas < bdpruebas.sql
// Realizamos la importación de la tabla alumno
D:\wamp\Mysql5\bin>mysql -u root -p pruebas < alumno.sql
```

Las acciones de exportación e importación tienen sus propias aplicaciones de línea de comandos, **mysqldump** y **mysqlimport** (aunque este utiliza un formato bastante particular). Las tareas de importación también se pueden realizar mediante la invocación en modo batch del cliente **mysql**.

phpMyAdmin utiliza estos mismos comandos para realizar las importaciones y exportaciones, por lo tanto no vamos a profundizar más en este tema. Se puede consultar la ayuda de los comandos o el manual de referencia para obtener más información.