

CURSO PROGRAMACIÓN WEB FULL STACK

NIVEL 2

Módulo 4 - Bases de datos

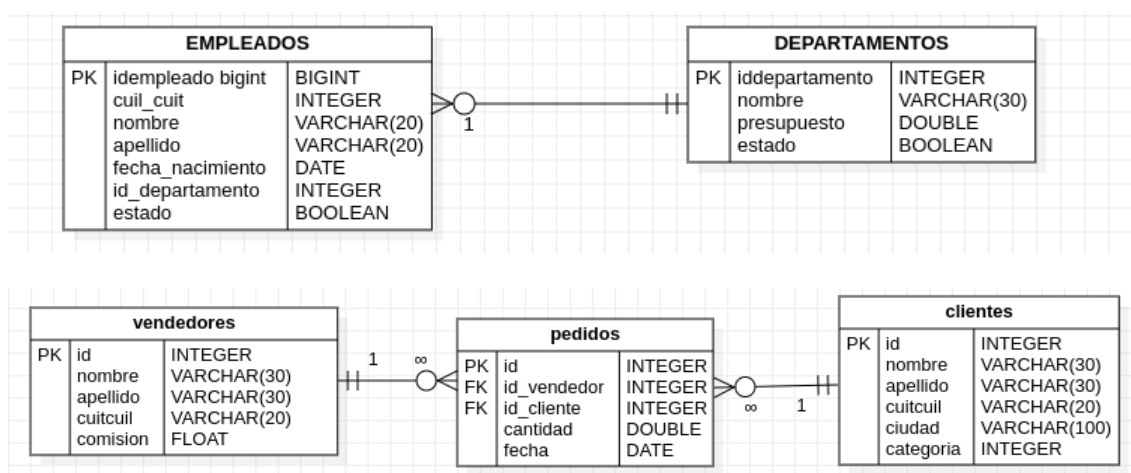
Ejercicio 1:

Partiendo del modelo entidad relación, ejecutar el script y resolver las consultas. Guardar en un archivo tp6-eje1.sql. Probar el script antes de subir a github. El script se debe ejecutar y no va fallar si los puntos están ordenados consecutivamente. En el caso de fallar localmente no lo commiten, es decir no lo suban al repositorio con errores.

Tener en cuenta que:

- Hay columnas que deben agregarle el tipo de dato.
- Hay relaciones que se deben construir en base a los enunciados para que se cumpla pk y fk.
- Leer TODO el documento y los links de la bibliografía antes de iniciar.

Modelo ER



Script para mysql

```
DROP DATABASE IF EXISTS elsistema;
```

```
CREATE DATABASE elsistema CHARACTER SET utf8mb4;
USE elsistema;

CREATE TABLE departamentos (
  iddepartamento INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(100) NOT NULL,
  presupuesto DOUBLE UNSIGNED NOT NULL,
  estado boolean NOT NULL
);

CREATE TABLE empleados (
  idempleado INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  cuil_cuit VARCHAR(15) NOT NULL UNIQUE,
  nombre VARCHAR(100) NOT NULL,
  apellido VARCHAR(100) NOT NULL,
  id_departamento INT UNSIGNED,
  estado BOOLEAN,
  FOREIGN KEY (id_departamento) REFERENCES departamentos(iddepartamento)
);

CREATE TABLE clientes (
  id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(30) NOT NULL,
  apellido VARCHAR(30) NOT NULL,
  cuilcuit VARCHAR(20),
  ciudad VARCHAR(100),
  categoría INT UNSIGNED
);

CREATE TABLE vendedores (
  id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  nombre VARCHAR(30) NOT NULL,
  apellido VARCHAR(30) NOT NULL,
  cuilcuit VARCHAR(20),
  comisión FLOAT
);

CREATE TABLE pedidos (
  id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  cantidad DOUBLE NOT NULL,
  fecha DATE,
  id_cliente INT UNSIGNED NOT NULL,
  id_vendedor INT UNSIGNED NOT NULL,
```

```
FOREIGN KEY (id_cliente) REFERENCES clientes(id),  
FOREIGN KEY (id_vendedor) REFERENCES vendedores(id)  
);
```

Resolver las siguientes consultas utilizando la sintaxis SQL

1. Agregar las entidades paises (id, nombre), provincias (id, nombre, idpais) y localidades (id,nombre, cp,id_provincia). Tener en cuenta que:
 - En un país hay muchas provincias.
 - En una provincia hay muchas localidades.
 2. Modificar la tabla empleados usando ALTER TABLE y agregar los campos:
 - direccion (varchar)
 - id_localidad (pk localidad) –Esta es un relación con la tabla localidades
 - email
 - telefono
 - fecha_ingreso
 - tms (timestamp)
 3. Modificar la tabla de departamentos usando ALTER TABLE y agregar los campos:
 - gasto (double)
 - tms (timestamp)
 4. Insertar 5 registros en cada tabla de: paises, provincias, localidades, departamentos, empleados.
 5. Modificar el nombre de la tabla “pedidos” por “movimientos”. RENAME TABLE
 6. Agregar las entidades:
 - Productos (id, nombre, descripcion, id_marca fk, stock, precio, estado, tms)
 - Marcas (id, nombre, descripción, imagen, id_proveedor, estado, tms)
 - Proveedores (id, razon_social, nombre, apellido, naturaleza (fisica o juridica), cuit,id_localidad fk, estado,tms)
 - Cajas (id,horainicio(datetime),horacierre(datetime), estado, tms)
- Notas: Muchos productos tienen una sola marca, o una marca tiene uno o muchos productos.
Un proveedor está en una localidad.
7. Insertar 5 registros en cada tabla del punto 6. Tener en cuenta que el script debe ejecutarse secuencialmente y no fallar.
 8. Listar el nombre, presupuesto, gastos y diferencia(presupuesto-gasto) de todos los departamentos con estado activo o 1.
 9. Listar todas las localidades agrupadas por pais. En la vista se debería ver el nombre del pais y el nombre de la localidad
 10. Modificar (UPADTE):
 - el telefono de un empleado cuando el id es igual a uno que hayan declarado.
 - el fecha_ingreso y la localidad de otro empleado.
 11. Insertar 5 vendedores.

12. Modificar la tabla movimientos y agregar los campos: id_producto fk, estado, tms(timestamp), tipo_movimiento (ingreso, egreso, pedido)
13. Insertar 5 movimientos distintos.
14. Borrar lógicamente (UPDATE de la columna estado):
 - 2 movimientos que fueron cargados mal
 - un país que tenga al menos 3 localidades
15. Modificar el campo stock de la tabla productos teniendo en cuenta la cantidad de la tabla de movimientos. Sumar el stock si es un ingreso, restar si es un egreso. Esto hacerlo de manera manual en base los 5 movimientos insertados en el punto 13. Es decir deben haber 5 updates de la tabla producto.
16. Cear la tabla parametros (id, tms,cosas(json), id_usuario)
17. Insertar en la tabla parametros teniendo en cuenta la columna cosas:
 - {"idDeLaCosa": 101, "permisos": "PUT, GET"}
 - {"vistasPermitidas": "menuPrincipal,menuSecundario,ventas,estadisticaVentas,listaCliente", "grupo": "ventas"}
 - {"zonaHoraria": "America/Argentina/BuenosAires"}
 - {"fechaInicioActividades": "01/01/2019", "mesAperturaCaja": "Enero", "mesCierreCaja": "Diciembre"}
 - {"balancesAniosAnteriores": {"2019": {"ingreso": "7374901.93", "egreso": "3732538,75"}, "2020": {"ingreso": "27442665,12", "egreso": "8522331,82"}, "2021": {"ingreso": "31634912,57", "egreso": "9757142,66"}}}

Nota: Rellenar a criterio los campos id, tms,id_usuario

Ejercicio 2:

Subir al repositorio público apellido_nombre las respuestas tp6-eje1.sql.

Subir al repositorio publicó una captura de pantalla en jpg o png del esquema completo. El archivo debe llamarse diagrama_bd-tp6-eje1.jpg|.png

Bibliografía

https://www.w3schools.com/mysql/mysql_update.asp

https://www.w3schools.com/mysql/mysql_insert.asp

https://www.w3schools.com/mysql/mysql_create_table.asp

https://www.w3schools.com/mysql/mysql_alter.asp

https://www.w3schools.com/mysql/mysql_dates.asp

https://www.w3schools.com/mysql/mysql_datatypes.asp

https://www.w3schools.com/mysql/mysql_orderby.asp

<https://stackoverflow.com/questions/40930896/how-to-create-and-insert-a-json-object-using-mysql-queries>