TALLER MySQL

PROGRAMA TÉCNICO EN PROGRAMACIÓN DE SOFTWARE

SERVICIO NACIONAL DE APRENDIZAJE SENA

CENTRO NACIONAL COLOMBO ALEMÁN

BARRANQUILLA (ATL.)

2018

ACTIVIDAD DE TRANSFERENCIA No 5 – GUÍA DE APRENDIZAJE BASE DE DATOS.

Para el desarrollo de la actividad, utilizaras la BD world.sql y deberás desarrollar una consulta que dé solución a cada situación planteada, contenerlos en un único archivo PDF.

RESUELVE

1. Muestre todos los países de los continentes: África, Europa, Oceanía.
2. Muestre todas las ciudades de Indonesia (IDN).
3. Muestre todas las ciudades del distrito: West Java
4. Muestre los lenguajes de Zimbabwe (ZWE), que no sean oficiales.
5. Muestre las ciudades donde se hable Inglés con un porcentaje mayor a 3.8.
6. Muestre todas las ciudades que hablan francés, que sean oficiales.
7. Muestre ID, Name, District, CountryCode de las ciudades con población mayor a 2173831;
8. Muestre todos los países de la región de North América.
9. Muestre en que países se habla catalán.
10. Muestre la población, superficie y jefe de estado de Syria.
11. Muestre todas las ciudades con CountryCode (PHL).
12. Muestre todos los países de la región Caribbean en la cual la esperanza de vida sea mayor o igual a 70.5.
13. Muestre el país con menor superficie.
14. Muestre los tres países con mayor superficie.
15. Muestre todos los países con código ALB, USA. En los cuales el lenguaje no sea oficial con un porcentaje menor de 4.2.
16. Muestre cuantos países hay registrados en la base de datos.
17. Muestre cuantas ciudades hay registradas en la base de datos.
18. Muestre cuantas ciudades de México (MEX) están registradas en la base de datos.
19. Muestre las ciudades de Arábia Saudita (SAU) del distrito de al-Sharqiya.
20. Muestre los países donde los jefes de estado sean:
    - Alfonso Portillo Cabrera
    - Nursultan Nazarbajev
    - Benjamin William Mkapa
    - Mswati III
    - John Bani
    - Islam Karimov
    - Charles Taylor
21. Muestre todos los países que son Republica en su forma de gobierno.

" LA TIENDA DE INFORMATICA"

INSTRUCCIONES: En el siguiente ejercicio se describen cada uno de los comandos para llevar a cabo cada una de las consultas solicitadas, ejecuta cada uno de ellos en el compilador de MySQL.

ACTIVIDADES A REALIZAR

1. Crea una base de datos llamada TIENDA

```
CREATE DATABASE TIENDA;
```

2. Muestra las Bases de datos almacenadas

```
SHOW DATABASES;
```

3. Habilita la Base de datos TIENDA

```
USE TIENDA;
```

4. Genera las siguientes tablas:

FABRICANTES		
	NOMBRE DE COLUMNA	TIPO DE DATO
Clave Principal	Clave_fabricante	Int
	Nombre	Varchar(30)

```
CREATE TABLE FABRICANTES (Clave_fabricante INT NOT NULL, Nombre VARCHAR(30), PRIMARY KEY (Clave_fabricante));
```

ARTICULOS		
	NOMBRE DE COLUMNA	TIPO DE DATO
Clave Principal	Clave_articulo	Int
	Nombre	Varchar(30)
	Precio	Int
Clave Foránea	Clave_fabricante	Int

```
CREATE TABLE ARTICULOS (Clave_articulo INT NOT NULL, Nombre VARCHAR(30), Precio INT, Clave_fabricante INT, PRIMARY KEY (Clave_articulo), FOREING KEY(Clave_fabricante) REFERENCES FABRICANTES (Clave_fabricante));
```

5. Muestra las tablas de la Base de datos TIENDA

```
SHOW TABLES;
```

6. Muestra los atributos de la tabla ARTICULOS;

```
DESCRIBE ARTICULOS;
```

7. Introduce los siguientes datos en cada tabla:

TABLA: FABRICANTES	
CLAVE_FABRICANTE	**NOMBRE**
1	Kingston
2	Adata
3	Logitech
4	Lexar
5	Seagate

```
INSERT INTO FABRICANTES VALUES ( 1 , 'Kingston');
```

TABLA: ARTICULOS			
CLAVE_ARTICULO	**NOMBRE**	**PRECIO**	**CLAVE_FABRICANTE**
1	Teclado	$ 100	3
2	Disco duro 300 Gb	$ 500	5
3	Mouse	$ 80	3
4	Memoria USB	$ 140	4
5	Memoria RAM	$ 290	1
6	Disco duro extraíble 250 Gb	$ 650	5
7	Memoria USB	$ 279	1
8	DVD Rom	$ 450	2
9	CD Rom	$ 200	2
10	Tarjeta de red	$ 180	3

```
INSERT INTO ARTICULOS VALUES (  1 , 'Teclado', 100 , 3 );
```

8. Genera las siguientes consultas:

a) Obtener todos los datos de los productos de la tienda

```
SELECT * FROM ARTICULOS;
```

b) Obtener los nombres de los productos de la tienda

```
SELECT Nombre FROM ARTICULOS;
```

c) Obtener los nombres y precio de los productos de la tienda

```
SELECT Nombre,Precio  FROM ARTICULOS;
```

d) Obtener los nombres de los artículos sin repeticiones

```
SELECT DISTINCT  Nombre  FROM ARTICULOS;
```

e) Obtener todos los datos del artículo cuya clave de producto es '5'

```
SELECT  *  FROM ARTICULOS WHERE Clave_articulo=5;
```

f) Obtener todos los datos del artículo cuyo nombre del producto es ''Teclado"

```
SELECT  *  FROM ARTICULOS WHERE Nombre='Teclado';
```

g) Obtener todos los datos de la Memoria RAM y memorias USB

```
SELECT  *  FROM ARTICULOS WHERE Nombre='Memoria RAM' OR Nombre='Memoria USB' ;
```

h) Obtener todos los datos de los artículos que empiezan con 'M'

```
SELECT * FROM ARTICULOS WHERE Nombre LIKE 'M%';
```

i) Obtener el nombre de los productos donde el precio sea $ 100

```
SELECT Nombre FROM ARTICULOS WHERE Precio = 100;
```

j) Obtener el nombre de los productos donde el precio sea mayor a $ 200

```
SELECT Nombre FROM ARTICULOS WHERE Precio > 200;
```

k) Obtener todos los datos de los artículos cuyo precio este entre $100 y $350

```
/* OPERADOR AND */

    SELECT * FROM ARTICULOS WHERE Precio >= 100 AND Precio<=350;

/* OPERADOR BETWEEN */

    SELECT * FROM ARTICULOS WHERE Precio BETWEEN 100 AND 350;
```

l) Obtener el precio medio de todos los productos

```
SELECT AVG(Precio) FROM ARTICULOS;
```

m) Obtener el precio medio de los artículos cuyo código de fabricante sea 2

```
SELECT AVG(Precio) FROM ARTICULOS WHERE Clave_fabricante=3;
```

n) Obtener el nombre y precio de los artículos ordenados por Nombre

```
SELECT Nombre, Precio FROM ARTICULOS ORDER BY Nombre;
```

o) Obtener todos los datos de los productos ordenados descendentemente por Precio

```
SELECT * FROM ARTICULOS ORDER BY Precio DESC;
```

p) Obtener el nombre y precio de los artículos cuyo precio sea mayor a $ 250 y ordenarlos descendentemente por precio y luego ascendentemente por nombre

```
SELECT Nombre, Precio FROM ARTICULOS
WHERE Precio >= 250 ORDER BY Precio DESC, Nombre;
```

q) Obtener un listado completo de los productos, incluyendo por cada articulo los datos del articulo y del fabricante

```
SELECT * FROM ARTICULOS, FABRICANTES
WHERE ARTICULOS.Clave_fabricante=FABRICANTES.Clave_fabricante
```

r) Obtener la clave de producto, nombre del producto y nombre del fabricante de todos los productos en venta

```
SELECT ARTICULOS.Clave_articulo, ARTICULOS.Nombre, FABRICANTES.Nombre
FROM ARTICULOS, FABRICANTES
WHERE ARTICULOS.Clave_fabricante=FABRICANTES.Clave_fabricante
```

s) Obtener el nombre y precio de los artículos donde el fabricante sea Logitech ordenarlos alfabéticamente por nombre del producto

```
SELECT ARTICULOS.Nombre, ARTICULOS.Precio FROM ARTICULOS, FABRICANTES
WHERE FABRICANTES.Nombre='Logitech' AND
 ARTICULOS.Clave_fabricante=FABRICANTES.Clave_fabricante
ORDER BY ARTICULOS.Nombre;
```

t) Obtener el nombre, precio y nombre de fabricante de los productos que son marca Lexar o Kingston ordenados descendentemente por precio

```
SELECT ARTICULOS.Nombre, ARTICULOS.Precio, FABRICANTES.Nombre
FROM ARTICULOS, FABRICANTES
WHERE FABRICANTES.Nombre='Lexar' OR FABRICANTES.Nombre='Kingston'
AND ARTICULOS.Clave_fabricante=FABRICANTES.Clave_fabricante
ORDER BY ARTICULOS.Precio DESC;
```

u) Añade un nuevo producto: Clave del producto 11, Altavoces de $ 120 del fabricante 2

```
INSERT INTO ARTICULOS VALUES (11 ,'Altavoces', 120, 2);
```

v) Cambia el nombre del producto 6 a 'Impresora Laser'

```
UPDATE ARTICULOS SET Nombre='Impresora Laser'
WHERE Clave_articulo = 8
```

w) Aplicar un descuento del 10% a todos los productos.

```
UPDATE ARTICULOS SET Precio=Precio *0.10
```

x) Aplicar un descuento de $ 10 a todos los productos cuyo precio sea mayor o igual a $ 300

```
UPDATE ARTICULOS SET Precio=Precio – 10
WHERE  Precio >= 300
```

y) Borra el producto numero 6

```
DELETE FROM ARTICULOS WHERE Clave_articulo= 6
```

ACTIVIDAD Nº 7

COMPETENCIA

ANALIZAR LOS REQUISITOS DEL CLIENTE PARA CONSTRUIR EL SISTEMA DE INFORMACION

PROYECTO

DESARROLLO E IMPLEMENTACIÓN DE SISTEMA DE INFORMACIÓN PARA EL MIPYME

INSTRUCTOR

RICARDO LLANOS TEJERA

PROGRAMA DE FORMACIÓN TÉCNICO EN PROGRAMACIÓN DE SOFTWARE

CENTRO NACIONAL COLOMBO ALEMÁN

SERVICIO NACIONAL DE APRENDIZAJE SENA

BARRANQUILLA ATLANTICO

2020

EJERCICIO PROPUESTOS

Debido a la contingencia de salud y las medidas tomadas para la continuidad del programa el grupo debe trabajar de forma colaborativa en casa en grupos máximo de 2 Aprendices y realizar su aporte a la entrega unificada de la actividad por classroom.

1. Crea un programa que pida por pantalla la cantidad de países y a continuación tres ciudades de cada uno de esos países. Los nombres de ciudades deben almacenarse en un Array multidimensional cuyo primer índice sea el número asignado a cada país y el segundo índice el número asignado a cada ciudad.
   Ejemplo de resultados que debe mostrar el programa:

País: España	Ciudades:	Madrid	Lugo	Sevilla
País: Francia	Ciudades:	Paris	Niza	Lyon
País: Italia	Ciudades:	Roma	Napoles	Sicilia

GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL PROCEDIMIENTO DESARROLLO CURRICULAR GUÍA DE APRENDIZAJE

1. IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE:

DENOMINACIÓN DEL PROGRAMA DE FORMACIÓN:

TÉCNICO EN PROGRAMACIÓN DE SOFTWARE.

CÓDIGO DEL PROGRAMA DE FORMACIÓN:

228120.

NOMBRE DEL PROYECTO:

DESARROLLO E IMPLEMENTACIÓN DE SISTEMA DE INFORMACIÓN PARA EL MIPYME

FASE DEL PROYECTO:

PLANEACIÓN Y LEVANTAMIENTO DE REQUERIMIENTOS DEL SISTEMA DE INFORMACIÓN.

ACTIVIDAD DE PROYECTO:

CONSTRUIR DIAGRAMA ENTIDAD RELACION DE LA BASE DE DATOS UTILIZADA EN LA APLICACIÓN.

COMPETENCIA:

220501032 - ANALIZAR LOS REQUISITOS DEL CLIENTE PARA CONSTRUIR EL SISTEMA DE INFORMACIÓN.

RESULTADOS DE APRENDIZAJE ALCANZAR:

1. INTERPRETAR LOS DIAGRAMAS DE CASO DE USO, DE OBJETOS, DE ESTADO, DE SECUENCIA, DE PAQUETES O COMPONENTES, DE DESPLIEGUE, DE COLABORACIÓN SEGÚN EL DISEÑO ENTREGADO.
2. COMPRENDER FRASES Y VOCABULARIO HABITUAL SOBRE TEMAS DE INTERÉS PERSONAL Y TEMAS TÉCNICOS.
3. ASUMIR RESPONSABLEMENTE LOS CRITERIOS DE PRESERVACIÓN Y CONSERVACIÓN DEL MEDIO AMBIENTE Y DE DESARROLLO SOSTENIBLE, EN EL EJERCICIO DE SU DESEMPEÑO LABORAL Y SOCIAL.
4. RECONOCER EL ROL DE LOS PARTICIPANTES EN EL PROCESO FORMATIVO, EL PAPEL DE LOS AMBIENTES DE APRENDIZAJE Y LA METODOLOGÍA DE FORMACIÓN, DE ACUERDO CON LA DINÁMICA ORGANIZACIONAL DEL SENA.

Duración de la Guía: 120 horas para lograr los resultados de aprendizaje: 96 horas de trabajo directo y 24 horas de trabajo independiente.

2. PRESENTACIÓN:







La tecnología a nivel mundial avanza a gran escala, claramente podemos observar como el desarrollo tecnológico se acelera, trayendo con ello múltiples beneficios para el ser humano, quien cada vez cuenta con muchas más herramientas tecnológicas que ayudan a facilitar la realización de sus actividades cotidianas, ya sea en su hogar, ambientes públicos o su lugar de trabajo. Conceptos como las ***aplicaciones móviles*** y la ***Cloud Computing***, se posicionan en el mundo, moviendo constantemente grandes cantidades de información, y es por eso que es importante crear bases de conocimientos que nos permitan entender los procesos de captación, almacenamiento estructurado y manipulación de la información, para ello en el transcurrir de esta guía, apoyándonos en un aprendizaje colaborativo desarrollaremos las temáticas de: Bases de datos (BD), Sistema Gestor de Base de Datos (SGBD), Modelo Entidad Relación (MER) y Diccionario de Datos, con sus respectiva conceptualización y aplicabilidad.

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE:

3.1. ACTIVIDADES DE REFLEXIÓN INICIAL:

3.1.1. ACTIVIDAD No 1: reflexionar sobre la introducción al mundo de las bases de datos.

En esta primera actividad de reflexión inicial, el instructor técnico a cargo de la formación basado en su experiencia realizará una breve introducción al mundo de las bases de datos, debes estar muy atento a dicha intervención e ir tomando nota de los conceptos que te llamen la atención, luego el instructor brindará un espacio para que puedas despejar todas las dudas que hayan surgido.

AMBIENTE DE APRENDIZAJE TIPIFICADO: para la realización de esta actividad se requiere de:

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Marcadores borrables.
  - Borrador para tablero acrílico.
  - Hojas de papel.
  - Lápiz o lapicero.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 tablero acrílico.

3.1.2. ACTIVIDAD No 2: valorar el concepto de bases de datos en el desarrollo de esta formación.

La actividad propone la lectura y reflexión sobre ***"las bases de datos como parte fundamental para el desarrollo de sistemas de información y como fuente de incursión en el campo laboral"***, ***y*** para llevar a cabo una correcta resolución, se deberá realizar la lectura completa del documento adjunto, y luego cuando el instructor de formación lo indique, varios aprendices elegidos al azar podrán exponer su punto de vista con respecto al tema.

AMBIENTE DE APRENDIZAJE TIPIFICADO: para la realización de esta actividad se requiere de:

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Hojas de papel.
  - Lápiz o lapiceros.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.

3.2. ACTIVIDADES DE CONTEXTUALIZACIÓN:

3.2.1. ACTIVIDAD No 1: aplicar técnica de conocimientos previos y posteriores para la temática de Bases de Datos.

Para la resolución de esta actividad, inicialmente se debe contestar los cuestionamientos planteados en el documento de apoyo adjunto, en la sección de conocimientos previos, luego podrás apoyarte en un contenido multimedia, extraído desde el sitio web YouTube, en el cual se definen algunos conceptos referentes al tema, seguido podrás debatir con el instructor de formación los puntos de vistas planteados y recibir la retroalimentación pertinente, por ultimo sacar las conclusiones a las que haya lugar las cuales podrás relacionar en la sección de conocimientos posteriores.

AMBIENTE DE APRENDIZAJE TIPIFICADO: para la realización de esta actividad se requiere de:

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Marcadores borrables.
  - Borrador para tablero acrílico.
  - Hojas de papel.
  - Lápiz o lapiceros.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 tablero acrílico.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.3. ACTIVIDADES DE APROPIACIÓN:

3.3.1. ACTIVIDAD No 1: Introducción a las Bases de Datos: conocer un poco de historia sobre las Bases de Datos.

Realizar comprensión lectora de la información pertinente al tema referenciada a continuación: "Introducción a las Bases de Datos –BD– y Sistemas de Gestión de Bases de Datos –SGBD–", luego deberás resolver el cuestionario ubicado al final del documento adjunto, para finalizar la solución a los interrogantes planteados, debe quedar contenido en un único archivo PDF, el cual deberá ser almacenado en su portafolio.

AMBIENTE DE APRENDIZAJE TIPIFICADO: para la realización de esta actividad se requiere de:

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.

3.3.2. ACTIVIDAD No 2: Introducción a las Bases de Datos: definir algunos conceptos.

Realizar un trabajo de investigación dándoles respuesta a múltiples interrogantes planteados a continuación: "Investigar e interiorizar algunas definiciones", luego prepárate para la socialización de lo investigado, y para concluir se debe contener la investigación realizada en un único archivo PDF el cual deberá ser almacenado en su portafolio.

AMBIENTE DE APRENDIZAJE TIPIFICADO: para la realización de esta actividad se requiere de:

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.

3.3.3. ACTIVIDAD No 3: reflexionar sobre los modelos de Bases de Datos.

Realizar detenidamente la lectura de toda la información referenciada en el documento: ***"modelos de Bases de Datos"*** (realizar comprensión lectora, apoyarse con anotaciones de información relevante que se vaya encontrando), dar solución a los interrogantes referenciados al final del documento, cuando tengas los puntos resueltos, vas a reunirte con 2 aprendices más y juntos van a desarrollar una presentación multimedia (*.pptx) donde se plasmara un único planteamiento consensuado sobre los cuestionamientos propuestos, luego el instructor de formación seleccionará al azar dos grupos los cuales expondrán su presentación.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Hojas de papel.
  - Lápiz o lapiceros.

- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.3.4. ACTIVIDAD No 4: revisar de manera exprés el concepto de SQL.

Observar el siguiente material multimedia extraído desde el sitio web YouTube, donde a groso modo se desarrolla un concepto de SQL.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.

3.3.5. ACTIVIDAD No 5: revisar la conceptualización de modelo ER y la transformación al modelo relacional.

Para el desarrollo de esta actividad se deberá realizar detenidamente la lectura de toda la información referenciada en el documento: "Diseños de BD - MER" (realizar comprensión lectora, apoyarse con anotaciones de información relevante que se vaya encontrando), dar solución a los casos referenciados al final del documento, la aplicación "Lucidchart" te podrá ser útil para ello, podrás reunirte con un aprendices más y trabajar juntos.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.

3.3.6. ACTIVIDAD No 6: observar y complementar los conocimientos adquiridos hasta el momento sobre el modelo entidad-relación.

Observar el siguiente material multimedia extraído desde el sitio web YouTube, con el fin de afianzar mucho más los conceptos sobre el modelo entidad-relación:

1. Material 1.
2. Material 2.
3. Material 3.
4. Material 4.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.

3.3.7. ACTIVIDAD No 7: observar y aplicar el proceso de instalación de MySQL.

Observar el siguiente material multimedia extraído desde el sitio web YouTube, o el documento paso a paso, donde se muestra el proceso de instalación de MySQL, luego debes colocar en práctica lo observado y realizar la instalación, al finalizar explorar ***MySQL Command Line Client*** y ***MySQL Workbench***.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.

3.3.8. ACTIVIDAD No 8: revisar la conceptualización de las BD en MySQL.

Avanzar en la lectura del documento e ir interiorizando los conceptos mientras vas transcribiendo y verificando el comportamiento de las secuencias que se definen en cada punto, utilizando MySQL Command Line Client, e iras tomando las capturas de pantalla para casa secuencia transcrita y su respectivo resultado, esto se ira concentrando en un único archivo PDF, y al final el instructor de formación brindara un espacio para la socialización.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.

3.3.9. ACTIVIDAD No 9: afianzar un poco más los conceptos de MySQL.

Sigue los pasos descritos en el siguiente vídeo: https://youtu.be/yBSlAjMoJGw, seguido deberás resolver el siguiente taller: taller MySQL, deberás documentar de forma general los pasos realizados por ti con capturas de pantalla las cuales quedaran concentradas en un único archivo PDF el cual deberás adjuntar a tus evidencias.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.

3.3.10. profundizar un poco más en el conocimiento de MySQL.

Realiza un cuadro sinóptico teniendo en cuenta cada uno de los siguientes materiales de aprendizaje:

Instalación de Sistemas Manejadores de Bases de Datos.
Arquitectura de red.
Bases de datos distribuidas.
Sistemas Operativos.
Sistemas Manejadores de Bases de Datos

Los cuales debes descargar, descomprimir cada archivo y en cada caso ejecutar el archivo “index.html”, documentos_rar.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.3.11. Colocar en práctica nuevamente algunos conceptos.

Importe a su SGBD (**MySQL**) la base de datos **"TPS.sql"**, y realice las siguientes consultas:

- ✓ Muestre código y nombre de todas las competencias.
- ✓ Muestre los resultados de aprendizaje de cada competencia.
- ✓ Diga cuantos resultados de aprendizaje tiene la competencia: "**CONSTRUIR EL SISTEMA QUE CUMPLA CON LOS REQUISITOS DE LA SOLUCIÓN INFORMÁTICA**.".
- ✓ Diga cuantos resultados de aprendizaje tiene la competencia: "**ANALIZAR LOS REQUISITOS DEL CLIENTE PARA CONSTRUIR EL SISTEMA DE INFORMACION**.".
- ✓ Muestre los conocimientos de conceptos y principios de cada resultado de aprendizaje.
- ✓ Elimine las competencias con código: **240201500** y **240201501**.
- ✓ Muestre los conceptos y principios que tengan que ver con bases de datos.

Para este punto usted deberá tomar capturas de pantalla como evidencias de la consulta que realizó, así mismo escribir la consulta en su documento de Word para mayor legibilidad.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.3.12. Realiza la práctica gestión de usuarios y privilegios que encontrará en el siguiente link: taller_usuarios_privilegios.

Recuerda que se debe contener en un único archivo PDF todas las acciones realizadas en el desarrollo de la actividad.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**

- Ninguno.

- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.3.13. Revisar la conceptualización sobre diccionario de datos.

Se deberá revisar ciertos conocimientos básicos y necesarios para un correcto entendimiento del concepto de diccionario de datos:

- Revisa el material **"Introducción al diccionario de datos".**
- Analizar el siguiente vídeo en el cual se explica las funciones, características y como hacer un diccionario de datos: **https://youtu.be/6iKROsw3br8**
- Analizar la siguiente presentación en la cual se explican los tipos de diccionarios de datos y ejemplos: **https://prezi.com/zigtfiyggxgc/diccionario-de-datos/**

Con los conocimientos adquiridos realice un mapa conceptual utilizando la herramienta: https://bubbl.us/, tomará una captura de pantalla del trabajo realizado como evidencia de este punto.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.4. ACTIVIDADES DE TRANSFERENCIA:

3.4.1. ACTIVIDAD No 1: diseñar tu primer MER.

Diseña tu primer MER, para este punto deberás reunirte con 2 aprendices más y de forma consensuada, plantear un tema a trabajar, se deberá e investigar sobre un modelo ER a desarrollar, como también diseñarlo (podrás apoyarte en la herramienta web: https://www.draw.io/) y socializarlo con los compañeros de formación, con:

a. Enunciado.
b. Diseño.

Utiliza el formato de presentaciones multimedia (*.pptx) del SENA para la presentación.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Hojas de papel.
  - Lápiz o lapiceros.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.4.2. ACTIVIDAD No 2: analizar cada MER relacionado en cada caso de estudio e interpretarlo.

Analizar e interpretar el caso de estudio para cada modelo ER que encontrarás en el siguiente documento, contén tus entendimientos en un archivo PDF y prepárate para la socialización con el resto de los aprendices.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.

- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.4.3. ACTIVIDAD No 3: analizar cada caso de estudio y plantear el MER correspondiente.

Analizar e interpretar cada caso de estudio planteado en el siguiente documento, seguido deberás proponer un modelo ER por caso y contenerlos en un único archivo PDF, y al final el instructor de formación brindara un espacio para la socialización.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.4.4. ACTIVIDAD No 4: colocar en práctica los conceptos aprendidos sobre MySQL.

Esta actividad se divide en 4 partes secuenciales, no puedes saltarte el orden en que están definidas, en cada una de ellas se plantea una actividad la cual se debe desarrollar y atender a las especificaciones definidas.

a. Actividad 4 parte No 1.
b. Actividad 4 parte No 2.
c. Actividad 4 parte No 3.
d. Actividad 4 parte No 4.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.4.5. Colocar en práctica lo aprendido.

Importa la base de datos "**world.sql**" y realiza las consultas que encontrarás a continuación: **taller MySQL**.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.4.6. Colocar en práctica los conceptos sobre diccionario de datos.

Desarrolle el diccionario de datos para los casos estudios vistos anteriormente: **link taller,** luego se dará el espacio para realizar la respectiva solicialización.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

3.4.7. Mezcla de conocimientos en práctica.

Seleccione un caso estudio cualquiera y diseñe el Modelo entidad relación y el diccionario de datos correspondiente, se deberá ser explícito en el desarrollo de este teniendo en cuenta:

- ✓ Nombre del caso estudio.
- ✓ Descripción del caso estudio.
- ✓ Identificación de entidades.
- ✓ Atributos.
- ✓ Relaciones.
- ✓ Diccionario de datos.
- ✓ Deberá identificar de su caso estudio mínimo 15 entidades.
- ✓ Prepare todo el material que será explicado a todos sus compañeros.

AMBIENTE: Ambiente de formación con capacidad para 30 aprendices, con luminarias adecuadas, aire acondicionado.

MATERIALES DE FORMACIÓN:

- **CONSUMIBLES:**
  - Ninguno.
- **DEVOLUTIVOS:**
  - 30 computadores completos (CPU, monitor, teclado y mouse) o en su defecto, computadores portátiles, con programas de ofimática instalados y con conexión a internet.
  - 30 estaciones de trabajo.
  - 30 sillas.
  - 1 video beam.
  - 1 cable VGA o HDMI.

4. ACTIVIDADES DE EVALUACIÓN

Evidencias de Aprendizaje	**Criterios de Evaluación**	**Técnicas e Instrumentos de Evaluación**
Evidencias de Producto: Entrega de todos los puntos desarrollados Evidencias de Desempeño	DISEÑA LA SOLUCION BASADA EN LOS REQUERIMIENTOS	**TECNICA:** Valoración de Productos **INSTRUMENTO:** Lista de Chequeo de producto

5. GLOSARIO DE TERMINOS:

ADMINISTRADOR DE BD: Tipo de usuario especial que realiza funciones de administración y control de la BD, que aseguran que la explotación de la BD es correcta.

BASE DE DATOS: Conjunto estructurado de datos que representa entidades y sus interrelaciones. La representación será única e integrada, a pesar de que debe permitir diversas utilizaciones. Sigla: *BD.*

BD: Ved: *base de datos*.

CLIENTE/SERVIDOR
Tecnología habitual para distribuir datos. La idea es que dos procesos diferentes, que se pueden ejecutar en un mismo sistema o en sistemas separados, actúan de modo que uno actúa de cliente o peticionario de un servicio y el otro, como servidor. Un proceso cliente puede pedir servicios a distintos servidores. Un servidor puede recibir peticiones de muchos clientes. En general, un proceso *A* que actúa como cliente pidiendo un servicio a otro proceso *B* puede hacer también de servidor de un servicio que le pida otro proceso *C*. Sigla: *C/S.*

C/S: Ved: *cliente/servidor*.

DATA DEFINITION LANGUAGE: Lenguaje especializado en la escritura de esquemas; es decir, en la descripción de BD. Sigla: *DDL.*

DATA MANIPULATION LANGUAGE: Lenguaje especializado en la utilización de BD (consultas y mantenimiento). Sigla: *DML.*

SGBD: Ved: *sistema de gestión de BD*.

SISTEMA DE GESTIÓN DE BD: *Software* que gestiona y controla BD. Sus principales funciones son facilitar la utilización de la BD a muchos usuarios simultáneos y de tipos diferentes, independizar al usuario del mundo físico y mantener la integridad de los datos. Sigla: SGBD

ATRIBUTO DE UNA ENTIDAD: Propiedad que interesa de una entidad.

ATRIBUTO DE UNA INTERRELACIÓN: Propiedad que interesa de una interrelación.

CONECTIVIDAD DE UNA INTERRELACIÓN: Expresión del tipo de correspondencia entre las ocurrencias de entidades asociadas con la interrelación.

DISEÑO CONCEPTUAL: Etapa del diseño de una base de datos que obtiene una estructura de la información de la futura BD independiente de la tecnología que se quiere utilizar.

DISEÑO FÍSICO: Etapa del diseño de una base de datos que transforma la estructura obtenida en la etapa del diseño lógico con el objetivo de conseguir una mayor eficiencia y que, además, la completa con aspectos de implementación física que dependerán del SGBD que se debe utilizar diseño lógico Etapa del diseño de una base de datos que parte del resultado del diseño conceptual y lo transforma de modo que se adapte al modelo del SGBD con el que se desea implementar la base de datos.

ENTIDAD: Objeto del mundo real que podemos distinguir del resto de los objetos y del cual nos interesan algunas propiedades.

ENTIDAD ASOCIATIVA: Entidad resultante de considerar una interrelación entre entidades como una nueva entidad.

GRADO DE UNA INTERRELACIÓN: Número de entidades que asocia la interrelación.

INTERRELACIÓN: Asociación entre entidades.

6. REFERENTES BILBIOGRAFICOS:

1. SOFTWARE LIBRE – BASE DE DATOS:
   Primera edición: mayo 2005.
   © Fundació per a la Universitat Oberta de Catalunya.

2. Plataforma de biblioteca virtual del Servicio Nacional de Aprendizaje – SENA
   http://biblioteca.sena.edu.co/.

3. Página web: http://www.dofactory.com/sql/tutorial.

7. CONTROL DEL DOCUMENTO:

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	**IVÁN DAVID LÓPEZ HOYOS**	**INSTRUCTOR**	**TPS**	**15/10/2019**
	IRWING FONTALVO NIETO	**INSTRUCTOR**	**TPS**	**15/10/2018**

8. CONTROL DE CAMBIOS:

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)					

EJERCICIOS RESUELTOS Y COMENTADOS PARA EL APRENDIZAJE DE LA INSTRUCCIÓN SELECT

1. Conociendo la base de datos. 3
   1.1. Conexión con la línea de comando. 3
   1.2. Conexión con el Administrador. 3
   1.3. MySQL Migration Toolkit. 4
   1.4. Sentencia USE. 5
   1.5. Sentencia SHOW TABLES. 5
   1.6. Sentencia DESCRIBE. 5
   1.7. Tipos de datos 7
2. Sentencia SELECT 7
   2.1. Cláusula WHERE. 11
   2.2. Cláusula ORDER BY. 14
   2.3. Cláusula LIMIT. 16
   2.4. Ejecución de un script en la línea de comando. 17
   2.5. Otros problemas resueltos. 17
3. SELECT (como calculadora) 22
   3.1. Otros problemas resueltos. 25
4. Funciones de agregados. 30
   4.1. SUM(). 30
   4.2. COUNT(*). 30
   4.3. MAX(). 31
   4.4. MIN(). 31
   4.5. AVG(). 32
5. Subconsultas. 32
   5.1. Problemas resueltos. 32
6. Agrupación de datos. 35
   6.1. Cláusula GROUP BY. 35
   6.2. Cláusula HAVING. 36
   6.3. Problemas resueltos 37
7. Vinculaciones entre tablas. 42
   7.1. Cláusula ON. 42
   7.2. Búsqueda de texto. 52
   7.3. Unión de dos tablas de resultados . 55
   7.4. Subconsultas Escalares 57
      7.4.1. IN y NOT IN 58
      7.4.2. ALL y ANY 59
   7.5. Subconsultas relacionadas. 59
      7.5.1. EXISTS y NOT EXISTS 60
8. Salida a Excel. 67
9. El funcionamiento de SELECT. 69
10. Glosario. 69

1. Conociendo la base de datos.

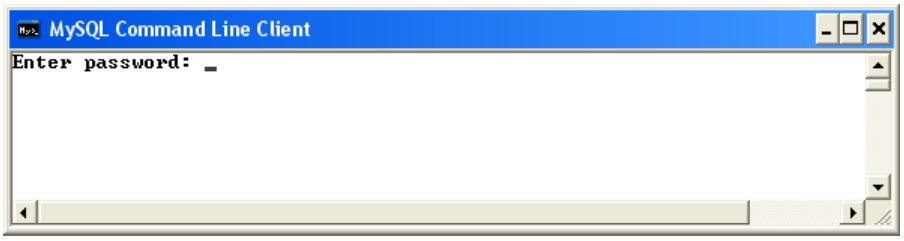
En primer lugar, antes de empezar a estudiar la instrucción SELECT exploraremos la base de datos.

Podemos trabajar de dos formas:

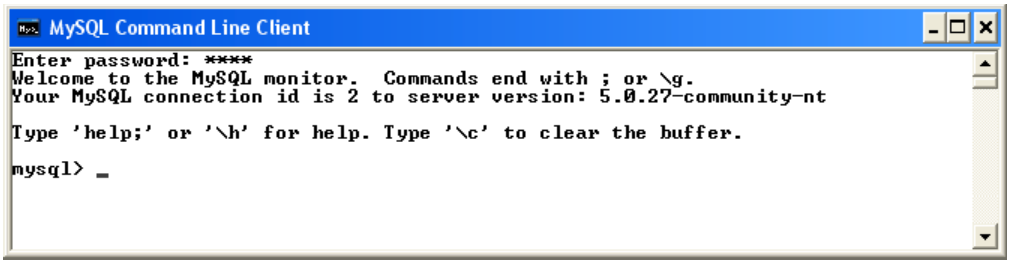
1) Con el cliente mysql a través de la línea de comando.
2) O con la herramienta MySQL Query Browser.

1.1. Conexión con la línea de comando.

Si ejecutamos desde Programa/MySQL/MySQL Server 5.0/MySQL Command Line Client aparecerá

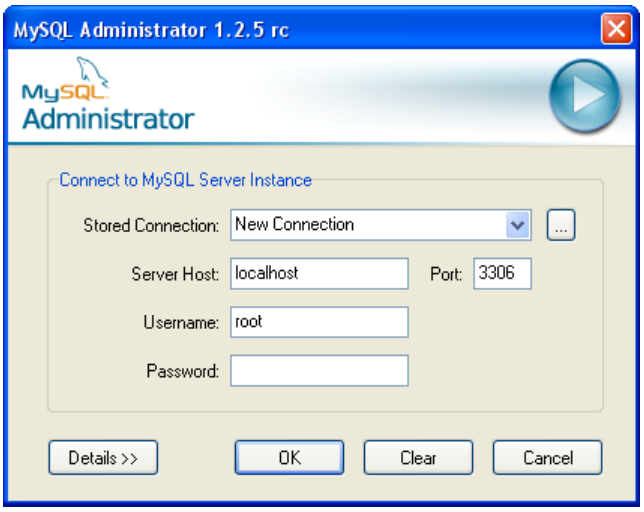


Y después de introducir el password

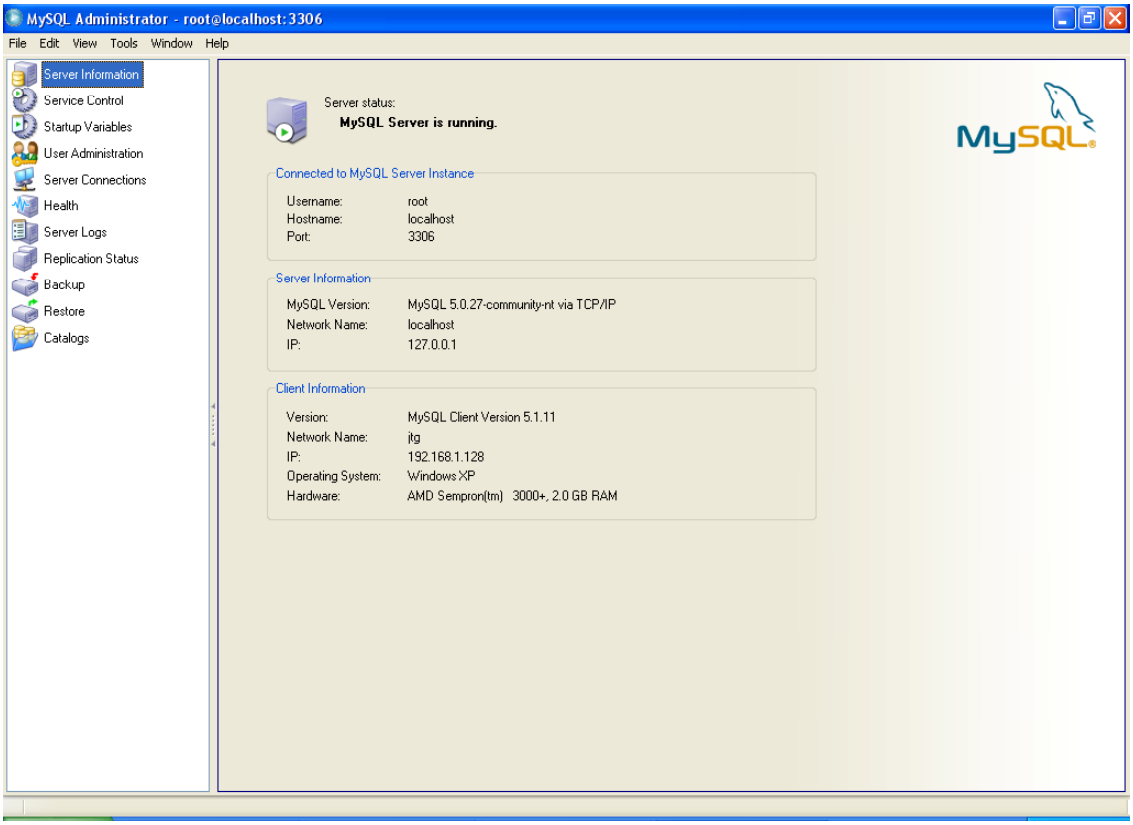


1.2. Conexión con el Administrador.

Si ejecutamos desde Programa/MySQL/MySQL Administrador aparecerá



Si nos identificamos y hacemos OK aparecerá



Desde la opción Tools/MySQL Command Line podemos ir a la línea de comando. También, desde dicha opción, podemos ir a la utilidad MySQL Query Browser, desde donde podemos seguir todo el ejercicio sin mas que ejecutar la sentencias una a una, con la ventaja de disponer de una interfaz más amigable. En estos apuntes utilizaremos la Línea de Comando para no hacer tan extenso los propios apuntes.

1.3. MySQL Migration Toolkit.

1.4. Sentencia USE.

Para acceder a la base de datos utilizamos la sentencia USE.

```
mysql> USE Empresa;
Database changed
```

Recibimos un mensaje "Database changed" (base de datos cambiada). Una base de datos está formada como ya sabemos por tablas.

1.5. Sentencia SHOW TABLES.

Para ver qué tablas tenemos escribimos la sentencia SHOW TABLES.

```
mysql> SHOW TABLES;
+-------------------+
| Tables_in_empresa |
+-------------------+
| articulos         |
| clientes          |
| conformes         |
| empleados         |
| facturas          |
| grupos            |
| lineas_factura    |
| sucursales        |
| ventas            |
+-------------------+
9 rows in set (0.03 sec)
```

Para ver qué contiene la tabla sucursales, o seleccionar todas sus filas escribimos nuestra primera instrucción SELECT :

```
mysql> select * from sucursales;
+----+-------------+
| id | descripcion |
+----+-------------+
|  1 | Centro      |
|  2 | Unión       |
|  3 | Marvín      |
+----+-------------+
3 rows in set (0.00 sec)
```

Esta tabla está formada por dos columnas: Id y Descripcion. Los nombres de columnas tampoco pueden llevar tildes. Debajo de los nombres están los valores. Tenemos una fila, por ejemplo, donde Id vale 2 y Descripcion es "Unión". Para averiguar más sobre esta tabla escribimos:

1.6. Sentencia DESCRIBE.

```
mysql> describe sucursales;
+-------------+------------------+------+-----+---------+----------------+
| Field       | Type             | Null | Key | Default | Extra          |
+-------------+------------------+------+-----+---------+----------------+
| id          | int(10) unsigned | NO   | PRI | NULL    | auto_increment |
| descripcion | varchar(15)      | NO   |     |         |                |
+-------------+------------------+------+-----+---------+----------------+
2 rows in set (0.01 sec)
```

La instrucción "DESCRIBE sucursales" describe la definición de la estructura de latabla.

El varchar(15) que se encuentra en la 3ª fila, 2ºcolumna significa que la columna Descripcion puede tener un número variable de caracteres, hasta 15. Análogamente el int(10) que se encuentra en la 2ª fila, 2ª columna indica que Id es un número entero.

La palabra PRI que se encuentra en la 2ªfila, 4ª columna significa que Id caracteriza de manera única una fila. Es decir, cada fila de sucursales va a tener un número que va a estar en la columna Id. Diferentes filas tendrán diferentes números. De manera que un valor de Id, si está en la tabla, determina una fila. PRI es una abreviatura de PRIMARY KEY , clave primaria. Esta es una expresión usada en computación para indicar una columna que se usa para identificar las filas.

Aún podríamos tener mas información de la tabla si ejecutamos la sentencia:

```
mysql> show create table sucursales;
+------------+------------------------------------------------------------
----------------------------------------------------------------------------
----------------------------------------------------------------------------
-------------------------+
| Table      | Create Table


                         |
+------------+------------------------------------------------------------
----------------------------------------------------------------------------
----------------------------------------------------------------------------
-------------------------+
| sucursales | CREATE TABLE `sucursales` (
  `id` int(10) unsigned NOT NULL auto_increment,
  `descripcion` varchar(15) collate latin1_spanish_ci NOT NULL default '',
  PRIMARY KEY  (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=4 DEFAULT CHARSET=latin1 COLLATE=latin1_spanish_ci
|
+------------+------------------------------------------------------------
----------------------------------------------------------------------------
----------------------------------------------------------------------------
-------------------------+
1 row in set (0.02 sec)
```

Se sugiere al alumnado que explore, usando SELECT * FROM ... y DESCRIBE ..., sobre las diversas tablas de la base de datos empresa.

1.7. Tipos de datos

En una tabla cada columna es de un tipo de datos. Para nuestros fines hay 3 grandes tipos de datos: numéricos, cadenas de caracteres y fechas.

Los datos numéricos pueden ser enteros (int) o decimales (decimal). Los decimales tienen una parte entera y una fraccionaria. Por ejemplo, un decimal puede valer 120.40.

Las cadenas de caracteres (varchar) representan texto, generalmente con un largo variable, hasta un máximo dado. Pueden contener letras, números y otros caracteres como '=','+', etc.

Ya vimos un ejemplo con la columna Descripcion de la tabla sucursales. Las columnas de tipo fecha tienen, en su descripción, la palabra "date". Cualquier columna puede, en principio, tomar el valor NULL, que representa un valor desconocido o inexistente. Sin embargo, cuando se crearon las tablas el administrador de la base de datos normalmente especificó que la mayoría de las columnas no pueden tomar valores NULL. Las que sí pueden aparecen con la palabra YES en la columna NULL de la tabla que se obtiene haciendo "DESCRIBE nombre_de_tabla" . Vemos que por ejemplo en la tabla Sucursales la base de datos no aceptará un valor NULL ni en Id ni en Descripción.

2. Sentencia SELECT

Para extraer información de una base de datos, utilizamos una consulta a través de una instrucción SELECT.

Para presentar los valores de una columna determinada de una tabla, usamos

SELECT Nombre_de_columna FROM Nombre_de_tabla;

Ejemplo :

```
mysql> SELECT descripcion FROM sucursales;
+-------------+
| descripcion |
+-------------+
| Centro      |
| Unión       |
| Marvín      |
+-------------+
3 rows in set (0.00 sec)
```

Para seleccionar más de una columna,

SELECT Nombre_de_columna_1, Nombre_de_columna_2,... FROM Nombre_de_tabla;

Ejemplo:

```
mysql> SELECT nombre, direccion FROM clientes;
+-------------+---------------------+
| nombre      | direccion           |
+-------------+---------------------+
| Matt Design | NULL                |
| Diana Pérez | Brito del Pino 1120 |
| John Smith  | Zum Felde 2024      |
+-------------+---------------------+
3 rows in set (0.00 sec)
```

Para seleccionar todas las columnas de una tabla,

SELECT * FROM Nombre_de_tabla;

Ejemplo:

```
mysql> SELECT * FROM ventas;
+----+---------+----------+---------+----------+----------+----------+------------+
| id | cliente | vendedor | importe | articulo | cantidad | sucursal | fecha      |
+----+---------+----------+---------+----------+----------+----------+------------+
|  1 |       1 |        1 |  178.50 |        1 |        1 |        1 | 2004-10-01 |
|  2 |       2 |        1 |  195.00 |        2 |        1 |        2 | 2004-10-05 |
|  3 |       3 |        1 |  178.50 |        1 |        1 |     NULL | 2004-10-07 |
|  4 |       1 |        1 |   16.00 |        7 |        2 |        2 | 2004-11-01 |
|  5 |       1 |        2 |  220.00 |        3 |        1 |        1 | 2004-12-02 |
|  6 |       1 |        1 |    5.50 |        5 |        1 |        1 | 2004-10-02 |
|  7 |       1 |        2 |   11.00 |        5 |        2 |        1 | 2004-10-03 |
|  8 |       1 |        2 |    5.50 |        5 |        1 |        1 | 2005-01-01 |
|  9 |       2 |        1 |  200.00 |        2 |        1 |        2 | 2004-10-12 |
+----+---------+----------+---------+----------+----------+----------+------------+
9 rows in set (0.00 sec)
```

Obsérvese que los clientes, vendedores, artículos y sucursales son identificados por su Id. El importe representa el total de la venta, no el precio unitario. También se pueden hacer cálculos con una columna numérica.

Ejemplos:

```
mysql> SELECT importe, 0.10*importe FROM ventas;
+---------+--------------+
| importe | 0.10*importe |
+---------+--------------+
|  178.50 |      17.8500 |
|  195.00 |      19.5000 |
|  178.50 |      17.8500 |
|   16.00 |       1.6000 |
|  220.00 |      22.0000 |
|    5.50 |       0.5500 |
|   11.00 |       1.1000 |
|    5.50 |       0.5500 |
|  200.00 |      20.0000 |
+---------+--------------+
9 rows in set (0.00 sec)
```

Tal como puede verse la expresión 0.10*importe constituye lo que se denomina un campo calculado que se obtiene a partir de un campo/columna de la tabla.

Se pueden agregar a la salida columnas constantes o calculadas, que no tienen o (no tienen) relación con las tablas.

Ejemplo:

```
mysql> SELECT "Descuento: ", 0.05*importe FROM ventas;
+-------------+--------------+
| Descuento:  | 0.05*importe |
+-------------+--------------+
| Descuento:  |       8.9250 |
| Descuento:  |       9.7500 |
| Descuento:  |       8.9250 |
| Descuento:  |       0.8000 |
| Descuento:  |      11.0000 |
| Descuento:  |       0.2750 |
| Descuento:  |       0.5500 |
| Descuento:  |       0.2750 |
| Descuento:  |      10.0000 |
+-------------+--------------+
9 rows in set (0.00 sec)
```

En general es deseable no tener títulos de columna complicados como "0.10*Importe". Para esos casos se usa la instrucción AS (que significa, entre otras cosas, "como").

Ejemplo:

```
mysql> SELECT 0.05*importe AS 'Descuento: ' FROM ventas;
+-------------+
| Descuento:  |
+-------------+
|      8.9250 |
|      9.7500 |
|      8.9250 |
|      0.8000 |
|     11.0000 |
|      0.2750 |
|      0.5500 |
|      0.2750 |
|     10.0000 |
+-------------+
9 rows in set (0.00 sec)
```

Se puede solicitar que no haya repeticiones en las filas seleccionadas. Basta agregar DISTINCT después del SELECT.

Ejemplo:

```
mysql> SELECT DISTINCT cliente FROM ventas;
+---------+
| cliente |
+---------+
|       1 |
|       2 |
|       3 |
+---------+
3 rows in set (0.00 sec)
```

Los resultados de cálculos, si no se toma alguna medida al respecto, a veces salen con demasiados decimales. En esos casos, conviene usar la función ROUND. Esta función redondea los números a la cantidad deseada de decimales después del punto. Por ejemplo, ROUND(2.343,2) produce el resultado 2.34. Análogamente ROUND(2.347,2) produce 2.35, y ROUND(2.245,2) produce 2.34. Para redondear a un entero alcanza con escribir ROUND(número). Por ejemplo, ROUND(2.345) produce 2.

Ejemplo:

```
mysql> SELECT ROUND(importe/3,2) FROM ventas;
+--------------------+
| ROUND(importe/3,2) |
+--------------------+
|              59.50 |
|              65.00 |
|              59.50 |
|               5.33 |
|              73.33 |
|               1.83 |
|               3.67 |
|               1.83 |
|              66.67 |
+--------------------+
9 rows in set (0.02 sec)
```

En el último punto hemos visto por primera vez una función. Una función consta de un nombre, seguido de paréntesis, entre los cuales según los casos puede no haber nada, o haber una o más variables. La función realiza ciertos cálculos con los valores de dichas variables, y genera otra cantidad, llamada el valor de la función. Una consideración práctica es que no se pueden dejar espacios entre el nombre de la función y el primer paréntesis después del mismo. Siempre escribiremos paréntesis después del nombre de una función.

2.1. Cláusula WHERE.

Esta cláusula sirve para seleccionar filas, dentro de las columnas seleccionadas. WHERE significa "donde". Se pueden seleccionar filas donde una columna tiene un valor determinado.

Ejemplo:

```
mysql> SELECT * FROM clientes WHERE id=1;
+----+-------------+-----------+----------+
| id | nombre      | direccion | vendedor |
+----+-------------+-----------+----------+
|  1 | Matt Design | NULL      |        1 |
+----+-------------+-----------+----------+
1 row in set (0.00 sec)
```

Vemos que de entre todos los clientes hemos obtenido solamente aquél registro para el que el identificador id es igual a uno.

Se puede seleccionar filas donde una columna tiene un valor mayor (o menor) que uno dado.

Ejemplo:

```
mysql> SELECT * FROM clientes WHERE id < 3;
+----+-------------+---------------------+----------+
| id | nombre      | direccion           | vendedor |
+----+-------------+---------------------+----------+
|  1 | Matt Design | NULL                |        1 |
|  2 | Diana Pérez | Brito del Pino 1120 |     NULL |
+----+-------------+---------------------+----------+
2 rows in set (0.00 sec)
```

Las igualdades y desigualdades también se aplican a cadenas de caracteres. Una cadena es menor que otra cuando es previa en orden alfabético. Por ejemplo "azzl" < "baa" . No se distingue entre mayúsculas y minúsculas (aunque, si se lo desea, es posible configurar el servidor MySQL para que sí distinga). Es decir, "A" < "b" < "c" < "D" y "a"="A".

Ejemplo:

```
mysql> SELECT * FROM clientes WHERE nombre < "i";
+----+-------------+---------------------+----------+
| id | nombre      | direccion           | vendedor |
+----+-------------+---------------------+----------+
|  2 | Diana Pérez | Brito del Pino 1120 |     NULL |
+----+-------------+---------------------+----------+
1 row in set (0.00 sec)
```

Este SELECT listará los clientes con nombre empezando en una letra anterior a "i" en el alfabeto.

También se aplican a fechas la igualdad y desigualdad. Para referirse a una fecha, se le escribe como "Año con 4 cifras-Mes-Día". Por ejemplo, "2004-12-20" indica el 20 de Diciembre de 2004.

Ejemplo:

```
mysql> SELECT * FROM ventas WHERE fecha < '2004-12-20';
+----+---------+----------+---------+----------+----------+----------+------------
+
| id | cliente | vendedor | importe | articulo | cantidad | sucursal | fecha
|
+----+---------+----------+---------+----------+----------+----------+------------
+
|  1 |       1 |        1 |  178.50 |        1 |        1 |        1 | 2004-10-01
|
|  2 |       2 |        1 |  195.00 |        2 |        1 |        2 | 2004-10-05
|
|  3 |       3 |        1 |  178.50 |        1 |        1 |     NULL | 2004-10-07
|
|  4 |       1 |        1 |   16.00 |        7 |        2 |        2 | 2004-11-01
|
|  5 |       1 |        2 |  220.00 |        3 |        1 |        1 | 2004-12-02
|
|  6 |       1 |        1 |    5.50 |        5 |        1 |        1 | 2004-10-02
|
|  7 |       1 |        2 |   11.00 |        5 |        2 |        1 | 2004-10-03
|
|  9 |       2 |        1 |  200.00 |        2 |        1 |        2 | 2004-10-12
|
+----+---------+----------+---------+----------+----------+----------+------------
+
8 rows in set (0.00 sec)
```

Para determinar si un dato es NULL se usa la condición 'IS NULL'. Para saber si no es NULL, se usa la condición 'IS NOT NULL'.

Ejemplo:

```
mysql> SELECT * FROM ventas WHERE sucursal IS NULL;
+----+---------+----------+---------+----------+----------+----------+------------
+
| id | cliente | vendedor | importe | articulo | cantidad | sucursal | fecha
|
+----+---------+----------+---------+----------+----------+----------+------------
+
|  3 |       3 |        1 |  178.50 |        1 |        1 |     NULL | 2004-10-07
|
+----+---------+----------+---------+----------+----------+----------+------------
+
1 row in set (0.00 sec)
```

No debe usarse <> NULL. Produce resultados equivocados, en su lugar habría que usar IS NOT NULL.

Llamamos condiciones simples a las siguientes

Expresión	Significado
a = b	a es igual a b
a = b	a es igual a b
a <> b	a es distinto de b
a < b	a es menor que b
a > b	a es mayor que b
a <= b	a es menor o igual a b
a >= b	a es mayor o igual a b
a IS NULL	a es NULL
a IS NOT NULL	a no es NULL

Las cantidades a y b pueden ser números, cadenas de caracteres o fechas, en todos los casos. En una cláusula WHERE se puede usar cualquier condición simple. La cláusula WHERE selecciona aquellas filas en que la condición es verdadera.

Las condiciones simples pueden aparecer combinadas por operadores lógicos. Los operadores lógicos son AND, OR y NOT. Aquí E y F representan condiciones.

Expresión	Significado	Es verdadera cuando ...
X AND Y	X y Y	X es verdadera y Y es verdadera
X OR Y	X o Y o ambos	X es verdadera o Y lo es o ambas
NOT X	No X	X es falsa

NOTA : El operador NOT requiere paréntesis. Es decir se debe escribir WHERE NOT (salario > 50) mientras que es incorrecto WHERE NOT salario > 50. Se entiende que buscamos filas con salarios no mayores que 50.

Ejemplos:

Listar los empleados cuya fecha de ingreso sea anterior al 2004, o cuyo salario sea mayor que 50 (o ambas cosas).

```
mysql> SELECT * FROM empleados WHERE fecha_ingreso < '2004-1-' OR salario > 50;
+----+-----------------+---------------+---------+
| id | nombre          | fecha_ingreso | salario |
+----+-----------------+---------------+---------+
|  1 | Carlos Zaltzman | 1999-12-10    |   10000 |
|  2 | Juan Fernández  | 2000-01-03    |   12000 |
+----+-----------------+---------------+---------+
2 rows in set, 1 warning (0.00 sec)
```

Listar los clientes cuyos nombres empiecen con una letra entre ‘c’ y ‘l’.

```
mysql> SELECT * FROM clientes WHERE "C" <= nombre AND nombre < "M";
+----+-------------+---------------------+----------+
| id | nombre      | direccion           | vendedor |
+----+-------------+---------------------+----------+
|  2 | Diana Pérez | Brito del Pino 1120 |     NULL |
|  3 | John Smith  | Zum Felde 2024      |        1 |
+----+-------------+---------------------+----------+
2 rows in set (0.00 sec)
```

Obsérvese que la condición apropiada para obtener los nombre que empiezan con ‘l’ es nombre <’M’. Si escribiésemos nombre < =‘L’ no obtendríamos un nombre como ‘Luís Zúñiga’ porque ‘L’<’Luís Zúñiga’. Se pueden usar incluso condiciones más complicadas. Por ejemplo, supongamos que ahora queremos listar los empleados cuya fecha de ingreso es anterior al 2004, o cuyo salario sea mayor que 50, pero no ambas cosas a la vez. Entonces, debemos escribir :

```
mysql> SELECT * FROM empleados WHERE (fecha_ingreso < '2004-1-1' OR salario > 50)
    -> AND NOT (fecha_ingreso < '2004-1-1' AND salario > 50);
Empty set (0.00 sec)
```

El resultado es "Empty set", es decir, no hay filas que cumplan la condición. En el caso de condiciones más complicadas se recomienda un amplio uso de paréntesis.

2.2. Cláusula ORDER BY.

La cláusula ORDER BY produce una ordenación de las filas de salida del Query o consulta. Se puede ordenar por una columna seleccionada.

```
mysql> SELECT cliente, articulo FROM ventas ORDER BY cliente;
+---------+----------+
| cliente | articulo |
+---------+----------+
|       1 |        1 |
|       1 |        7 |
|       1 |        3 |
|       1 |        5 |
|       1 |        5 |
|       1 |        5 |
|       2 |        2 |
|       2 |        2 |
|       3 |        1 |
+---------+----------+
9 rows in set (0.02 sec)
```

También se puede ordenar por varias columnas.

```
mysql> SELECT cliente, vendedor, articulo FROM ventas ORDER BY cliente, vendedor;
+---------+----------+----------+
| cliente | vendedor | articulo |
+---------+----------+----------+
|       1 |        1 |        1 |
|       1 |        1 |        7 |
|       1 |        1 |        5 |
|       1 |        2 |        3 |
|       1 |        2 |        5 |
|       1 |        2 |        5 |
|       2 |        1 |        2 |
|       2 |        1 |        2 |
|       3 |        1 |        1 |
+---------+----------+----------+
9 rows in set (0.00 sec)
```

Cuando se ordena por varias columnas, por ejemplo 3, el procedimiento es básicamente el que sigue : Se ordena por la primera columna. Si hay valores repetidos en la primera columna, para cada grupo de valores repetidos se ordenan las filas por el valor de la 2ª columna. Si hay valores repetidos de las dos primeras columnas en conjunto, se ordenan las filas correspondientes por la 3ª columna.

Ejemplo: Supongamos una tabla con las siguientes columnas y valores:

a	b	c	d

125	Diana Pérez	1/1/2004	12313
486	Alejandro Bentancourt	30/12/2005	45646
222	Jorge Rodríguez	2/5/2004	78987
125	Diana Pérez	3/8/2004	654654
222	Adriana Salgado	1/3/2002	12312

Si ordenamos esta tabla por las columnas a, b y c, obtenemos

125	Diana Pérez	1/1/2004	12313
125	Diana Pérez	3/8/2004	654654

222	Adriana Salgado	1/3/2002	12312
222	Jorge Rodríguez	2/5/2004	78987
486	Alejandro Bentancourt	30/12/2005	45646

Por último, se puede ordenar por una cantidad calculada a partir de una o varias columnas.

```
mysql> SELECT articulo, importe, cantidad FROM ventas
    -> ORDER BY articulo, importe/cantidad;
+----------+---------+----------+
| articulo | importe | cantidad |
+----------+---------+----------+
|        1 |  178.50 |        1 |
|        1 |  178.50 |        1 |
|        2 |  195.00 |        1 |
|        2 |  200.00 |        1 |
|        3 |  220.00 |        1 |
|        5 |    5.50 |        1 |
|        5 |   11.00 |        2 |
|        5 |    5.50 |        1 |
|        7 |   16.00 |        2 |
+----------+---------+----------+
9 rows in set (0.00 sec)
```

2.3. Cláusula LIMIT.

La preparación de una consulta complicada implica normalmente un proceso de prueba y error. Aunque no se cometan errores, siempre se empieza escribiendo consultas que sólo realizan una parte de lo que se desea alcanzar. Luego, se van mejorando gradualmente hasta llegar al objetivo buscado.

Cuando se trabaja con tablas auténticas con muchos cientos o miles de filas, puede ser demasiado engorroso ir obteniendo repetidas salidas con cientos o miles de filas. Es obvio que no se pueden observar en la pantalla del cliente mysql. Por otra parte, en su instalación puede haber otros clientes que operen con MySQL.

De todas maneras, interesa una cláusula sencilla que limite el número de filas que produce el SELECT. Esa es la función de LIMIT. Si, por ejemplo, escribimos

```
mysql> SELECT articulo, cliente FROM ventas LIMIT 3;
+----------+---------+
| articulo | cliente |
+----------+---------+
|        1 |       1 |
|        2 |       2 |
|        1 |       3 |
+----------+---------+
3 rows in set (0.00 sec)
```

Vemos 3 filas en la salida, a pesar que hay varias más en Ventas.

El orden de las cláusulas.

Las cláusulas mencionadas, SELECT...FROM.. , WHERE, ORDER BY, y LIMIT deben escribirse, si aparecen, en ese orden. SELECT siempre aparece y va en primer lugar. Las otras 3 son optativas.

2.4. Ejecución de un script en la línea de comando.

Hasta ahora hemos venido escribiendo a continuación del prompt mysql> del cliente mysql.

Este procedimiento presenta inconvenientes cuando tenemos que ejecutar un número importantes de instrucciones o sentencias de forma repetitiva. Un procedimiento sencillo para Windows, es el siguiente:

1. Inicie el programa Bloc de Notas. Suele estar en Inicio, Programas, Accesorios.
2. Escriba en él su consulta. (select * from sucursales;)
3. Guárdelo por ejemplo en C:/MYSQL/MySQL Server5.0/bin/pruebas/pruebas.txt con el nombre pruebas.txt
4. Vaya al cliente mysql y escriba

```
mysql> SOURCE C:/MySQL/MySQL Server 5.0/bin/pruebas/pruebas.txt;
+----+-------------+
| id | descripcion |
+----+-------------+
|  1 | Centro      |
|  2 | Unión       |
|  3 | Marvín      |
+----+-------------+
3 rows in set (0.00 sec)
```

Si la consulta no dio el resultado esperado, ahora no es preciso escribir todo de nuevo. Simplemente corríjalo en el Bloc de Notas. y vuelva a guardarlo. Repita los pasos hasta obtener el resultado buscado. La opción anterior tiene verdaderamente sentido cuando tenemos que ejecutar o probar un grupo importante de sentencias que finalmente conduzcan a un script de mantenimiento o administración de la base de datos diaria, semanal, mensual, etc…

2.5. Problemas resueltos.

1. Obtener todos los datos de la tabla Empleados.

Solución:

```
mysql> SELECT * FROM Empleados;
+----+-----------------+---------------+---------+
| id | nombre          | fecha_ingreso | salario |
+----+-----------------+---------------+---------+
|  1 | Carlos Zaltzman | 1999-12-10    |   10000 |
|  2 | Juan Fernández  | 2000-01-03    |   12000 |
+----+-----------------+---------------+---------+
2 rows in set (0.00 sec)
```

2. Obtener los nombres y las fechas de ingreso de los empleados.

Solución : Formalmente, queremos obtener las columnas Nombre, Fecha_ingreso de todas las filas de la tabla Empleados. La consulta es:

```
mysql> SELECT  Nombre, Fecha_ingreso FROM Empleados;
+-----------------+---------------+
| Nombre          | Fecha_ingreso |
+-----------------+---------------+
| Carlos Zaltzman | 1999-12-10    |
| Juan Fernández  | 2000-01-03    |
+-----------------+---------------+
2 rows in set (0.00 sec)
```

3. Obtener los datos de los empleados que ganan más de 10500,50.

Solución: Queremos obtener todas las columnas de las filas de Empleados para las cuales Salario es mayor que 10500,50.

```
mysql> SELECT * FROM Empleados
    -> WHERE Salario > 10500.50;
+----+----------------+---------------+---------+
| id | nombre         | fecha_ingreso | salario |
+----+----------------+---------------+---------+
|  2 | Juan Fernández | 2000-01-03    |   12000 |
+----+----------------+---------------+---------+
1 row in set (0.00 sec)
```

Obsérvese el uso del punto decimal. Las comas no se pueden utilizar ni en lugar del punto decimal ni para separar los miles.

4. Queremos obtener los datos del funcionario Carlos Zaltzman.

Solución: Se trata de obtener una fila de Empleados, aquella en la cual el nombre del funcionario es Carlos Zaltzman.

```
mysql> SELECT  * FROM  Empleados
    -> WHERE Nombre = 'Carlos Zaltzman';
+----+-----------------+---------------+---------+
| id | nombre          | fecha_ingreso | salario |
+----+-----------------+---------------+---------+
|  1 | Carlos Zaltzman | 1999-12-10    |   10000 |
+----+-----------------+---------------+---------+
1 row in set (0.00 sec)
```

5. ¿Qué empleados ingresaron el 3 de enero del 2000?.

Solución: Se trata de obtener filas de Empleados, aquellas en las cuales la fecha de ingreso sea '2000-1-3'. El formato apropiado para la fecha es Año-Mes-Día. La consulta es:

```
mysql> SELECT  * FROM Empleados
    -> WHERE Fecha_ingreso = '2000-1-3';
+----+---------------+---------------+---------+
| id | nombre        | fecha_ingreso | salario |
+----+---------------+---------------+---------+
|  2 | Juan Fernández | 2000-01-03    |   12000 |
+----+---------------+---------------+---------+
1 row in set (0.00 sec)
```

6. ¿Qué empleados ingresaron en fecha posterior al 1º de enero del 2000?

Solución: Queremos obtener las filas de Empleados con fecha de ingreso mayor que '2000-1-1'. La consulta es:

```
mysql> SELECT  * FROM Empleados
    -> WHERE Fecha_ingreso>'2000-1-1';
+----+---------------+---------------+---------+
| id | nombre        | fecha_ingreso | salario |
+----+---------------+---------------+---------+
|  2 | Juan Fernández | 2000-01-03    |   12000 |
+----+---------------+---------------+---------+
1 row in set (0.00 sec)
```

7. ¿Qué empleados ingresaron el 5 de enero del 2000 o en una fecha posterior?

Solución: La fecha de ingreso debe ser mayor o igual a '2001-1-5'.

```
mysql> SELECT  * FROM Empleados
    -> WHERE Fecha_ingreso>='2000-1-5';
Empty set (0.00 sec)
```

8. Obtener una lista de los nombres de los clientes de los cuales no se tiene dirección.

Solución: Se trata de obtener la columna Nombre de las filas de la tabla Clientes cuya dirección es NULL.

```
mysql> SELECT Nombre FROM Clientes
    -> WHERE Direccion IS NULL;
+-------------+
| Nombre      |
+-------------+
| Matt Design |
+-------------+
1 row in set (0.00 sec)
```

9. Sacar una lista de los nombres y direcciones de los clientes de los cuales sí se tiene la dirección.

Solución: Se trata de obtener la columnas Nombre, Direccion de las filas de la tabla Clientes que tienen direcciones que no sean NULL.

```
mysql> SELECT  Nombre,Direccion FROM Clientes
    -> WHERE Direccion IS NOT NULL;
+-------------+---------------------+
| Nombre      | Direccion           |
+-------------+---------------------+
| Diana Pérez | Brito del Pino 1120 |
| John Smith  | Zum Felde 2024      |
+-------------+---------------------+
2 rows in set (0.00 sec)
```

10. Obtener una lista de los diferentes salarios que se pagan en la empresa.

Solución:

```
mysql> SELECT DISTINCT Salario FROM Empleados;
+---------+
| Salario |
+---------+
|   10000 |
|   12000 |
+---------+
2 rows in set (0.00 sec)
```

11. Obtener una lista ordenada alfabéticamente de los nombres y direcciones de los clientes, ordenados por nombre.

Solución:

```
mysql> SELECT Nombre,Direccion FROM Clientes
    -> ORDER BY Nombre;
+-------------+---------------------+
| Nombre      | Direccion           |
+-------------+---------------------+
| Diana Pérez | Brito del Pino 1120 |
| John Smith  | Zum Felde 2024      |
| Matt Design | NULL                |
+-------------+---------------------+
3 rows in set (0.00 sec)
```

12. Obtener una lista de todos los datos de los empleados ordenados por nombre.

Solución:

```
mysql> SELECT * FROM Empleados
    -> ORDER BY Nombre;
+----+-----------------+---------------+---------+
| id | nombre          | fecha_ingreso | salario |
+----+-----------------+---------------+---------+
|  1 | Carlos Zaltzman | 1999-12-10    |   10000 |
|  2 | Juan Fernández  | 2000-01-03    |   12000 |
+----+-----------------+---------------+---------+
2 rows in set (0.00 sec)
```

13. Obtener una lista de los datos de los empleados ordenados por salario en forma descedente. Los salarios deben ser formateados con comas cada 3 dígitos y dos decimales después del punto y alineados a la derecha, por ejemplo, 1,200,340.50.

Solución:

```
mysql> SELECT Id, Nombre, Fecha_ingreso, LPAD( FORMAT(Salario,2) , 12 , ' ' )
    -> AS Salario FROM Empleados ORDER BY Salario DESC;
+----+-----------------+---------------+-------------+
| Id | Nombre          | Fecha_ingreso | Salario     |
+----+-----------------+---------------+-------------+
|  2 | Juan Fernández  | 2000-01-03    |   12,000.00 |
|  1 | Carlos Zaltzman | 1999-12-10    |   10,000.00 |
+----+-----------------+---------------+-------------+
2 rows in set (0.00 sec)
```

14. Igual al anterior pero se quiere que los números salgan formateados con puntos cada 3 dígitos y coma decimal, en vez de punto. Por ejemplo, 1.200.340,50 .

Solución:

```
mysql> SELECT Id, Nombre, Fecha_ingreso,
    -> REPLACE(
    -> REPLACE(
    -> REPLACE(LPAD(FORMAT(Salario,2),12,' '),
    ->    '.','!'),
    ->    ',','.'),
    ->    '!',',')
    -> AS Salario
    ->     FROM Empleados ORDER BY Salario DESC;
+----+-----------------+---------------+--------------+
| Id | Nombre          | Fecha_ingreso | Salario      |
+----+-----------------+---------------+--------------+
|  2 | Juan Fernández  | 2000-01-03    |    12.000,00 |
|  1 | Carlos Zaltzman | 1999-12-10    |    10.000,00 |
+----+-----------------+---------------+--------------+
2 rows in set (0.00 sec)
```

Si bien es un procedimiento complicado, para usarlo en otro caso sólo hay que copiar la parte que comienza con REPLACE y termina en el paréntesis antes de AS Salario y hacer los siguientes cambios:

1. Cambiar la columna Salario por la que se vaya a usar.
2. Eventualmente cambiar el número de decimales después de la coma, que figura dentro de la función FORMAT, de 2 al valor que se desee.
3. Si es conveniente, variar el ancho de la columna. Este ancho aparece dentro de la función LPAD. En este ejemplo vale 12. El ancho debe ser suficiente para poder escribir todos los valores que aparezcan en la columna, con el número de decimales que se haya solicitado.

3. SELECT (como calculadora)

Se puede usar la cláusula SELECT para hacer cálculos aritméticos.

Ejemplos:

```
mysql> SELECT 1+1;
+-----+
| 1+1 |
+-----+
|   2 |
+-----+
1 row in set (0.00 sec)
```

O este otro:

```
mysql> SELECT 4.5*(3.86+2.34);
+-----------------+
| 4.5*(3.86+2.34) |
+-----------------+
|          27.900 |
+-----------------+
1 row in set (0.00 sec)
```

usado COUNT (DISTINCT ....) para contar cuántos elementos distintos hay en una expresión calculada en base a una columna.

7.2. Búsqueda de texto.

A menudo tenemos columnas que son cadenas de caracteres, y queremos buscar las cadenas que contienen cierta palabra. Esto se realiza a través de un nuevo tipo de condición: Nombre_de_columna LIKE cadena_de_caracteres.

Por ejemplo, Nombre LIKE ‘Carlos’.

Estas condiciones pueden usarse, como todas, en una cláusula WHERE o en una cláusula HAVING. La condición Nombre_de_columna LIKE cadena_de_caracteres (donde cadena_de_caracteres no contiene ni ‘%’ ni ‘_’) es verdadera cuando el valor de Nombre_de_columna coincide con cadena_de_caracteres (salvo que no se distingue entre mayúsculas y minúsculas).

Veamos los valores de la condición A LIKE ’Carlos’ donde A es una columna:

Valor de la columna A	
'Carlos'	Verdadera
'carlos'	Verdadera
'Carlos '	Falsa
'Juan'	Falsa
'Juan Carlos'	Falsa

Un % dentro de cadena_de_caracteres representa cualquier cadena de caracteres, incluso una sin caracteres. Entonces A LIKE ‘Carlos%’ tiene los siguientes valores:

Valor de la columna A	
'Carlos'	Verdadera
'carlos'	Verdadera
'Carlos Zaltzman'	Verdadera
'Carlos '	Verdadera
'Juan'	Falsa
'Juan Carlos'	Falsa

Valores de A LIKE ‘%Carlos’:

Valor de la columna A	
'Carlos'	Verdadera
'carlos'	Verdadera
'Juan carlos'	Verdadera
'Juan Carlos Rodríguez’	Falsa
'Juan Carlos '	Falsa

Valores de A LIKE ‘%Carlos%’:

Valor de la columna A	
'Carlos'	Verdadera
'carlos'	Verdadera
' carlos '	Verdadera
'Juan Carlos Rodríguez’	Verdadera
'Juan'	Falsa

El carácter '_' representa un carácter cualquiera. A diferencia de ‘%’ representa un carácter y uno sólo. Luego dos’_’ seguidos representan dos caracteres cualquiera, etc.

Valores de A LIKE ‘__Carlos’

Valor de la columna A	
'Carlos'	Falsa
'12Carlos'	Verdadera
'xxCarlos'	Verdadera
' Carlos'	Verdadera
'xxcarlos'	Verdadera
'Juan Carlos'	Falsa

NOT LIKE es simplemente la negación de LIKE: es verdadera cuando LIKE es falsa y recíprocamente.

Problemas resueltos

1. Listar el artículo cuyo código es ’mon20'.

Solución:

```
mysql> SELECT * FROM Articulos WHERE Codigo LIKE ‘mon20’;
+----+-------------+--------+--------+-------+
| id | nombre      | precio | codigo | grupo |
+----+-------------+--------+--------+-------+
|  2 | Monitor 20" | 200.00 | mon20  |     1 |
+----+-------------+--------+--------+-------+
1 row in set (0.16 sec)
```

2. Hacer una lista de todos los datos de los artículos en cuyo nombre figura la palabra ‘monitor’.

Solución:

```
mysql> SELECT  * FROM  Articulos WHERE  Nombre LIKE ‘%monitor%’;
+----+-------------+--------+--------+-------+
| id | nombre      | precio | codigo | grupo |
+----+-------------+--------+--------+-------+
|  1 | Monitor 16" | 178.50 | mon16  |     1 |
|  2 | Monitor 20" | 200.00 | mon20  |     1 |
|  3 | Monitor 22" | 220.00 | mon22  |     1 |
+----+-------------+--------+--------+-------+
3 rows in set (0.00 sec)
```

Obsérvese que aunque la palabra monitor esté en la columna Nombre con mayúscula, igualmente aparece en el resultado de la consulta.

3. Listar todos los datos de los artículos cuyo código comience por ‘mon’.

Solución:

```
mysql> SELECT * FROM Articulos WHERE Codigo LIKE ‘mon%’;
+----+-------------+--------+--------+-------+
| id | nombre      | precio | codigo | grupo |
+----+-------------+--------+--------+-------+
|  1 | Monitor 16" | 178.50 | mon16  |     1 |
|  2 | Monitor 20" | 200.00 | mon20  |     1 |
|  3 | Monitor 22" | 220.00 | mon22  |     1 |
+----+-------------+--------+--------+-------+
3 rows in set (0.00 sec)
```

4. Listar los artículos cuyo código termine en ‘20’.

Solución:

```
mysql> SELECT * FROM Articulos WHERE Codigo LIKE '%20';
+----+--------------+--------+--------+-------+
| id | nombre       | precio | codigo | grupo |
+----+--------------+--------+--------+-------+
|  2 | Monitor 20"  | 200.00 | mon20  |     1 |
|  7 | Diskettes 20 |   8.00 | D20    |     2 |
+----+--------------+--------+--------+-------+
2 rows in set (0.00 sec)
```

5. Listar los artículos cuyo código tenga exactamente dos caracteres.

Solución:

```
mysql> SELECT * FROM Articulos WHERE Codigo LIKE "__";
Empty set (0.00 sec)
```

No hay ninguno. Entonces MySQL responde "Empty set", es decir, "Conjunto vacío".

6. Listar todos los artículos en cuyo nombre NO figure la palabra "monitor".

Solución:

```
mysql> SELECT * FROM Articulos WHERE Nombre NOT LIKE "%monitor%";
+----+----------------+--------+--------+-------+
| id | nombre         | precio | codigo | grupo |
+----+----------------+--------+--------+-------+
|  4 | Motherboard FX |  99.50 | mthFX  |     1 |
|  5 | Papel A4-500   |   5.50 | PA4500 |     2 |
|  6 | Diskettes 10   |   4.50 | D10    |     2 |
|  7 | Diskettes 20   |   8.00 | D20    |     2 |
+----+----------------+--------+--------+-------+
4 rows in set (0.00 sec)
```

7.3. Unión de dos tablas de resultados .

UNION ALL indica que se haga la unión de dos resultados, simplemente escribiendo una tabla debajo de la otra. Por ejemplo, si queremos los artículos y cantidades de las ventas a los clientes 1 y 2, basta escribir

```
(SELECT Articulo, Cantidad FROM Ventas
 WHERE Cliente=1)
UNION ALL
(SELECT Articulo, Cantidad FROM Ventas
 WHERE Cliente=2);
```

" LA TIENDA DE INFORMATICA"

INSTRUCCIONES: En el siguiente ejercicio se describen cada uno de los comandos para llevar a cabo cada una de las consultas solicitadas, ejecuta cada uno de ellos en el compilador de MySQL.

ACTIVIDADES A REALIZAR

1. Crea una base de datos llamada TIENDA

```
CREATE DATABASE TIENDA;
```

2. Muestra las Bases de datos almacenadas

```
SHOW DATABASES;
```

3. Habilita la Base de datos TIENDA

```
USE TIENDA;
```

4. Genera las siguientes tablas:

FABRICANTES		
	NOMBRE DE COLUMNA	TIPO DE DATO
Clave Principal	Clave_fabricante	Int
	Nombre	Varchar(30)

```
CREATE TABLE FABRICANTES (Clave_fabricante INT NOT NULL, Nombre VARCHAR(30), PRIMARY KEY (Clave_fabricante));
```

ARTICULOS		
	NOMBRE DE COLUMNA	TIPO DE DATO
Clave Principal	Clave_articulo	Int
	Nombre	Varchar(30)
	Precio	Int
Clave Foránea	Clave_fabricante	Int

```
CREATE TABLE ARTICULOS (Clave_articulo INT NOT NULL, Nombre VARCHAR(30), Precio INT, Clave_fabricante INT, PRIMARY KEY (Clave_articulo), FOREING KEY(Clave_fabricante) REFERENCES FABRICANTES (Clave_fabricante));
```

5. Muestra las tablas de la Base de datos TIENDA

```
SHOW TABLES;
```

6. Muestra los atributos de la tabla ARTICULOS;

```
DESCRIBE ARTICULOS;
```

7. Introduce los siguientes datos en cada tabla:

TABLA: FABRICANTES	
CLAVE_FABRICANTE	**NOMBRE**
1	Kingston
2	Adata
3	Logitech
4	Lexar
5	Seagate

```
INSERT INTO FABRICANTES VALUES ( 1 , 'Kingston');
```

TABLA: ARTICULOS			
CLAVE_ARTICULO	**NOMBRE**	**PRECIO**	**CLAVE_FABRICANTE**
1	Teclado	$ 100	3
2	Disco duro 300 Gb	$ 500	5
3	Mouse	$ 80	3
4	Memoria USB	$ 140	4
5	Memoria RAM	$ 290	1
6	Disco duro extraíble 250 Gb	$ 650	5
7	Memoria USB	$ 279	1
8	DVD Rom	$ 450	2
9	CD Rom	$ 200	2
10	Tarjeta de red	$ 180	3

```
INSERT INTO ARTICULOS VALUES (  1 , 'Teclado', 100 , 3 );
```

8. Genera las siguientes consultas:

a) Obtener todos los datos de los productos de la tienda

```
SELECT * FROM ARTICULOS;
```

b) Obtener los nombres de los productos de la tienda

```
SELECT Nombre FROM ARTICULOS;
```

c) Obtener los nombres y precio de los productos de la tienda

```
SELECT Nombre,Precio  FROM ARTICULOS;
```

d) Obtener los nombres de los artículos sin repeticiones

```
SELECT DISTINCT  Nombre  FROM ARTICULOS;
```

e) Obtener todos los datos del artículo cuya clave de producto es '5'

```
SELECT  *  FROM ARTICULOS WHERE Clave_articulo=5;
```

f) Obtener todos los datos del artículo cuyo nombre del producto es ''Teclado"

```
SELECT  *  FROM ARTICULOS WHERE Nombre='Teclado';
```

g) Obtener todos los datos de la Memoria RAM y memorias USB

```
SELECT  *  FROM ARTICULOS WHERE Nombre='Memoria RAM' OR Nombre='Memoria USB' ;
```

h) Obtener todos los datos de los artículos que empiezan con 'M'

```
SELECT * FROM ARTICULOS WHERE Nombre LIKE 'M%';
```

i) Obtener el nombre de los productos donde el precio sea $ 100

```
SELECT Nombre FROM ARTICULOS WHERE Precio = 100;
```

j) Obtener el nombre de los productos donde el precio sea mayor a $ 200

```
SELECT Nombre FROM ARTICULOS WHERE Precio > 200;
```

k) Obtener todos los datos de los artículos cuyo precio este entre $100 y $350

```
/* OPERADOR AND */

    SELECT * FROM ARTICULOS WHERE Precio >= 100 AND Precio<=350;

/* OPERADOR BETWEEN */

    SELECT * FROM ARTICULOS WHERE Precio BETWEEN 100 AND 350;
```

l) Obtener el precio medio de todos los productos

```
SELECT AVG(Precio) FROM ARTICULOS;
```

m) Obtener el precio medio de los artículos cuyo código de fabricante sea 2

```
SELECT AVG(Precio) FROM ARTICULOS WHERE Clave_fabricante=3;
```

n) Obtener el nombre y precio de los artículos ordenados por Nombre

```
SELECT Nombre, Precio FROM ARTICULOS ORDER BY Nombre;
```

o) Obtener todos los datos de los productos ordenados descendentemente por Precio

```
SELECT * FROM ARTICULOS ORDER BY Precio DESC;
```

p) Obtener el nombre y precio de los artículos cuyo precio sea mayor a $ 250 y ordenarlos descendentemente por precio y luego ascendentemente por nombre

```
SELECT Nombre, Precio FROM ARTICULOS
WHERE Precio >= 250 ORDER BY Precio DESC, Nombre;
```

q) Obtener un listado completo de los productos, incluyendo por cada articulo los datos del articulo y del fabricante

```
SELECT  *  FROM ARTICULOS, FABRICANTES
WHERE ARTICULOS.Clave_fabricante=FABRICANTES.Clave_fabricante
```

r) Obtener la clave de producto, nombre del producto y nombre del fabricante de todos los productos en venta

```
SELECT  ARTICULOS.Clave_articulo, ARTICULOS.Nombre, FABRICANTES.Nombre
FROM ARTICULOS, FABRICANTES
WHERE ARTICULOS.Clave_fabricante=FABRICANTES.Clave_fabricante
```

s) Obtener el nombre y precio de los artículos donde el fabricante sea Logitech ordenarlos alfabéticamente por nombre del producto

```
SELECT  ARTICULOS.Nombre, ARTICULOS.Precio FROM ARTICULOS, FABRICANTES
WHERE FABRICANTES.Nombre='Logitech' AND
 ARTICULOS.Clave_fabricante=FABRICANTES.Clave_fabricante
ORDER BY ARTICULOS.Nombre;
```

t) Obtener el nombre, precio y nombre de fabricante de los productos que son marca Lexar o Kingston ordenados descendentemente por precio

```
SELECT  ARTICULOS.Nombre, ARTICULOS.Precio, FABRICANTES.Nombre
FROM ARTICULOS, FABRICANTES
WHERE FABRICANTES.Nombre='Lexar' OR  FABRICANTES.Nombre='Kingston'
AND  ARTICULOS.Clave_fabricante=FABRICANTES.Clave_fabricante
ORDER BY ARTICULOS.Precio DESC;
```

u) Añade un nuevo producto: Clave del producto 11, Altavoces de $ 120 del fabricante 2

```
INSERT INTO ARTICULOS VALUES (11 ,'Altavoces', 120, 2);
```

v) Cambia el nombre del producto 6 a 'Impresora Laser'

```
UPDATE ARTICULOS SET Nombre='Impresora Laser'
WHERE Clave_articulo = 8
```

w) Aplicar un descuento del 10% a todos los productos.

```
UPDATE ARTICULOS SET Precio=Precio *0.10
```

x) Aplicar un descuento de $ 10 a todos los productos cuyo precio sea mayor o igual a $ 300

```
UPDATE ARTICULOS SET Precio=Precio – 10
WHERE  Precio >= 300
```

y) Borra el producto numero 6

```
DELETE FROM ARTICULOS WHERE Clave_articulo= 6
```

ANALIZAR LOS REQUISITOS DEL CLIENTE PARA CONSTRUIR EL SISTEMA DE INFORMACIÓN

Taller MySQL

BASES DE DATOS

TÉCNICO EN PROGRAMACIÓN DE SOFTWARE

REGIONAL ATLÁNTICO

SERVICIO NACIONAL DE APRENDIZAJE SENA

BARRANQUILLA

1. Desarrolle la siguiente práctica, para ello ingrese al siguiente link:

http://sqlfiddle.com

1. Crear una tabla llamada paciente y otra llamada atenciones, las cuales llevarán las siguientes estructuras:

Tabla: **Atención – Estructura y tipos de datos**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
codigo	int	11		☑	☑	☐	☐	☐	
id_paciente	int	11		☐	☐	☐	☐	☐	
fecha	varchar	20		☐	☐	☐	☐	☐	
hora	varchar	20		☐	☐	☐	☐	☐	
observacion	varchar	100		☐	☐	☐	☐	☐	

Registros para la tabla **Atención**

codigo	id_paciente	fecha	hora	observacion
1	1	19/11/2016	2:30	FIEBRE
2	1	19/11/2016	4:31	MALESTAR GENERAL
3	2	19/11/2017	10:20	DOLOR

Tabla: **Paciente – Estructura y tipos de datos**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?
id	int	11		☑	☑	☐	☐	☐
nombres	varchar	20		☐	☐	☐	☐	☐
apellidos	varchar	20		☐	☐	☐	☐	☐

Registros para la tabla **paciente**

id	nombres	apellidos
1	CARLOS	RIOS
2	MARIA	PEREZ
(NULL)	*(NULL)*	*(NULL)*

- Relacione las tablas (*Paciente y Atenciones*), teniendo en cuenta que el campo **id_paciente** de la tabla atenciones debe referenciar el campo **id** de la tabla paciente, creando una relación **1:N**.
- Realice una consulta donde muestre nombres, apellidos, fecha, hora, observación de “MARIA PEREZ”, para este punto utilizar la cláusula **INNER JOIN**.
- Inserte un paciente llamado “JHON DOE”, y registre 5 atenciones para este paciente (*Las fechas y las horas pueden ser inventadas*), las observaciones deberán ser las siguientes:

  Dolor de cabeza, Caída desde su propia altura, Fiebre, Intoxicación, Quemadura.
- Realice una consulta donde muestre: nombres y apellidos (*Concatenados*), fecha, hora y observación del paciente “JHON DOE”, para este punto utilizar la cláusula **INNER JOIN**.
- Realice una consulta donde muestre: nombres y apellidos (*Concatenados*) de los pacientes que fueron atendidos por Fiebre, para este punto utilizar la cláusula **INNER JOIN**.

Nota: Recuerda realizar tu informe con todos los puntos detallados en esta guía para soportar tu evidencia de aprendizaje y súbela en el link dispuesto para esta actividad en la plataforma.

ANALIZAR LOS REQUISITOS DEL CLIENTE PARA CONSTRUIR EL SISTEMA DE INFORMACIÓN

Taller MySQL

BASES DE DATOS

TÉCNICO EN PROGRAMACIÓN DE SOFTWARE

REGIONAL ATLÁNTICO

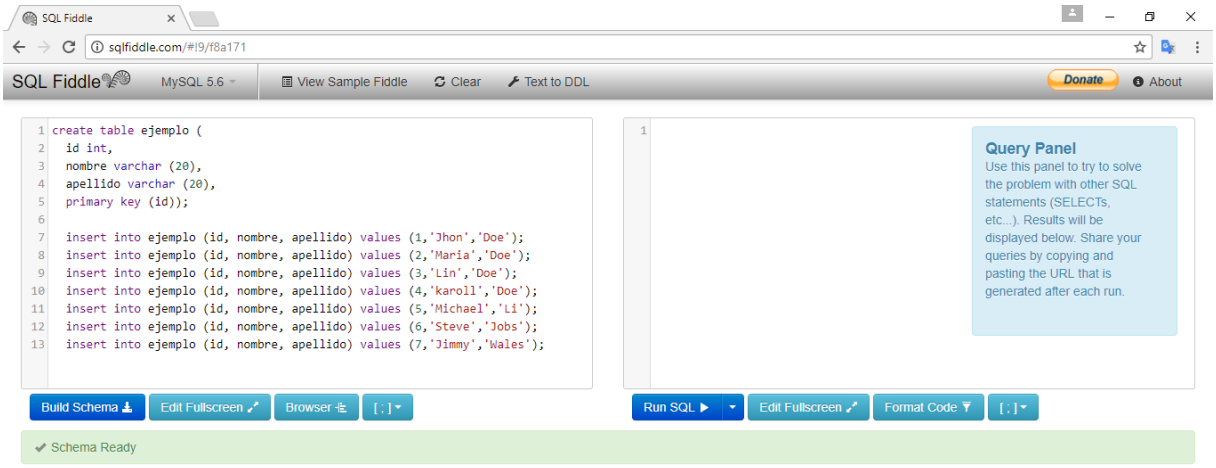
SERVICIO NACIONAL DE APRENDIZAJE SENA

BARRANQUILLA

1. Desarrolle la siguiente práctica, para ello ingrese al siguiente link:

http://sqlfiddle.com

Escriba las siguientes líneas en el campo mostrado en la siguiente imagen:



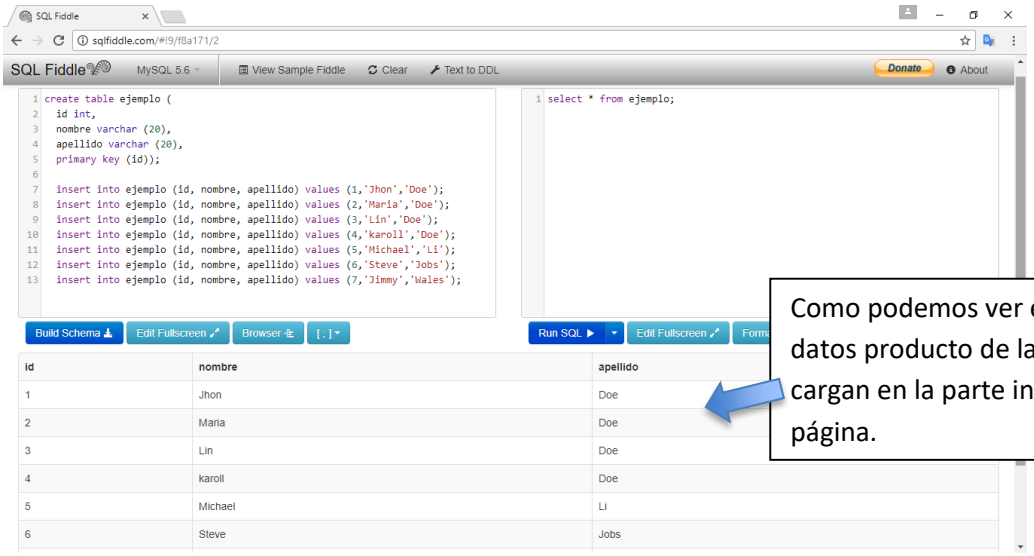
Responda el siguiente interrogante:

- Explique línea por línea que hace cada comando que acaba de escribir.

Al terminar de escribir los comandos presione **Build Schema** (*Debe aparecer el mensaje: Shema Ready tal como se ve en la imagen anterior*).

Realizando consultas

Mostrar todos los registros de la tabla ejemplo:



2. Realice las siguientes consultas SQL, para este punto tome capturas de pantalla para cada consulta, donde muestre el conjunto de resultados y la consulta utilizada.

- Ordene todos los registros de forma ascendente y descendente.
- Diga cuantos registros existen en la tabla.
- Muestre todos los nombres y apellidos concatenados de la tabla.
- Diga cuantas personas tienen apellido Doe.
- Realice una suma de todos los ID de la tabla.
- Muestre el promedio de todos los ID que están en la tabla.
- Muestre las personas que tienen id mayor a 4.
- Muestre las personas que tienen el id menor o igual a 6.
- Muestre todos los nombres de las personas con apellido Doe.
- Muestre nombres y apellidos concatenados de las personas con apellido diferente a Doe.
- Muestre id y apellidos de las personas con id mayor o igual a 4 y que su apellido sea diferente de “Jobs”.
- Muestre las personas cuyo nombre empiecen con M.
- Muestre las personas cuyo nombre contenga “ar” o que contenga “ae” o que contenga “on”.
- Diga cuantas personas su apellido termina con “es”.
- Diga cuantas personas su apellido termina en “e”.
- Muestre nombre y apellidos de las personas que su id está entre 3 y 6.
- Muestre la suma de los “id” de las personas que tienen apellido “Doe”.
- Muestre el promedio del campo “id”, de las personas que en su nombre contenga “ar”.
- Muestre los apellidos distintos de la tabla.
- Ordene de forma ascendente y descendente según el id todos los registros de la tabla.
- Inserte 10 registros más y muéstrelos (*Solo los que insertó*).

3. Crear una tabla llamada paciente y otra llamada atenciones, las cuales llevarán las siguientes estructuras:

Tabla: **Atención – Estructura y tipos de datos**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?	Comment
codigo	int	11		☑	☑	☐	☐	☐	
id_paciente	int	11		☐	☐	☐	☐	☐	
fecha	varchar	20		☐	☐	☐	☐	☐	
hora	varchar	20		☐	☐	☐	☐	☐	
observacion	varchar	100		☐	☐	☐	☐	☐	

Registros para la tabla **Atención**

codigo	id_paciente	fecha	hora	observacion
1	1	19/11/2016	2:30	FIEBRE
2	1	19/11/2016	4:31	MALESTAR GENERAL
3	2	19/11/2017	10:20	DOLOR

Tabla: **Paciente – Estructura y tipos de datos**

Column Name	Data Type	Length	Default	PK?	Not Null?	Unsigned?	Auto Incr?	Zerofill?
id	int	11		☑	☑	☐	☐	☐
nombres	varchar	20		☐	☐	☐	☐	☐
apellidos	varchar	20		☐	☐	☐	☐	☐

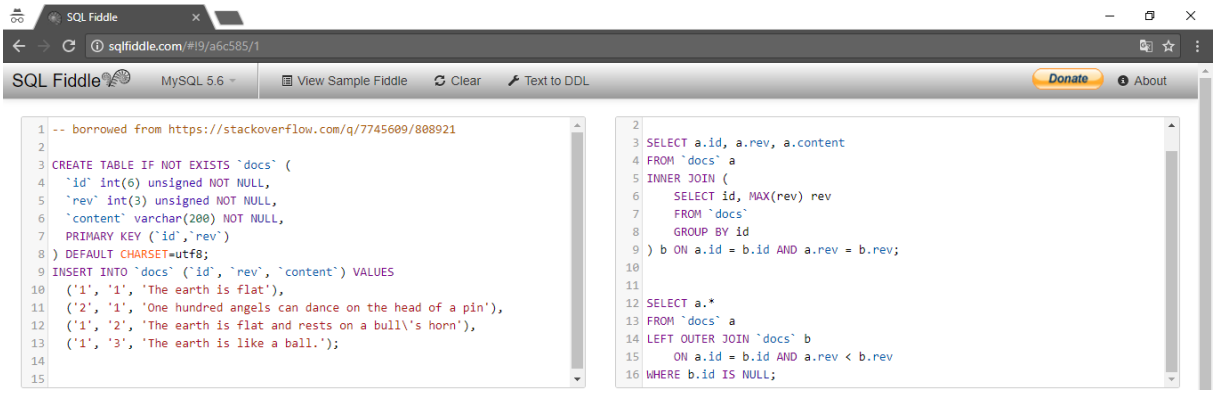
Registros para la tabla **paciente**

id	nombres	apellidos
1	CARLOS	RIOS
2	MARIA	PEREZ
(NULL)	*(NULL)*	*(NULL)*

- Relacione las tablas (*Paciente y Atenciones*), teniendo en cuenta que el campo **id_paciente** de la tabla atenciones debe referenciar el campo **id** de la tabla paciente, creando una relación **1:N**.
- Realice una consulta donde muestre nombres, apellidos, fecha, hora, observación de “MARIA PEREZ”, para este punto utilizar la cláusula **INNER JOIN**.
- Inserte un paciente llamado “JHON DOE”, y registre 5 atenciones para este paciente (*Las fechas y las horas pueden ser inventadas*), las observaciones deberán ser las siguientes:

  Dolor de cabeza, Caída desde su propia altura, Fiebre, Intoxicación, Quemadura.
- Realice una consulta donde muestre: nombres y apellidos (*Concatenados*), fecha, hora y observación del paciente “JHON DOE”, para este punto utilizar la cláusula **INNER JOIN**.
- Realice una consulta donde muestre: nombres y apellidos (*Concatenados*) de los pacientes que fueron atendidos por Fiebre, para este punto utilizar la cláusula **INNER JOIN**.

4. Explica detalladamente que se realiza en las siguientes consultas SQL:



Nota: Recuerda realizar tu informe con todos los puntos detallados en esta guía para soportar tu evidencia de aprendizaje y súbela en el link dispuesto para esta actividad en la plataforma.