

**ESCUELA POLITÉCNICA NACIONAL**

**ESCUELA DE FORMACIÓN DE TECNÓLOGOS**

**TECNOLOGÍA EN ANÁLISIS DE SISTEMAS INFORMATICOS**

**PERÍODO ACADÉMICO:**

**ASIGNATURA:** BASES DE DATOS

**PROFESOR:** MSc. Marina Vintimilla

**TIPO DE INSTRUMENTO: EJERCITARIO REFUERZO**

**Ejercicio propuesto:**

**Suponga las siguientes tablas de una base de datos BANCO:**

*SUCURSAL (sucursal\_nombre, sucursal\_ciudad, valor\_activos)*

|  |  |  |
| --- | --- | --- |
| **sucursal\_nombre** | **sucursal\_ciudad** | **valor\_activos** |
| Brighton | Brooklyn | 7100000 |
| Downtown | Brooklyn | 9000000 |
| Mianus | Horseneck | 400000 |
| North Town | Rye | 3700000 |
| Perryridge | Horseneck | 1700000 |
| Pownal | Bennington | 300000 |
| Redwood | Palo Alto | 2100000 |
| Round Hill | Horseneck | 8000000 |

*CLIENTE (cliente\_nombre, cliente\_calle, cliente\_ciudad)*

|  |  |  |
| --- | --- | --- |
| **cliente\_nombre** | **cliente\_calle** | **ciudad** |
| Adams | Spring | Pittsfield |
| Brooks | Senator | Brooklyn |
| Curry | North | Rye |
| Glenn | Sand Hill | Woodside |
| Green | Walnut | Stanford |
| Hayes | Main | Harrison |
| Johnson | Alma | Palo Alto |
| Jones | Main | Harrison |
| Lindsay | Park | Pittsfield |
| Smith | North | Rye |
| Turner | Putnam | Stanford |
| Williams | Nassau | Rye |

CUENTA  *(cuenta\_numero, sucursal\_nombre, balance)*

|  |  |  |
| --- | --- | --- |
| cuenta\_numero | sucursal\_nombre | balance |
| A-101 | Downtown | 500 |
| A-102 | Perryridge | 500 |
| A-201 | Brighton | 900 |
| A-215 | Downtown | 700 |
| A-217 | Brighton | 750 |
| A-222 | Perryridge | 700 |
| A-305 | Brighton | 350 |

PRESTAMO  *(prestamo\_numero, sucursal\_nombre, monto)*

|  |  |  |
| --- | --- | --- |
| prestamo\_numero | sucusal\_nombre | monto |
| L-11 | Brighton | 900 |
| L-14 | Downtown | 1500 |
| L-15 | Perryridge | 1500 |
| L-16 | Perryridge | 1300 |
| L-17 | Downtown | 1000 |
| L-23 | Perryridge | 2000 |
| L-93 | Downtown | 500 |

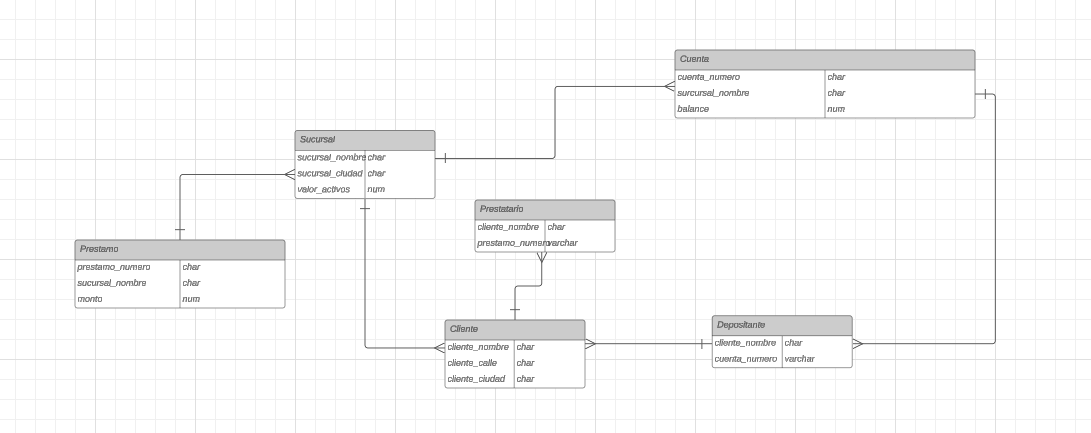
DEPOSITANTE  *(cliente\_nombre, cuenta\_numero)*

|  |  |
| --- | --- |
| cliente\_nombre | cuenta\_numero |
| Hayes | A-102 |
| Johnson | A-101 |
| Johnson | A-201 |
| Jones | A-217 |
| Lindsay | A-222 |
| Smith | A-215 |
| Turner | A-305 |

PRESTATARIO  *(cliente\_nombre, prestamo\_numero)*

|  |  |
| --- | --- |
| cliente\_nombre | prestamo\_numero |
| Adams | L-16 |
| Curry | L-93 |
| Hayes | L-15 |
| Jackson | L-14 |
| Jones | L-17 |
| Smith | L-11 |
| Smith | L-23 |
| Williams | L-17 |

1. Con la información proporcionada genere el **modelo lógico**  (inserte el gráfico)



1. Implemente el modelo e incluya los datos proporcionados para cada tabla.

CREATE DATABASE Banco;

USE Banco;

CREATE TABLE SUCURSAL (

sucursal\_nombre VARCHAR(50),

sucursal\_ciudad VARCHAR(50),

valor\_activos INT,

PRIMARY KEY (sucursal\_nombre)

);

CREATE TABLE CLIENTE (

cliente\_nombre VARCHAR(50),

cliente\_calle VARCHAR(50),

ciudad VARCHAR(50),

PRIMARY KEY (cliente\_nombre)

);

CREATE TABLE CUENTA (

cuenta\_numero VARCHAR(10),

sucursal\_nombre VARCHAR(50),

balance INT,

PRIMARY KEY (cuenta\_numero),

FOREIGN KEY (sucursal\_nombre) REFERENCES SUCURSAL(sucursal\_nombre)

);

CREATE TABLE PRESTATARIO (

cliente\_nombre VARCHAR(50),

prestamo\_numero VARCHAR(10),

PRIMARY KEY (cliente\_nombre, prestamo\_numero),

FOREIGN KEY (cliente\_nombre) REFERENCES CLIENTE(cliente\_nombre),

FOREIGN KEY (prestamo\_numero) REFERENCES PRESTAMO(prestamo\_numero)

);

CREATE TABLE PRESTAMO (

prestamo\_numero VARCHAR(10),

sucursal\_nombre VARCHAR(50),

monto INT,

PRIMARY KEY (prestamo\_numero),

FOREIGN KEY (sucursal\_nombre) REFERENCES SUCURSAL(sucursal\_nombre)

);

CREATE TABLE DEPOSITANTE (

cliente\_nombre VARCHAR(50),

cuenta\_numero VARCHAR(10),

PRIMARY KEY (cliente\_nombre, cuenta\_numero),

FOREIGN KEY (cliente\_nombre) REFERENCES CLIENTE(cliente\_nombre),

FOREIGN KEY (cuenta\_numero) REFERENCES CUENTA(cuenta\_numero)

);

INSERT INTO SUCURSAL (sucursal\_nombre, sucursal\_ciudad, valor\_activos) VALUES

('Brighton', 'Brooklyn', 7100000),

('Downtown', 'Brooklyn', 9000000),

('Mianus', 'Horseneck', 400000),

('North Town', 'Rye', 3700000),

('Perryridge', 'Horseneck', 1700000),

('Pownal', 'Bennington', 300000),

('Redwood', 'Palo Alto', 2100000),

('Round Hill', 'Horseneck', 8000000);

INSERT INTO CLIENTE (cliente\_nombre, cliente\_calle, ciudad) VALUES

('Adams', 'Spring', 'Pittsfield'),

('Brooks', 'Senator', 'Brooklyn'),

('Curry', 'North', 'Rye'),

('Glenn', 'Sand Hill', 'Woodside'),

('Green', 'Walnut', 'Stanford'),

('Hayes', 'Main', 'Harrison'),

('Johnson', 'Alma', 'Palo Alto'),

('Jones', 'Main', 'Harrison'),

('Lindsay', 'Park', 'Pittsfield'),

('Smith', 'North', 'Rye'),

('Turner', 'Putnam', 'Stanford'),

('Williams', 'Nassau', 'Rye');

INSERT INTO CUENTA (cuenta\_numero, sucursal\_nombre, balance) VALUES

('A-101', 'Downtown', 500),

('A-102', 'Perryridge', 500),

('A-201', 'Brighton', 900),

('A-215', 'Downtown', 700),

('A-217', 'Brighton', 750),

('A-222', 'Perryridge', 700),

('A-305', 'Brighton', 350);

INSERT INTO PRESTAMO (prestamo\_numero, sucursal\_nombre, monto) VALUES

('L-11', 'Brighton', 900),

('L-14', 'Downtown', 1500),

('L-15', 'Perryridge', 1500),

('L-16', 'Perryridge', 1300),

('L-17', 'Downtown', 1000),

('L-23', 'Perryridge', 2000),

('L-93', 'Downtown', 500);

INSERT INTO DEPOSITANTE (cliente\_nombre, cuenta\_numero) VALUES

('Hayes', 'A-102'),

('Johnson', 'A-101'),

('Johnson', 'A-201'),

('Jones', 'A-217'),

('Lindsay', 'A-222'),

('Smith', 'A-215'),

('Turner', 'A-305');

INSERT INTO PRESTATARIO (cliente\_nombre, prestamo\_numero) VALUES

('Adams', 'L-16'),

('Curry', 'L-93'),

('Hayes', 'L-15'),

('Jackson', 'L-14'),

('Jones', 'L-17'),

('Smith', 'L-11'),

('Smith', 'L-23'),

('Williams', 'L-17');

1. Resuelva las siguientes consultas:
2. Encuentre todos los préstamos cuyo monto sea mayor a $1200

SELECT \* FROM PRESTAMO WHERE monto > 1200;

1. Encuentre el número de préstamo de cada préstamo con un monto mayor a $1200

SELECT prestamo\_numero FROM PRESTAMO WHERE monto > 1200;

1. Encuentre los nombres de las sucursales con valor en activos mayor que $100000.

SELECT sucursal\_nombre FROM SUCURSAL WHERE valor\_activos > 100000;

1. Encuentre los nombres de todos los clientes que tienen depósitos o préstamos o ambos.

SELECT DISTINCT cliente\_nombre FROM (

SELECT cliente\_nombre FROM DEPOSITANTE

UNION

SELECT cliente\_nombre FROM PRESTATARIO

) AS CombinedNames;

1. Encuentre los nombres de todos los clientes que tienen un préstamo en la sucursal Perryridge.

SELECT DISTINCT CLIENTE.cliente\_nombre

FROM CLIENTE

JOIN PRESTATARIO ON CLIENTE.cliente\_nombre = PRESTATARIO.cliente\_nombre

WHERE PRESTATARIO.sucursal\_nombre = 'Perryridge';

1. Encuentre los nombres de los clientes que tienen préstamos con monto mayores a $1000

SELECT DISTINCT CLIENTE.cliente\_nombre

FROM CLIENTE

JOIN PRESTATARIO ON CLIENTE.cliente\_nombre = PRESTATARIO.cliente\_nombre

WHERE PRESTATARIO.monto > 1000;

1. Presente la lista de ciudades de donde vienen los clientes.

SELECT DISTINCT ciudad FROM CLIENTE;

1. Encuentre los nombres de todos los clientes que tienen un préstamo en la sucursal Perryridge, pero no tienen una cuenta en ninguna sucursal del banco.

SELECT DISTINCT CLIENTE.cliente\_nombre

FROM CLIENTE

JOIN PRESTATARIO ON CLIENTE.cliente\_nombre = PRESTATARIO.cliente\_nombre

WHERE PRESTATARIO.sucursal\_nombre = 'Perryridge'

AND CLIENTE.cliente\_nombre NOT IN (SELECT cliente\_nombre FROM DEPOSITANTE);

1. Encuentre a los depositantes con cuentas en la sucursal Brighton.

SELECT cliente\_nombre, cuenta\_numero

FROM DEPOSITANTE

WHERE cuenta\_numero IN (SELECT cuenta\_numero FROM CUENTA WHERE sucursal\_nombre = 'Brighton');

1. Obtenga la información de cada cliente: sus datos personales, sus cuentas, balance de sus cuentas, prestamos que registra

SELECT

C.cliente\_nombre,

C.cliente\_calle,

C.ciudad,

D.cuenta\_numero,

COALESCE(CU.balance, 0) AS balance\_cuenta,

P.prestamo\_numero,

COALESCE(PR.monto, 0) AS monto\_prestamo

FROM CLIENTE C

LEFT JOIN DEPOSITANTE D ON C.cliente\_nombre = D.cliente\_nombre

LEFT JOIN CUENTA CU ON D.cuenta\_numero = CU.cuenta\_numero

LEFT JOIN PRESTATARIO P ON C.cliente\_nombre = P.cliente\_nombre

LEFT JOIN PRESTAMO PR ON P.prestamo\_numero = PR.prestamo\_numero;

1. Encuentre el valor más alto de balance de cuenta.

SELECT MAX(balance) AS max\_balance FROM CUENTA;

1. Cuántos clientes tiene cada sucursal

SELECT sucursal\_nombre, COUNT(\*) AS cantidad\_clientes

FROM DEPOSITANTE

GROUP BY sucursal\_nombre;

1. Cuál es el monto de préstamos que se han realizado en cada sucursal

SELECT sucursal\_nombre, SUM(monto) AS monto\_total\_prestamos

FROM PRESTAMO

GROUP BY sucursal\_nombre;

1. Cuál es el valor promedio de los préstamos que se han realizado

SELECT AVG(monto) AS promedio\_prestamos

FROM PRESTAMO;

1. Cuántos depósitos se tiene por cada cuenta

SELECT cuenta\_numero, COUNT(\*) AS cantidad\_depositos

FROM DEPOSITANTE

GROUP BY cuenta\_numero;

1. Qué sucursales no registran ningún préstamo

SELECT S.sucursal\_nombre

FROM SUCURSAL S

LEFT JOIN PRESTAMO P ON S.sucursal\_nombre = P.sucursal\_nombre

WHERE P.sucursal\_nombre IS NULL;

1. Cuáles son los préstamos que tienen la misma sucursal que el número de préstamo
   1. L-15

SELECT prestamo\_numero

FROM PRESTAMO

WHERE sucursal\_nombre = (SELECT sucursal\_nombre FROM PRESTAMO WHERE prestamo\_numero = 'L-15');

1. Cuáles son los clientes que tienen la misma sucursal que el cliente Adams

SELECT cliente\_nombre

FROM CLIENTE

WHERE ciudad = (SELECT ciudad FROM CLIENTE WHERE cliente\_nombre = 'Adams');

1. Qué clientes tienen el mismo monto de préstamo que el cliente Jackson

SELECT cliente\_nombre

FROM PRESTATARIO

WHERE monto = (SELECT monto FROM PRESTATARIO WHERE cliente\_nombre = 'Jackson');

1. Cuáles son los clientes que tienen más de un préstamo

SELECT cliente\_nombre

FROM PRESTATARIO

GROUP BY cliente\_nombre

HAVING COUNT(\*) > 1;

1. Cuáles son los clientes que tienen más de un préstamo y cuáles son sus montos

SELECT cliente\_nombre, COUNT(\*) AS cantidad\_prestamos, SUM(monto) AS total\_montos

FROM PRESTATARIO

GROUP BY cliente\_nombre

HAVING COUNT(\*) > 1;

1. Qué clientes viven en la misma ciudad del cliente Smith

SELECT cliente\_nombre

FROM CLIENTE

WHERE ciudad = (SELECT ciudad FROM CLIENTE WHERE cliente\_nombre = 'Smith');

1. Seleccione las sucursales cuyo total de préstamos supere los 3000

SELECT sucursal\_nombre

FROM PRESTAMO

GROUP BY sucursal\_nombre

HAVING SUM(monto) > 3000;

1. Seleccione los clientes cuyo promedio de préstamos sea mayor a 1500

SELECT cliente\_nombre

FROM (

SELECT cliente\_nombre, AVG(monto) AS promedio\_prestamos

FROM PRESTATARIO

GROUP BY cliente\_nombre

) AS promedios

WHERE promedio\_prestamos > 1500;

1. Seleccione el cliente cuyo préstamo sea el mayor de todos y el cliente con el préstamo menor de todos (debe resolverlo en solo select)

SELECT

MAX(cliente\_max\_monto.cliente\_nombre) AS cliente\_con\_max\_monto,

MIN(cliente\_min\_monto.cliente\_nombre) AS cliente\_con\_min\_monto

FROM (

SELECT cliente\_nombre, ROW\_NUMBER() OVER (ORDER BY monto DESC) AS rn\_desc,

ROW\_NUMBER() OVER (ORDER BY monto ASC) AS rn\_asc

FROM PRESTATARIO

) AS cliente\_max\_monto

WHERE cliente\_max\_monto.rn\_desc = 1

OR cliente\_max\_monto.rn\_asc = 1;