



Módulo
Bases de Datos

**Desarrollador de aplicaciones
Full Stack**

Python Trainee



Objetivo de la jornada

- Realiza la conexión a una base de datos utilizando las herramientas utilitarias para su posterior operación.

3.1.- Bases de datos relacionales

3.1.1.- El rol de las bases de datos relacionales

Las aplicaciones de hojas de cálculo, como Microsoft Excel, se utilizan ampliamente como una forma de almacenar e inspeccionar datos. Es fácil ordenar los datos de diferentes maneras y ver las características y patrones en estos con solo mirarlos.

Desafortunadamente, las personas a menudo confunden una herramienta que es buena para inspeccionar y manipular datos con una herramienta adecuada para almacenar y compartir datos complejos y quizás críticos para el negocio. Las dos necesidades suelen ser muy diferentes.

La mayoría de las personas están familiarizadas con una o más hojas de cálculo y se sentirán cómodas con los datos organizados en un conjunto de filas y columnas. LibreOffice es otra buena alternativa y hay bastantes más.

C2			f _x	María Andrade
	A	B	C	D
1	Nombre	Apellido	Nombre Completo	
2	María	Andrade	María Andrade	
3	Juan	Sandoval	Juan Sandoval	
4	Margarita	Murillo	Margarita Murillo	
5	Miguel	Vera	Miguel Vera	
6	Verónica	Castillo	Verónica Castillo	
7	Franciso	Rivera	Franciso Rivera	
8	Elizabeth	Barroso	Elizabeth Barroso	
9	Antonio	Corona	Antonio Corona	
10	Leticia	García	Leticia García	
11				

Esta sencilla hoja de cálculo, indicada en la imagen anterior, incorpora varias características que serán útiles de recordar cuando comencemos a diseñar



bases de datos. Por ejemplo, el nombre y los apellidos se mantienen en columnas separadas, lo que facilita la clasificación de los datos por apellido si es necesario. Entonces, ¿qué hay de malo en almacenar información del cliente en una hoja de cálculo? Las hojas de cálculo están bien, siempre y cuando:

- No tenga demasiados clientes
- No tenga muchos detalles complejos para cada cliente
- No sea necesario almacenar ninguna otra información repetida, como los distintos pedidos que ha realizado cada cliente.
- No quiera que varias personas puedan actualizar la información simultáneamente
- Asegurarse de que se realice una copia de seguridad de la hoja de cálculo con regularidad si contiene datos importantes

Las hojas de cálculo son una idea fantástica y son excelentes herramientas para muchos tipos de problemas. Sin embargo, así como no intentaría (o al menos no debería) intentar clavar un clavo con un destornillador, a veces las hojas de cálculo no son la herramienta adecuada para el trabajo.

Imagínese cómo sería si una gran empresa, con decenas de miles de clientes, mantuviera la copia maestra de su lista de clientes en una simple hoja de cálculo. En una gran empresa, es probable que varias personas necesiten actualizar la lista. Aunque el bloqueo de archivos puede garantizar que solo una persona actualice la lista a la vez, a medida que aumenta el número de personas que intentan actualizar la lista, pasarán más y más tiempo esperando su turno para editar la lista. Lo que nos gustaría es permitir que muchas personas lean, actualicen, agreguen y eliminen filas simultáneamente, y que la computadora se asegure de que no haya conflictos. Claramente, el bloqueo simple de archivos no será adecuado para manejar este problema de manera eficiente.

Otro problema con las hojas de cálculo son sus estrictas dos dimensiones. Supongamos que también quisiéramos almacenar detalles de cada pedido que realizó un cliente. Podríamos comenzar a poner la información de los pedidos junto a cada cliente, pero a medida que aumentara la cantidad de pedidos por cliente, la hoja de cálculo se volvería cada vez más compleja.

Un sistema de administración de bases de datos (DBMS) suele ser un conjunto de bibliotecas, aplicaciones y utilidades que alivian al desarrollador de aplicaciones de la carga de preocuparse por los detalles de almacenamiento y administración de datos. También proporciona facilidades



para buscar y actualizar registros. Los DBMS vienen en varios tipos desarrollados a lo largo de los años para resolver tipos particulares de problemas de almacenamiento de datos.

Cuando se mira superficialmente, una base de datos relacional (RDB), como PostgreSQL, MySQL u Oracle, tiene muchas similitudes con una hoja de cálculo. Sin embargo, cuando se conoce la estructura subyacente de una base de datos, puede ver que es mucho más flexible, principalmente debido a su capacidad para relacionar tablas de formas complejas. Puede almacenar de manera eficiente datos mucho más complejos que una hoja de cálculo, y también tiene muchas otras características que lo convierten en una mejor opción como almacén de datos. Por ejemplo, una base de datos puede administrar varios usuarios simultáneamente.

Pensemos en almacenar nuestra lista de clientes simple de una sola hoja en una base de datos, para ver qué beneficios podría tener.

Las bases de datos se componen de tablas, o en terminología más formal, relaciones. Nos ceñiremos al uso del término tablas. Una tabla contiene filas de datos y cada fila de datos consta de varias columnas o atributos.

3.1.2. Características de un RDBMS

Varias reglas importantes definen un sistema de gestión de bases de datos relacionales (RDBMS). Todas las filas deben seguir el mismo patrón, ya que todas tienen el mismo número y tipo de componentes. A continuación, se muestra un ejemplo de un conjunto de filas:

```
{"Chile", "CLP", 1000,56}  
{"Bélgica", "EUR", 1,34}
```

Cada una de estas filas tiene tres atributos: un nombre de país (cadena), una moneda (cadena) y un tipo de cambio (un número de punto flotante). En una base de datos relacional, todos los registros que se agregan a este conjunto, o tabla, deben seguir el mismo formulario, por lo que no se permiten los siguientes:

```
{"Germany", "EUR"}
```

Esto tiene muy pocos atributos.

```
{"Switzerland", "CHF", "French", "German", "Italian", "Romansch"}
```



Esto tiene demasiados atributos.

<code>{1936.27, "EUR", "Italy"}</code>
--

Esto tiene tipos de atributos incorrectos (orden incorrecto).

Además, en cualquier tabla de filas, no debe haber duplicados. Esto significa que en cualquier tabla en una base de datos relacional correctamente diseñada, no puede haber filas o registros idénticos. Esto podría parecer una restricción bastante draconiana. Por ejemplo, en un sistema que registra los pedidos realizados por los clientes, parecería no permitir que el mismo cliente pida el mismo producto dos veces. Más adelante, veremos que hay una manera fácil de evitar este tipo de problemas/requisitos agregando un atributo.

Cada atributo de un registro debe ser "atómico"; es decir, debe ser un solo dato, no otro registro o una lista de otros atributos. Además, el tipo de atributos correspondientes en cada registro de la tabla (elementos de la columna) debe ser el mismo. Técnicamente, esto significa que deben extraerse del mismo conjunto de valores o dominio. En términos prácticos, significa que todos serán una cadena, un número entero, un valor de punto flotante o algún otro tipo admitido por el sistema de base de datos.

El atributo (o atributos) que se utilizan para distinguir un registro en particular en una tabla de todos los otros registros se llama clave primaria o clave principal (primary key). En una base de datos relacional, cada relación o tabla debe tener una clave principal para cada registro para que sea único, diferente de todos los demás en esa tabla.

3.1.3.- Conociendo las herramientas para consultar una base de datos

Los programas que permiten administrar bases de datos, organizan y almacenan los datos de manera tal que las tablas están indexadas y se pueden contestar preguntas. Estas herramientas están disponibles en un número de configuraciones diferentes y escalables para ser usadas por un individuo o una corporación global. Algunas son fáciles de usar, con bases de datos a manera de interfaz visual, mientras que otras, al final del espectro empresarial, requieren entrenamiento y herramientas especializadas para ser útiles.



A continuación se indicarán las más conocidas y utilizadas en la actualidad.

- **Microsoft Access:** Es un sistema de base de datos personal de Microsoft. Se trata de un producto de software orientado hacia lo visual, lo que hace que quienes no sean programadores puedan crear bases de datos útiles con facilidad. Si bien la estructura de base de datos de Access puede ampliarse para satisfacer las necesidades empresariales, su uso más común es para pequeñas bases de datos individuales o en programas multiusuario de uso limitado. Access integra el lenguaje Visual Basic para aplicaciones, por lo que es un entorno de desarrollo completo.
- **MySQL:** es una base de datos basada en servidor que permite a varios usuarios acceder a múltiples bases de datos. El software funciona en múltiples plataformas, incluyendo la mayoría de las variedades de UNIX y Windows. Ofrece usabilidad de primer plano limitada y está diseñado como un servidor de base de datos back-end. MySQL se diferencia de otros productos de base de datos por sus costos, la versión no empresarial se distribuye de forma gratuita.
- **SQL Server:** es un servidor de base de datos a nivel empresarial escalable. Este producto se diferencia de la base de datos personal al no proporcionar las herramientas para el usuario que proporciona un producto de base de datos individual. El motor de base de datos se centra en responder rápidamente a las solicitudes del cliente en el formulario de consultas SQL. Estas consultas se pueden generar directamente en SQL Server, o por medio de una interfaz de usuario independiente desarrollada en una variedad de lenguajes de programación. SQL Server está diseñado para manejar bases de datos con millones de registros.
- **Oracle:** es otra base de datos escalable a nivel empresarial. Soporta bases de datos corporativas distribuidas, que permiten al usuario acceder a los datos de forma local o desde bases de datos remotas en una transacción transparente. Las bases de datos distribuidas ayudan a superar las limitaciones físicas de un entorno informático físico.

3.1.4.- Instalando la base de datos y sus herramientas utilitarias

Cada motor o software que permite administrar bases de datos tiene su procedimiento particular, dependiendo también del sistema operativo en el que se haga.

En este caso se indicará cómo instalar MySQL en su versión 8.0 en un equipo local, habilitando también una herramienta que permitirá consultar los datos.



1. Acceder al sitio de descarga de complementos de MySQL.
 - a. <https://www.mysql.com/downloads/> [2]
2. En la parte central, hacer clic en el enlace "MySQL Community (GPL) Downloads »".
3. En la página que se desplegará, seleccionar la opción "MySQL Installer for Windows".
4. Aparecerán dos opciones de descarga; se debe elegir la segunda de ellas, la que pesa poco más de 400 MB. Esto se hace presionando el botón "Download".
5. Con lo anterior se desplegará un botón de login y otro de registro. En realidad no es necesario presionar ninguno de ellos, así que puedes hacer clic en el enlace "No thanks, just start my download".
6. Se descargará un archivo con extensión *.msi, qué es el instalable para Windows; se debe hacer doble clic sobre él.
7. En la sección que permite elegir un tipo de instalación, se debe seleccionar la primera opción de las cuatro, esto es, "Developer Default". Esta opción no solo instalará el motor de MySQL en su última versión, sino también MySQL Workbench y otros complementos que te permitirán crear, consultar y administrar bases de datos en un entorno local, e incluso si lo deseas de manera remota. Una vez seleccionada la opción presiona "Next".
8. En las secciones siguientes indicará datos para la customización del entorno. Se recomienda dejar estos datos tal y como están, especialmente el puerto sobre el cual operará este aplicativo en modo local.
9. Otro aspecto que se solicitará en la instalación es la clave de administrador del servidor local, denominado comúnmente "root". No debes olvidar esta clave.
10. Una vez que se hayan instalado todas las herramientas necesarias, aparecerá un botón "Finish", el cual al ser presionado terminará el proceso.

3.1.5.- Creando una conexión a la base de datos

Gracias a la instalación del paso anterior, se instalará MySQL como motor de base de datos, y Workbench como herramienta de gestión. Además, es necesario considerar que junto con la instalación de estos aplicativos, se cargarán datos de prueba que permitirán hacer pruebas de concepto.

Para acceder a estos datos se debe seguir las siguientes acciones:



- Ingresa al programa MySQL Workbench (En Windows: Inicio MySQL MySQL Workbench).
- En primera instancia, solo existirá una conexión activa, que corresponde a la instancia creada en el ambiente local. Para ingresar en ella, se debe presionar el botón con nombre "Local instance MySQL 80". Por cierto, puedes agregar más conexiones haciendo clic en el botón "+" ubicado antes de este recuadro.
- En caso que no se haya ingresado antes a la instancia, te solicitará el usuario y la clave respectiva. Recuerda que no es obligación conectarte siempre con el usuario "root", también lo puedes hacer con un usuario especialmente creado con menos permisos que el usuario administrador.
- A fin de probar consultas sobre una base de datos, existen las bases "sakila" y "world". Para ingresar a alguna de ella, se debe seleccionar la pestaña Schemas ubicada en el menú de la izquierda, junto con el menú "Administration". Ya dentro de esta pestaña, se debe presionar la flecha a un costado izquierdo y con ello desplegar su contenido.
- En cada base de datos, al momento de desplegar sus componentes, se despliegan las opciones:
 - o **Tablas:** es un conjunto de datos que contienen los mismos registros. Todos los registros (llamados comúnmente filas) de una tabla tienen los mismos datos que los representan; estos últimos se conocen como campos.
 - o **Vistas:** es una definición lógica de una tabla, formada por la unión de uno o más campos de diversas tablas. Las vistas pueden ser lógicas o bien materializadas.
 - o **Procedimientos almacenados:** son un conjunto de instrucciones que realiza un conjunto de tareas específica entre una o más tablas, y que retorna un resultado.
 - o **Funciones:** una función es una rutina o conjunto de acciones que aplican sobre un dato específico, transformándolo o bien aplicando operaciones de cálculo y/o transformación sobre el mismo.



Anexo: Referencias

1.- ¿Qué es una base de datos?

Referencia: <https://www.hn.cl/blog/para-que-sirven-la-bases-de-datos/>

2.- Instalación de MySQL en Windows 10

Referencia:

<https://www.solvetic.com/tutoriales/article/9129-instalar-mysql-8-0-22-ser-ver-en-windows-10/>