
PRÁCTICA 5: CREACIÓN DE TABLAS

- Crea en MySQL la base de datos **PRACTICAS**.

```
CREATE DATABASE PRACTICAS;
```

- Crea la tabla EJEMPLO definiendo las columnas NIF y NOMBRE como no nulas. Dale un nombre a la restricción no nula de la columna NOMBRE. Después de crearla, añade una nueva columna EDAD con un valor por defecto de 18.

```
CREATE TABLE EJEMPLO(  
    NIF VARCHAR(9) NOT NULL,  
    NOMBRE VARCHAR(50) NOT NULL  
)ENGINE=innodb;
```

```
ALTER TABLE EJEMPLO ADD EDAD tinyint DEFAULT 18;
```

- Crea la tabla EJEMPLO1 definiendo las columnas DNI, NOMBRE y FECHA, asignando a la columna FECHA una fecha por defecto.

```
CREATE TABLE EJEMPLO1(  
    DNI VARCHAR(9) ,  
    NOMBRE VARCHAR(50),  
    FECHA DATE DEFAULT '2012-11-12'  
)ENGINE=innodb;
```

- Crea la tabla EJEMPLO2 con las columnas DNI, NOMBRE y USUARIO; y asigna por defecto a la columna NOMBRE el literal 'No definido'.

```
CREATE TABLE EJEMPLO2(  
    DNI VARCHAR(9) ,  
    NOMBRE VARCHAR(50) DEFAULT 'NO DEFINIDO',  
    USUARIO VARCHAR(30)  
)ENGINE=innodb;
```

- Crea la tabla EJEMPLO3. Las columnas son

DNI	Alfanumérico(10)
NOMBRE	Alfanumérico(30)
EDAD	Númérico entero(2)
CURSO	Númérico entero(1)

y las restricciones son:

- ✗ El DNI no puede ser nulo.
- ✗ La clave primaria es el DNI
- ✗ El NOMBRE no puede ser nulo
- ✗ La EDAD ha de estar comprendida entre 5 y 20 años

- ✗ El curso sólo puede almacenar 1, 2 ó 3.

```
CREATE TABLE EJEMPLO3(
  DNI VARCHAR(9) PRIMARY KEY,
  NOMBRE VARCHAR(30) NOT NULL,
  EDAD tinyint CHECK (5>EDAD>20),
  CURSO ENUM('1','2','3')
)engine=innodb;
```

- Modifica la definición de tabla anterior y cambia la definición de la columna EDAD para que no pueda ser nula.

```
ALTER TABLE EJEMPLO3 change edad EDAD tinyint NOT NULL CHECK (5>EDAD>20);
```

- Crea las siguientes tablas con las restricciones definidas.

FABRICANTES

Nombre de columna	Tipo
COD_FABRICANTE	Numérico entero (2)
NOMBRE	Alfanumérico(15)
PAIS	Alfanumérico(15)

ARTICULOS

Nombre de columna	Tipo
ARTICULO	Alfanumérico(20)
COD_FABRICANTE	Numérico entero(3)
PESO	Numérico real(3,2)
CATEGORIA	Alfanumérico(10)
PRECIO_VENTA	Real (6,2)
PRECIO_COSTO	Real (6,2)
EXISTENCIAS	Numérico entero(5)

- ✗ Restricciones para la tabla FABRICANTES:

- La clave primaria es COD_FABRICANTE
- La columna NOMBRE no debe duplicar valores para poder definir un índice a partir de ella.

- ✗ Restricciones para la tabla ARTICULOS:

- La clave primaria está formada por las columnas: ARTICULO, COD_FABRICANTE, PESO Y CATEGORIA
- COD_FABRICANTE es clave ajena que referencia a FABRICANTES.
- PRECIO_VENTA, PRECIO_COSTO han de ser igual a 100 por defecto.
- CATEGORÍA ha de ser 'Primera', 'Segunda' o 'Tercera'.

```
create table fabricantes (
  cod_fabricante tinyint primary key,
  nombre varchar(15) unique,
  pais varchar(15)
)engine=innodb;
```

```
create table articulos (  
  articulo varchar(20),  
  cod_fabricante tinyint,  
  peso decimal(3,2),  
  categoria enum('Primera','Segunda','Tercera'),  
  precio_venta decimal(6,2) default 100.0,  
  precio_costo decimal(6,2) default 100.0,  
  existencias smallint,  
  primary key (articulo,cod_fabricante,peso, categoria),  
  constraint fk_af foreign key (cod_fabricante)  
  references fabricantes(cod_fabricante) on delete no action  
  on update no action  
)engine=innodb;
```

- Modifica la tabla EJEMPLO3 para añadir dos columnas:

SEXO Carácter(1) con la restricción NOT NULL

IMPORTE Numérico real (5,2)

```
alter table ejemplo3 add sexo char(1) not null;
```

```
alter table ejemplo3 add importe decimal(5,2);
```

- Elimina las dos columnas añadidas a la tabla EJEMPLO3 en el apartado anterior (SEXO e IMPORTE).

```
alter table ejemplo3 drop column sexo;
```

```
alter table ejemplo3 drop column importe;
```

- Cambia el nombre a la tabla EJEMPLO3 para que pase a llamarse ALUMNO.

```
RENAME TABLE ejemplo3 TO alumno;
```

- Crea la tabla TIENDAS sin restricciones, cuya descripción es la siguiente:

NIF	Alfanumérico(10)
NOMBRE	Alfanumérico(20)
DIRECCION	Alfanumérico(20)
POBLACION	Alfanumérico(20)
PROVINCIA	Alfanumérico(20)
CODPOSTAL	Numérico entero(5)

- ✗ Después añada las siguientes restricciones:

- La clave primaria es NIF
- PROVINCIA no puede tomar valores nulos.
- Cambia la longitud de NOMBRE a 30 caracteres y no nulo.

```
create table tiendas (
nif varchar(9),
nombre varchar(20),
direccion varchar(20),
poblacion varchar(20),
provincia varchar(20),
codPostal smallint
)engine=innodb;
```

```
alter table tiendas add primary key (nif);
alter table tiendas modify provincia varchar(20) not null;
alter table tiendas modify nombre varchar(30) not null;
```

- Crea las siguientes tablas con las restricciones definidas.

PEDIDOS

Nombre de columna	Tipo
NIF	Numérico entero (10)
ARTICULO	Alfanumérico(20)
COD_FABRICANTE	Numérico entero(3)
PESO	Numérico real(3,2)
CATEGORIA	Alfanumérico(10)
FECHA_PEDIDO	Fecha
UNIDADES_PEDIDAS	Numérico entero(4)

VENTAS

Nombre de columna	Tipo
NIF	Alfanumérico(10)
ARTICULO	Alfanumérico(20)
COD_FABRICANTE	Numérico entero(3)
PESO	Numérico real(3,2)
CATEGORIA	Alfanumérico(10)
FECHA_VENTA	Fecha
UNIDADES_VENDIDAS	Numérico entero(4)

✗ Restricciones para la tabla PEDIDOS:

- La clave primaria está formada por las columnas: NIF, ARTICULO, COD_FABRICANTE, PESO, CATEGORIA y FECHA_PEDIDO
- COD_FABRICANTE es clave ajena que referencia a la tabla FABRICANTES.
- UNIDADES_PEDIDAS ha de ser 5 por defecto.
- CATEGORIA ha de ser 'Primera', 'Segunda' o 'Tercera'.
- Las columnas ARTICULO, COD_FABRICANTE, PESO y CATEGORIA son clave ajena y referencian a la tabla ARTICULOS. Realiza un borrado en cascada.
- NIF es clave ajena y referencia a la tabla TIENDAS

✗ Restricciones para la tabla VENTAS:

- La clave primaria está formada por las columnas: NIF, ARTICULO, COD_FABRICANTE, PESO, CATEGORIA y FECHA_VENTA.
- COD_FABRICANTE es clave ajena que referencia a la tabla FABRICANTES.
- UNIDADES_VENDIDAS ha de ser 5 por defecto.
- CATEGORIA ha de ser 'Primera', 'Segunda' o 'Tercera'.
- Las columnas ARTICULO, COD_FABRICANTE, PESO y CATEGORIA son clave ajena y referencian a la tabla ARTICULOS. Realiza un borrado en cascada.
- NIF es clave ajena y referencia a la tabla TIENDAS

```
create table pedidos (  
  nif varchar(9),  
  articulo varchar(20),  
  cod_fabricante tinyint,  
  peso decimal(3,2),  
  categoria enum('Primera','Segunda','Tercera'),  
  fecha_pedido date,  
  unidades_pedidas tinyint default 5,  
  primary key (nif,articulo,cod_fabricante,peso,categoria,fecha_pedido),  
  constraint fk_p_t foreign key (nif) references tiendas(nif),  
  foreign key (articulo,cod_fabricante,peso,categoria) references  
  articulos(articulo,cod_fabricante,peso,categoria) on delete cascade,  
  foreign key (cod_fabricante) references fabricantes(cod_fabricante)  
  )engine=innodb;
```

```
create table ventas(  
  nif varchar(9),  
  articulo varchar(20),  
  cod_fabricante integer(3),  
  peso decimal(3,2),  
  categoria enum('Primera','Segunda','Tercera'),  
  fecha_venta date,  
  unidades_vendidas tinyint default 5,  
  primary key (nif,articulo,cod_fabricante,peso,categoria,fecha_venta),  
  foreign key (nif) references tiendas(nif),  
  foreign key (articulo,cod_fabricante,peso,categoria) references  
  articulos(articulo,cod_fabricante,peso,categoria) on delete cascade,  
  foreign key (cod_fabricante) references fabricantes(cod_fabricante)  
  )engine=innodb;
```

- Visualiza las restricciones definidas para las tablas anteriores.

```
SELECT * FROM INFORMATION_SCHEMA.TABLE_CONSTRAINTS  
  
WHERE TABLE_SCHEMA LIKE 'PRUEBA';
```

- Modifica las columnas de las tablas PEDIDOS y VENTAS para que las UNIDADES_VENDIDAS y las UNIDADES_PEDIDAS puedan almacenar cantidades numéricas de 6 dígitos.

```
Alter table pedidos modify unidades_pedidas mediumint default 5;
```

```
Alter table ventas modify unidades_vendidas mediumint default 5;
```

- Añade a las tablas PEDIDOS y VENTAS una nueva columna para que almacenen el PVP del artículo.

```
Alter table pedidos add PVP decimal (6,3);
```

```
Alter table ventas add PVP decimal (6,3);
```