

Índice

PhpMyAdmin en MySQL	2
Introducción	2
Como cambiar la contraseña de root de MySQL en XAMPP	3
Cómo crear un usuario	5
Cómo crear una Base de Datos con phpMyAdmin en MySQL	8
Crear una tabla	10
Altas, bajas y modificaciones en bases de datos phpMyAdmin	13
Dar de alta un registro	15
Ver un listado con datos	16
Modificar un registro	16
Borrar un registro	17
Copia de seguridad: Cómo exportar una tabla y volver a importarla	17
Cómo agregar una clave foránea en phpMyAdmin	19

PhpMyAdmin en MySQL

Introducción

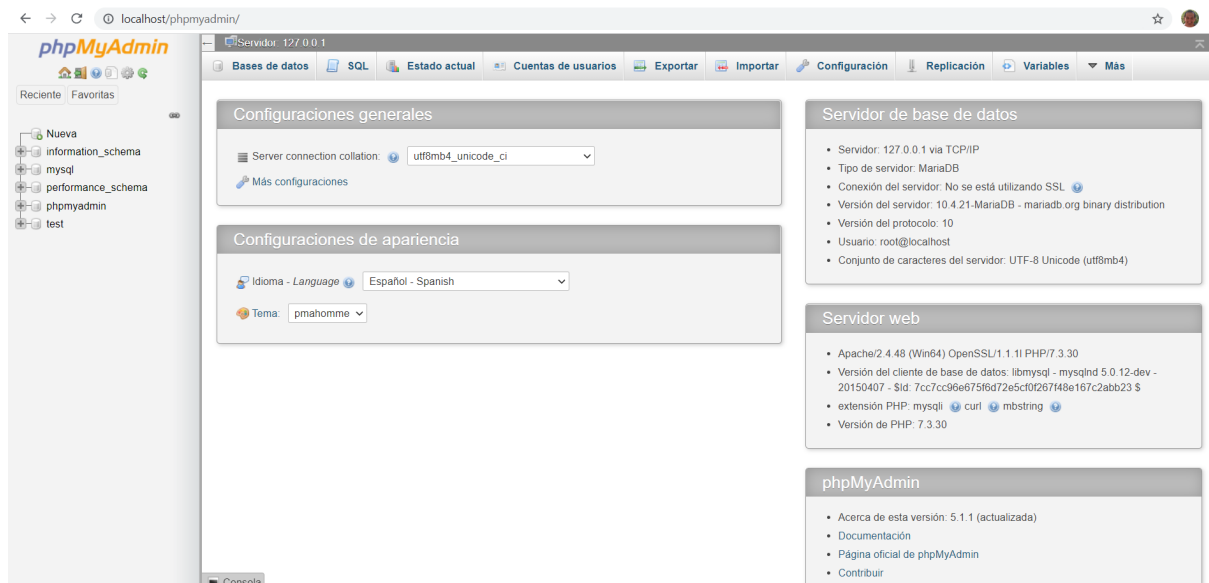
MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base de datos de código abierto más popular del mundo y una de las más populares en general junto a Oracle y Microsoft SQL Server, sobre todo para entornos de desarrollo web.

PhpMyAdmin es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de páginas web, utilizando un navegador web.



Después de haber iniciado el servidor XAMPP, tanto el servicio de Apache como el de MySQL, se debe ingresar a la siguiente url:

`http://localhost/phpmyadmin`



La versión de phpMyAdmin se encuentra en la carpeta **/xampp/phpMyAdmin**. También podemos ponerla en marcha haciendo lo siguiente:

Abrir el panel de administración web (**http://localhost**) y hacer clic en **phpMyAdmin**:



Esto nos abre la dirección <http://localhost/phpmyadmin/>.

Como cambiar la contraseña de root de MySQL en XAMPP

El phpMyAdmin de XAMPP está configurado inicialmente para conectarse directamente a MySQL/MariaDB como usuario **root** sin contraseña, ya que esta es la configuración inicial de MySQL cuando se instala XAMPP. Tener una base de datos con acceso root sin contraseña es algo demasiado peligroso. Vamos a ver por tanto qué tenemos que hacer para poner una contraseña al usuario root.

IMPORTANTE: Vamos a indicar los pasos para cambiar la contraseña, pero como el ordenador es compartido por más de un alumno y es probable que se haga uso de MySQL en otros módulos, no vamos a proceder a cambiar dicha contraseña.

Formas de cambiar la contraseña del usuario root:

1. Cambio de contraseña de usuario de MySQL/MariaDB con phpMyAdmin

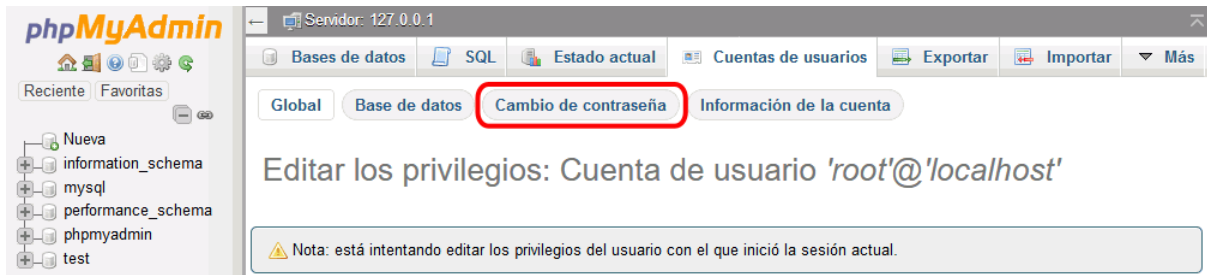
- Haga clic en **Cuentas de usuario**. Se mostrará la lista de cuentas de usuario. Haga clic en **Editar privilegios** de la cuenta root en localhost:

Vista global de las cuentas de usuario

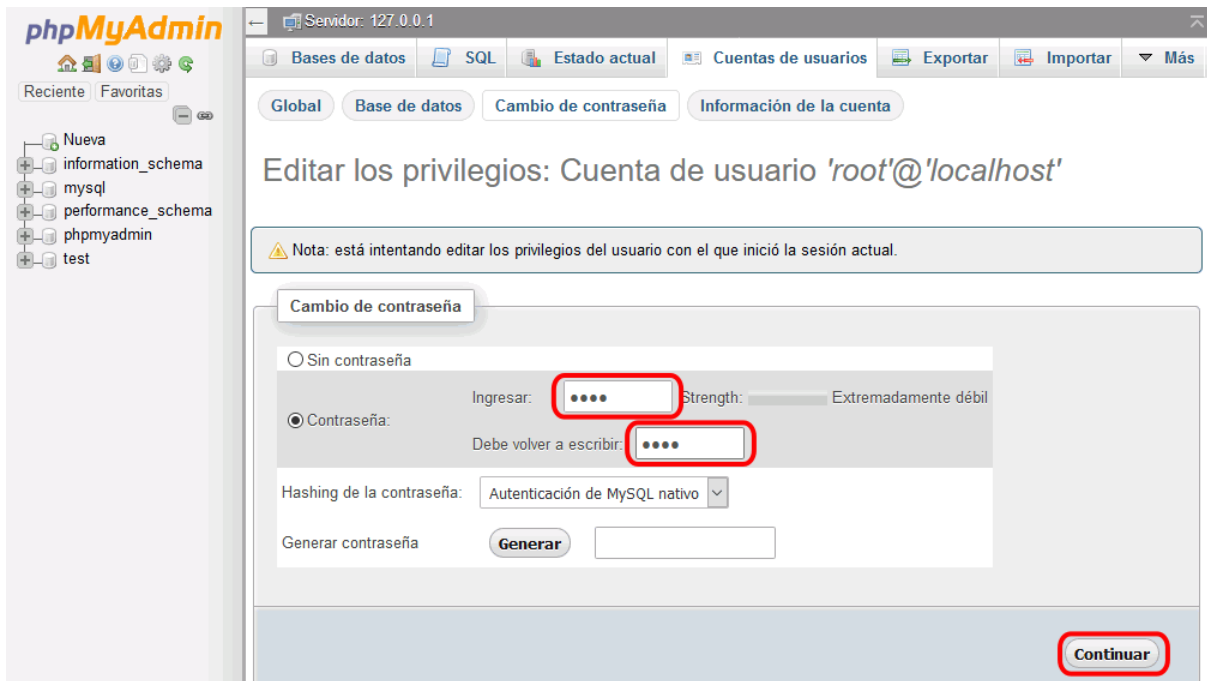
Existe una cuenta de usuario que permite a cualquier usuario de localhost conectarse. Esto evitará conectarse a otros usuarios, si la parte del host de su cuenta permite una conexión desde cualquier host (%).

Nombre de usuario	Nombre del servidor	Contraseña	Privilegios globales	Grupo de usuario	Conceder	Acción
<input type="checkbox"/> cualquiera	%	No	USAGE		No	Editar privilegios Exportar
<input type="checkbox"/> cualquiera	localhost	No	USAGE		No	Editar privilegios Exportar
<input type="checkbox"/> pma	localhost	No	USAGE		No	Editar privilegios Exportar
<input type="checkbox"/> root	127.0.0.1	No	ALL PRIVILEGES	Sí		Editar privilegios Exportar
<input type="checkbox"/> root	:::1	No	ALL PRIVILEGES	Sí		Editar privilegios Exportar
<input type="checkbox"/> root	localhost	No	ALL PRIVILEGES	Sí		Editar privilegios Exportar

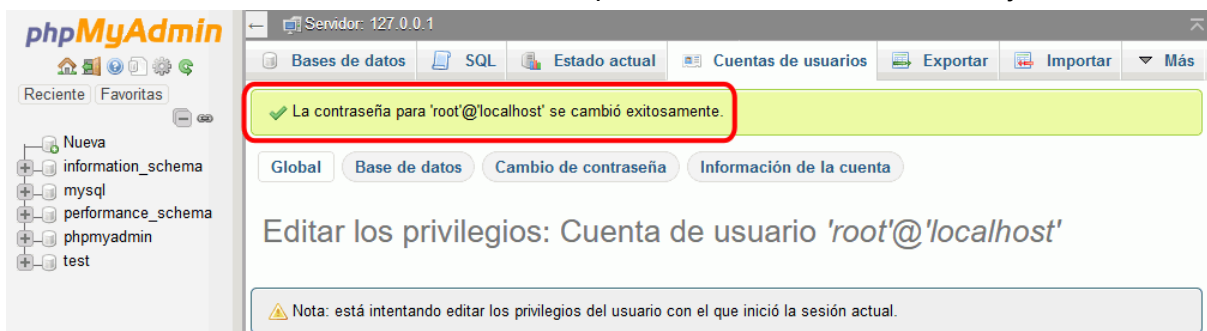
- Haga clic en **Cambio de contraseña**:



- Escriba la nueva contraseña y haga clic en **Continuar**:



- Si el cambio de contraseña se produce, se mostrará un mensaje de éxito:



2. Cambio de contraseña de usuario de MySQL/MariaDB en archivo de configuración de phpMyAdmin:

- Abra con un editor de texto el archivo de configuración **config.inc.php**:

```

17
18  /* Authentication type and info */
19  $cfg['Servers'][$i]['auth type'] = 'config';
20  $cfg['Servers'][$i]['user'] = 'root';
21  $cfg['Servers'][$i]['password'] = '';
22  $cfg['Servers'][$i]['extension'] = 'mysqli';
23  $cfg['Servers'][$i]['AllowNoPassword'] = true;
24  $cfg['Lang'] = '';
25

```

- Escriba la contraseña en la variable `$cfg['Servers'][$i]['password']`:

```

17
18  /* Authentication type and info */
19  $cfg['Servers'][$i]['auth type'] = 'config';
20  $cfg['Servers'][$i]['user'] = 'root';
21  $cfg['Servers'][$i]['password'] = 'root'; // Barto 2018-12-11 Cambio de contraseña
22  $cfg['Servers'][$i]['extension'] = 'mysqli';
23  $cfg['Servers'][$i]['AllowNoPassword'] = true;
24  $cfg['Lang'] = '';
25

```

- Guarde el archivo de configuración modificado.

Cómo crear un usuario

Cualquier aplicación web que se cree lo normal es que utilice una base de datos. Por motivos de seguridad, en servidores reales no se recomienda que las aplicaciones se conecten con el usuario **root**. Para cada aplicación por tanto, es más recomendable crear un usuario que administre la base de datos de dicha aplicación.

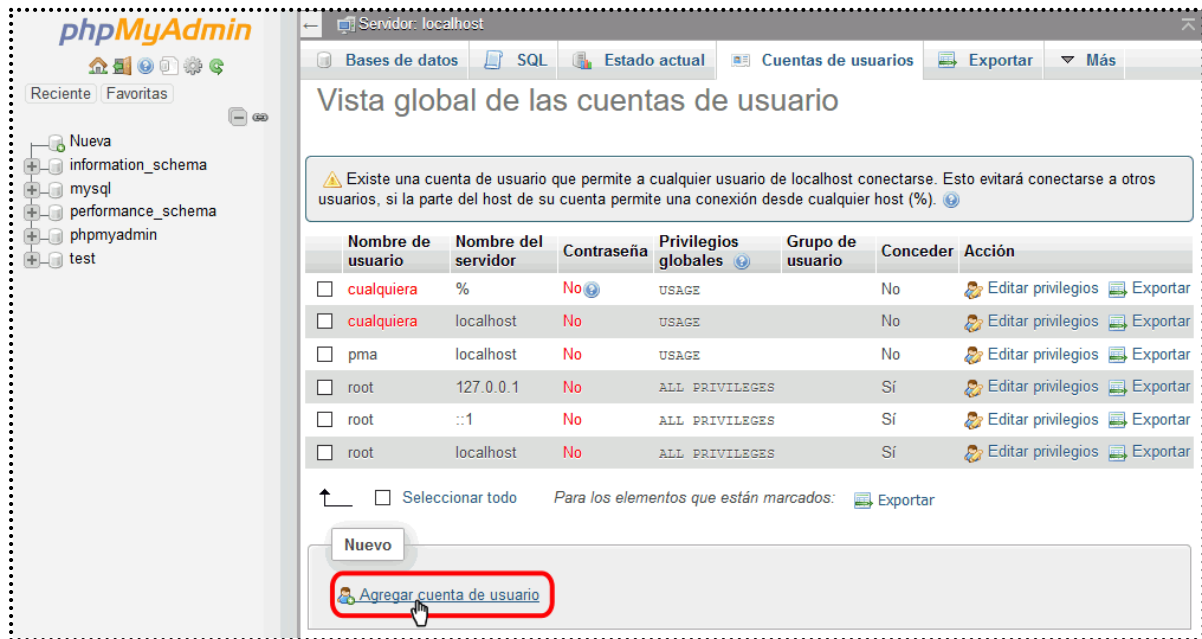
Nosotros no vamos a crear un usuario por cada una de las aplicaciones con las que vamos a trabajar, pero sí vamos a crear un usuario para diferenciar al alumnado de la mañana y de la tarde.

A continuación vamos a ver el procedimiento para crear usuarios en MySQL, Para crear esos usuarios y, en general, administrar MySQL, se utiliza la aplicación phpMyAdmin.

En la página principal, haga clic en el enlace **Cuentas de usuarios**:



Para crear un nuevo usuario, haga clic en el enlace **Agregar cuenta de usuario**:



En la pantalla Agregar cuenta de usuario, agregue un nuevo usuario con la siguiente configuración y haga clic en el botón “Continuar”:

- nombre de usuario: **dwes_manana**
- nombre de host: **localhost**
- contraseña: **dwes_2223**
- base de datos para la cuenta de usuario: **Crear base de datos con el mismo nombre (dwes_manana)**
- sin otorgar todos los privilegios

- Se mostrará el mensaje de creación de usuario y en la lista de bases de datos lateral se mostrará la base de datos **dwes_manana**.

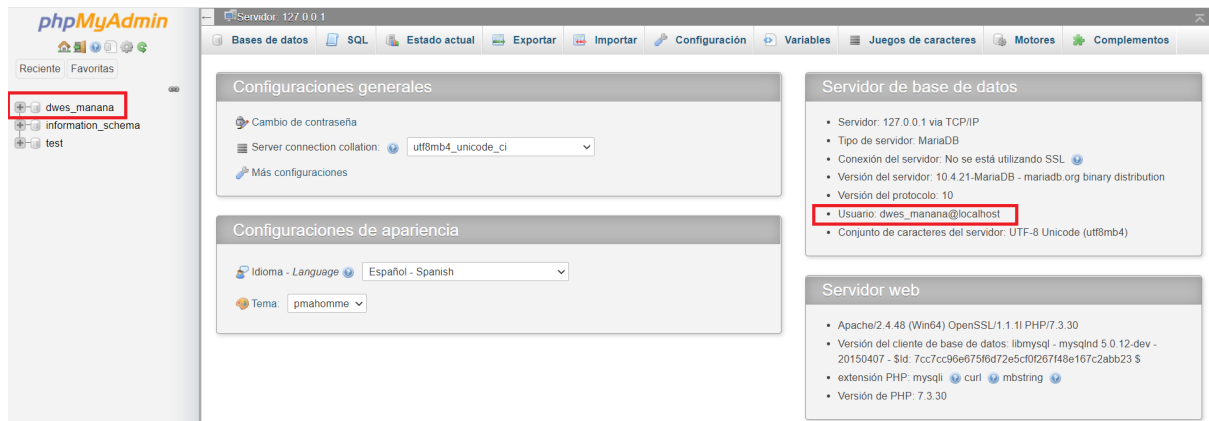


NOTA: Para que al acceder a phpMyAdmin no entremos directamente con el usuario root tenemos que hacer una pequeña modificación en el fichero **config.inc.php**

```
/* Authentication type and info */
/* 2022-09-15. Antonio. He cambiado tipo de autenticación de
phpMyAdmin */
$cfg['Servers'][$i]['auth_type'] = 'cookie';
// $cfg['Servers'][$i]['auth_type'] = 'config';
$cfg['Servers'][$i]['user'] = 'root';
$cfg['Servers'][$i]['password'] = '';
$cfg['Servers'][$i]['extension'] = 'mysqli';
```

```
$cfg['Servers'][$i]['AllowNoPassword'] = true;
$cfg['Lang'] = '';
```

- Al entrar como usuario **dwes_manana**, en el lateral izquierdo se puede ver la base de datos **dwes_manana**.



- Si hacemos clic en la opción **Bases de datos** comprobamos que no se tiene privilegios para crear más bases de datos



NOTA: Una vez identificados, se nos permitirá el acceso al phpMyAdmin. Hay que tener en cuenta que si no utilizamos la interfaz web que gestiona la base de datos por más de 1440 segundos (24 minutos,), nos pedirá que ingresemos nuevamente nuestro usuario y clave para MySQL.

IMPORTANTE: Para entornos de aprendizaje no es cómodo tener que estar creando un usuario por cada base de datos que vayamos a usar. Por lo tanto, el usuario anterior **dwes_manana** lo borraremos y crearemos uno nuevo con el mismo nombre pero con todos los privilegios, marcar la opción "Otorgar todos los privilegios al nombre que contiene comodín (username_%)".

- nombre de usuario: **dwes_manana**
- nombre de host: **localhost**
- contraseña: **dwes_2223**
- Otorgar todos los privilegios al nombre que contiene comodín (username_%)

Información de la cuenta

Nombre de usuario: Use el campo de text

Nombre de Host: Use el campo de text

Contraseña: Use el campo de text Strength: Bueno

Debe volver a escribir:

Authentication plugin: Autenticación de MySQL nativo

Generar contraseña:

Base de datos para la cuenta de usuario

☐ Crear base de datos con el mismo nombre y otorgar todos los privilegios.

☒ Otorgar todos los privilegios al nombre que contiene comodín (username_%).

Cómo crear una Base de Datos con phpMyAdmin en MySQL

Con el **phpMyAdmin** abierto, lo que vamos a hacer ahora es crear una nueva base de datos. En nuestro servidor, de pruebas, lo suyo es crear una nueva base de datos para cada proyecto en el que estemos trabajando.

Para crear una nueva base de datos, dentro del phpMyAdmin escribiremos (en la zona que se resalta a continuación) el nombre que le queremos dar. Para este ejemplo vamos a crear una base de datos con nombre "**dwes_manana_pruebas**",

NOTA: Todas las bases de datos con las que vamos a trabajar a lo largo del curso van a empezar por la coletilla "dwes_manana" para que no haya confusión sobre a qué grupo pertenecen.



Dentro del campo de texto escribimos **dwes_manana_pruebas** (en minúsculas), pues ese será el nombre de nuestra nueva base de datos.

Luego de escribir el nombre, elegimos el juego de caracteres que almacenaremos (para textos en español o normal será el `utf8mb4_spanish_ci`, que corresponde al español tradicional), aunque a modo general se puede utilizar el `utf8mb4_general_ci`.



Finalmente, pulsamos el botón Crear y, a continuación, el nombre de la base recién creada aparecerá en la columna de la izquierda, debajo del mundo de selección que nos muestra todas las bases de datos que tengamos en nuestro servidor, así como también aparece el nombre de la base de datos activa en la ruta superior (miga de pan) que siempre nos indica dónde estamos:



Antes de seguir avanzando, vamos a comprobar que sucedió en el nivel "físico" al crear esta nueva base.

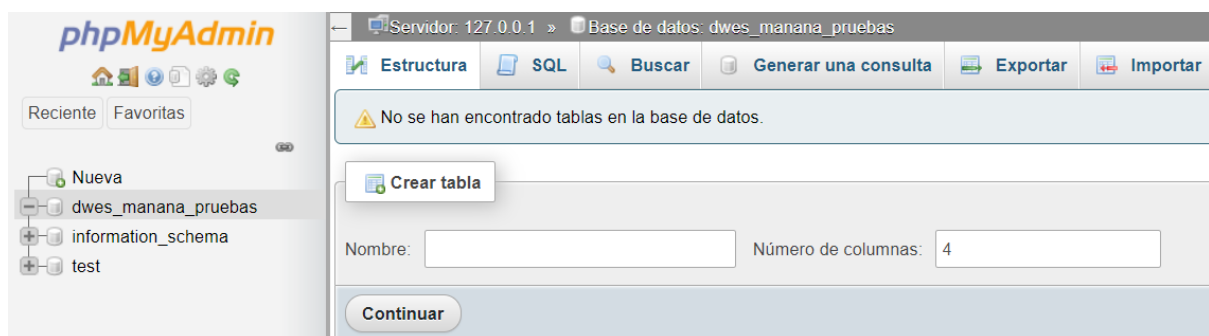
Si estamos usando XAMPP y siempre que hayamos dejado la ruta de instalación por defecto, en **C:\xampp\mysql\data** encontraremos una carpeta por cada base de datos que hayamos creado; en este caso, vemos, al lado de las bases que vienen por defecto, nuestra nueva base **"dwes_manana_pruebas"**:

Crear una tabla

En este punto, ya estamos listos para crear nuestra primera tabla dentro de nuestra base de datos (recordemos que una base de datos es una simple carpeta que organiza nuestras tablas, pero los lugares donde se almacenan realmente los datos son las tablas).

Para ello, primero haremos un clic en la columna izquierda, sobre el nombre de la base dentro de la cual queremos crear una tabla (nuestra base de datos con nombre **"dwes_manana_pruebas"** aún no tiene ninguna tabla creada).

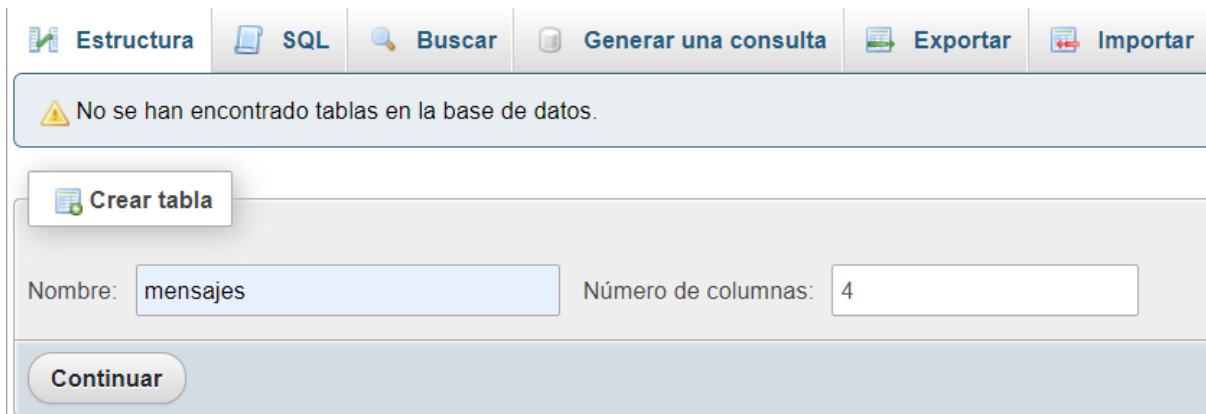
Esto recargará la parte derecha de la pantalla, y veremos un mensaje avisando que todavía no hay tablas en la base:



Ahora podremos crear una tabla muy fácilmente en la base de datos, simplemente escribiendo el nombre de la tabla que creamos y la cantidad de campos (columnas) que deseamos que posea.

Como ejemplo, crearemos una tabla llamada "mensajes" cuyo fin será almacenar el nombre, el correo electrónico y un mensaje que irán dejando los usuarios en un típico formulario de consultas de nuestro sitio web.

A primera vista parecería que la nueva tabla sólo necesitaría tener tres campos, uno para cada dato que almacenará (nombre, correo electrónico y mensaje); pero en las bases de datos las tablas casi siempre utilizan un campo extra, cuyo valor debe ser único en cada registro, siempre diferente, constituyéndose en un código que permite identificar cada registro de forma inequívoca e irrepetible. A este campo extra se lo suele denominar **id** (ya que cumple la función de identificador de cada registro), por lo cual tendremos cuatro campos: id, nombre, email y mensaje.



Estructura SQL Buscar Generar una consulta Exportar Importar

⚠ No se han encontrado tablas en la base de datos.

Crear tabla

Nombre: mensajes Número de columnas: 4

Continuar

Tras pulsar el botón Continuar, aparecerá una pantalla, en la que tendremos que escribir los nombres de cada uno de los cuatro campos o columnas que tendrá nuestra tabla.

1. En el primer campo de texto, justo debajo del título que dice "Nombre", escribiremos el nombre de cada campo (id, nombre, email, mensaje), en ese orden, uno debajo de otro, todos en la primera columna.



Nombre de la tabla: mensajes Agregar 1 column

Nombre	Tipo	Longitud/Valores	Predeterminado
id	INT		Ninguno
nombre	INT		Ninguno
email	INT		Ninguno
mensaje	INT		Ninguno

2. En la segunda columna, denominada Tipo, elegiremos el tipo de dato que podrá almacenar cada uno de estos campos. Ya iremos viendo otros tipos de datos posibles, pero por ahora adelantemos que los tipos de datos normalmente más utilizados son INT (integer, es decir, números enteros, sin decimales, como los que precisa el campo id), VARCHAR (variable carácter o caracteres variables, que almacena letras y números, hasta un máximo de 255 caracteres, como los necesarios para los campos nombre y email), y TEXT (para textos mayores de 255 caracteres, como los de nuestro campo mensaje). Así que elegiremos estos tipos de datos en la columna "Tipo":

Nombre de la tabla: Agregar column

Nombre	Tipo	Longitud/Valores	Predeterminado
<input type="text" value="id"/>	<input type="text" value="INT"/>	<input type="text"/>	<input type="text" value="Ninguno"/>
<input type="text" value="nombre"/>	<input type="text" value="VARCHAR"/>	<input type="text"/>	<input type="text" value="Ninguno"/>
<input type="text" value="email"/>	<input type="text" value="VARCHAR"/>	<input type="text"/>	<input type="text" value="Ninguno"/>
<input type="text" value="mensaje"/>	<input type="text" value="TEXT"/>	<input type="text"/>	<input type="text" value="Ninguno"/>

- En tercera columna, definiremos la cantidad máxima de caracteres que almacenará cada campo (cuatro dígitos para el id si por ejemplo suponemos que no tendremos nunca más de 9999 mensajes), 60 dígitos para cada "nombre" y cada "email". En los campos de tipo TEXT como "mensaje" no deberemos poner nada en longitud, ya que debe quedar vacía.

Nombre	Tipo	Longitud/Valores	Predeterminado
<input type="text" value="id"/>	<input type="text" value="INT"/>	<input type="text" value="4"/>	<input type="text" value="Ninguno"/>
<input type="text" value="nombre"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="60"/>	<input type="text" value="Ninguno"/>
<input type="text" value="email"/>	<input type="text" value="VARCHAR"/>	<input type="text" value="60"/>	<input type="text" value="Ninguno"/>
<input type="text" value="mensaje"/>	<input type="text" value="TEXT"/>	<input type="text"/>	<input type="text" value="Ninguno"/>

- Ahora nos desplazamos hacia la derecha de la pantalla. En la columna "Nulo", si dejamos sin marcar la casilla de selección, haremos que este campo sea **NOT NULL**; es decir, será obligatorio introducir algún valor cuando agreguemos un registro.

Si no queremos que esto sea obligatorio y que se pueda dejar vacío y que se añada igual el registro completo con el resto de campos que si se hubieran completado, entonces marcamos esa casilla de selección, lo que equivale a definir ese campo como potencialmente **NULL**, o sea que pueda ser nulo o vacío.

Por ahora, determinamos todos los campos como NOT NULL, es decir, no se permitirá valores nulos (vacíos) en ninguno de los campos cuando pretendamos

insertar un nuevo registro. Para eso, no tenemos nada que hacer, ya que por defecto las casillas están desmarcadas:

Predeterminado	Cotejamiento	Atributos	Nulo	Índice	A_I
Ninguno			<input type="checkbox"/>	PRIMARY	<input checked="" type="checkbox"/>
Ninguno			<input type="checkbox"/>	---	<input type="checkbox"/>
Ninguno			<input type="checkbox"/>	---	<input type="checkbox"/>
Ninguno			<input type="checkbox"/>	---	<input type="checkbox"/>

- Ahora, exclusivamente en la fila a la que pertenece al campo id (el primero) debemos elegir en la columna Índice la opción **Primary**, tal como vemos en la imagen anterior, lo que indica que ese campo será el que identifica cada registro de forma única, será su clave primaria.

Además, marcaremos la casilla con la abreviatura A_I (Auto Increment), que hace que el contenido o "valor" de este campo id, sea completado automáticamente cada vez que agreguemos un registro, con números que se irán incrementando de uno en uno, sin repetirse nunca.

- Por último pulsamos el botón "Guardar".

Altas, bajas y modificaciones en bases de datos phpMyAdmin

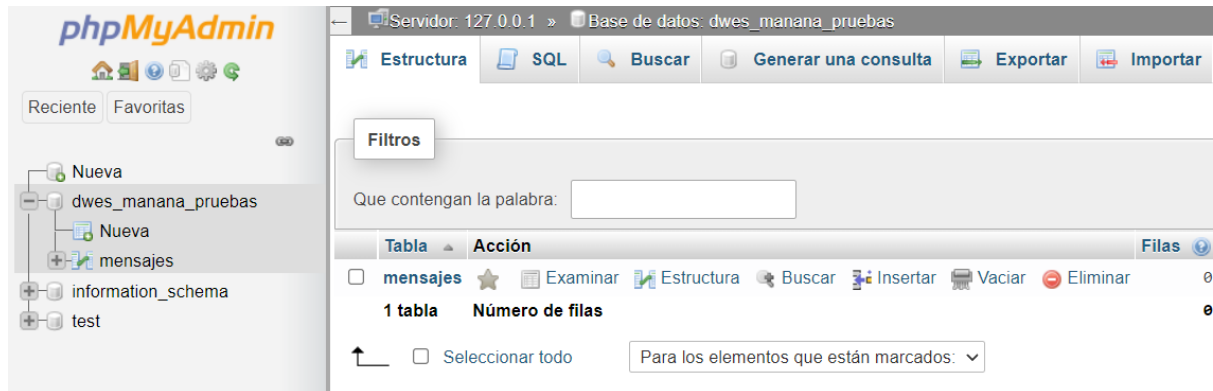
Como hemos visto anteriormente la creación de la estructura de una tabla se puede realizar con una interfaz visual como phpMyAdmin. Esta forma es más sencilla que con código SQL.

Las tareas de manipulación de datos dentro de la tabla ya creada, las haremos desde páginas PHP creadas a tal efecto. Pero, hasta tanto no aprendamos lo necesario como para hacer esas tareas mediante códigos del lenguaje SQL, utilizaremos momentáneamente el phpMyAdmin y realizaremos "visualmente" las principales tareas de administración del contenido de una tabla, que son:

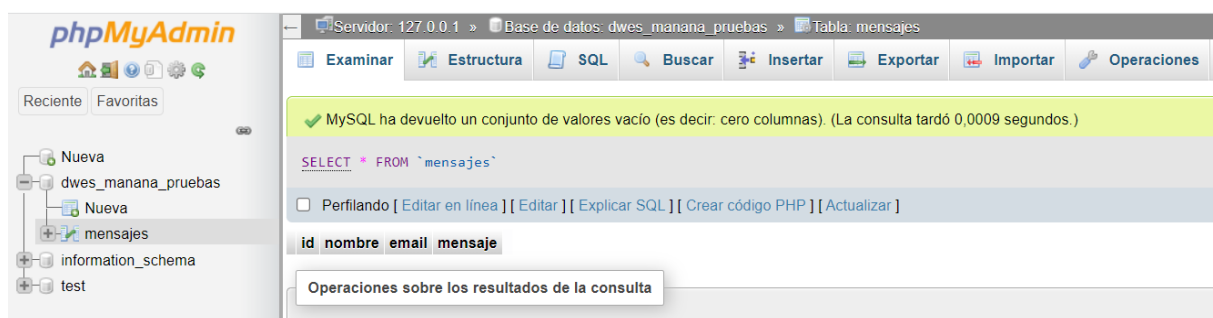
- Agregar nuevos registros.
- Borrar registros.
- Modificar registros.
- Ver un listado de los registros.

También agregamos otra tarea: realizar una copia de seguridad (backup) y restaurarla.

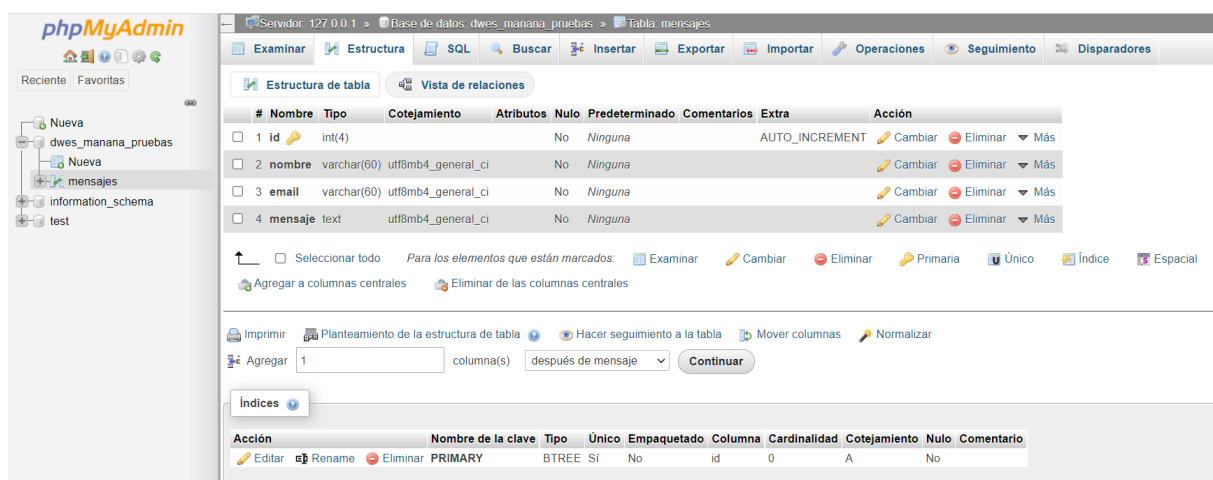
Comenzamos entrando al phpMyAdmin y seleccionando la base de datos **dwes_manana_pruebas**:



Vemos que en el menú de la columna izquierda hay un listado con las tablas que contiene esta base de datos (por ahora hay una sola llamada mensaje). Si pulsamos sobre el nombre de la tabla vemos que aún no tenemos datos



Y si pulsamos sobre "estructura", veremos la estructura de la tabla, (sus campos de datos, etc.), pero no su "contenido".



Dar de alta un registro

Para agregar un registro con datos dentro de los campos de nuestra tabla, pulsamos en el enlace "Insertar". Al pulsar este enlace aparecerá un formulario desde el que podemos agregar uno o dos registros (el segundo es optativo):

Formulario de inserción de registros en la tabla 'mensajes'.

Columna	Tipo	Función	Nulo	Valor
id	int(4)			
nombre	varchar(60)			
email	varchar(60)			
mensaje	text			

En nuestro caso sólo vamos a agregar un registro

Formulario de inserción de registros con datos ingresados.

Columna	Tipo	Función	Nulo	Valor
id	int(4)			
nombre	varchar(60)			Juan
email	varchar(60)			juan@email.com
mensaje	text			Deseo realizar una consulta sobre el curso de Desarrollo Web Entorno Servidor

Continuar

En el campo id no es necesario ingresar ningún valor ya que el propio programa gestor de MySQL le asigna un valor único, sucesivo, automáticamente (al valor más alto existente del

campo id le sumará 1, y ese será el valor que le pondrá al campo id del nuevo registro). Esto se debe a que hemos definido este campo como "A_I" (auto-increment).

Si pulsamos en "Continuar", veremos un mensaje de que se insertó la fila, y ya podremos ver los datos en la pestaña Examinar.



Ver un listado con datos

Si pulsamos en el enlace "Examinar" visualizamos el contenido de nuestra tabla:



En el extremo izquierdo del registro (antes del valor 1 perteneciente al campo id) observaremos dos botones, un lápiz y un símbolo en rojo.

El primero "**Editar**" (modifica) ese registro, el segundo "**Borrar**" elimina el registro seleccionado.

Modificar un registro

Si queremos modificar el valor de alguno de los campos de un registro, pulsaremos el lápiz correspondiente al registro (renglón) que queremos editar, y aparecerá un formulario con los valores de cada campo de este registro ya escritos; en ese momento, escribiremos o borraremos lo que queramos, pulsando, finalmente, en "**Continuar**" para actualizar el registro definitivamente.

Columna	Tipo	Función	Nulo	Valor
id	int(4)			1
nombre	varchar(60)			Juan
email	varchar(60)			juan@email.com
mensaje	text			Deseo realizar una consulta sobre el curso de Desarrollo Web Entorno Servidor

Borrar un registro

Una vez que añadamos algún registro, ya podemos borrar uno, simplemente pulsando el símbolo en rojo ubicado en el renglón del registro que queremos eliminar.

Al borrar un registro nos aparecerá una advertencia que pedirá nuestra confirmación

Mostrando filas 0 - 0 (total de 1, La consulta tardó 0,0011 segundos.)

```
SELECT * FROM `mensajes`
```

Perfilando [Editar en línea] [Explicar SQL] [Crear código PHP] [Actualizar]

Mostrar todo | Número de filas: 25 | Filtrar filas: Buscar en esta tabla

id	nombre	email
1	Juan	juan@email.com

Operaciones sobre los resultados de la consulta

Imprimir | Copiar al portapapeles | Exportar | Mostrar gráfico | Crear vista

Confirmar

¿Realmente desea ejecutar "DELETE FROM `mensajes` WHERE `mensajes`.`id` = 1"?

OK Cancelar

Una vez que le damos en Aceptar, el registro desaparecerá definitivamente de nuestra tabla, así que use esta opción con precaución.

Copia de seguridad: Cómo exportar una tabla y volver a importarla

Una tarea muy necesaria cuando trabajamos con base de datos, es la posibilidad de llevarnos tanto la estructura como los datos en una tabla hacia otro ordenador (hacer un backup de esa base de datos).

Para poder hacer una copia de seguridad tenemos que pulsar en el enlace **"Exportar"**.

The screenshot shows the phpMyAdmin interface for exporting data from the 'mensajes' table. The top navigation bar includes links for 'Examinar', 'Estructura', 'SQL', 'Buscar', 'Insertar', 'Exportar', 'Importar', 'Operaciones', 'Seguimiento', and 'Disparadores'. The main heading is 'Exportando filas de la tabla "mensajes"'. Below this, there are sections for 'Exportar plantillas:', 'Método de exportación:', 'Formato:', and 'Filas:'. The 'Exportar plantillas:' section has a 'Nueva plantilla:' area with a text input and a 'Crear' button, and a 'Plantillas existentes:' area with a dropdown menu and 'Actualizar' and 'Borrar' buttons. The 'Método de exportación:' section has two radio buttons: 'Rápido - mostrar sólo el mínimo de opciones de configuración' (selected) and 'Personalizado - mostrar todas las opciones de configuración posibles'. The 'Formato:' section has a dropdown menu set to 'SQL'. The 'Filas:' section has a radio button for 'Volcar algunas filas' (selected), with input fields for 'Número de filas:' (set to 1) and 'Fila con la que comenzar:' (set to 0).

En la exportación lo normal es seleccionar el Formato SQL y todas las filas pero se pueden exportar a otro formato y sólo algunas filas. Si dejamos las opciones por defecto y pulsamos el botón "Continuar" en nuestro ordenador se genera un archivo de texto con el nombre de la tabla que estamos exportando, con la extensión .sql. Para la tabla mensajes el fichero sería: mensajes.sql.

Este archivo contiene todas las órdenes escritas en lenguaje SQL, necesarias para volver a crear en otro servidor una tabla idéntica a la que hemos exportado.

Este mismo archivo podemos "Importarlo" en otro phpMyAdmin, o en caso de usarlo como copia de seguridad luego de algún borrado de datos, en este mismo servidor, simplemente yendo al menú Importar, y eligiendo este archivo mensaje.sql, que hemos dejado a salvo como copia de respaldo.

Esto reconstruirá la tabla con todos sus campos y con idénticos tipos de datos, y cargará dentro de ella todos los datos (registros) que hubiera tenido almacenados la tabla original en el momento en el que la exportamos.

The screenshot shows the phpMyAdmin interface for importing data into the 'mensajes' table. The top navigation bar includes links like 'Examinar', 'Estructura', 'SQL', 'Buscar', 'Insertar', 'Exportar', 'Importar', 'Operaciones', 'Seguimiento', and 'Disparadores'. The main heading is 'Importando en la tabla "mensajes"'. Under 'Archivo a importar:', there's a text box for the file path with a 'Seleccionar archivo' button highlighted by a red rectangle. Below this, it says 'Buscar en su ordenador:' and 'También puede arrastrar un archivo en cualquier página.' The 'Conjunto de caracteres del archivo:' is set to 'utf-8'. Under 'Importación parcial:', there's a checked checkbox for 'Permitir la interrupción de una importación...' and a text box for 'Omitir esta cantidad de consultas (en SQL) desde la primera:' set to '0'. Under 'Otras opciones:', there's a checked checkbox for 'Habilite la revisión de las claves foráneas'. At the bottom, the 'Formato:' is set to 'SQL'.

NOTA: Tanto la exportación como la importación se puede realizar tanto a nivel de tabla como a nivel de base de datos.

Cómo agregar una clave foránea en phpMyAdmin

¿Qué es una clave externa? En el lenguaje de base de datos SQL en general, y en particular en PHPMyAdmin, una clave externa es un campo de una tabla de la base de datos que depende del campo de otra base de datos.

Esto significa que no es posible crear una entrada en una tabla que haga referencia a través de una clave externa a otra tabla, si la otra tabla no contiene una entrada correspondiente a la clave externa que se está insertando.

Por ejemplo, esto es como unir una botella y su corcho: no se puede usar el corcho si la botella no está allí.

Al agregar una clave externa PHPMyAdmin a una tabla, se asegurará de que cualquier entrada en esa tabla tenga una referencia directa a una entrada de entrada única de otra tabla. Una buena práctica generalmente aceptada es crear una clave externa PHPMyAdmin para hacer referencia a otros campos indexados de tablas, como claves primarias, claves únicas o claves indexadas, para asegurarse de que una sola entrada sea referenciada de manera única por la clave externa PHPMyAdmin creada.

Agregar una clave foránea en phpMyAdmin es bastante simple, pero la opción puede ser difícil de encontrar.

Supongamos las siguientes tablas:

Tabla1

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/> 1	id	int(11)			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/> 2	campo	int(11)			No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 3	FK_tabla2	int(11)			No	Ninguna			Cambiar Eliminar Más

Tabla2

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/> 1	id	int(11)			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/> 2	campo	int(11)			No	Ninguna			Cambiar Eliminar Más

Abra la tabla en la que desea agregar la clave externa (tabla1). Vaya a la pestaña **Estructura**, en la que encontrará un submenú llamado **Vista de relación**.

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Comentarios	Extra	Acción
<input type="checkbox"/> 1	id	int(11)			No	Ninguna		AUTO_INCREMENT	Cambiar Eliminar Más
<input type="checkbox"/> 2	campo	int(11)			No	Ninguna			Cambiar Eliminar Más
<input type="checkbox"/> 3	FK_tabla2	int(11)			No	Ninguna			Cambiar Eliminar Más

En el submenú “vista de relaciones” tiene que seleccionar la columna que contendrá la clave externa y la tabla y columna a las que hace referencia esta clave.

Restricciones de clave foránea

Acciones	Propiedades de la restricción	Columna	Restricción de clave foránea (INNODB)		
			Base de datos	Tabla	Columna
	Nombre de la restricción	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
	ON DELETE	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
	ON UPDATE	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

+ Añadir restricción



También es posible agregar una clave externa con el siguiente código SQL:

```
ALTER TABLE `table1` ADD FOREIGN KEY (`FK_table2`) REFERENCES `table2`(`ID`) ON DELETE RESTRICT ON UPDATE RESTRICT;
```

La creación de una clave externa permite varias opciones, incluido el asegurarse de que una columna solo contenga entradas existentes en otra tabla, la posibilidad de evitar la eliminación de entradas en la otra tabla, o de poner en cascada la eliminación en las tablas a las que se hace referencia.