



El empleo
es de todos

Mintrabajo

DESARROLLO DE BASE DE DATOS CON MYSQL



@SENAcomunica

www.sena.edu.co



DESARROLLO DE BASE DE DATOS CON MYSQL



@SENAcomunica

www.sena.edu.co

9

USO DE SENTENCIAS DDL

OBJETIVOS



Después de completar esta lección usted estará en la capacidad de:

- **Clasificar los objetos principales de la base de datos.**
- **Revisar la estructura de tabla**
- **Mostrar los tipos de datos que están disponibles para las columnas.**
- **Crear una tabla simple.**
- **Comprender cómo se crean las restricciones en el momento de la creación la tabla.**
- **Describir cómo funcionan los objetos de la base de datos .**

OBJETOS DE BASE DE DATOS



Objeto	Descripción
Tabla	Unidad básica de almacenamiento; compuesta por filas
Vista	Representación de forma lógica subconjuntos de datos de una o más tablas.
Indice	Mejora el rendimiento de algunas consultas
Sinónimo	Proporciona nombres alternativos a objetos

REGLAS DE NOMENCLATURA



Los nombres de las tablas y columnas:

- **Deben comenzar por una letra.**
- **Solo deben contener A-Z, a-z, 0-9, _, \$, #.**
- **No debe duplicar el nombre de otro objeto que sea propiedad del mismo usuario.**
- **No deben ser palabras reservadas**

REFERENCIA A TABLAS DE OTRO USUARIO



- Las tablas pertenecientes a otros usuarios no están en el esquema del usuario.
- Debe utilizar el nombre del propietario como prefijo de dichas tablas.

Desde el usuario A

Select * from B.employees;

Desde el usuario B

Select * from A.employees;

OPCION DEFAULT



- Especifica un valor por defecto para una columna durante una inserción.

```
MariaDB [hr]> CREATE TABLE HIRE_DATES(  
    -> ID INTEGER(8),  
    -> HIRE_DATE DATETIME DEFAULT CURRENT_TIMESTAMP  
    -> );  
Query OK, 0 rows affected (0.07 sec)
```

```
MariaDB [hr]> DESC HIRE_DATES;
```

Field	Type	Null	Key	Default	Extra
ID	int(8)	YES		NULL	
HIRE_DATE	datetime	YES		CURRENT_TIMESTAMP	

2 rows in set (0.04 sec)

TIPOS DE DATOS NUMERICOS

Tipo	Descripción
TINYINT	Datos numéricos de un byte -128
SMALLINT	Datos numéricos de dos bytes -32768
MEDIUMINT	Datos numéricos de tres bytes -8388608
INT o INTEGER	Datos numéricos de cuatro bytes -2147483648
BIGINT	Número entero con o sin signo. Con signo el rango de valores va desde -9.223.372.036.854.775.808 a 9.223.372.036.854.775.807.
FLOAT	Número pequeño en coma flotante de precisión simple. Los valores válidos van desde -3.402823466E+38 a -1.175494351E-38, 0 y desde 1.175494351E-38 a 3.402823466E+38.

TIPOS DE DATOS NUMERICOS

Tipo	Descripción
xReal, Double	Número en coma flotante de precisión doble. Los valores permitidos van desde -1.7976931348623157E+308 a -2.2250738585072014E-308, 0 y desde 2.2250738585072014E-308 a 1.7976931348623157E+308
DOUBLE	Datos numéricos de ocho bytes
DECIMAL(M,D)	M+2 bytes sí D > 0, M+1 bytes sí D = 0

TIPOS DE DATOS FECHA

Tipo	Descripción
Date	Tipo fecha, almacena una fecha. El rango de valores va desde el 1 de enero del 1001 al 31 de diciembre de 9999. El formato de almacenamiento es de año-mes-día
DateTime	Combinación de fecha y hora. El rango de valores va desde el 1 de enero del 1001 a las 0 horas, 0 minutos y 0 segundos al 31 de diciembre del 9999 a las 23 horas, 59 minutos y 59 segundos. El formato de almacenamiento es de año-mes-día horas:minutos:segundos
TimeStamp	Combinación de fecha y hora. El rango va desde el 1 de enero de 1970 al año 2037. El formato de almacenamiento depende del tamaño del campo
Time	Almacena una hora. El rango de horas va desde -838 horas, 59 minutos y 59 segundos a 838, 59 minutos y 59 segundos. El formato de almacenamiento es de 'HH:MM:SS'
Year	Almacena un año. El rango de valores permitidos va desde el año 1901 al año 2155. El campo puede tener tamaño dos o tamaño 4 dependiendo de si queremos almacenar el año con dos o cuatro dígitos.

TIPOS DE DATOS DE CADENA

Tipo	Descripción
VARCHAR(tamaño)	Datos de caracteres de longitud variable.
CHAR(tamaño)	Datos de caracteres de longitud fija.
TinyText y TinyBlob	Columna con una longitud máxima de 255 caracteres.
Blob y Text	Un texto con un máximo de 65535 caracteres.
MediumBlob y MediumText	Un texto con un máximo de 16.777.215 caracteres.
LongBlob y LongText	Un texto con un máximo de caracteres 4.294.967.295. Hay que tener en cuenta que debido a los protocolos de comunicación los paquetes pueden tener un máximo de 16 Mb.
Enum	Campo que puede tener un único valor de una lista que se especifica. El tipo Enum acepta hasta 65535 valores distintos
Set	Un campo que puede contener ninguno, uno ó varios valores de una lista. La lista puede tener un máximo de 64 valores.

RESTRICCIONES (CONSTRAINT)



- Las restricciones fuerzan las reglas en el nivel de tablas
- Las restricciones evitan el borrado de una tabla si hay dependencias.
- Son válidos los siguientes tipos de restricciones:
 - NOT NULL
 - UNIQUE
 - PRIMARY KEY
 - FOREIGN KEY
 - CHECK

RESTRICCIONES (CONSTRAINT)



- Restricciones a nivel de columna

```
MariaDB [hr]> CREATE TABLE EMPLEADOS(  
  -> EMPLEADO_ID INTEGER PRIMARY KEY,  
  -> FIRST_NAME VARCHAR(20)  
  -> );  
Query OK, 0 rows affected (0.05 sec)
```

- Restricciones a nivel de tabla

```
MariaDB [hr]> CREATE TABLE EMPLEADOS1(  
  -> EMPLEADO_ID INTEGER,  
  -> FIRST_NAME VARCHAR(20),  
  -> CONSTRAINT EMPLEADOS1PK PRIMARY KEY(EMPLEADO_ID)  
  -> );  
Query OK, 0 rows affected (0.04 sec)
```

RESTRICCIONES (NOT NULL)



- Garantiza que no permita valores nulos en la columna

```
MariaDB [hr]> CREATE TABLE EMPLEADOS2(  
    -> EMPLEADO_ID INTEGER,  
    -> FIRST_NAME VARCHAR(20) NOT NULL,  
    -> CONSTRAINT EMPLEADOS2PK PRIMARY KEY(EMPLEADO_ID)  
    -> );  
Query OK, 0 rows affected (0.04 sec)
```

RESTRICCIONES (UNIQUE)



- Garantiza que no permita valores repetidos en la columna

```
MariaDB [hr]> CREATE TABLE EMPLEADOS3(  
  -> EMPLEADO_ID INTEGER,  
  -> FIRST_NAME VARCHAR(20) NOT NULL,  
  -> EMAIL VARCHAR(25),  
  -> CONSTRAINT EMPLEADOS3PK PRIMARY KEY(EMPLEADO_ID),  
  -> CONSTRAINT EMPLEADOS3EMAILUK UNIQUE(EMAIL)  
  -> );  
Query OK, 0 rows affected (0.03 sec)
```


RESTRICCIONES (FOREIGN KEY)



- **Es la lista de los posibles valores que puede tomar un campo**
- **Estos valores deben estar en otra tabla**
- **En campos de llave primaria**
- **Pueden aceptar valores nulos**
- **Toda relación es llave foránea.**

RESTRICCIONES (FOREIGN KEY)



- **FOREIGN KEY:** Define la columna(s) de la tabla secundaria en el nivel de restricción de tabla.
- **REFERENCES:** Identifica la tabla y la(s) columna(s) de la tabla a enlazar.
- Las columnas a enlazar deben ser de la llave primaria de la tabla a referenciar.

```
MariaDB [hr]> CREATE TABLE EMPLEADOS4(  
-> EMPLEADO_ID INTEGER,  
-> FIRST_NAME VARCHAR(20) NOT NULL,  
-> EMAIL VARCHAR(25),  
-> DEPARTMENT_ID INT(11) UNSIGNED,  
-> CONSTRAINT EMPLEADOS4PK PRIMARY KEY(EMPLEADO_ID),  
-> CONSTRAINT EMPLEADOS4EMAILUK UNIQUE(EMAIL),  
-> CONSTRAINT EMPLEADOS4DEPARTMANETOFK FOREIGN KEY(DEPARTMENT_ID)  
-> REFERENCES DEPARTMENTS(DEPARTMENT_ID)  
-> );  
Query OK, 0 rows affected (0.06 sec)
```

RESTRICCIONES (CHECK)



```
MariaDB [hr]> CREATE TABLE EMPLEADOS5(  
  -> EMPLEADO_ID INTEGER,  
  -> FIRST_NAME VARCHAR(20) NOT NULL,  
  -> EMAIL VARCHAR(25),  
  -> DEPARTMENT_ID INT(11) UNSIGNED,  
  -> SALARIO INTEGER,  
  -> CONSTRAINT EMPLEADOS5PK PRIMARY KEY(EMPLEADO_ID),  
  -> CONSTRAINT EMPLEADOS5EMAILUK UNIQUE(EMAIL),  
  -> CONSTRAINT EMPLEADOS5DEPARTMANETOFK FOREIGN KEY(DEPARTMENT_ID)  
  -> REFERENCES DEPARTMENTS(DEPARTMENT_ID),  
  -> CONSTRAINT EMPLEADOS5SALARIOCHK CHECK(SALARIO >0)  
  -> );  
Query OK, 0 rows affected (0.05 sec)
```

- NO FUNCIONA

ALTERANDO OBJETOS (ALTER)



Utilice la sentencia ALTER TABLE para:

- **Agregar una nueva columna**
- **Modificar una columna existente.**
- **Definir un valor por defecto para la nueva columna.**
- **Borrar una columna**
- **Agregar una restricción a una columna.**
- **Borrar una restricción a una tabla;**

ALTERANDO OBJETOS (ALTER)



- ADICIONAR UN CAMPO EN UNA TABLA

```
MariaDB [hr]> ALTER TABLE EMPLEADOS5 ADD COLUMN LAST_NAME VARCHAR(20) NOT NULL;  
Query OK, 0 rows affected (0.11 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
MariaDB [hr]> DESC EMPLEADOS5;
```

Field	Type	Null	Key	Default	Extra
EMPLEADO_ID	int(11)	NO	PRI	NULL	
FIRST_NAME	varchar(20)	NO		NULL	
EMAIL	varchar(25)	YES	UNI	NULL	
DEPARTMENT_ID	int(11) unsigned	YES	MUL	NULL	
SALARIO	int(11)	YES		NULL	
LAST_NAME	varchar(20)	NO		NULL	

```
6 rows in set (0.02 sec)
```

ALTERANDO OBJETOS (ALTER)



- MODIFICAR UN CAMPO EN UNA TABLA

```
MariaDB [hr]> ALTER TABLE EMPLEADOS5 MODIFY COLUMN LAST_NAME VARCHAR(30) NOT NULL;  
Query OK, 1 row affected (0.15 sec)  
Records: 1 Duplicates: 0 Warnings: 0
```

```
MariaDB [hr]> DESC EMPLEADOS5;
```

Field	Type	Null	Key	Default	Extra
EMPLEADO_ID	int(11)	NO	PRI	NULL	
FIRST_NAME	varchar(20)	NO		NULL	
EMAIL	varchar(25)	YES	UNI	NULL	
DEPARTMENT_ID	int(11) unsigned	YES	MUL	NULL	
SALARIO	int(11)	YES		NULL	
LAST_NAME	varchar(30)	NO		NULL	

```
6 rows in set (0.03 sec)
```


ALTERANDO OBJETOS (ALTER)



- RENOMBRAR UN CAMPO EN UNA TABLA

```
MariaDB [hr]> ALTER TABLE EMPLEADOS5 CHANGE LAST_NAME APELLIDO VARCHAR(30);  
Query OK, 0 rows affected (0.12 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
MariaDB [hr]> DESC EMPLEADOS5;
```

Field	Type	Null	Key	Default	Extra
EMPLEADO_ID	int(11)	NO	PRI	NULL	
FIRST_NAME	varchar(20)	NO		NULL	
EMAIL	varchar(25)	YES	UNI	NULL	
DEPARTMENT_ID	int(11) unsigned	YES	MUL	NULL	
SALARIO	int(11)	YES		NULL	
APELLIDO	varchar(30)	YES		NULL	

```
6 rows in set (0.02 sec)
```

ALTERANDO OBJETOS (ALTER)



- RENOMBRAR UNA TABLA

```
MariaDB [hr]> ALTER TABLE EMPLEADOS5 RENAME EMPLEADITOS;  
Query OK, 0 rows affected (0.05 sec)
```

```
MariaDB [hr]> DESC EMPLEADOS5;  
ERROR 1146 (42S02): Table 'hr.empleados5' doesn't exist  
MariaDB [hr]> DESC EMPLEADITOS;
```

Field	Type	Null	Key	Default	Extra
EMPLEADO_ID	int(11)	NO	PRI	NULL	
FIRST_NAME	varchar(20)	NO		NULL	
EMAIL	varchar(25)	YES	UNI	NULL	
DEPARTMENT_ID	int(11) unsigned	YES	MUL	NULL	
SALARIO	int(11)	YES		NULL	
APELLIDO	varchar(30)	YES		NULL	

```
6 rows in set (0.02 sec)
```


ALTERANDO OBJETOS (ALTER)



- BORRANDO UN CONSTRAINT UNIQUE DE UNA TABLA

```
MariaDB [hr]> ALTER TABLE EMPLEADITOS DROP KEY EMPLEADOS5EMAILUK;  
Query OK, 0 rows affected (0.04 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

```
MariaDB [hr]> DESC EMPLEADITOS;
```

Field	Type	Null	Key	Default	Extra
EMPLEADO_ID	int(11)	NO	PRI	NULL	
FIRST_NAME	varchar(20)	NO		NULL	
EMAIL	varchar(25)	YES		NULL	
DEPARTMENT_ID	int(11) unsigned	YES	MUL	NULL	
SALARIO	int(11)	YES		NULL	
APELLIDO	varchar(30)	YES		NULL	

```
6 rows in set (0.03 sec)
```

ALTERANDO OBJETOS (ALTER)



- BORRANDO UN CONSTRAINT FOREIGN KEY DE UNA TABLA

```
MariaDB [hr]> ALTER TABLE EMPLEADITOS DROP FOREIGN KEY EMPLEADOS5DEPARTMANETOFK;  
Query OK, 0 rows affected (0.02 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

ALTERANDO OBJETOS (ALTER)



- BORRANDO UN CONSTRAINT PRIMARY KEY DE UNA TABLA

```
MariaDB [hr]> ALTER TABLE EMPLEADITOS DROP PRIMARY KEY;  
Query OK, 1 row affected (0.11 sec)  
Records: 1 Duplicates: 0 Warnings: 0
```

```
MariaDB [hr]> DESC EMPLEADITOS;
```

Field	Type	Null	Key	Default	Extra
EMPLEADO_ID	int(11)	NO		NULL	
FIRST_NAME	varchar(20)	NO		NULL	
EMAIL	varchar(25)	YES		NULL	
DEPARTMENT_ID	int(11) unsigned	YES	MUL	NULL	
SALARIO	int(11)	YES		NULL	
APELLIDO	varchar(30)	YES		NULL	

```
6 rows in set (0.02 sec)
```

ALTERANDO OBJETOS (ALTER)



- BORRANDO UNA TABLA

```
MariaDB [hr]> DROP TABLE EMPLEADITOS;  
Query OK, 0 rows affected (0.05 sec)
```

```
MariaDB [hr]> SELECT * FROM EMPLEADITOS;  
ERROR 1146 (42S02): Table 'hr.empleaditos' doesn't exist
```



GRACIAS

Línea de atención al ciudadano: 018000 910270
Línea de atención al empresario: 018000 910682



@SENAcomunica

www.sena.edu.co

CRÉDITOS



Realizado por el instructor José Fernando Galindo Suárez
jgalindos@sena.edu.co 2020

