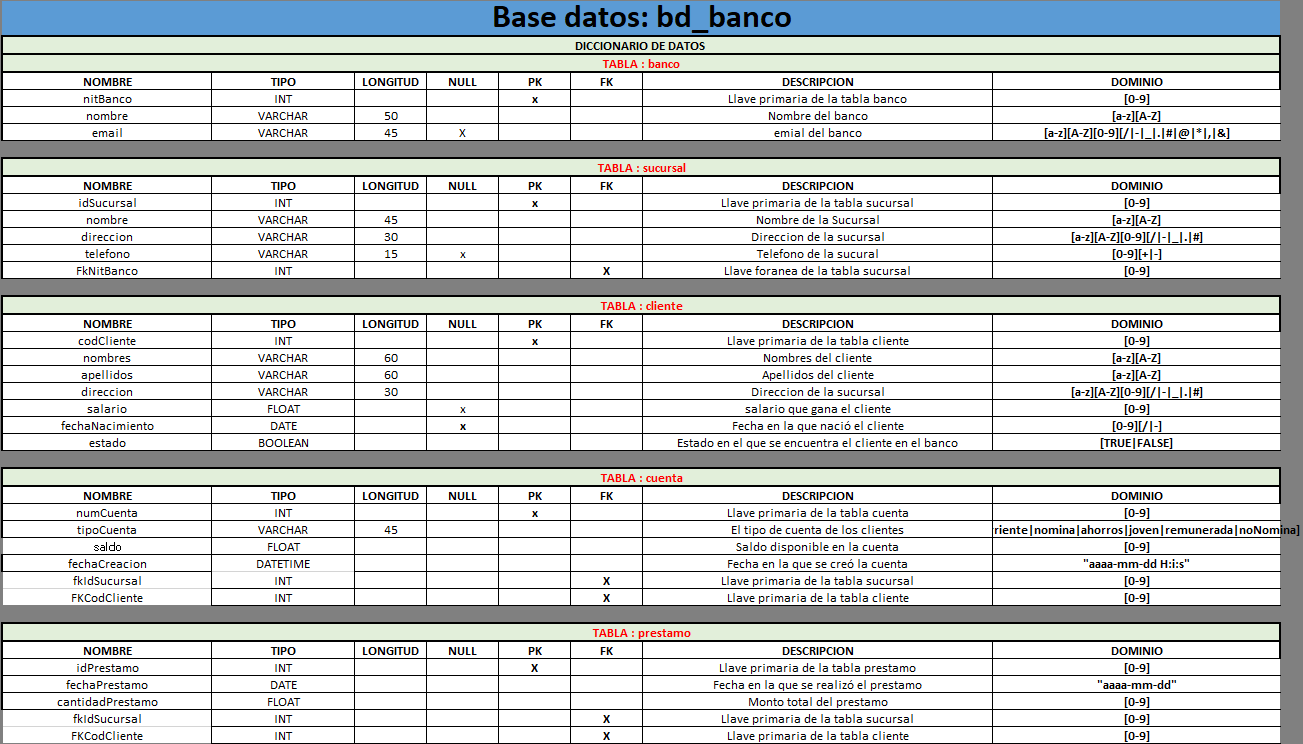
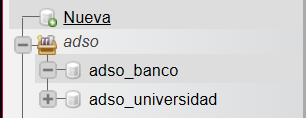
# Realizar el diccionario de datos con base al diagrama de tablas anterior.



# Con comandos SQL, crear una base de datos en MySQL llamada: **adso\_banco**.

CREATE DATABASE adso\_banco;



# Con base al diccionario de datos y utilizando comandos SQL, crear las tablas de la base de datos

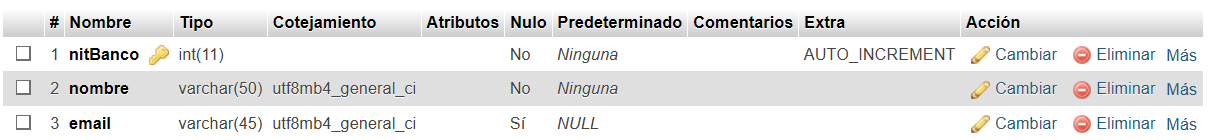
CREATE TABLE banco (

nitBanco INT PRIMARY KEY AUTO\_INCREMENT,

nombre VARCHAR(50) NOT NULL,

email VARCHAR(45)

);



CREATE TABLE sucursal (

idSucursal INT PRIMARY KEY AUTO\_INCREMENT,

nombre VARCHAR(45) NOT NULL,

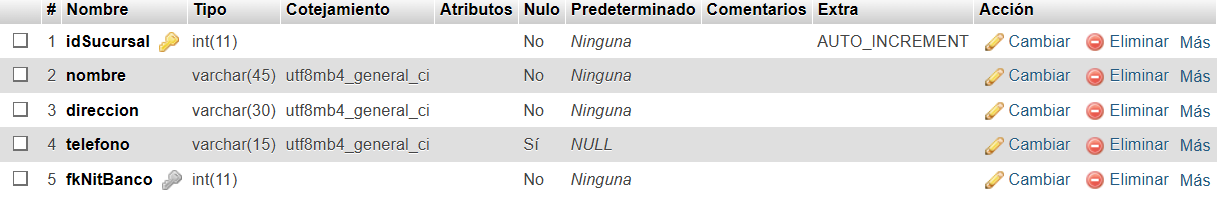
direccion VARCHAR(30) NOT NULL,

telefono VARCHAR(15),

fkNitBanco INT NOT NULL,

FOREIGN KEY (fkNitBanco) REFERENCES banco(nitBanco)

);



CREATE TABLE cliente (

codCliente INT PRIMARY KEY AUTO\_INCREMENT,

nombres VARCHAR(60) NOT NULL,

apellidos VARCHAR(60) NOT NULL,

direccion VARCHAR(30) NOT NULL,

salario FLOAT,

fechaNacimiento DATE,

estado BOOLEAN NOT NULL

);



CREATE TABLE cuenta (

numCuenta INT PRIMARY KEY AUTO\_INCREMENT,

tipoCuenta VARCHAR(45) NOT NULL,

saldo FLOAT NOT NULL,

fechaCreacion DATETIME NOT NULL,

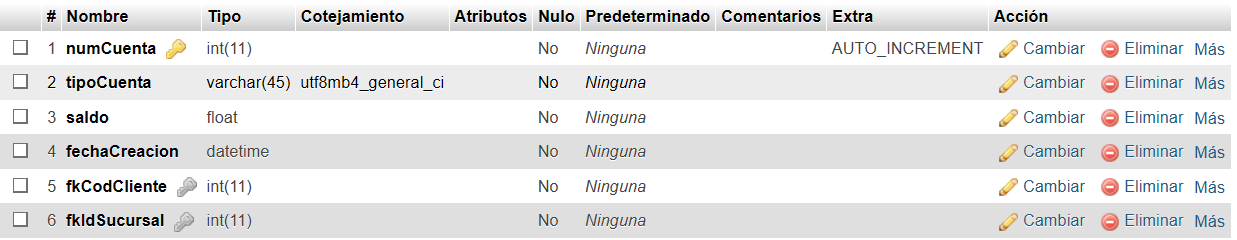
fkCodCliente INT NOT NULL,

fkIdSucursal INT NOT NULL,

FOREIGN KEY (fkCodCliente) REFERENCES cliente(codCliente),

FOREIGN KEY (fkIdSucursal) REFERENCES sucursal(idSucursal)

);



CREATE TABLE prestamo (

idPrestamo INT PRIMARY KEY AUTO\_INCREMENT,

fechaPrestamo DATE NOT NULL,

cantidadPrestamo FLOAT NOT NULL,

fkCodCliente INT NOT NULL,

fkIdSucursal INT NOT NULL,

FOREIGN KEY (fkCodCliente) REFERENCES cliente(codCliente),

FOREIGN KEY (fkIdSucursal) REFERENCES sucursal(idSucursal)

);



# Insertar los siguientes registros:

## 4 Clientes.

INSERT INTO cliente (nombres,apellidos,direccion,salario,fechaNacimiento,estado)

VALUES ('Isabela','Rodriguez','Viveros #12-34',450000,'2004-11-24',1);

INSERT INTO cliente (nombres,apellidos,direccion,salario,fechaNacimiento,estado)

VALUES ('Daniel','Duque','San Sebastian',1450000,'2004-04-03',1);

INSERT INTO cliente (nombres,apellidos,direccion,salario,fechaNacimiento,estado)

VALUES ('Viviana','Cardenas','Kra 48D-54',1300000,'1985-11-12',1);

INSERT INTO cliente (nombres,apellidos,direccion,salario,fechaNacimiento,estado)

VALUES ('Moralba','Marin','Apto 114113 BQ 123',2550000,'1990-3-20',1);



## 5 Cuentas (asociar 2 cuentas a un solo cliente y el resto de cuentas a clientes diferentes.

INSERT INTO cuenta (tipoCuenta,saldo,fechaCreacion,fkCodCLiente,fkIdSucursal)

VALUES ('ahorros',2000000,'2014-05-14',1,1);

INSERT INTO cuenta (tipoCuenta,saldo,fechaCreacion,fkCodCLiente,fkIdSucursal)

VALUES ('corriente',500000,'2014-05-14',1,2);

INSERT INTO cuenta (tipoCuenta,saldo,fechaCreacion,fkCodCLiente,fkIdSucursal)

VALUES ('ahorros',2500000,'2010-11-30',2,3);

INSERT INTO cuenta (tipoCuenta,saldo,fechaCreacion,fkCodCLiente,fkIdSucursal)

VALUES ('nomina',1500000,'2020-10-10',3,4);

INSERT INTO cuenta (tipoCuenta,saldo,fechaCreacion,fkCodCLiente,fkIdSucursal)

VALUES ('estudiantil',10000000,'2007-01-15',4,5);



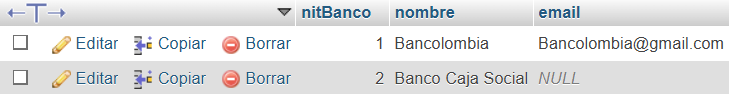
## 2 Bancos

INSERT INTO banco (nombre,email)

VALUES ('Bancolombia','Bancolombia@gmail.com');

# INSERT INTO banco (nombre)

# VALUES ('Banco Caja Social');



## 5 sucursales (asociar 3 sucursales a un banco y 2 al otro banco)

INSERT INTO sucursal (nombre,direccion,telefono,fkNitBanco)

VALUES ('Bancolombia de la 22','Cra-22','6542495205',1);

INSERT INTO sucursal (nombre,direccion,telefono,fkNitBanco)

VALUES ('Bancolombia de la santander','Av-Santander','6523148975',1);

INSERT INTO sucursal (nombre,direccion,telefono,fkNitBanco)

VALUES ('Bancolombia Plaza Bolivar','Plaza de Bolivar','6321478942',1);

INSERT INTO sucursal (nombre,direccion,telefono,fkNitBanco)

VALUES ('Caja Social Cra 23','Cra-23','3562465872',2);

INSERT INTO sucursal (nombre,direccion,telefono,fkNitBanco)

VALUES ('Caja Social Santander','Av-Santander','3564215875',2);



## 4 prestamos

INSERT INTO prestamo (fechaPrestamo,cantidadPrestamo,fkCodCliente,fkIdSucursal)

VALUES ('2014-06-15',1500000,1,1);

INSERT INTO prestamo (fechaPrestamo,cantidadPrestamo,fkCodCliente,fkIdSucursal)

VALUES ('2014-07-20',2500000,1,1);

INSERT INTO prestamo (fechaPrestamo,cantidadPrestamo,fkCodCliente,fkIdSucursal)

VALUES ('2013-04-10',1000000,2,3);

INSERT INTO prestamo (fechaPrestamo,cantidadPrestamo,fkCodCliente,fkIdSucursal)

VALUES ('2010-05-04',2300000,4,5);



# Taller 2 – Segunda Parte

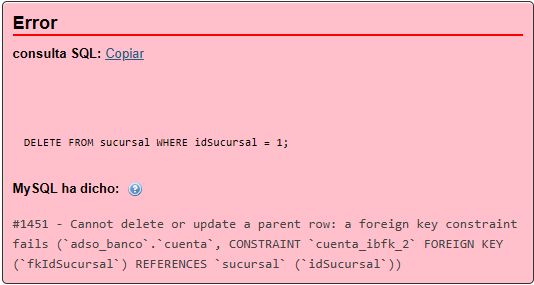
## Actualice el “nombre”, “fechaNacimiento” y “salario” de un Cliente (en una sola consulta)

UPDATE cliente SET nombres = 'Santiago', salario = 600000, fechaNacimiento ='2000-02-03' WHERE codCliente = 1;



## Intente eliminar una Sucursal específica por medio del atributo “idSucursal” ¿Es válido realizar esta acción? Explique

DELETE FROM sucursal WHERE idSucursal = 1;



POR QUÉ : Ocurre cuando intentas eliminar o actualizar una fila que está referenciada por una clave foránea en otra tabla.

## 3. Ordenar los clientes por nombre y mostrar solamente los dos últimos (utilizando LIMIT)

SELECT \* FROM cliente

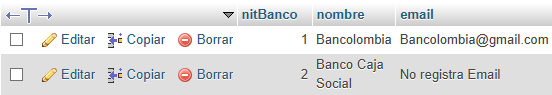
ORDER BY nombres DESC

LIMIT 2;



## 4. Actualice el campo “email” de la tabla Banco de todos los bancos que no tengan correo electrónico, es decir, que tengan valor NULL en dicho campo. El campo se debe actualizar con el valor "No Registra email".

UPDATE banco SET email = 'No registra Email' WHERE nitBanco = 2;

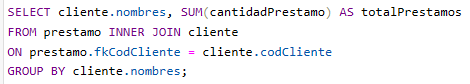


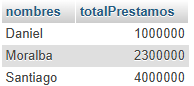
## 5. Actualice el campo "fechaPrestamo” de un Préstamo. Esta fecha actualícela con la fecha actual del sistema (buscar la forma de obtener la fecha actual con MySQL).

UPDATE prestamo SET fechaPrestamo = CURDATE() WHERE IdPrestamo = 4;



## 6. Mostrar el valor total de préstamos por cada cliente.





## 7. Mostrar los clientes que tengan estado activo, un salario mayor a 1'000.000 y fecha de nacimiento mayor 01/07/1990

SELECT \* FROM cliente

WHERE estado = 1 AND salario > 1000000 AND fechaNacimiento > '1990-07-01';



## 8. Calcular la cantidad promedio de todos los préstamos de un cliente específico

SELECT AVG(cantidadPrestamo) FROM prestamo WHERE fkCodCliente = 1;



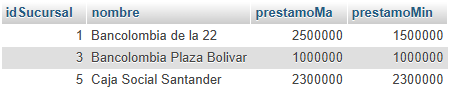
## 9. Obtener el préstamo con mayor valor y menor valor por sucursal.

SELECT idSucursal, nombre, MAX(cantidadPrestamo) AS prestamoMa, MIN(cantidadPrestamo) AS prestamoMin

FROM sucursal INNER JOIN prestamo

ON sucursal.idSucursal = prestamo.fkIdSucursal

GROUP BY nombre;



## 10. Mostrar la cantidad de préstamos realizados por un cliente específico en el último año. Para esta consulta se debe mostrar Nombre del cliente y la cantidad de préstamos

SELECT c.nombres, COUNT(p.idPrestamo) AS cantidadPrestamos

FROM cliente c

JOIN prestamo p ON c.codCliente = p.fkCodCliente

WHERE c.codCliente = 1

AND p.fechaPrestamo >= 2024-01-01

GROUP BY c.nombres;



## 11. Obtener en qué fecha se realizó el primer préstamo para la sucursal 'x'

SELECT MIN(fechaPrestamo) AS primerPrestamo FROM prestamo WHERE fkIdSucursal = 1;



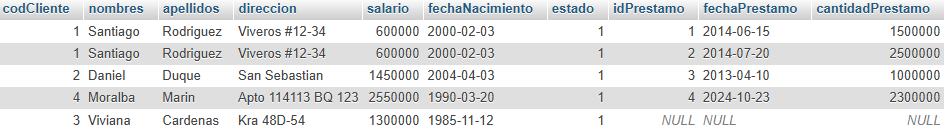
## 12. Mostrar el listado completo de clientes con la información de los préstamos realizados. Se debe tener en cuenta que haya realizado o no préstamos, debe mostrarse la información básica del cliente en el listado.

SELECT c.codCliente, c.nombres, c.apellidos, c.direccion, c.salario, c.fechaNacimiento, c.estado,

p.idPrestamo, p.fechaPrestamo, p.cantidadPrestamo

FROM cliente c

LEFT JOIN prestamo p ON c.codCliente = p.fkCodCliente;



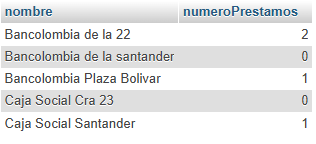
### 13. Listar cuántos préstamos se han realizado por cada sucursal. Se debe mostrar el nombre de la sucursal y el número de préstamos.

SELECT s.nombre, COUNT(p.idPrestamo) AS numeroPrestamos

FROM sucursal s

LEFT JOIN prestamo p ON s.idSucursal = p.fkIdSucursal

GROUP BY s.nombre;



## 14. Mostrar el valor total en préstamos para el banco 'x' desde el 1/01/2024 hasta el 30/01/2024.

## 