

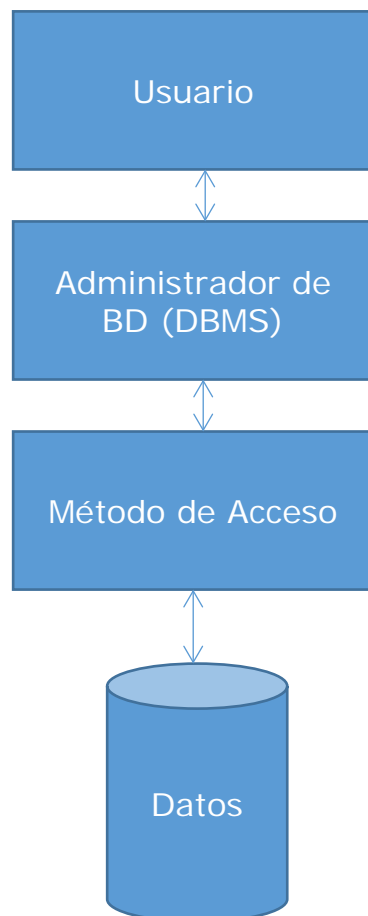
Tecnicatura Superior en Análisis de Sistemas

PROGRAMACIÓN V

Tema 12- Base de datos - MySQL.

Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta. Actualmente, y debido al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos están en formato digital, siendo este un componente electrónico, por tanto se ha desarrollado y se ofrece un amplio rango de soluciones al problema del almacenamiento de datos.

Existen programas denominados sistemas gestores de bases de datos, abreviado SGBD (del inglés Database Management System o DBMS), que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada. Las propiedades de estos DBMS, así como su utilización y administración, se estudian dentro del ámbito de la informática.



Definición

Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular.

Entre las principales características de los sistemas de base de datos podemos mencionar:

- Independencia lógica y física de los datos.
- Redundancia mínima.
- Acceso concurrente por parte de múltiples usuarios.
- Integridad de los datos.
- Consultas complejas optimizadas.
- Seguridad de acceso y auditoría.
- Respaldo y recuperación.
- Acceso a través de lenguajes de programación estándar.

Los Sistemas de Gestión de Base de Datos (en inglés DataBase Management System) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.

En los sistemas de bases de datos todos estos ficheros están integrados, por lo que no se almacenan varias copias de los mismos datos. Sin embargo, en una base de datos no se puede eliminar la redundancia completamente, ya que en ocasiones es necesaria para modelar las relaciones entre los datos.

Eliminando o controlando las redundancias de datos se reduce en gran medida el riesgo de que haya inconsistencias. Si un dato está almacenado una sola vez, cualquier actualización se debe realizar sólo una vez, y está disponible para todos los usuarios inmediatamente. Si un dato está duplicado y el sistema conoce esta redundancia, el propio sistema puede encargarse de garantizar que todas las copias se mantienen consistentes.

En los sistemas de ficheros, los ficheros pertenecen a las personas o a los departamentos que los utilizan. Pero en los sistemas de bases de datos, la base de datos pertenece a la empresa y puede ser compartida por todos los usuarios que estén autorizados.

Gracias a la integración es más fácil respetar los estándares necesarios, tanto los establecidos a nivel de la empresa como los nacionales e internacionales. Estos estándares pueden establecerse sobre el formato de los datos para facilitar su intercambio, pueden ser estándares de documentación, procedimientos de actualización y también reglas de acceso.

La integridad de la base de datos se refiere a la validez y la consistencia de los datos almacenados. Normalmente, la integridad se expresa mediante restricciones o reglas que no se pueden violar. Estas restricciones se pueden aplicar tanto a los datos, como a sus relaciones, y es el SGBD quien se debe encargar de mantenerlas.

La seguridad de la base de datos es la protección de la base de datos frente a usuarios no autorizados. Sin unas buenas medidas de seguridad, la integración de

datos en los sistemas de bases de datos hace que éstos sean más vulnerables que en los sistemas de ficheros.

Muchos **SGBD** proporcionan lenguajes de consultas o generadores de informes que permiten al usuario hacer cualquier tipo de consulta sobre los datos, sin que sea necesario que un programador escriba una aplicación que realice tal tarea.

El hecho de disponer de estas funciones permite al programador centrarse mejor en la función específica requerida por los usuarios, sin tener que preocuparse de los detalles de implementación de bajo nivel.

Sin embargo, los SGBD separan las descripciones de los datos de las aplicaciones. Esto es lo que se conoce como independencia de datos, gracias a la cual se simplifica el mantenimiento de las aplicaciones que acceden a la base de datos.

En este caso, todo el trabajo realizado sobre los datos desde que se hizo la última copia de seguridad se pierde y se tiene que volver a realizar. Sin embargo, los SGBD actuales funcionan de modo que se minimiza la cantidad de trabajo perdido cuando se produce un fallo.

MySQL

Es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base de datos open source 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.

MySQL fue inicialmente desarrollado por MySQL AB (empresa fundada por David Axmark, Allan Larsson y Michael Widenius). MySQL A.B. fue adquirida por **Sun Microsystems** en 2008, y ésta a su vez fue comprada por **Oracle Corporation** en 2010, la cual ya era dueña desde 2005 de Innobase Oy, empresa finlandesa desarrolladora del motor InnoDB para MySQL.

MySQL es usado por muchos sitios web grandes y populares, como **Wikipedia**, **Google** (aunque no para búsquedas), **Facebook**, **Twitter**, **Flickr**, y **YouTube**.

MySQL es muy utilizado en aplicaciones web, como **Joomla**, **Wordpress**, **Drupal** o **phpBB**, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a **PHP**, que a menudo aparece en combinación con **MySQL**.

Sus inicios

MySQL fue creado por una compañía sueca MySQL AB en 1995. Los desarrolladores de la plataforma fueron Michael Widenius (@montywi), David Axmark y Allan Larsson. El objetivo principal era ofrecer opciones eficientes y fiables de gestión de datos para los usuarios domésticos y profesionales. Más de la mitad de una docena de versiones alfa y beta de la plataforma fueron lanzados en 2000. Estas versiones son compatibles con casi todas las principales plataformas.

Open Source

Originalmente la propiedad era de **MySQL AB**, la plataforma de código abierto fue a partir del 2000 y comenzó a seguir los términos de GPL. Ser open source (código abierto) dio lugar a una disminución significativa de los ingresos, sin embargo, se

recuperó con el tiempo. La naturaleza de código abierto de MySQL ha hecho que terceros desarrolladores contribuyan al proyecto.

Expansión de Negocios

MySQL fue ganado constante popularidad entre los usuarios domésticos y profesionales, y en 2001, la plataforma tenía 2 millones de instalaciones activas. En 2002, la compañía amplió su alcance y abrió sede en EE.UU. Además de la sede en Suecia. El mismo año, se anunció que el número de miembros de las plataformas era más de 3 millones de usuarios con unos ingresos por valor de \$ 6.500.000.

Cambio en la Estrategia

La plataforma continúa ganando popularidad en relación a fines de 2003, se podría presumir el ingreso total de \$ 12 millones, con 4 millones de instalaciones activas. En 2004, la empresa decidió centrarse más en los ingresos del usuario final en lugar de cuota de licencia por instalación. La estrategia resultó ser rentable y el año terminó con un ingreso neto de \$ 20 millones.

Adquisición de Innobase por Oracle

En el 2005, Oracle compró Innobase, la empresa que gestiona el almacenamiento backend Innobase de MySQL. Este motor de almacenamiento de MySQL permite la implementación de funciones importantes, como las transacciones y claves foráneas. El mismo año, MySQL Network desarrolla en las líneas de RedHat Red, esto dio lugar a MySQL 5, que amplió considerablemente el conjunto de características disponibles para los usuarios de la empresa. Después de años, se renovó el contrato entre MySQL y Innabose.

MySQL adquirida por Sun Microsystems

En enero de 2008, MySQL fue adquirida por Sun Microsystems por \$ 1 mil millones. La decisión fue criticada por Michael Widenius y David Axmark, los co-fundadores de MySQL AB. En ese momento MySQL ya era la primera opción de las grandes corporaciones, bancos y empresas de telecomunicaciones. El CEO de Sun Microsystems, Jonathan Schwartz, llamó a MySQL “la acción de la raíz” (“The root stock”) de la economía de Internet.

La adquisición de Sun y MySQL por Oracle

La adquisición de MySQL por Sun no resultó muy fructífera y en abril de 2009, se llegó a un acuerdo entre Sun Microsystems y Oracle Corporation, según la cual Oracle fue compra a Sun Microsystems, junto con los derechos de autor y marca registrada de MySQL. El acuerdo fue aprobado por el gobierno de los EE.UU el 20 de Agosto de 2009. Como resultado de la petición en línea iniciada por uno de los fundadores de MySQL Monty Widenius, Oracle se enfrentó a algunas complicaciones legales con la Comisión Europea. Sin embargo, los problemas se resolvieron y en Enero de 2010, la adquisición de MySQL por parte de Oracle se convirtió oficial.

MySQL Forks

Michael Widenius dejó Sun Microsystems después de que fuera adquirida por Oracle y con el tiempo desarrolló una copia (mini-copia) (fork) de MySQL llamado MariaDB. Los Forks son proyectos relacionados que se pueden considerar mini-versiones de MySQL estándar. Hasta la fecha, varios de tales versiones se han puesto en marcha cuyo objetivo es proporcionar una funcionalidad específica. Maria DB es un fork de propiedad comunitaria que significa que no tendría restricciones de licencia habituales que tiene la versión estándar de MySQL. Es compatible con MySQL binary Library de modo que no hay ninguna diferencia entre los comandos y las API.

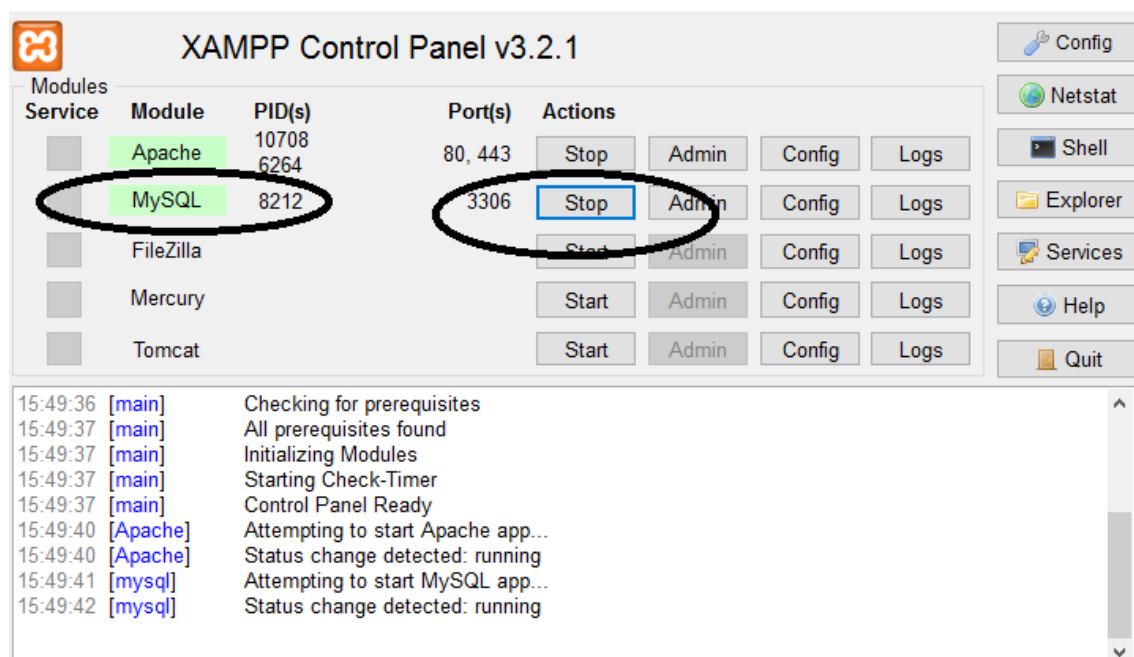
MySQL and Cloud Computing

Las versiones antiguas de MySQL sólo se han desarrollado para máquinas convencionales. Sin embargo, con el advenimiento de la computación en la nube (cloud computing), MySQL se hizo también compatible con diversos servicios de computación en la nube, como Amazon EC2. Varios modelos de implementación se han utilizado para la implementación de MySQL en plataformas de cloud computing. Tal vez el más popular de estos modelos es 'imágenes en máquinas virtuales', que permite el uso de una imagen ya hecha máquina donde esta MySQL preinstalado.

Un segundo modelo de cloud computing es gestionar MySQL cloud hosting donde la base de datos no está disponible como un servicio pero que está alojado y administrado en el nombre del propietario. Este modo, sin embargo, se ofrece sólo para algunas empresas. Con la expansión de la computación en nube y la tecnología relacionada, también se espera que las versiones de **MySQL** para la computación en nube puedan aumentar en número.

Configuración (primeros pasos)

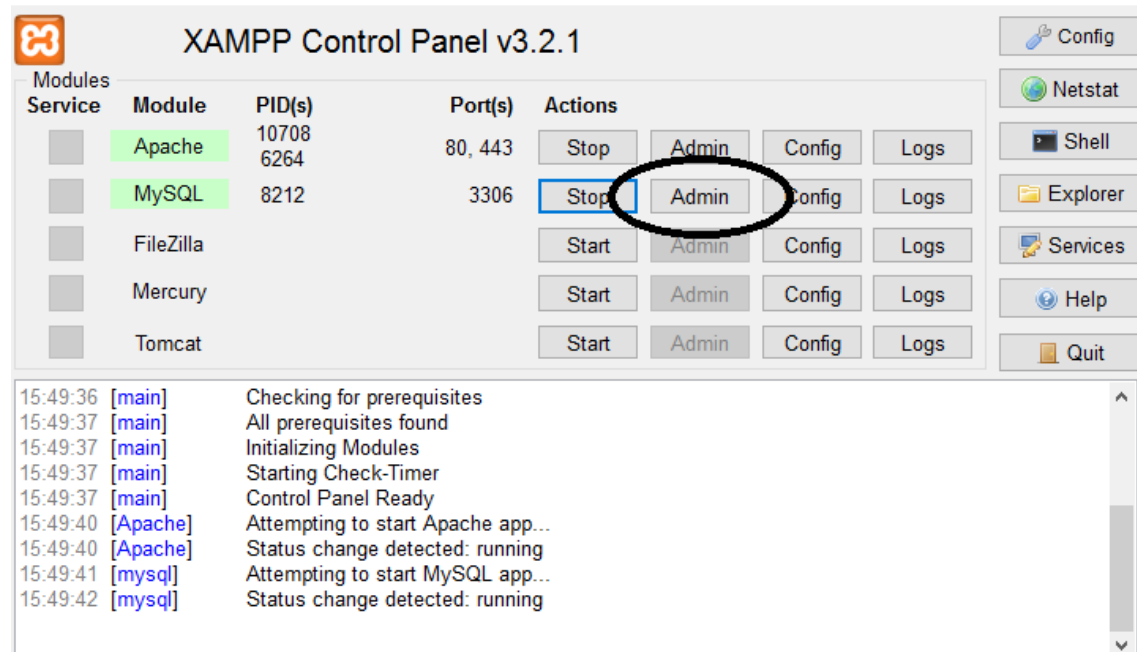
Para comenzar a utilizar MySQL, debemos tener los servicios instalados. Si instalaron correctamente XAMPP, esos servicios se instalan solos. Lo que sí debemos verificar es que el servicio este activo. Para esto vamos al control panel de XAMPP y verificamos que el servicio este activo, caso contrario hacemos clic en el botón **Start** para iniciar el mismo. Si está activo debemos tener una imagen similar a la siguiente figura:



Luego debemos ir al administrador para configurar el usuario y la base de datos.

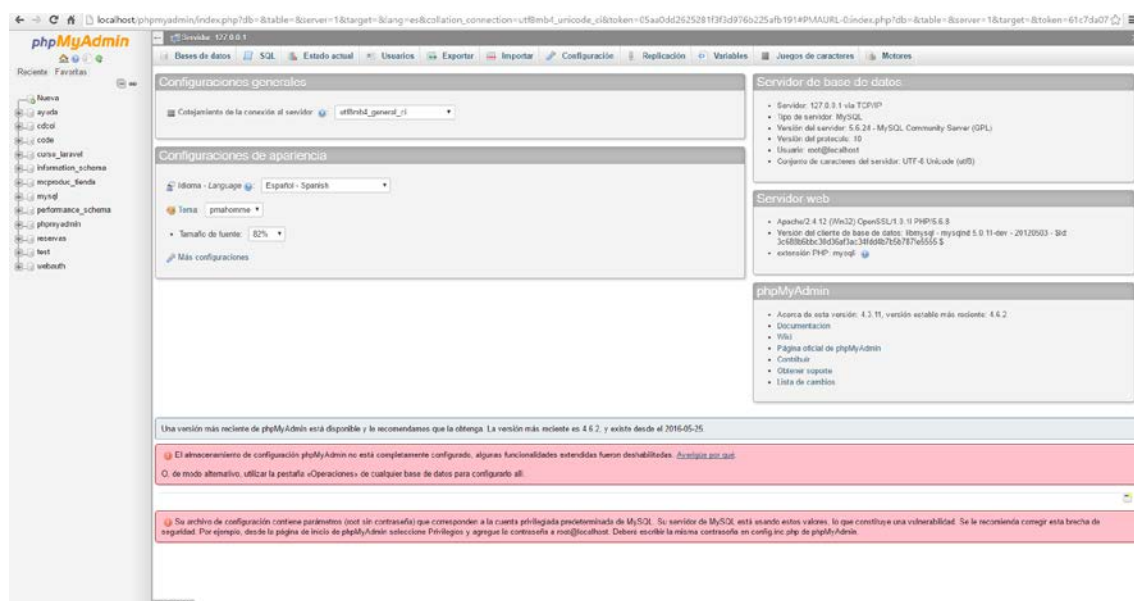
Podemos utilizar un administrador propio como puede ser MySQLFront, que lo podemos bajar de mi página <http://www.murcielagoblanco.com.ar/index.php/descargas> o bien utilizar PHPMyAdmin, que ya viene disponible en el paquete que instalamos con XAMPP. Hay muchos más, cualquiera es bueno.

Nosotros vamos a utilizar PHPMyAdmin, ya que lo tenemos instalado. Para esto podemos ir al panel de control de XAMPP y hacer clic en la opción admin:



Otra opción, más rápida (si el servicio esta iniciado) es desde el navegador, ir directamente al sitio: <http://localhost/phpmyadmin>

Y de acuerdo a la versión que tengamos instalada, vamos a ver algo parecido a esto (digo “parecido” porque cambian algunas cosas de acuerdo a la versión):



Desde el administrador vamos a poder administrar usuarios, crear base de datos, tablas, hacer consultas entre otras cosas. Nosotros sólo vamos a hacer la mayoría de las cosas desde un script en PHP, pero tengan en cuenta que muchas cosas se pueden hacer simplemente desde el administrador.

Para comenzar a utilizar base de datos, es decir conectar desde un script vamos a necesitar crear un usuario que cuente con todos los privilegios.

De acuerdo a la versión que tengamos instalada del administrador, debemos dirigirnos a la solapa “usuarios” o “privilegios” para poder crear un usuario nuevo.



Vista global de usuarios

Usuario	Servidor	Contraseña	Privilegios globales	Conceder	Acción
<input type="checkbox"/> cualquiera	%	--	USAGE	No	Editar los privilegios Exportar
<input type="checkbox"/> cualquiera	localhost	No	USAGE	No	Editar los privilegios Exportar



Vista global de usuarios

Usuario	Servidor	Contraseña	Privilegios globales	Grupo de usuario	Conceder	Acción
<input type="checkbox"/> cualquiera	%	--	USAGE		No	Editar los privilegios Exportar
<input type="checkbox"/> cualquiera	localhost	No	USAGE		No	Editar los privilegios Exportar

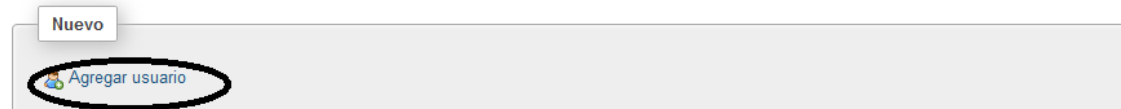
Generalmente el usuario que viene por defecto es **root** y de contraseña nada, es decir sin contraseña. Algunos caso me ha tocado usuario: “**root**” y contraseña: “**1111**”. Pero como dije anteriormente depende mucho de la versión que tengamos instalada.

Para crear un usuario nuevo vamos a “**Vista global de usuarios**” y hacemos clic en **Nuevo -> Agregar usuario**.

Vista global de usuarios

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

↑ ☐ Marcar todos Para los elementos que están marcados: Exportar



Al hacer clic se nos va a abrir una ventana nueva con toda la configuración del usuario nuevo:

Agregar usuario

Información de la cuenta

Nombre de usuario:

Servidor:

Contraseña:

Debe volver a escribir:

Generar contraseña:

Base de datos para el usuario

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

☐ Otorgar todos los privilegios al nombre que contiene comodín (usuario%_%)

Privilegios globales

Ver los nombres de los privilegios de rights están representados en right

Datos

- ☐ SELECT
- ☐ INSERT
- ☐ UPDATE
- ☐ DELETE
- ☐ TRIGGER
- ☐ EXECUTE

Estructura

- ☐ CREATE
- ☐ ALTER
- ☐ DROP
- ☐ RENAME
- ☐ CREATE TEMPORARY TABLES
- ☐ SHOW VIEW
- ☐ CREATE VIEW
- ☐ ALTER VIEW
- ☐ EXECUTE
- ☐ CREATE VIEW
- ☐ EXECUTE
- ☐ TRIGGER

Administración

- ☐ GRANT
- ☐ REVOKE
- ☐ ANALYZE
- ☐ OPTIMIZE
- ☐ LOCK TABLES
- ☐ UNLOCK TABLES
- ☐ REPLICATION CLIENT
- ☐ REPLICATION SLAVE
- ☐ CREATE USER

Límites de recursos

Para el campo de parámetro de hora (hora a 2 dígitos), verifique el tipo

MAX CONNECTIONS PER HOUR:

MAX CONNECTIONS PER MINUTE:

MAX CONNECTIONS PER SECOND:

MAX USER_CONNECTIONS:

Continuar

En el campo **nombre de usuario** asignamos un nombre al usuario nuevo, en **servidor** si es local debemos escribir **localhost** o bien desde el combo elegimos la opción **local** y agregamos una **contraseña**.

En **Base de datos para el usuario**, hacemos clic en la opción **Crear base de datos con el mismo nombre y otorgar todos los privilegios**. Y en **Privilegios globales** hacemos clic en **Marcar todos**, de esta forma le damos todos los privilegios al usuario nuevo.

De esta forma al hacer clic en **continuar**, ya tendremos nuestro usuario creado. **Recordemos el usuario y contraseña porque es lo que vamos a utilizar al crear el script de conexión a la base.**

Actividades

1. Iniciar el servicio de base de datos MySQL y entrar al administrador de la misma.
2. Crear un usuario nuevo con todos los privilegios de root.

Autoevaluación

1. ¿Qué es MySQL?
2. ¿Actualmente que empresa es propietaria de MySQL?
3. ¿Defina las formas de entrar al administrador de la base?
4. ¿Es necesario crear un usuario nuevo al ingresar?
5. Diga por qué a la respuesta de la pregunta 4.