

Manejo de Tablas y Llaves Sencillas

El Motor de Almacenamiento

El motor de almacenamiento (storage-engine) se encarga de almacenar, manejar y recuperar información de una tabla. Los motores más conocidos son MyISAM e InnoDB.

Principalmente InnoDB se utiliza cuando se tiene un diseño Relacional y se usan muchas sentencias para modificar la información, mientras que MyISAM se recomienda cuando se trabajan principalmente sentencias de consulta. El motor por defecto de MySQL antes de la versión 5.5 era MyISAM,. En versiones posteriores se cambió a InnoDB.

Para conocer que motor de almacenamiento se está utilizando en una determinada base de datos, se ejecutan las siguientes instrucciones

```
USE nombre_tabla;  
SHOW TABLE STATUS;
```

Para cambiar el motor de almacenamiento, se utilizará la siguiente instrucción:

```
ENGINE=InnoDB
```

Consulta de Tablas en MySQL

Para saber que tablas tiene una base de datos se utiliza la sentencia.

```
SHOW TABLES FROM nombre_base
```

Para conocer las propiedades de las columnas de una tabla, ya con una base de datos seleccionada se utiliza algunas de las siguientes sentencias.

```
DESCRIBE nombre_tabla  
SHOW COLUMNS FROM nombre_tabla
```

Creación de Tablas

La sentencia básica para la creación de Tablas es:

```
CREATE TABLE nombre_tabla(  
nombre_columna TIPO  
) ENGINE=InnoDB
```

Se debe de crear al menos una columna en la tabla. Si se quieren tener más columnas, estas se deben separar por coma (,). Después del último atributo no se coloca coma.

```
CREATE TABLE nombre_tabla(  
nombre_columna_1 TIPO,  
nombre_columna_2 TIPO  
) ENGINE=InnoDB
```

Valores No Nulos

Se puede especificar si se desea permitir o no valores nulos en una columna

```
CREATE TABLE nombre_tabla (  
nombre_columna_1 TIPO NOT NULL,  
columna_dos TIPO NULL  
) ENGINE=InnoDB
```

Valores por Defecto

Se puede especificar un valor por defecto en las columnas, si se permite un valor nulo, el valor por defecto será NULL, en caso contrario el valor por defecto será el especificado.

```
CREATE TABLE nombre_tabla (  
nombre_columna TIPO DEFAULT valor  
) ENGINE=InnoDB
```

```
CREATE TABLE nombre_tabla (  
nombre_columna TIPO NULL DEFAULT valor  
) ENGINE=InnoDB
```

Llaves Primarias

Para especificar una llave primaria, se debe utilizar la palabra clave **KEY** O **PRIMARY KEY** después del tipo del atributo que se desea usar como llave, una llave primaria nunca debe ser nula, aunque no es necesario especificar esto, al declararla como primaria, automáticamente tendrá esta propiedad

```
CREATE TABLE nombre_tabla (  
nombre_columna_1 TIPO PRIMARY KEY,  
nombre_columna_2 TIPO  
) ENGINE=InnoDB
```

Llaves Primarias Compuestas

Si se quiere utilizar dos o más atributos como llave primaria, se puede especificar al momento de crear la tabla, colocando entre paréntesis el nombre de los atributos separados por coma después de la palabra **PRIMARY KEY**, en este caso la palabra **PRIMARY KEY** no se coloca después de ningún atributo, esto no crea una columna extra, sino que maneja los atributos como **PK**, es decir, no pueden

ser nulos y no puede haber combinación de valores repetidos. Estos atributos no deben permitir valores nulos.

```
CREATE TABLE nombre_tabla (  
nombre_columna_1 TIPO NOT NULL,  
nombre_columna_2 TIPO NOT NULL,  
nombre_columna_3 TIPO,  
PRIMARY KEY (nombre_columna_1, nombre_columna_2)  
) ENGINE=InnoDB
```

Llaves Auto Incrementables

Este tipo de llaves se utiliza principalmente para declarar llaves primarias cuando no hay un atributo que se pueda utilizar como tal, se coloca la palabra clave **AUTO_INCREMENT** después del tipo del atributo que se desea que sea auto incrementable, normalmente va seguido de **PRIMARY KEY**

```
CREATE TABLE nombre_tabla (  
nombre_columna TIPO AUTO_INCREMENT PRIMARY KEY,  
nombre_columna_2 TIPO  
) ENGINE=InnoDB
```

Índices

Los índices son muy útiles cuando se tienen tablas con muchas tuplas, se pueden utilizar para optimizar las búsquedas creando índices para determinadas columnas que mejoren el desempeño de la búsqueda. No son llaves primarias, ya que pueden repetirse o tener valores nulos.

```
CREATE TABLE nombre_tabla (  
nombre_columna TIPO,  
nombre_columna_indice TIPO,  
INDEX (nombre_columna_indice)  
) ENGINE=InnoDB
```

Llaves Únicas

Otra manera de identificar a una tupla sin que esta sea una llave primaria como tal, es la de crear uno de sus atributos como único, aquí no se pueden repetir valores, salvo los que sean nulos, lo que la distingue de una llave primaria.

```
CREATE TABLE nombre_tabla (  
nombre_columna TIPO,  
nombre_columna_unica TIPO,  
UNIQUE (nombre_columna_unico)  
) ENGINE=InnoDB
```

Consultando las Tablas de una Base de Datos

Para consultar las tablas de una base de datos

```
SHOW FULL TABLES FROM nombre_base
```

Modificar el Nombre de una Tabla

Para cambiar el nombre a una Tabla, se tiene la instrucción ALTER TABLE y RENAME, en caso de que haya tablas relacionadas a esta con llaves foráneas, el cambio en la referencia será automático.

```
ALTER TABLE nombre_tabla RENAME nuevo_nombre
```

Eliminar una Tabla

Para eliminar una tabla ya creada, se debe utilizar la instrucción DROP TABLE junto con el nombre de la tabla a eliminar.

```
DROP TABLE nombre_tabla
```

Modificando las Columnas

Consultar los Nombres y Columnas de una Tabla

Es posible consultar el nombre y el tipo de las columnas de una tabla

```
SHOW COLUMNS FROM nombre_tabla
```

Consultar los Nombres y Columnas de Todas las Tablas

Es posible consultar los nombres de todas las columnas y tablas que existen en la base de datos

```
SELECT COLUMN_NAME, table_name, data_type FROM  
INFORMATION_SCHEMA.COLUMNS  
WHERE table_schema = 'nombre_tabla'
```

Cambiar el Nombre a una Columna

Es posible cambiar el nombre de una columna

```
ALTER TABLE nombre_tabla CHANGE nombre_columna nombre_columna TIPO
```

Cambiar Nombre y Tipo a una Columna

También se puede cambiar nombre y tipo al mismo tiempo.

```
ALTER TABLE nombre_tabla CHANGE nombre_columna nuevo_nombre NUEVO_TIPO
```

Al momento de poner nuevamente la especificación de la columna a modificar, se pueden quitar atributos como si acepta o no valores nulos .

Eliminando Atributos

Para eliminar un atributo de llave primaria que no es auto incrementable.

```
ALTER TABLE nombre_tabla DROP PRIMARY KEY
```

Para eliminar el atributo de llave primaria que es auto incrementable.

```
ALTER TABLE nombre_tabla MODIFY nombre_columna_1 INT NOT NULL  
ALTER TABLE nombre_tabla DROP PRIMARY KEY
```

Eliminar Columnas

Para eliminar una columna se utiliza la instrucción ALTER TABLE junto con la instrucción DROP.

```
ALTER TABLE nombre_tabla DROP columna_a_borrar
```

Agregando Columnas

Es posible agregar columnas a una tabla ya creada, esto con las instrucciones ALTER TABLE y ADD COLUMN seguido de la definición de la columna (nombre, tipo y modificadores si son necesarios).

```
ALTER TABLE nombre_tabla ADD COLUMN nombre_columna TIPO
```

Esto agregará la columna al final del resto, también se puede especificar si se quiere agregar al inicio o después de una determinada columna.

```
ALTER TABLE nombre_tabla ADD COLUMN nombre_columna TIPO FIRST
```

```
ALTER TABLE nombre_tabla ADD COLUMN nombre_columna TIPO AFTER  
columna_anterior
```