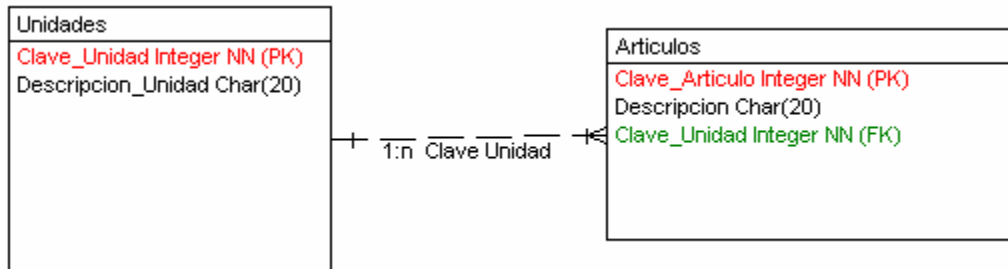


El objetivo de la siguiente practica, consiste en comprender como llevar un modelo relacional a la práctica, también podremos introducir registros para verificar la integridad en base al manejador, así como la forma simple de combinación de datos, todo esto será a través de MySQL.

El ejemplo que introduciremos será basado en las siguientes entidades, nombraremos a nuestra aplicación Almacén, y estará integrada por dos tablas relacionadas.



Accedemos al manejador.

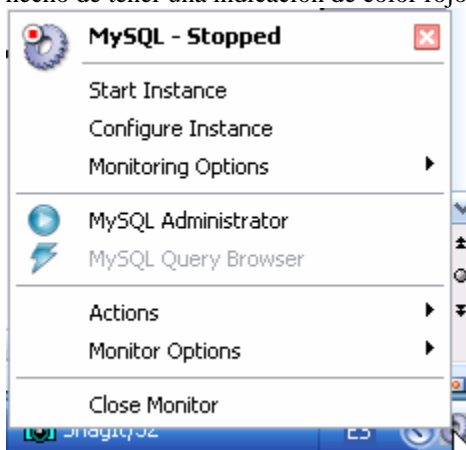
Si no sabemos si esta en ejecución del menú donde instalamos el Manejador ejecutamos la siguiente aplicación.



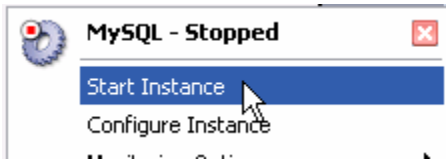
Esto agregara en la barra de tareas el siguiente icono.



Podemos oprimir el botón derecho del ratón para verificar el estatus del manejador, que de antemano por el hecho de tener una indicación de color rojo, nos indica que están detenidos los servicios.



Para ejecutar los servicios seleccionamos “Star Instance”



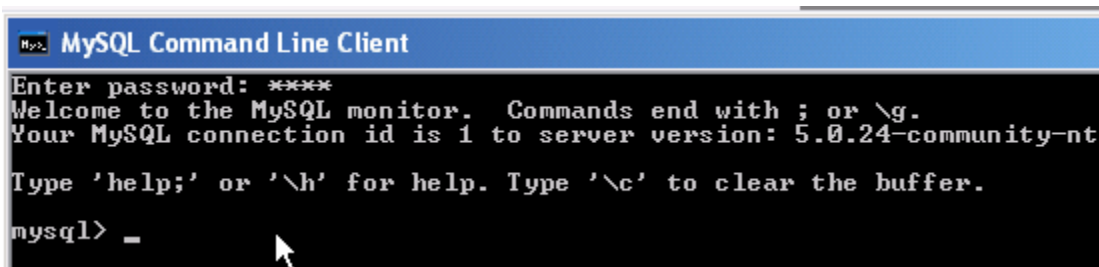
Ahora el icono tendrá un detalle en la esquina izquierda de color verde, lo que indica que los servicios están disponibles.



Ejecutamos el cliente en ambiente de texto.



Introducimos la contraseña que ustedes indicaron al momento de instalar.



Basado en el problema que planteamos, lo primero que tenemos que hacer es nombrar la aplicación, para esto usaremos el comando **create database nombre_de_la_base;** en este caso la base de datos se llamara almacén y estará conformada por las dos tablas.

```
mysql> create database almacen
-> ;
Query OK, 1 row affected (0.11 sec)
mysql>
```

En este ejemplo olvide introducir el punto y coma (;) al final de la instrucción el shell no se mostrara, lo que indicara que nos falta algo para completar la instrucción, no importa, podemos teclearlo en este momento y dar <Enter>, la forma mas común seria haber tecleado el comando completo.

create database almacén;

Ahora para indicarle que usaremos esta base tecleamos el comando **use Nombre_de_la_base;**

```
mysql> use almacen;_
```

Creación de la tabla de las unidades.

El comando para hacer esto es

```
create table Nombre_de_la_tabla ( campo1 tipo1, campo2 tipo2,..., primary key (campos) );
```

```
mysql> create table unidades ( clv_unid int(3) zerofill,
-> descrip_unid char(10),
-> primary key (clv_unid));
Query OK, 0 rows affected (0.06 sec)
mysql>
```

Explicación de la línea insertada en el shell.

create table unidades → Permite crear la tabla llamada unidades
 clv_unid int(3) zerofill → Crea el campo “clv_unid” (Clave de la unidad) del tipo entero de longitud tres, y además tendrá como característica que rellenara con ceros a la izquierda de la clave como 1 = 001, 2 = 002, 3 = 003, ... , 20 = 020, 21= 021,....., 100 = 100.
 descrip_unid char (10) Para la descripción de las unidades, esta será alfanumérica de longitud diez.
 primary key (clv_unid) → Le indicara al manejador que la llave primaria esta formada por el campo clv_unid (Clave de la unidad).

Ahora vamos a introducir información a la tabla, usaremos el comando

```
insert into Nombre_de_la_table values ( 'val_campo1','val_campo2',.....,'val_campon');
```

```
mysql> insert into unidades values (1,'Pieza'),(2,'Lata'), (3,'Metro')
-> ,(4,'Pulgada');
Query OK, 4 rows affected (0.00 sec)
Records: 4 Duplicates: 0 Warnings: 0
```

Explicación de la línea introducida

insert into unidades values → Permite introducir (insert), dentro (into) de la tabla unidades, los siguientes valores (values)
 (1,'pieza') → Clave unidad = 001, Descripción de la unidad = Pieza
 (2,'Lata') → Clave unidad = 002, Descripción de la unidad = Lata
 (3,'Metro') → Clave unidad = 003, Descripción de la unidad = Metro
 (4,'Pulgada') → Clave unidad = 004, Descripción de la unidad = Pulgada

Hemos introducido hasta el momento cuatro registro, para listarlos mediante el Editor de línea (shell) de MySQL, usamos la instrucción

```
select * from unidades ;
```

```
mysql> select * from unidades;
+-----+-----+
| clv_unid | descrip_unid |
+-----+-----+
| 001 | Pieza |
| 002 | Lata |
| 003 | Metro |
| 004 | Pulgada |
+-----+-----+
4 rows in set (0.00 sec)
```

Debemos interpretar la línea como sigue; seleccionar (select), todos los campos (*) de (from) la tabla unidades, esta línea permite ver toda la información de la tabla.

Vamos a introducir más elementos a esta tabla.

```
mysql> insert into unidades values (5,'MiliMetro'),(6,'Caja'), (7,'Gramo');
Query OK, 3 rows affected (0.01 sec)
Records: 3 Duplicates: 0 Warnings: 0
```

Ahora vamos a visualizar todos los datos.

```
mysql> select * from unidades;
+-----+-----+
| clv_unid | descrip_unid |
+-----+-----+
| 001 | Pieza |
| 002 | Lata |
| 003 | Metro |
| 004 | Pulgada |
| 005 | MiliMetro |
| 006 | Caja |
| 007 | Gramo |
+-----+-----+
7 rows in set (0.03 sec)
```

Vamos a crear rangos de visualización de datos, por ejemplo
Selecciona solo el campo de la clave de las unidades para su visualización.

```
mysql> select clv_unid from unidades;
+-----+
| clv_unid |
+-----+
| 001 |
| 002 |
| 003 |
| 004 |
| 005 |
| 006 |
| 007 |
+-----+
7 rows in set (0.00 sec)
```

O para solo ver la descripción de la tabla de las unidades, usaremos.

```
mysql> select descrip_unid from unidades;
+-----+
| descrip_unid |
+-----+
| Pieza |
| Lata |
| Metro |
| Pulgada |
| MiliMetro |
| Caja |
| Gramo |
+-----+
7 rows in set (0.00 sec)
```

La siguiente sentencia seleccionara (select), todos los campos (*) desde (from) la tabla unidades, donde (where) la clave de la unidad sea mayor o igual a dos y además la clave de la unidad sea menor que seis.

```
mysql> select * from unidades where clv_unid >= 2 and clv_unid < 6;
+-----+-----+
| clv_unid | descrip_unid |
+-----+-----+
| 002 | Lata |
| 003 | Metro |
| 004 | Pulgada |
| 005 | MiliMetro |
+-----+-----+
4 rows in set (0.02 sec)
```

Considero que en este punto ya quedo claro el concepto de la visualización del contenido de las tablas en términos de datos registro a registro, además ya podemos hacer consultas sencillas en base a ciertos criterios. Ahora vamos a tratar de introducir un dato que ya es repetido y que por definición le indicamos al manejador no se debe repetir por ser una llave primaria.

```
mysql> insert into unidades values (2, 'javier');
ERROR 1062 (23000): Duplicate entry '002' for key 1
```

En este ejemplo trate de introducir la clave numero dos (002), pero recordar que en la definición de la tabla, le indicamos que la llave primaria estaria formada por el campo clave de la unidad, así es que el manejador evita la duplicidad de información evitando registrar información, esto lo hace automáticamente, sin tener nosotros que validar absolutamente nada, este error no afecta la información, ya que jamás se almaceno, el proceso es simple, el manejador antes de registrar la información valida los datos en función de la definición formal realizada en la declaración de las tablas, por lo que rechaza indicando una violación, en este caso nos indica que la entrada con clave "002" esta duplicada para la llave primaria 1, la cual sabemos es la clave de la unidad.

Ahora crearemos la tabla para los artículos.

Para la clave de la unidad que será una llave débil usare clv_unid_arti para identificarla.

```
mysql> create table articulos (clv_arti int(4), descrip_arti char(20),
-> clv_unid_arti int(3), primary key (clv_arti));
Query OK, 0 rows affected (0.00 sec)
```

Observen que la clave de la unidad no la declare del tipo zerofill, ¿ afectara esto ?, ahora introduciremos algunos registros, recuerden que esto es manual y no hay forma de validar en este momento los datos entre las dos tablas, sabemos que el rango de las claves validas es entre uno y siete.

Si tenemos alguna duda podemos volver a ver los datos, recordar que ya sabemos el comando.

```
mysql> select * from unidades;
+-----+-----+
| clv_unid | descrip_unid |
+-----+-----+
| 001 | Pieza |
| 002 | Lata |
| 003 | Metro |
| 004 | Pulgada |
| 005 | MiliMetro |
| 006 | Caja |
| 007 | Gramo |
+-----+-----+
7 rows in set (0.00 sec)
```

Bien ahora introduciremos algunos registros con rangos validos de datos en el campo de las unidades.

```
mysql> insert into articulos values ('1','Coca Cola','1'),
-> ('2','Marcador','1'),('9','Lapiz','1'),('10','Aceite','2'),
-> ('5','Pepsi','6'),('100','Vaso','1'),('20','Sopa','6');
Query OK, 7 rows affected (0.02 sec)
Records: 7 Duplicates: 0 Warnings: 0
```

Observe que no existe orden alguno en los valores, claro sabemos que no podemos repetir los datos, de lo contrario obtendríamos un error, lo que evitaría registrar la información, además a diferencia de la captura de datos en la entidad de unidades todos los valores los indicamos con apostrofes, los valores enteros pueden ir o no con este símbolo, así es que no afecta lo usemos o no, pero recuerde que solamente los numero tienen esta cualidad, las cadenas de texto jamás.

Vamos a ver los datos que usamos

```
mysql> select * from articulos;
+-----+-----+-----+
| clv_arti | descrip_arti | clv_unid_arti |
+-----+-----+-----+
| 1 | Coca Cola | 1 |
| 2 | Marcador | 1 |
| 5 | Pepsi | 6 |
| 9 | Lapiz | 1 |
| 10 | Aceite | 2 |
| 20 | Sopa | 6 |
| 100 | Vaso | 1 |
+-----+-----+-----+
7 rows in set (0.00 sec)
```

Los datos ya están ordenados, lógicamente por la llave primaria en orden ascendente, esta es la opción por default, la cual podemos cambiar y veremos después, ya sabemos que podemos ver todos los campos o solo algunos de ellos.

Ejemplo para ver solamente los datos por el campo de la clave del artículo.

```
mysql> select clv_arti from articulos;
+-----+
| clv_arti |
+-----+
| 1 |
| 2 |
| 5 |
| 9 |
| 10 |
| 20 |
| 100 |
+-----+
7 rows in set (0.00 sec)
```

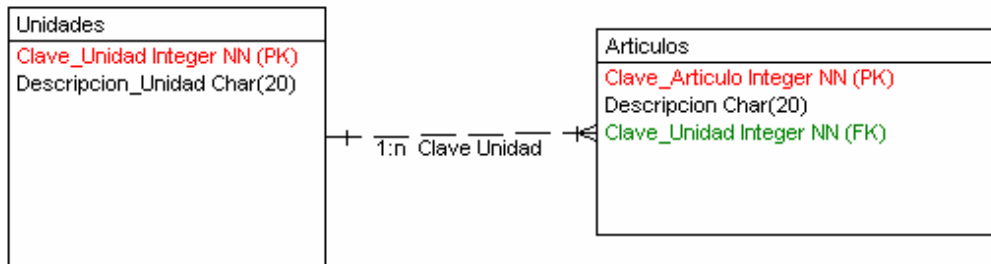
O para ver solo los datos de la clave del artículo y la descripción.

```
mysql> select clv_arti, descrip_arti from articulos;
+-----+-----+
| clv_arti | descrip_arti |
+-----+-----+
| 1 | Coca Cola |
| 2 | Marcador |
| 5 | Pepsi |
| 9 | Lapiz |
| 10 | Aceite |
| 20 | Sopa |
| 100 | Vaso |
+-----+-----+
7 rows in set (0.00 sec)
```

O la clave de la unidad primero, después la clave del artículo y al final la descripción.

```
mysql> select clv_unid_arti, clv_arti, descrip_arti from articulos;
+-----+-----+-----+
| clv_unid_arti | clv_arti | descrip_arti |
+-----+-----+-----+
|          1   |        1 | Coca Cola   |
|          1   |        2 | Marcador   |
|          6   |        5 | Pepsi      |
|          1   |        9 | Lapiz      |
|          2   |       10 | Aceite     |
|          6   |       20 | Sopa      |
|          1   |      100 | Vaso      |
+-----+-----+-----+
7 rows in set (0.00 sec)
```

Ahora vamos a combinar visualmente los datos para comprender la esencia total de esta práctica y entender como es que están relacionados los datos entre las entidades. El problema es el siguiente, queremos listar una relación de los artículos mostrando la clave del artículo, la descripción del mismo, la clave de la unidad del artículo y la descripción de la unidad, tenemos el diagrama inicial.



El comando quedaría de la siguiente forma.

Selecciona (select) la clave del artículo (clv_arti), la descripción del artículo (descrip_arti), la clave de la unidad (clv_unid_arti) y la descripción de la unidad que sabemos esta en la entidad unidades (unidades.descripcion_unid), desde las tablas (from) artículos y unidades, donde (where) la clave de la unidad del artículo sea igual a la clave de la unidad en la tabla de las unidades (unidades.clv_unid)

```
mysql> select clv_arti, descrip_arti, clv_unid_arti, unidades.descripcion_unid
-> from articulos,unidades where articulos.clv_unid_arti=unidades.clv_unid;
+-----+-----+-----+-----+
| clv_arti | descrip_arti | clv_unid_arti | descripcion_unid |
+-----+-----+-----+-----+
|        1 | Coca Cola   |          1   | Pieza           |
|        2 | Marcador   |          1   | Pieza           |
|        5 | Pepsi      |          6   | Caja            |
|        9 | Lapiz      |          1   | Pieza           |
|       10 | Aceite     |          2   | Lata            |
|       20 | Sopa      |          6   | Caja            |
|      100 | Vaso      |          1   | Pieza           |
+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

El proceso que realiza busca por cada registro de la entidad artículos el registro que equivale dentro de la entidad unidades y lo muestra para que visualmente nos muestre su descripción, no estamos copiando n realizando ningún tipo de operación sobre los datos, solamente los estamos mostrando combinados.

¿Que harán los siguientes comandos?.

```
mysql> select clv_arti, descrip_arti, clv_unid_arti, descrip_unid from articulos
, unidades where clv_unid_arti=clv_unid;
```

No existe ninguna referencia del campo a que tabla corresponde, cuando los campos de las tablas son distintos entre si, MySQL asume la referencia hacia su tabla de origen automáticamente, sin ningun problema, pero si los campos de la clave de la unidad de las dos entidades fueran iguales, tendríamos que hacer forzosamente la referencia para distinguirlos adecuadamente sin error, por ejemplo suponer que los campos de la clave de la unidad en ambas entidades fuera clv_unid, el comando tendria que integrarse asi.

```
select clv_arti, descrip_arti, articulos.clv_uni descrip_unid from articulos, unidades
where articulos.clv_unid = unidades.clv_unid ;
```

Referencia a la entidad de los **articulos**

Referencia a la entidad de **unidades**

Este comando (order by campo) nos permite cambiar el orden en la visualizacion, para este ejemplo ordenaremos los datos por el campo de la descripción en orden ascendente (A-Z).

```
mysql> select * from articulos order by descrip_arti;
+-----+-----+-----+
| clv_arti | descrip_arti | clv_unid_arti |
+-----+-----+-----+
|      10 | Aceite       |              2 |
|       1 | Coca Cola   |              1 |
|       9 | Lapis       |              1 |
|       2 | Marcador    |              1 |
|       5 | Pepsi       |              6 |
|      20 | Sopa        |              6 |
|     100 | Vaso        |              1 |
+-----+-----+-----+
7 rows in set (0.00 sec)
```

Para verlos en orden descendente.

```
mysql> select * from articulos order by descrip_arti desc;
+-----+-----+-----+
| clv_arti | descrip_arti | clv_unid_arti |
+-----+-----+-----+
|     100 | Vaso        |              1 |
|      20 | Sopa        |              6 |
|       5 | Pepsi       |              6 |
|       2 | Marcador    |              1 |
|       9 | Lapis       |              1 |
|       1 | Coca Cola   |              1 |
|      10 | Aceite       |              2 |
+-----+-----+-----+
7 rows in set (0.00 sec)
```