

DATA SCIENCE



Asignatura: Bootcamp SQL SERVER

Tema del proyecto: “Gestión del Proceso de Crédito Vehicular en una Entidad Financiera”

Grupo: N°2

Integrantes:

- Durand Galindo Josué Yamil
- Pachas Luis Felipe Lachira
- Pasache Mendoza, Eduardho Jair

Docente:

- Ing. Kevin Rivera Vergaray

Lima, Perú

2025

INDICE

1) Iniciación: Definición de los objetivos y el alcance inicial de la base de datos. .	3
❖ Título del Problema:	3
❖ Descripción del Problema:	3
❖ Requerimientos Funcionales:	3
❖ Entidades Identificadas:	4
❖ Tablas y Campos Relevantes:	4
❖ Desafíos:	6
❖ Beneficios Esperados:	6
2) Análisis: Diseño Conceptual de la base de datos	7
3) Diseño: Definición del Modelo Lógico de la Base de Datos.	7
4) Construcción: Desarrollo de las sentencias SQL que permiten la construcción de las tablas y demás estructuras de la Base de Datos en SQL SERVER.	8
5) Producto: Diseño Físico, Script de la Base de datos con inserción de datos (Mínimo 5 registros por tablas independientes o no transaccionales y mínimo 15 registros por cada tabla transaccional).....	12
❖ Diseño físico	12
6) Inserciones.....	13
❖ Inserción de datos a tablas independientes	13
❖ Inserción de datos a tablas transaccionales	20
7) Anexos	37
❖ Ejercicios.....	37

1)Iniciación: Definición de los objetivos y el alcance inicial de la base de datos.

❖ Título del Problema:

Gestión del Proceso de Crédito Vehicular en una Entidad Financiera

❖ Descripción del Problema:

Una entidad financiera necesita gestionar eficientemente los créditos vehiculares solicitados por clientes a través de concesionarios afiliados. El proceso inicia desde que el cliente contacta al dealer, pasa por distintos niveles de validación y control dentro de la entidad (asesor, gestor, analista) hasta llegar al desembolso del préstamo y la constitución de una prenda vehicular como garantía.

El sistema debe permitir un seguimiento detallado y estructurado de todas las etapas del crédito, asegurando la trazabilidad y eficiencia del proceso, así como la correcta gestión de los recursos y mejora de la experiencia del cliente.

❖ Requerimientos Funcionales:

- Registrar la información de clientes (nombre, apellido, correo, teléfono).
- Gestionar las cuentas de los clientes (número de cuenta, tipo, saldo).
- Registrar y vincular concesionarios y sus sucursales.
- Administrar los productos (vehículos) ofrecidos.
- Registrar y gestionar solicitudes de crédito vehicular.
- Controlar las tasas de interés, monto, fechas de inicio y vencimiento del crédito.
- Gestionar las garantías prendarias asociadas al crédito.
- Vincular créditos con garantías específicas.
- Consultar el stock de productos por sucursal.
- Generar reportes de créditos activos, vencidos y solicitudes pendientes.

❖ Entidades Identificadas:

- Cliente: Persona que solicita el crédito vehicular.
- Entidad Financiera: Institución que ofrece el crédito y controla el proceso.
- Cuenta: Registro financiero del cliente.
- Producto: Vehículo ofertado como objeto del crédito.
- Concesionario: Punto de venta del vehículo.
- Sucursal: Oficina del concesionario afiliada a la entidad.
- Crédito: Registro de la solicitud y aprobación del préstamo.
- Garantía: Prenda vehicular asociada al crédito.
- Relaciones N:M: Producto-Sucursal y Garantía-Crédito.

❖ Tablas y Campos Relevantes:

cliente

- id_cliente (PK)
- nombre
- apellido
- email
- telefono

entidad_financiera

- id_entidadFinanciera (PK)
- nombre
- direccion
- telefono

cuenta

- id_cuenta (PK)

- id_cliente (FK)
- id_entidadFinanciera (FK)
- numero_cuenta
- tipo
- saldo

producto

- id_producto (PK)
- nombre
- tipo
- descripcion
- precio

concesionario

- id_concesionario (PK)
- nombre
- direccion
- telefono

sucursal

- id_sucursal (PK)
- id_concesionario (FK)
- nombre
- direccion

producto_sucursal

- id_producto (FK)
- id_sucursal (FK)
- stock

credito

- id_credito (PK)
- id_cuenta (FK)

- id_producto (FK)
- monto
- interes
- fecha_inicio
- fecha_vencimiento

garantia

- id_garantia (PK)
- tipo
- descripcion
- valor

garantia_credito

- id_credito (FK)
- id_garantia (FK)

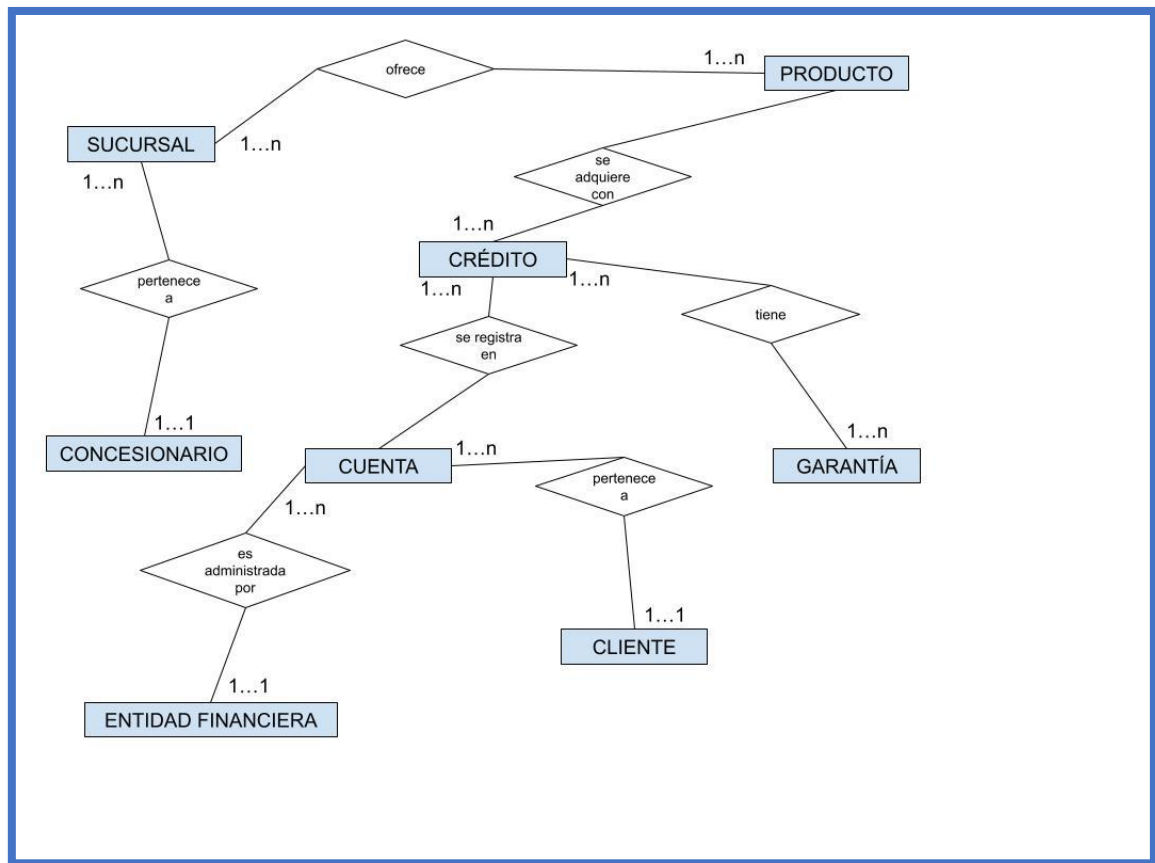
❖ Desafíos:

- Trazabilidad del proceso: Requiere asegurar que todas las etapas del crédito queden correctamente registradas.
- Asignación correcta de productos y sucursales: Debe reflejar adecuadamente la relación N:M y el stock disponible.
- Gestión de riesgos: Es vital controlar las garantías para minimizar riesgos crediticios.
- Escalabilidad y control: El sistema debe permitir la gestión simultánea de múltiples créditos, sucursales y clientes.
- Satisfacción del cliente: Se debe asegurar una experiencia fluida y sin errores.

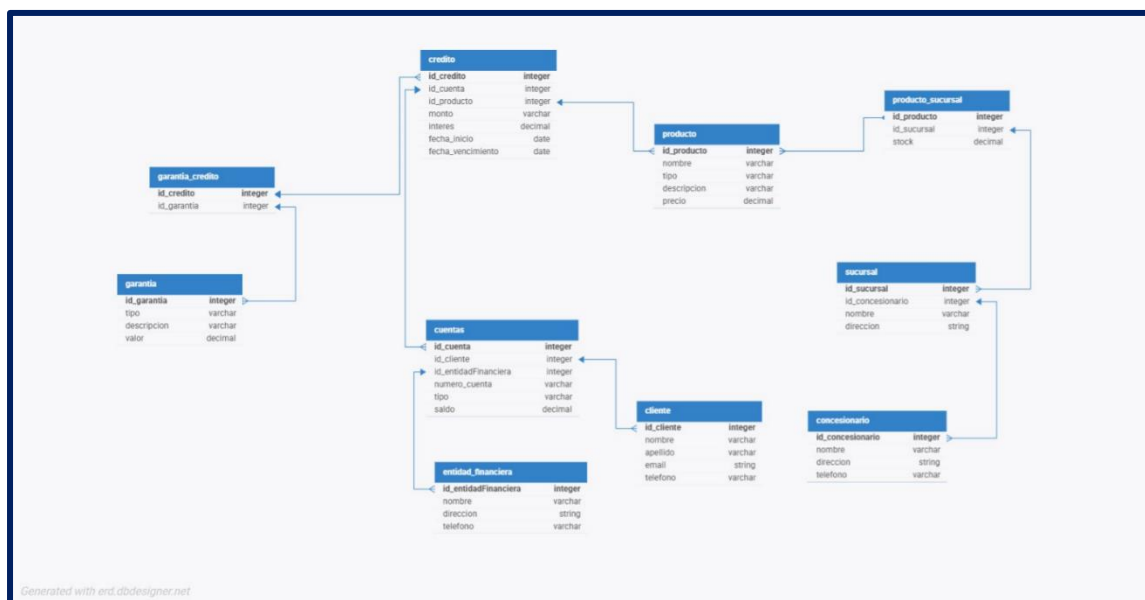
❖ Beneficios Esperados:

- Mejor control del proceso crediticio.
- Mayor eficiencia operativa en las aprobaciones y desembolsos.
- Trazabilidad clara y control de riesgos con garantías registradas.
- Mejora de la experiencia del cliente.
- Información precisa para toma de decisiones estratégicas.

2)Análisis: Diseño Conceptual de la base de datos



3)Diseño: Definición del Modelo Lógico de la Base de Datos.



4) Construcción: Desarrollo de las sentencias SQL que permiten la construcción de las tablas y demás estructuras de la Base de Datos en SQL SERVER.

- Creación de la tabla **cliente**

```
-- Tabla cliente
id_cliente INT IDENTITY(1,1) PRIMARY KEY,
nombre VARCHAR(100),
apellido VARCHAR(100),
email VARCHAR(100),
telefono VARCHAR(20)
);
```

- Creación de la tabla **entidad_financiera**

```
-- Tabla entidad_financiera
CREATE TABLE entidad_financiera (
    id_entidadFinanciera INT IDENTITY(1,1) PRIMARY KEY,
    nombre VARCHAR(100),
    direccion VARCHAR(200),
    telefono VARCHAR(20)
);
```

- Creación de la tabla **cuenta**

```
-- Tabla cuenta
CREATE TABLE cuenta (
    id_cuenta INT IDENTITY(1,1) PRIMARY KEY,
    id_cliente INT,
    id_entidadFinanciera INT,
    numero_cuenta VARCHAR(50),
    tipo VARCHAR(50),
    saldo DECIMAL(18,2)
);
```

```
-- FK en cuenta
ALTER TABLE cuenta
ADD CONSTRAINT fk_cuenta_cliente FOREIGN KEY (id_cliente) REFERENCES
cliente(id_cliente),
    CONSTRAINT fk_cuenta_entidad FOREIGN KEY (id_entidadFinanciera)
REFERENCES entidad_financiera(id_entidadFinanciera);
```


➤ Creación de la tabla **producto**

```
-- Tabla producto
CREATE TABLE producto (
  id_producto INT IDENTITY(1,1) PRIMARY KEY,
  nombre VARCHAR(100),
  tipo VARCHAR(50),
  descripcion VARCHAR(200),
  precio DECIMAL(18,2)
);
```

➤ Creación de la tabla **concesionario**

```
-- Tabla concesionario
CREATE TABLE concesionario (
  id_concesionario INT IDENTITY(1,1) PRIMARY KEY,
  nombre VARCHAR(100),
  direccion VARCHAR(200),
  telefono VARCHAR(20)
);
```

➤ Creación de la tabla **sucursal**

```
-- Tabla sucursal
CREATE TABLE sucursal (
  id_sucursal INT IDENTITY(1,1) PRIMARY KEY,
  id_concesionario INT,
  nombre VARCHAR(100),
  direccion VARCHAR(200)
);
```

```
-- FK en sucursal
ALTER TABLE sucursal
ADD CONSTRAINT fk_sucursal_concesionario FOREIGN KEY (id_concesionario)
REFERENCES concesionario(id_concesionario);
```

➤ Creación de la tabla **producto_sucursal**

```
-- Tabla producto_sucursal (relación N:M)
CREATE TABLE producto_sucursal (
  id_producto INT,
  id_sucursal INT,
  stock INT,
  PRIMARY KEY (id_producto, id_sucursal)
);
```

```
-- FK en producto_sucursal
ALTER TABLE producto_sucursal
ADD CONSTRAINT fk_productoSucursal_producto FOREIGN KEY (id_producto)
REFERENCES producto(id_producto),
  CONSTRAINT fk_productoSucursal_sucursal FOREIGN KEY (id_sucursal)
REFERENCES sucursal(id_sucursal);
```

➤ Creación de la tabla **crédito**

```
-- Tabla credito
CREATE TABLE credito (
  id_credito INT IDENTITY(1,1) PRIMARY KEY,
  id_cuenta INT,
  id_producto INT,
  monto DECIMAL(18,2),
  interes DECIMAL(5,2),
  fecha_inicio DATE,
  fecha_vencimiento DATE
);
```

```
-- FK en credito
ALTER TABLE credito
ADD CONSTRAINT fk_credito_cuenta FOREIGN KEY (id_cuenta) REFERENCES
cuenta(id_cuenta),
  CONSTRAINT fk_credito_producto FOREIGN KEY (id_producto) REFERENCES
producto(id_producto);
```

- Creación de la tabla **garantía**

```
-- Tabla garantia
CREATE TABLE garantia (
  id_garantia INT IDENTITY(1,1) PRIMARY KEY,
  tipo VARCHAR(100),
  descripcion VARCHAR(200),
  valor DECIMAL(18,2)
);
```

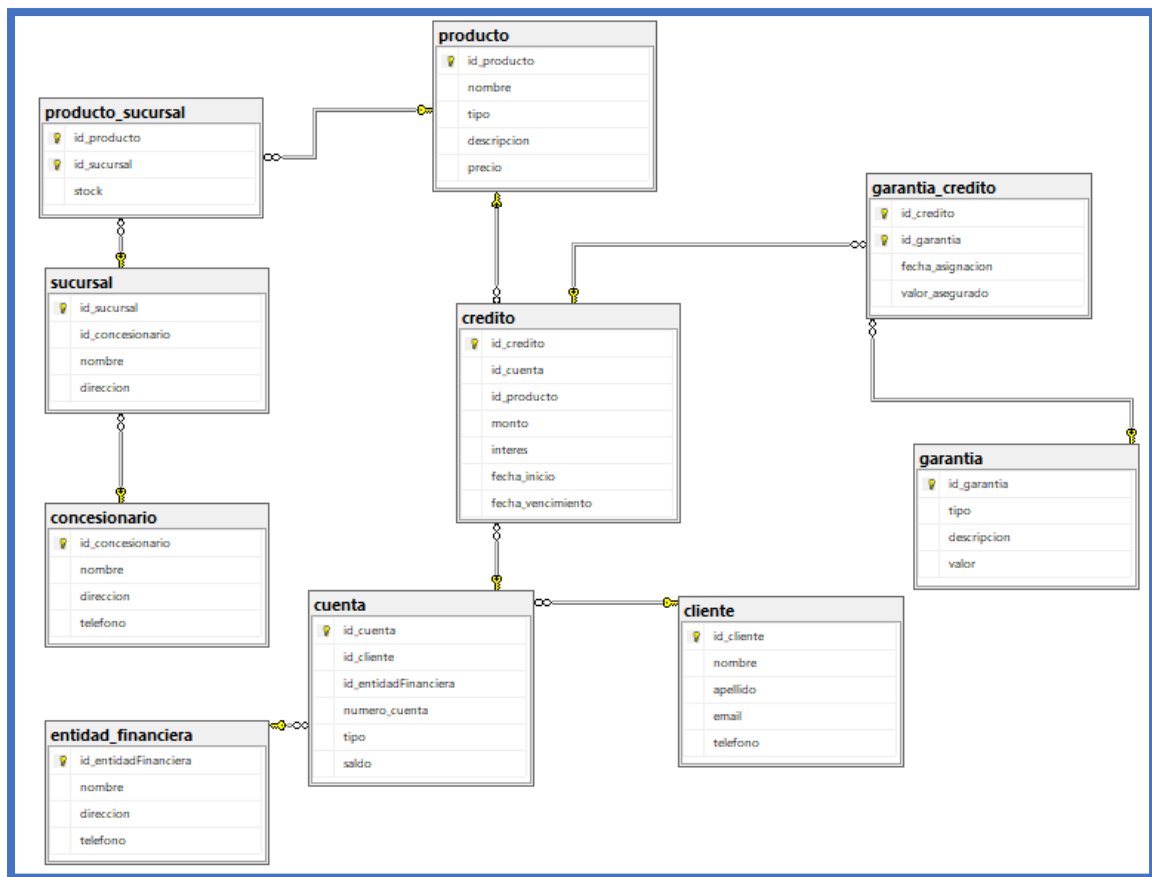
- Creación de la tabla **garantía_credito**

```
-- Tabla garantia_credito (relación N:M)
CREATE TABLE garantia_credito (
  id_credito INT,
  id_garantia INT,
  PRIMARY KEY (id_credito, id_garantia)
);
```

```
-- FK en garantia_credito
ALTER TABLE garantia_credito
ADD CONSTRAINT fk_garantiaCredito_credito FOREIGN KEY (id_credito)
REFERENCES credito(id_credito),
  CONSTRAINT fk_garantiaCredito_garantia FOREIGN KEY (id_garantia)
REFERENCES garantia(id_garantia);
```

5)Producto: Diseño Físico, Script de la Base de datos con inserción de datos (Mínimo 5 registros por tablas independientes o no transaccionales y mínimo 15 registros por cada tabla transaccional).

❖ Diseño físico



- Link del SCRIPT

Se adjunta link de nuestro repositorio en **GITHUB** donde está incluido el Script:

https://github.com/DurandGalindo/BDD_creditos_vehiculares.git

6) Inserciones

❖ Inserción de datos a tablas independientes

➤ Tabla **clientes**:

```
--Tabla cliente
SELECT * FROM cliente;
DELETE FROM cliente
DBCC CHECKIDENT ('cliente', RESEED, 0);
GO
INSERT INTO cliente (nombre, apellido, email, telefono) VALUES
('Diego', 'Gomez', 'dgomez@unmsm.edu.pe', '984621098'),
('Lucia', 'Garcia', 'lucia.g@unmsm.edu.pe', '984126350'),
('Ana', 'Perez', 'ana.perez@gmail.com', '923187645'),
('Carlos', 'Lopez', 'clopez@hotmail.com', '949112287'),
('Valeria', 'Morales', 'valeria.m@gmail.com', '987612530'),
('Pedro', 'Soto', 'pedro.soto@unmsm.edu.pe', '957489621'),
('Flor', 'Mendoza', 'flor.m@unmsm.edu.pe', '913476281'),
('Luis', 'Vargas', 'l.vargas@unmsm.edu.pe', '937416205'),
('Gabriela', 'Silva', 'g.silva@gmail.com', '917438612'),
('Carmen', 'Delgado', 'carmen.d@gmail.com', '974562013'),
('Ricardo', 'Silva', 'ricardo.silva@gmail.com', '983214875'),
('Jorge', 'Cruz', 'jorge.c@gmail.com', '961875230'),
('Pedro', 'Lopez', 'pedro.lopez@hotmail.com', '918736402'),
('Lucia', 'Vargas', 'lucia.vargas@gmail.com', '989764321'),
('Raul', 'Cruz', 'raulc@hotmail.com', '999283615'),
('Juan', 'Lopez', 'juanlopez@unmsm.edu.pe', '963142578'),
('Gabriela', 'Chavez', 'gabriela.ch@unmsm.edu.pe', '976431209'),
('Carmen', 'Cruz', 'carmen.cruz@hotmail.com', '981432672'),
('Jorge', 'Salazar', 'jorge.salazar@unmsm.edu.pe', '952348713'),
('Diego', 'Cruz', 'diego.cruz@hotmail.com', '918726354'),
('Valeria', 'Ramos', 'valeria.ramos@hotmail.com', '913562479'),
('Luis', 'Rojas', 'lrojas@gmail.com', '998124578'),
('Maria', 'Soto', 'm.soto@unmsm.edu.pe', '997134258'),
('Flor', 'Rojas', 'f.rojas@gmail.com', '942875163'),
('Juan', 'Silva', 'juan.silva@gmail.com', '984712639'),
('Elena', 'Mendoza', 'elena.m@unmsm.edu.pe', '931875210'),
('Ricardo', 'Torres', 'rtorres@hotmail.com', '965789431'),
('Sandra', 'Gomez', 'sandrag@unmsm.edu.pe', '996412783'),
('Victor', 'Vargas', 'victor.v@gmail.com', '989112540'),
('Ana', 'Garcia', 'ana.garcia@hotmail.com', '951237846'),
('Diego', 'Salazar', 'dsalazar@hotmail.com', '958742693'),
('Sofia', 'Garcia', 'sofia.garcia@gmail.com', '975216384'),
('Luis', 'Gomez', 'l.gomez@hotmail.com', '987452133'),
('Alfredo', 'Soto', 'alfredo.soto@gmail.com', '954682193'),
('Carlos', 'Torres', 'torres.carlos@gmail.com', '951334467'),
('Alfredo', 'Chavez', 'alf.chavez@unmsm.edu.pe', '998731205'),
('Victor', 'Garcia', 'v.garcia@hotmail.com', '931287436'),
```

```
( 'Maria', 'Gomez', 'm.gomez@unmsm.edu.pe', '977289163'),
( 'Jorge', 'Ramirez', 'j.ramirez@gmail.com', '962741085'),
( 'Elena', 'Perez', 'eperez@hotmail.com', '949875132'),
( 'Sofia', 'Martinez', 'sofia.m@unmsm.edu.pe', '926543817'),
( 'Raul', 'Martinez', 'raul.martinez@unmsm.edu.pe', '982146389'),
( 'Maria', 'Ramirez', 'maria.ramirez@hotmail.com', '934712849'),
( 'Sandra', 'Ramirez', 'sandra.ramirez@hotmail.com', '978321754'),
( 'Ana', 'Rojas', 'arojas@unmsm.edu.pe', '915823744'),
( 'Sandra', 'Lopez', 's.lopez@unmsm.edu.pe', '915637240'),
( 'Juan', 'Torres', 'juan.torres@gmail.com', '987165431'),
( 'Alfredo', 'Chavez', 'alf.chavez@unmsm.edu.pe', '998731205'),
( 'Lucia', 'Ramos', 'lucia.ramos@hotmail.com', '996314289'),
( 'Pedro', 'Rojas', 'p.rojas@gmail.com', '948762309');
```

➤ Tabla **entidad_financiera**:

```
--Tabla entidad_financiera
SELECT * FROM entidad_financiera;
GO
DELETE FROM entidad_financiera
DBCC CHECKIDENT ('entidad_financiera', RESEED, 0);
INSERT INTO entidad_financiera (nombre, direccion, telefono) VALUES
( 'Banco Uno', 'Av. Central 123', '900000001'),
( 'Caja Lima', 'Av. La Marina 200', '900000002'),
( 'Financiera ABC', 'Calle Amazonas 222', '900000003'),
( 'Banco del Pueblo', 'Av. San Juan 320', '900000004'),
( 'CreditoSeguro', 'Jr. Cusco 789', '900000005'),
( 'Financiera Capital', 'Av. Grau 870', '900000006'),
( 'Caja Arequipa', 'Av. Cultura 900', '900000007'),
( 'Banco Familiar', 'Calle Apurímac 210', '900000008'),
( 'MicroBanco', 'Av. Brasil 515', '900000009'),
( 'Caja Metropolitana', 'Calle Bolívar 720', '900000010'),
( 'Banco Uno', 'Jr. Cajamarca 875', '900000011'),
( 'Caja Lima', 'Av. Libertad 800', '900000012'),
( 'Financiera ABC', 'Av. Ferrocarril 450', '900000013'),
( 'Banco del Pueblo', 'Av. República 315', '900000014'),
( 'CreditoSeguro', 'Calle Callao 808', '900000015'),
( 'Financiera Capital', 'Calle Tacna 345', '900000016'),
( 'Caja Arequipa', 'Calle Piura 556', '900000017'),
( 'Banco Familiar', 'Av. Universitaria 1050', '900000018'),
( 'MicroBanco', 'Av. Independencia 159', '900000019'),
( 'Caja Metropolitana', 'Av. Salaverry 300', '900000020'),
( 'Banco Uno', 'Av. Túpac Amaru 890', '900000021'),
( 'Caja Lima', 'Av. Panamericana 1420', '900000022'),
( 'Financiera ABC', 'Av. Los Próceres 490', '900000023'),
( 'Banco del Pueblo', 'Av. Bolívar 1200', '900000024'),
( 'CreditoSeguro', 'Calle Sur 223', '900000025');
```

```
('Financiera Capital', 'Av. Lima 999', '900000026'),  
( 'Caja Arequipa', 'Jr. Miraflores 345', '900000027'),  
( 'Banco Familiar', 'Av. Piérola 200', '900000028'),  
( 'MicroBanco', 'Calle Ayacucho 111', '900000029'),  
( 'Caja Metropolitana', 'Av. Andes 222', '900000030'),  
( 'Banco Andino', 'Av. Central 123', '900000031'),  
( 'Banco Andino', 'Av. San Juan 320', '900000032'),  
( 'Banco Andino', 'Calle Amazonas 222', '900000033'),  
( 'Banco Andino', 'Jr. Cajamarca 875', '900000034'),  
( 'Banco Emprende', 'Av. Cultura 900', '900000035'),  
( 'Banco Emprende', 'Calle Bolívar 720', '900000036'),  
( 'Banco Emprende', 'Calle Callao 808', '900000037'),  
( 'Banco Emprende', 'Av. Grau 870', '900000038'),  
( 'Caja Cusco', 'Av. Universitaria 1050', '900000039'),  
( 'Caja Cusco', 'Av. La Marina 200', '900000040'),  
( 'Caja Cusco', 'Calle Piura 556', '900000041'),  
( 'Caja Cusco', 'Av. Salaverry 300', '900000042'),  
( 'Financiera Esperanza', 'Av. República 315', '900000043'),  
( 'Financiera Esperanza', 'Av. Bolívar 1200', '900000044'),  
( 'Financiera Esperanza', 'Calle Ayacucho 111', '900000045'),  
( 'Financiera Esperanza', 'Av. Ferrocarril 450', '900000046'),  
( 'Microfinanzas Andinas', 'Calle Tacna 345', '900000047'),  
( 'Microfinanzas Andinas', 'Av. Panamericana 1420', '900000048'),  
( 'Microfinanzas Andinas', 'Av. Libertad 800', '900000049'),  
( 'Microfinanzas Andinas', 'Av. Los Próceres 490', '900000050');
```

➤ Tabla **producto**:

```
--Tabla producto
SELECT * FROM producto;
GO
INSERT INTO producto (nombre, tipo, descripcion, precio) VALUES
('Sedan LX 2025', 'Carro', 'Sedan moderno con bajo consumo', 13500),
('Pickup Strong X', 'Camion', 'Pickup resistente para carga pesada',
26000),
('Moto Runner 150', 'Motocicleta', 'Moto ligera ideal para ciudad',
4900),
('Bus Escolar Max', 'Bus', 'Bus escolar con 40 asientos', 34000),
('SUV Explorer V6', 'Carro', 'SUV para familias grandes', 22000),
('Camion Turbo Carga', 'Camion', 'Camion de gran tonelaje', 39000),
('Scooter EcoLite', 'Motocicleta', 'Scooter electrico urbano', 3200),
('Bus Urbano XL', 'Bus', 'Bus urbano con aire acondicionado', 42000),
('Hatchback Swift', 'Carro', 'Auto compacto con gran maniobrabilidad',
11500),
('Camioneta 4x4 Sierra', 'Camion', 'Ideal para caminos rurales', 27500),
('Moto CrossX 200', 'Motocicleta', 'Motocicleta de motocross resistente',
6500),
('Minibus Ruta 7', 'Bus', 'Minibus ideal para rutas cortas', 25000),
('Coupe GT Turbo', 'Carro', 'Carro deportivo compacto', 18000),
('Camion CargaMax', 'Camion', 'Camion diesel para largas distancias',
41000),
('Moto Urbana RZ', 'Motocicleta', 'Moto economica para uso diario',
5400),
('Bus Interprovincial X1', 'Bus', 'Bus de viaje con butacas reclinables',
46000),
('Sedan Classic 1.6', 'Carro', 'Auto tradicional con estilo sobrio',
12800),
('Camion Boxer 300', 'Camion', 'Camion cerrado tipo furgon', 35000),
('Moto Retro 125', 'Motocicleta', 'Motocicleta con diseño clasico',
4700),
('Bus Ejecutivo Gold', 'Bus', 'Bus de lujo con servicios premium',
52000),
('Crossover Compact', 'Carro', 'Crossover para ciudad y carretera',
19000),
('Camioneta Carga Light', 'Camion', 'Camioneta para medianas cargas',
22000),
('Moto Electrica Volt', 'Motocicleta', 'Moto electrica con buena
autonomia', 7000),
('Bus Escolar Mini', 'Bus', 'Bus escolar compacto de 20 asientos',
29500),
('Hatchback Zippy', 'Carro', 'Hatchback joven y economico', 10500),
('Camion Basculante X2', 'Camion', 'Camion con tolva para construccion',
37000),
('Moto Sport FZ', 'Motocicleta', 'Moto deportiva de alta cilindrada',
8900),
```



```
( 'Bus Rural Linea', 'Bus', 'Bus para rutas rurales', 31000),
( 'Sedan Elegance', 'Carro', 'Auto con diseño sofisticado', 14500),
( 'Camion Frigorifico', 'Camion', 'Camion con refrigeracion integrada',
44000),
( 'Moto Trail 250', 'Motocicleta', 'Moto doble proposito para campo y
ciudad', 6000),
( 'Bus Hibrido Eco', 'Bus', 'Bus ecologico hibrido', 48500),
( 'Carro Deportivo RS', 'Carro', 'Carro de alto rendimiento', 25000),
( 'Camion Granelero', 'Camion', 'Camion especializado en transporte a
granel', 36500),
( 'Moto Naked 300', 'Motocicleta', 'Moto sin carenado de aspecto
agresivo', 7300),
( 'Bus Articulado XL', 'Bus', 'Bus articulado de gran capacidad', 55000),
( 'Compacto Familiar', 'Carro', 'Auto compacto para 5 personas', 13000),
( 'Camion Liviano Z1', 'Camion', 'Camion liviano para ciudad', 24000),
( 'Moto Touring 400', 'Motocicleta', 'Moto para viajes largos', 8500),
( 'Bus Ejecutivo SemiCama', 'Bus', 'Bus de viaje semi-cama', 47000),
( 'Hatchback EcoDrive', 'Carro', 'Hatchback ecologico con motor hibrido',
17000),
( 'Camion Mixer', 'Camion', 'Camion mezclador de concreto', 43000),
( 'Moto Scoot Fast', 'Motocicleta', 'Scooter de 125cc para ciudad', 5200),
( 'Bus Metropolitano', 'Bus', 'Bus para transporte masivo urbano', 50000),
( 'Sedan Hybrid Pro', 'Carro', 'Auto sedan con motor hibrido', 20000),
( 'Camion Tipper Pro', 'Camion', 'Camion volquete de 12 toneladas',
39500),
( 'Moto Enduro XT', 'Motocicleta', 'Moto para terrenos extremos', 7900),
( 'Bus Escolar Safe', 'Bus', 'Bus escolar reforzado con GPS', 33000),
( 'Coupe Elegance V8', 'Carro', 'Coupe deportivo con motor potente',
27000),
( 'Camioneta DualPower', 'Camion', 'Camioneta diesel/hibrida para trabajo
y ciudad', 31000);
```

➤ **Tabla concesionario:**

```
--Tabla concesionario
SELECT * FROM concesionario;
GO
INSERT INTO concesionario (nombre, direccion, telefono) VALUES
( 'CamionetasPlus', 'Av. La Marina 1800', '910000001'),
( 'AutoMarket', 'Calle Velocidad 404', '910000002'),
( 'MotoExpress', 'Av. Revolucion 501', '910000003'),
( 'Concesionaria Sur', 'Av. Brasil 300', '910000004'),
( 'SpeedCar', 'Av. Los Olivos 100', '910000005'),
( 'Ruedas del Sur', 'Jr. Carros 505', '910000006'),
( 'MotoExpress', 'Av. Aceleracion 150', '910000007'),
( 'VelocidadMax', 'Jr. Automovil 340', '910000008'),
( 'AutoMarket', 'Calle Llantas 880', '910000009'),
```

```
('Concesionaria Norte', 'Av. Maquinarias 970', '910000010'),
('MotoCity', 'Av. Central 123', '910000011'),
('SuperAutos', 'Jr. Motores 220', '910000012'),
('Concesionaria Lima', 'Av. Turbo 101', '910000013'),
('CamionetasPlus', 'Calle Tracción 210', '910000014'),
('Ruedas del Sur', 'Av. Libertad 345', '910000015'),
('SpeedCar', 'Av. Independencia 159', '910000016'),
('AutoAndino', 'Jr. Combustible 715', '910000017'),
('CamionetasPlus', 'Av. Brasil 300', '910000018'),
('VelocidadMax', 'Av. Pista Nueva 600', '910000019'),
('MotoExpress', 'Jr. Lima 202', '910000020'),
('SuperAutos', 'Calle Velocidad 404', '910000021'),
('AutoMarket', 'Jr. Carros 505', '910000022'),
('Ruedas del Sur', 'Av. Revolucion 501', '910000023'),
('Concesionaria Norte', 'Av. La Marina 1800', '910000024'),
('MotoCity', 'Av. Libertad 345', '910000025'),
('CamionetasPlus', 'Av. Turbo 101', '910000026'),
('Concesionaria Sur', 'Av. Central 123', '910000027'),
('AutoAndino', 'Jr. Lima 202', '910000028'),
('VelocidadMax', 'Calle Tracción 210', '910000029'),
('SpeedCar', 'Calle Llantas 880', '910000030'),
('AutoMarket', 'Av. Aceleracion 150', '910000031'),
('Concesionaria Lima', 'Jr. Combustible 715', '910000032'),
('MotoCity', 'Jr. Automovil 340', '910000033'),
('AutoAndino', 'Av. Maquinarias 970', '910000034'),
('SuperAutos', 'Av. Independencia 159', '910000035'),
('CamionetasPlus', 'Jr. Motores 220', '910000036'),
('Ruedas del Sur', 'Av. La Marina 1800', '910000037'),
('VelocidadMax', 'Av. Central 123', '910000038'),
('Concesionaria Lima', 'Calle Velocidad 404', '910000039'),
('AutoMarket', 'Av. Libertad 345', '910000040'),
('MotoCity', 'Jr. Automovil 340', '910000041'),
('SpeedCar', 'Jr. Carros 505', '910000042'),
('Concesionaria Sur', 'Av. Revolucion 501', '910000043'),
('SuperAutos', 'Jr. Lima 202', '910000044'),
('CamionetasPlus', 'Av. Pista Nueva 600', '910000045'),
('Concesionaria Norte', 'Av. Turbo 101', '910000046'),
('Ruedas del Sur', 'Calle Llantas 880', '910000047'),
('AutoAndino', 'Calle Tracción 210', '910000048'),
('MotoExpress', 'Jr. Motores 220', '910000049'),
('VelocidadMax', 'Jr. Combustible 715', '910000050');
```

➤ Tabla **garantia**:

```
--Tabla garantia
SELECT * FROM garantia;
GO
INSERT INTO garantia (tipo, descripcion, valor) VALUES
('Hipoteca', 'Casa como respaldo', 50000),
('Prenda Vehicular', 'Auto usado como garantia', 12000),
('Aval', 'Tercero responsable solidario', 10000),
('Seguro de Desgravamen', 'Cubre deuda en fallecimiento', 3000),
('Deposito en garantia', 'Monto retenido', 5000),
('Fianza', 'Garantia personal', 7000),
('Carta Fianza', 'Garantia bancaria', 15000),
('Garantia Bancaria', 'Respaldo financiero', 18000),
('Prenda Mercantil', 'Mercaderia como garantia', 11000),
('Hipoteca Rural', 'Terreno agricola como garantia', 25000),
('Garantia Fiduciaria', 'Transferencia de bienes en garantia', 22000),
('Caucion', 'Garantia para cumplimiento de contrato', 9000),
('Prenda Industrial', 'Equipo industrial en garantia', 14000),
('Seguro de Garantia', 'Seguro contra incumplimiento', 3500),
('Garantia Solidaria', 'Responsabilidad conjunta', 13000),
('Hipoteca Comercial', 'Inmueble comercial en garantia', 60000),
('Prenda Agraria', 'Equipo agricola como garantia', 17000),
('Aval Personal', 'Avalista individual', 8000),
('Garantia Patronal', 'Responsabilidad de empleador', 4000),
('Garantia Judicial', 'Respaldo en proceso legal', 12000),
('Hipoteca', 'Casa como respaldo', 52000),
('Prenda Vehicular', 'Auto usado como garantia', 12500),
('Aval', 'Tercero responsable solidario', 10500),
('Seguro de Desgravamen', 'Cubre deuda en fallecimiento', 3100),
('Deposito en garantia', 'Monto retenido', 5300),
('Fianza', 'Garantia personal', 7200),
('Carta Fianza', 'Garantia bancaria', 15500),
('Garantia Bancaria', 'Respaldo financiero', 18500),
('Prenda Mercantil', 'Mercaderia como garantia', 11500),
('Hipoteca Rural', 'Terreno agricola como garantia', 25500),
('Garantia Fiduciaria', 'Transferencia de bienes en garantia', 22500),
('Caucion', 'Garantia para cumplimiento de contrato', 9300),
('Prenda Industrial', 'Equipo industrial en garantia', 14500),
('Seguro de Garantia', 'Seguro contra incumplimiento', 3600),
('Garantia Solidaria', 'Responsabilidad conjunta', 13500),
('Hipoteca Comercial', 'Inmueble comercial en garantia', 62000),
('Prenda Agraria', 'Equipo agricola como garantia', 17500),
('Aval Personal', 'Avalista individual', 8400),
('Garantia Patronal', 'Responsabilidad de empleador', 4200),
('Garantia Judicial', 'Respaldo en proceso legal', 12500),
('Hipoteca', 'Casa como respaldo', 54000),
('Prenda Vehicular', 'Auto usado como garantia', 13000),
('Aval', 'Tercero responsable solidario', 11000),
```

```
( 'Seguro de Desgravamen', 'Cubre deuda en fallecimiento', 3200),
( 'Deposito en garantia', 'Monto retenido', 5600),
( 'Fianza', 'Garantia personal', 7400),
( 'Carta Fianza', 'Garantia bancaria', 16000),
( 'Garantia Bancaria', 'Respaldo financiero', 19000),
( 'Prenda Mercantil', 'Mercaderia como garantia', 12000),
( 'Hipoteca Rural', 'Terreno agricola como garantia', 26000);
```

❖ Inserción de datos a tablas transaccionales

➤ Tabla **cuenta**:

```
--TABLAS TRANSACCIONALES
use BDD_creditos_vehiculares

--Tabla cuenta
SELECT * FROM cuenta
INSERT INTO cuenta (id_cliente, id_entidadFinanciera, numero_cuenta,
tipo,saldo) VALUES
(1, 1, '900000002345678', 'corriente', 3200.00),
(2, 2, '900000003456789', 'ahorros', 580.40),
(3, 3, '900000004567890', 'corriente', 12000.00),
(4, 4, '900000005678901', 'ahorros', 230.10),
(5, 5, '900000006789012', 'corriente', 4520.90),
(6, 6, '900000007890123', 'ahorros', 980.00),
(7, 7, '900000008901234', 'corriente', 1100.50),
(8, 8, '900000009012345', 'ahorros', 745.25),
(9, 9, '900000010123456', 'corriente', 6700.00),
(10, 10, '900000011234567', 'ahorros', 230.00),
(11, 11, '900000012345678', 'corriente', 1820.15),
(12, 12, '900000013456789', 'ahorros', 910.70),
(13, 13, '900000014567890', 'corriente', 650.30),
(14, 14, '900000015678901', 'ahorros', 1400.90),
(15, 15, '900000016789012', 'corriente', 7800.50),
(16, 16, '900000017890123', 'ahorros', 300.00),
(17, 17, '900000018901234', 'corriente', 420.00),
(18, 18, '900000019012345', 'ahorros', 1250.25),
(19, 19, '900000020123456', 'corriente', 6800.00),
(20, 20, '900000021234567', 'ahorros', 190.45),
(21, 21, '900000022345678', 'corriente', 1500.00),
(22, 22, '900000023456789', 'ahorros', 245.80),
(23, 23, '900000024567890', 'corriente', 3200.75),
(24, 24, '900000025678901', 'ahorros', 700.50),
(25, 25, '900000026789012', 'corriente', 560.00),
(26, 26, '900000027890123', 'ahorros', 230.00),
(27, 27, '900000028901234', 'corriente', 4000.90),
(28, 28, '900000029012345', 'ahorros', 1100.10),
(29, 29, '900000030123456', 'corriente', 980.00),
```

```
(30, 30, '900000031234567', 'ahorros', 1340.00),
(31, 31, '900000032345678', 'corriente', 2700.00),
(32, 32, '900000033456789', 'ahorros', 750.25),
(33, 33, '900000034567890', 'corriente', 1020.40),
(34, 34, '900000035678901', 'ahorros', 890.50),
(35, 35, '900000036789012', 'corriente', 630.60),
(36, 36, '900000037890123', 'ahorros', 480.00),
(37, 37, '900000038901234', 'corriente', 7200.00),
(38, 38, '900000039012345', 'ahorros', 550.00),
(39, 39, '900000040123456', 'corriente', 1800.75),
(40, 40, '900000041234567', 'ahorros', 640.20),
(41, 41, '900000042345678', 'corriente', 300.00),
(42, 42, '900000043456789', 'ahorros', 960.50),
(43, 43, '900000044567890', 'corriente', 450.00),
(44, 44, '900000045678901', 'ahorros', 320.00),
(45, 45, '900000046789012', 'corriente', 2400.00),
(46, 46, '900000047890123', 'ahorros', 700.00),
(47, 47, '900000048901234', 'corriente', 1100.00),
(48, 48, '900000049012345', 'ahorros', 2300.00),
(49, 49, '900000050123456', 'corriente', 950.00),
(50, 50, '900000001234567', 'ahorros', 1500.75),
(1, 2, '900000002345679', 'corriente', 2900.00),
(2, 3, '900000003456780', 'ahorros', 780.40),
(3, 4, '900000004567891', 'corriente', 8200.00),
(4, 5, '900000005678902', 'ahorros', 430.10),
(5, 6, '900000006789013', 'corriente', 2520.90),
(6, 7, '900000007890124', 'ahorros', 780.00),
(7, 8, '900000008901235', 'corriente', 1600.50),
(8, 9, '900000009012346', 'ahorros', 845.25),
(9, 10, '900000010123457', 'corriente', 7700.00),
(10, 11, '900000011234568', 'ahorros', 130.00),
(11, 12, '900000012345679', 'corriente', 1820.15),
(12, 13, '900000013456780', 'ahorros', 1010.70),
(13, 14, '900000014567891', 'corriente', 750.30),
(14, 15, '900000015678902', 'ahorros', 1800.90),
(15, 16, '900000016789013', 'corriente', 8800.50),
(16, 17, '900000017890124', 'ahorros', 600.00),
(17, 18, '900000018901235', 'corriente', 920.00),
(18, 19, '900000019012346', 'ahorros', 1550.25),
(19, 20, '900000020123457', 'corriente', 7800.00),
(20, 21, '900000021234568', 'ahorros', 290.45),
(21, 22, '900000022345679', 'corriente', 1800.00),
(22, 23, '900000023456780', 'ahorros', 445.80),
(23, 24, '900000024567891', 'corriente', 4200.75),
(24, 25, '900000025678902', 'ahorros', 900.50),
(25, 26, '900000026789013', 'corriente', 1060.00),
(26, 27, '900000027890124', 'ahorros', 530.00),
(27, 28, '900000028901235', 'corriente', 5000.90),
(28, 29, '900000029012346', 'ahorros', 1400.10),
```

```
(29, 30, '900000030123457', 'corriente', 880.00),
(30, 31, '900000031234568', 'ahorros', 1140.00),
(31, 32, '900000032345679', 'corriente', 3700.00),
(32, 33, '900000033456780', 'ahorros', 950.25),
(33, 34, '900000034567891', 'corriente', 1220.40),
(34, 35, '900000035678902', 'ahorros', 990.50),
(35, 36, '900000036789013', 'corriente', 830.60),
(36, 37, '900000037890124', 'ahorros', 680.00),
(37, 38, '900000038901235', 'corriente', 8200.00),
(38, 39, '900000039012346', 'ahorros', 750.00),
(39, 40, '900000040123457', 'corriente', 2800.75),
(40, 41, '900000041234568', 'ahorros', 740.20),
(41, 42, '900000042345679', 'corriente', 400.00),
(42, 43, '900000043456780', 'ahorros', 1060.50),
(43, 44, '900000044567891', 'corriente', 550.00),
(44, 45, '900000045678902', 'ahorros', 420.00),
(45, 46, '900000046789013', 'corriente', 3400.00),
(46, 47, '900000047890124', 'ahorros', 900.00),
(47, 48, '900000048901235', 'corriente', 1300.00),
(48, 49, '900000049012346', 'ahorros', 2500.00),
(49, 50, '900000050123457', 'corriente', 1150.00),
(50, 1, '900000001234569', 'ahorros', 1150.75),
(1, 3, '900000002345680', 'corriente', 2600.00),
(2, 5, '900000003456781', 'ahorros', 880.40),
(3, 7, '900000004567892', 'corriente', 7200.00),
(4, 9, '900000005678903', 'ahorros', 530.10),
(5, 11, '900000006789014', 'corriente', 1520.90),
(6, 13, '900000007890125', 'ahorros', 680.00),
(7, 15, '900000008901236', 'corriente', 2100.50),
(8, 17, '900000009012347', 'ahorros', 945.25),
(9, 19, '900000010123458', 'corriente', 8700.00),
(10, 21, '900000011234569', 'ahorros', 230.00),
(11, 23, '900000012345680', 'corriente', 2020.15),
(12, 25, '900000013456781', 'ahorros', 1110.70),
(13, 27, '900000014567892', 'corriente', 850.30),
(14, 29, '900000015678903', 'ahorros', 1900.90),
(15, 31, '900000016789014', 'corriente', 9800.50),
(16, 33, '900000017890125', 'ahorros', 700.00),
(17, 35, '900000018901236', 'corriente', 1020.00),
(18, 37, '900000019012347', 'ahorros', 1750.25),
(19, 39, '900000020123458', 'corriente', 8800.00),
(20, 41, '900000021234569', 'ahorros', 390.45),
(21, 43, '900000022345680', 'corriente', 2100.00),
(22, 45, '900000023456781', 'ahorros', 545.80),
(23, 47, '900000024567892', 'corriente', 5200.75),
(24, 49, '900000025678903', 'ahorros', 1100.50),
(25, 1, '900000026789014', 'corriente', 1260.00),
(26, 2, '900000027890125', 'ahorros', 630.00),
(27, 4, '900000028901236', 'corriente', 6000.90),
```

```
(28, 6, '900000029012347', 'ahorros', 1500.10),
(29, 8, '900000030123458', 'corriente', 1080.00),
(30, 10, '900000031234569', 'ahorros', 1340.00),
(31, 12, '900000032345680', 'corriente', 4700.00),
(32, 14, '900000033456781', 'ahorros', 1150.25),
(33, 16, '900000034567892', 'corriente', 1420.40),
(34, 18, '900000035678903', 'ahorros', 1190.50),
(35, 20, '900000036789014', 'corriente', 1030.60),
(36, 22, '900000037890125', 'ahorros', 880.00),
(37, 24, '900000038901236', 'corriente', 9200.00),
(38, 26, '900000039012347', 'ahorros', 850.00),
(39, 28, '900000040123458', 'corriente', 3800.75),
(40, 30, '900000041234569', 'ahorros', 840.20),
(41, 32, '900000042345680', 'corriente', 500.00),
(42, 34, '900000043456781', 'ahorros', 1160.50),
(43, 36, '900000044567892', 'corriente', 650.00),
(44, 38, '900000045678903', 'ahorros', 520.00),
(45, 40, '900000046789014', 'corriente', 3700.00),
(46, 42, '900000047890125', 'ahorros', 1100.00),
(47, 44, '900000048901236', 'corriente', 1500.00),
(48, 46, '900000049012347', 'ahorros', 2700.00),
(49, 48, '900000050123458', 'corriente', 1350.00),
(50, 2, '900000001234568', 'ahorros', 1300.75);
```


➤ Tabla **credito**:

```
--Tabla credito
SELECT*FROM credito;

INSERT INTO credito (id_cuenta, id_producto, monto, interes,
fecha_inicio, fecha_vencimiento) VALUES
(1, 1, 17000, 6.0, '2023-07-24', '2027-07-17'),
(2, 2, 10000, 5.5, '2024-05-23', '2027-05-14'),
(3, 3, 9000, 6.5, '2022-05-10', '2025-05-04'),
(4, 4, 15000, 6.0, '2022-09-13', '2026-09-06'),
(5, 5, 20000, 6.0, '2023-08-19', '2027-08-12'),
(6, 6, 10000, 4.5, '2023-04-23', '2026-04-16'),
(7, 7, 11000, 4.5, '2022-02-26', '2026-02-19'),
(8, 8, 17000, 4.5, '2023-03-03', '2026-02-24'),
(9, 9, 10000, 6.0, '2022-01-09', '2026-01-02'),
(10, 10, 20000, 4.5, '2022-09-08', '2025-09-02'),
(11, 11, 17000, 5.5, '2022-01-14', '2025-01-08'),
(12, 12, 15000, 6.0, '2022-01-12', '2025-01-06'),
(13, 13, 10000, 6.0, '2023-10-21', '2026-10-14'),
(14, 14, 9000, 4.5, '2023-03-18', '2027-03-11'),
(15, 15, 11000, 6.0, '2022-02-15', '2026-02-08'),
(16, 16, 17000, 5.5, '2022-07-08', '2025-07-02'),
(17, 17, 20000, 5.0, '2022-03-03', '2025-02-25'),
(18, 18, 11000, 6.0, '2023-06-23', '2026-06-16'),
(19, 19, 12000, 5.0, '2024-03-14', '2027-03-07'),
(20, 20, 12000, 5.0, '2022-02-21', '2026-02-14'),
(21, 21, 9000, 4.5, '2022-01-15', '2025-01-09'),
(22, 22, 17000, 4.5, '2023-04-24', '2026-04-17'),
(23, 23, 17000, 5.5, '2022-05-10', '2026-05-04'),
(24, 24, 12000, 4.5, '2023-02-14', '2026-02-07'),
(25, 25, 20000, 5.5, '2023-02-15', '2027-02-08'),
(26, 26, 9000, 6.5, '2024-06-15', '2027-06-08'),
(27, 27, 12000, 5.0, '2022-09-19', '2026-09-13'),
(28, 28, 9000, 5.5, '2023-03-16', '2026-03-09'),
(29, 29, 12000, 6.0, '2023-09-11', '2026-09-04'),
(30, 30, 9000, 6.0, '2024-07-23', '2028-07-15'),
(31, 31, 15000, 6.0, '2023-09-08', '2027-09-01'),
(32, 32, 15000, 4.5, '2024-04-06', '2027-03-30'),
(33, 33, 20000, 6.5, '2023-07-07', '2026-06-30'),
(34, 34, 12000, 6.5, '2023-01-27', '2026-01-20'),
(35, 35, 10000, 4.5, '2022-09-10', '2026-09-03'),
(36, 36, 9000, 4.5, '2022-03-17', '2026-03-11'),
(37, 37, 15000, 4.5, '2023-11-08', '2026-11-01'),
(38, 38, 10000, 6.0, '2022-04-10', '2026-04-04'),
(39, 39, 9000, 6.5, '2023-04-19', '2026-04-12'),
(40, 40, 12000, 6.5, '2022-01-03', '2026-12-28'),
(41, 41, 10000, 6.0, '2022-10-18', '2025-10-12'),
(42, 42, 12000, 4.5, '2024-03-16', '2028-03-08'),
```


(43, 43, 17000, 5.5, '2022-03-14', '2026-03-08'),
(44, 44, 17000, 6.0, '2024-02-13', '2028-02-06'),
(45, 45, 10000, 5.0, '2023-08-06', '2027-07-30'),
(46, 46, 20000, 6.0, '2022-03-30', '2025-03-24'),
(47, 47, 9000, 5.0, '2022-10-30', '2026-10-24'),
(48, 48, 11000, 6.5, '2024-02-01', '2027-01-25'),
(49, 49, 17000, 5.0, '2022-06-15', '2026-06-09'),
(50, 50, 11000, 5.5, '2024-06-01', '2028-05-25'),
(51, 1, 18000, 6.5, '2024-01-10', '2028-01-03'),
(52, 2, 9500, 5.0, '2023-07-02', '2027-06-25'),
(53, 3, 14000, 5.5, '2023-12-24', '2026-12-17'),
(54, 4, 11000, 6.0, '2023-02-18', '2026-02-11'),
(55, 5, 13000, 5.0, '2022-11-25', '2025-11-18'),
(56, 6, 15000, 4.5, '2022-10-09', '2026-10-02'),
(57, 7, 9000, 4.5, '2024-02-17', '2027-02-10'),
(58, 8, 10000, 5.0, '2023-01-06', '2026-12-30'),
(59, 9, 12000, 6.0, '2022-07-19', '2025-07-13'),
(60, 10, 16000, 6.5, '2023-03-29', '2027-03-22'),
(61, 11, 19000, 6.0, '2022-08-21', '2025-08-15'),
(62, 12, 18000, 5.0, '2024-04-27', '2027-04-20'),
(63, 13, 20000, 6.5, '2022-06-10', '2026-06-03'),
(64, 14, 17000, 5.5, '2023-10-01', '2027-09-24'),
(65, 15, 9500, 4.5, '2022-04-20', '2026-04-13'),
(66, 16, 10000, 5.5, '2024-05-06', '2027-04-29'),
(67, 17, 12000, 5.0, '2022-11-14', '2025-11-07'),
(68, 18, 13000, 6.0, '2022-09-25', '2026-09-18'),
(69, 19, 9000, 6.5, '2023-12-02', '2027-11-25'),
(70, 20, 16000, 5.5, '2022-08-10', '2026-08-03'),
(71, 21, 17000, 6.0, '2023-06-08', '2026-06-01'),
(72, 22, 9500, 4.5, '2023-09-11', '2027-09-04'),
(73, 23, 15000, 5.5, '2022-02-19', '2026-02-12'),
(74, 24, 10000, 6.0, '2024-01-26', '2028-01-19'),
(75, 25, 13000, 5.0, '2023-05-31', '2027-05-24'),
(76, 26, 15000, 4.5, '2022-07-07', '2025-06-30'),
(77, 27, 9000, 4.5, '2024-03-03', '2027-02-24'),
(78, 28, 10000, 6.0, '2022-01-18', '2026-01-11'),
(79, 29, 18000, 6.5, '2023-11-05', '2027-10-29'),
(80, 30, 14000, 5.5, '2023-04-11', '2026-04-04'),
(81, 31, 9500, 6.0, '2024-06-21', '2028-06-13'),
(82, 32, 13000, 6.0, '2022-05-05', '2026-04-28'),
(83, 33, 20000, 5.0, '2023-10-12', '2026-10-05'),
(84, 34, 12000, 4.5, '2023-08-16', '2027-08-09'),
(85, 35, 15000, 5.0, '2022-03-27', '2026-03-20'),
(86, 36, 17000, 5.5, '2023-07-29', '2026-07-22'),
(87, 37, 14000, 6.0, '2022-10-23', '2025-10-17'),
(88, 38, 12000, 4.5, '2024-02-10', '2028-02-03'),
(89, 39, 9000, 6.0, '2023-01-30', '2026-01-23'),
(90, 40, 10000, 5.5, '2022-06-03', '2026-05-27'),
(91, 41, 15000, 6.5, '2022-12-09', '2026-12-02'),

```
(92, 42, 10000, 6.0, '2023-06-17', '2026-06-10'),
(93, 43, 20000, 5.0, '2024-05-01', '2027-04-24'),
(94, 44, 17000, 5.5, '2023-02-05', '2027-01-29'),
(95, 45, 18000, 4.5, '2022-09-14', '2025-09-07'),
(96, 46, 9000, 5.0, '2023-03-15', '2026-03-08'),
(97, 47, 13000, 6.0, '2023-05-19', '2026-05-12'),
(98, 48, 15000, 6.5, '2022-01-11', '2026-01-04'),
(99, 49, 9500, 4.5, '2022-08-07', '2026-07-31'),
(100, 50, 17000, 5.0, '2023-12-20', '2027-12-13'),
(101, 1, 16000, 5.5, '2022-10-22', '2026-10-15'),
(102, 2, 13000, 6.0, '2023-06-28', NULL),
(103, 3, 12000, 5.0, '2022-12-01', '2026-11-24'),
(104, 4, 9500, 4.5, '2023-08-05', '2027-07-29'),
(105, 5, 17000, 6.5, '2022-05-03', NULL),
(106, 6, 11000, 6.0, '2023-02-14', '2026-02-07'),
(107, 7, 10000, 4.5, '2023-04-01', NULL),
(108, 8, 14000, 5.5, '2022-01-06', '2025-12-30'),
(109, 9, 12000, 6.0, '2023-03-25', '2027-03-18'),
(110, 10, 15000, 5.0, '2024-01-30', NULL),
(111, 11, 18000, 5.5, '2022-09-12', '2026-09-05'),
(112, 12, 13000, 6.0, '2023-07-18', NULL),
(113, 13, 9000, 4.5, '2022-06-09', '2026-06-02'),
(114, 14, 17000, 6.5, '2023-11-03', '2027-10-27'),
(115, 15, 16000, 5.5, '2022-08-22', NULL),
(116, 16, 15000, 5.0, '2023-02-23', '2026-02-16'),
(117, 17, 9500, 6.0, '2022-03-14', NULL),
(118, 18, 10000, 4.5, '2024-06-11', '2027-06-04'),
(119, 19, 12000, 6.0, '2023-09-04', '2027-08-28'),
(120, 20, 13000, 5.5, '2022-04-18', NULL),
(121, 21, 9000, 5.0, '2023-01-03', '2026-12-27'),
(122, 22, 17000, 4.5, '2022-07-27', NULL),
(123, 23, 18000, 6.0, '2024-05-13', '2028-05-06'),
(124, 24, 15000, 6.5, '2023-06-05', NULL),
(125, 25, 10000, 5.0, '2022-10-26', '2026-10-19'),
(126, 26, 9500, 5.5, '2023-03-08', NULL),
(127, 27, 11000, 6.0, '2022-12-18', '2026-12-11'),
(128, 28, 14000, 5.0, '2022-05-22', NULL),
(129, 29, 20000, 6.0, '2023-10-07', '2027-09-30'),
(130, 30, 17000, 5.5, '2022-01-13', NULL),
(131, 31, 15000, 4.5, '2024-02-02', '2027-01-26'),
(132, 32, 10000, 6.0, '2023-07-14', NULL),
(133, 33, 9500, 5.5, '2022-02-11', '2026-02-04'),
(134, 34, 11000, 6.5, '2022-08-31', NULL),
(135, 35, 13000, 6.0, '2023-12-13', '2027-12-06'),
(136, 36, 12000, 5.0, '2022-06-20', NULL),
(137, 37, 18000, 6.5, '2023-02-12', '2026-02-05'),
(138, 38, 9000, 5.0, '2022-03-22', NULL),
(139, 39, 17000, 6.0, '2024-04-04', '2028-03-28'),
(140, 40, 15000, 5.5, '2023-05-17', NULL),
```

```
(141, 41, 10000, 6.5, '2022-09-28', '2026-09-21'),
(142, 42, 14000, 6.0, '2023-08-08', NULL),
(143, 43, 9500, 5.0, '2022-12-04', '2026-11-27'),
(144, 44, 11000, 6.0, '2022-07-03', NULL),
(145, 45, 13000, 4.5, '2024-01-01', '2027-12-25'),
(146, 46, 12000, 5.0, '2022-11-09', NULL),
(147, 47, 10000, 5.5, '2023-09-06', '2027-08-30'),
(148, 48, 16000, 6.0, '2022-05-17', NULL),
(149, 49, 18000, 6.5, '2023-11-30', '2027-11-23'),
(150, 50, 17000, 5.5, '2022-02-27', NULL);
```

➤ Tabla **garantía créditos:**

```
--Tabla garatia_credito
use BDD_creditos_vehiculares
select * from garantia_credito
INSERT INTO garantia_credito (id_credito, id_garantia, fecha_asignacion,
valor_asegurado) VALUES
(1, 1, '2024-01-15', 10500.50),
(2, 2, '2024-02-10', 15000.00),
(3, 3, '2024-03-05', 17500.75),
(4, 4, '2024-04-20', 9800.00),
(5, 5, '2024-05-15', 13450.25),
(6, 6, '2024-06-12', 19000.00),
(7, 7, '2024-07-08', 14250.00),
(8, 8, '2024-08-23', 16000.80),
(9, 9, '2024-09-10', 8000.00),
(10, 10, '2024-10-02', 12000.00),
(11, 11, '2024-10-25', 17000.00),
(12, 12, '2024-11-15', 18500.00),
(13, 13, '2024-12-05', 15000.00),
(14, 14, '2025-01-10', 19500.00),
(15, 15, '2025-02-18', 11000.00),
(16, 16, '2025-03-25', 12500.00),
(17, 17, '2025-04-12', 14000.00),
(18, 18, '2025-05-05', 13000.00),
(19, 19, '2025-05-20', 15500.00),
(20, 20, '2025-05-30', 16500.00),
(21, 21, '2024-01-07', 11500.00),
(22, 22, '2024-02-14', 12500.00),
(23, 23, '2024-03-18', 13500.00),
(24, 24, '2024-04-22', 14500.00),
(25, 25, '2024-05-26', 15500.00),
(26, 26, '2024-06-30', 16500.00),
(27, 27, '2024-07-04', 17500.00),
(28, 28, '2024-08-08', 18500.00),
(29, 29, '2024-09-12', 19500.00),
(30, 30, '2024-10-16', 20500.00),
```

```
(31, 31, '2024-11-20', 15000.00),
(32, 32, '2024-12-24', 15500.00),
(33, 33, '2025-01-28', 16000.00),
(34, 34, '2025-02-01', 16500.00),
(35, 35, '2025-03-05', 17000.00),
(36, 36, '2025-04-09', 17500.00),
(37, 37, '2025-05-13', 18000.00),
(38, 38, '2024-01-17', 18500.00),
(39, 39, '2024-02-21', 19000.00),
(40, 40, '2024-03-27', 19500.00),
(41, 41, '2024-04-30', 20000.00),
(42, 42, '2024-05-04', 20500.00),
(43, 43, '2024-06-08', 21000.00),
(44, 44, '2024-07-12', 21500.00),
(45, 45, '2024-08-16', 22000.00),
(46, 46, '2024-09-20', 22500.00),
(47, 47, '2024-10-24', 23000.00),
(48, 48, '2024-11-28', 23500.00),
(49, 49, '2024-12-02', 24000.00),
(50, 50, '2025-01-06', 24500.00),
(51, 2, '2025-02-10', 25000.00),
(52, 3, '2025-03-16', 25500.00),
(53, 4, '2025-04-20', 26000.00),
(54, 5, '2025-05-24', 26500.00),
(55, 6, '2024-01-28', 27000.00),
(56, 7, '2024-02-02', 27500.00),
(57, 8, '2024-03-08', 28000.00),
(58, 9, '2024-04-12', 28500.00),
(59, 10, '2024-05-16', 29000.00),
(60, 11, '2024-06-20', 29500.00),
(61, 12, '2024-07-24', 30000.00),
(62, 13, '2024-08-28', 30500.00),
(63, 14, '2024-09-01', 31000.00),
(64, 15, '2024-10-05', 31500.00),
(65, 16, '2024-11-09', 32000.00),
(66, 17, '2024-12-13', 32500.00),
(67, 18, '2025-01-17', 33000.00),
(68, 19, '2025-02-21', 33500.00),
(69, 20, '2025-03-27', 34000.00),
(70, 21, '2025-04-30', 34500.00),
(71, 22, '2024-01-03', 35000.00),
(72, 23, '2024-02-07', 35500.00),
(73, 24, '2024-03-13', 36000.00),
(74, 25, '2024-04-17', 36500.00),
(75, 26, '2024-05-21', 37000.00),
(76, 27, '2024-06-25', 37500.00),
(77, 28, '2024-07-29', 38000.00),
(78, 29, '2024-08-02', 38500.00),
(79, 30, '2024-09-06', 39000.00),
```

```
(80, 31, '2024-10-10', 39500.00),
(81, 32, '2024-11-14', 40000.00),
(82, 33, '2024-12-18', 40500.00),
(83, 34, '2025-01-22', 41000.00),
(84, 35, '2025-02-26', 41500.00),
(85, 36, '2025-03-02', 42000.00),
(86, 37, '2025-04-06', 42500.00),
(87, 38, '2025-05-10', 43000.00),
(88, 39, '2024-01-14', 43500.00),
(89, 40, '2024-02-18', 44000.00),
(90, 41, '2024-03-24', 44500.00),
(91, 42, '2024-04-28', 45000.00),
(92, 43, '2024-05-02', 45500.00),
(93, 44, '2024-06-06', 46000.00),
(94, 45, '2024-07-10', 46500.00),
(95, 46, '2024-08-14', 47000.00),
(96, 47, '2024-09-18', 47500.00),
(97, 48, '2024-10-22', 48000.00),
(98, 49, '2024-11-26', 48500.00),
(99, 50, '2024-12-30', 49000.00),
(100, 1, '2025-01-03', 49500.00),
(101, 3, '2025-02-07', 50000.00),
(102, 5, '2025-03-13', 50500.00),
(103, 7, '2025-04-17', 51000.00),
(104, 9, '2025-05-21', 51500.00),
(105, 11, '2024-01-25', 52000.00),
(106, 13, '2024-02-28', 52500.00),
(107, 15, '2024-03-05', 53000.00),
(108, 17, '2024-04-09', 53500.00),
(109, 19, '2024-05-13', 54000.00),
(110, 21, '2024-06-17', 54500.00),
(111, 23, '2024-07-21', 55000.00),
(112, 25, '2024-08-25', 55500.00),
(113, 27, '2024-09-29', 56000.00),
(114, 29, '2024-10-03', 56500.00),
(115, 31, '2024-11-07', 57000.00),
(116, 33, '2024-12-11', 57500.00),
(117, 35, '2025-01-15', 58000.00),
(118, 37, '2025-02-19', 58500.00),
(119, 39, '2025-03-25', 59000.00),
(120, 41, '2025-04-29', 59500.00),
(121, 43, '2024-01-02', 60000.00),
(122, 45, '2024-02-06', 60500.00),
(123, 47, '2024-03-12', 61000.00),
(124, 49, '2024-04-16', 61500.00),
(125, 1, '2024-05-20', 62000.00),
(126, 2, '2024-06-24', 62500.00),
(127, 4, '2024-07-28', 63000.00),
(128, 6, '2024-08-01', 63500.00),
```

```
(129, 8, '2024-09-05', 64000.00),
(130, 10, '2024-10-09', 64500.00),
(131, 12, '2024-11-13', 65000.00),
(132, 14, '2024-12-17', 65500.00),
(133, 16, '2025-01-21', 66000.00),
(134, 18, '2025-02-25', 66500.00),
(135, 20, '2025-03-31', 67000.00),
(136, 22, '2025-04-04', 67500.00),
(137, 24, '2025-05-08', 68000.00),
(138, 26, '2024-01-12', 68500.00),
(139, 28, '2024-02-16', 69000.00),
(140, 30, '2024-03-22', 69500.00),
(141, 32, '2024-04-26', 70000.00),
(142, 34, '2024-05-30', 70500.00),
(143, 36, '2024-06-03', 71000.00),
(144, 38, '2024-07-07', 71500.00),
(145, 40, '2024-08-11', 72000.00),
(146, 42, '2024-09-15', 72500.00),
(147, 44, '2024-10-19', 73000.00),
(148, 46, '2024-11-23', 73500.00),
(149, 48, '2024-12-27', 74000.00),
(150, 2, '2025-01-31', 74500.00);
```

➤ Tabla **sucursal**:

```
--Tabla sucursal
SELECT * FROM sucursal;
INSERT INTO sucursal (id_concesionario, nombre, direccion) VALUES
(1, 'Sucursal Central', 'Av. Siempre Viva 101'),
(2, 'Sucursal Norte', 'Calle Falsa 123'),
(3, 'Sucursal Sur', 'Jr. Los Olivos 456'),
(4, 'Sucursal Este', 'Psj. Las Rosas 789'),
(5, 'Sucursal Oeste', 'Av. Libertad 111'),
(6, 'Sucursal Metropolitana', 'Calle Unión 222'),
(7, 'Sucursal Express', 'Jr. Primavera 333'),
(8, 'Sucursal Industrial', 'Av. Perú 444'),
(9, 'Sucursal Ejecutiva', 'Calle Sol 555'),
(10, 'Sucursal Andina', 'Av. América 666'),
(11, 'Sucursal Pacífico', 'Jr. Amazonas 777'),
(12, 'Sucursal Sierra', 'Calle Luna 888'),
(13, 'Sucursal Selva', 'Av. Mar 999'),
(14, 'Sucursal Miraflores', 'Jr. Cielo 100'),
(15, 'Sucursal San Isidro', 'Psj. Estrella 200'),
(16, 'Sucursal Surco', 'Av. Los Pinos 300'),
(17, 'Sucursal La Molina', 'Calle 9 de Octubre 400'),
(18, 'Sucursal Callao', 'Jr. Ayacucho 500'),
(19, 'Sucursal Arequipa', 'Av. Grau 600'),
(20, 'Sucursal Cusco', 'Calle Arequipa 700'),
```

(21, 'Sucursal Trujillo', 'Av. La Marina 800'),
(22, 'Sucursal Chiclayo', 'Jr. Cusco 900'),
(23, 'Sucursal Piura', 'Psj. Loreto 1000'),
(24, 'Sucursal Tacna', 'Calle Trujillo 1100'),
(25, 'Sucursal Juliaca', 'Av. Tarapoto 1200'),
(26, 'Sucursal Iquitos', 'Av. San Martín 1300'),
(27, 'Sucursal Huancayo', 'Calle Amazonas 1400'),
(28, 'Sucursal Pucallpa', 'Jr. Puno 1500'),
(29, 'Sucursal Ayacucho', 'Psj. Lima 1600'),
(30, 'Sucursal Cajamarca', 'Av. Callao 1700'),
(31, 'Sucursal Ate', 'Av. Brasil 1800'),
(32, 'Sucursal Comas', 'Calle Argentina 1900'),
(33, 'Sucursal San Miguel', 'Jr. Bolívar 2000'),
(34, 'Sucursal Barranco', 'Psj. Sucre 2100'),
(35, 'Sucursal Chorrillos', 'Av. Venezuela 2200'),
(36, 'Sucursal Villa El Salvador', 'Av. Javier Prado 2300'),
(37, 'Sucursal San Borja', 'Calle Angamos 2400'),
(38, 'Sucursal Jesús María', 'Jr. Benavides 2500'),
(39, 'Sucursal Magdalena', 'Psj. Canadá 2600'),
(40, 'Sucursal Breña', 'Av. Colonial 2700'),
(41, 'Sucursal Los Olivos', 'Av. República 2800'),
(42, 'Sucursal Rímac', 'Calle Europa 2900'),
(43, 'Sucursal Pueblo Libre', 'Jr. Asia 3000'),
(44, 'Sucursal El Agustino', 'Psj. África 3100'),
(45, 'Sucursal La Victoria', 'Av. Oceanía 3200'),
(46, 'Sucursal Lince', 'Av. Inca 3300'),
(47, 'Sucursal Independencia', 'Calle Misti 3400'),
(48, 'Sucursal San Juan', 'Jr. Túpac Amaru 3500'),
(49, 'Sucursal Ventanilla', 'Psj. Pachacútec 3600'),
(50, 'Sucursal SMP', 'Av. Huayna Cápac 3700'),
(2, 'Sucursal Central', 'Calle Falsa 123'),
(3, 'Sucursal Norte', 'Jr. Los Olivos 456'),
(4, 'Sucursal Sur', 'Psj. Las Rosas 789'),
(5, 'Sucursal Este', 'Av. Libertad 111'),
(6, 'Sucursal Oeste', 'Calle Unión 222'),
(7, 'Sucursal Metropolitana', 'Jr. Primavera 333'),
(8, 'Sucursal Express', 'Av. Perú 444'),
(9, 'Sucursal Industrial', 'Calle Sol 555'),
(10, 'Sucursal Ejecutiva', 'Av. América 666'),
(11, 'Sucursal Andina', 'Jr. Amazonas 777'),
(12, 'Sucursal Pacífico', 'Calle Luna 888'),
(13, 'Sucursal Sierra', 'Av. Mar 999'),
(14, 'Sucursal Selva', 'Jr. Cielo 100'),
(15, 'Sucursal Miraflores', 'Psj. Estrella 200'),
(16, 'Sucursal San Isidro', 'Av. Los Pinos 300'),
(17, 'Sucursal Surco', 'Calle 9 de Octubre 400'),
(18, 'Sucursal La Molina', 'Jr. Ayacucho 500'),
(19, 'Sucursal Callao', 'Av. Grau 600'),
(20, 'Sucursal Arequipa', 'Calle Arequipa 700'),

(21, 'Sucursal Cusco', 'Av. La Marina 800'),
(22, 'Sucursal Trujillo', 'Jr. Cusco 900'),
(23, 'Sucursal Chiclayo', 'Psj. Loreto 1000'),
(24, 'Sucursal Piura', 'Calle Trujillo 1100'),
(25, 'Sucursal Tacna', 'Av. Tarapoto 1200'),
(26, 'Sucursal Juliaca', 'Av. San Martín 1300'),
(27, 'Sucursal Iquitos', 'Calle Amazonas 1400'),
(28, 'Sucursal Huancayo', 'Jr. Puno 1500'),
(29, 'Sucursal Pucallpa', 'Psj. Lima 1600'),
(30, 'Sucursal Ayacucho', 'Av. Callao 1700'),
(31, 'Sucursal Cajamarca', 'Av. Brasil 1800'),
(32, 'Sucursal Ate', 'Calle Argentina 1900'),
(33, 'Sucursal Comas', 'Jr. Bolívar 2000'),
(34, 'Sucursal San Miguel', 'Psj. Sucre 2100'),
(35, 'Sucursal Barranco', 'Av. Venezuela 2200'),
(36, 'Sucursal Chorrillos', 'Av. Javier Prado 2300'),
(37, 'Sucursal Villa El Salvador', 'Calle Angamos 2400'),
(38, 'Sucursal San Borja', 'Jr. Benavides 2500'),
(39, 'Sucursal Jesús María', 'Psj. Canadá 2600'),
(40, 'Sucursal Magdalena', 'Av. Colonial 2700'),
(41, 'Sucursal Breña', 'Av. República 2800'),
(42, 'Sucursal Los Olivos', 'Calle Europa 2900'),
(43, 'Sucursal Rímac', 'Jr. Asia 3000'),
(44, 'Sucursal Pueblo Libre', 'Psj. África 3100'),
(45, 'Sucursal El Agustino', 'Av. Oceanía 3200'),
(46, 'Sucursal La Victoria', 'Av. Inca 3300'),
(47, 'Sucursal Lince', 'Calle Misti 3400'),
(48, 'Sucursal Independencia', 'Jr. Túpac Amaru 3500'),
(49, 'Sucursal San Juan', 'Psj. Pachacútec 3600'),
(50, 'Sucursal Ventanilla', 'Av. Huayna Cápac 3700'),
(1, 'Sucursal SMP', 'Av. Siempre Viva 101'),
(3, 'Sucursal Central', 'Jr. Los Olivos 456'),
(5, 'Sucursal Norte', 'Psj. Las Rosas 789'),
(7, 'Sucursal Sur', 'Calle Unión 222'),
(9, 'Sucursal Este', 'Av. Perú 444'),
(11, 'Sucursal Oeste', 'Calle Sol 555'),
(13, 'Sucursal Metropolitana', 'Jr. Amazonas 777'),
(15, 'Sucursal Express', 'Av. Mar 999'),
(17, 'Sucursal Industrial', 'Psj. Estrella 200'),
(19, 'Sucursal Ejecutiva', 'Calle 9 de Octubre 400'),
(21, 'Sucursal Andina', 'Av. Grau 600'),
(23, 'Sucursal Pacífico', 'Calle Arequipa 700'),
(25, 'Sucursal Sierra', 'Av. La Marina 800'),
(27, 'Sucursal Selva', 'Jr. Cusco 900'),
(29, 'Sucursal Miraflores', 'Psj. Loreto 1000'),
(31, 'Sucursal San Isidro', 'Calle Trujillo 1100'),
(33, 'Sucursal Surco', 'Av. Tarapoto 1200'),
(35, 'Sucursal La Molina', 'Av. San Martín 1300'),
(37, 'Sucursal Callao', 'Calle Amazonas 1400'),


```

(39, 'Sucursal Arequipa', 'Jr. Puno 1500'),
(41, 'Sucursal Cusco', 'Psj. Lima 1600'),
(43, 'Sucursal Trujillo', 'Av. Callao 1700'),
(45, 'Sucursal Chiclayo', 'Av. Brasil 1800'),
(47, 'Sucursal Piura', 'Calle Argentina 1900'),
(49, 'Sucursal Tacna', 'Jr. Bolívar 2000'),
(2, 'Sucursal Juliaca', 'Psj. Sucre 2100'),
(4, 'Sucursal Iquitos', 'Av. Venezuela 2200'),
(6, 'Sucursal Huancayo', 'Av. Javier Prado 2300'),
(8, 'Sucursal Pucallpa', 'Calle Angamos 2400'),
(10, 'Sucursal Ayacucho', 'Jr. Benavides 2500'),
(12, 'Sucursal Cajamarca', 'Psj. Canadá 2600'),
(14, 'Sucursal Ate', 'Av. Colonial 2700'),
(16, 'Sucursal Comas', 'Av. República 2800'),
(18, 'Sucursal San Miguel', 'Calle Europa 2900'),
(20, 'Sucursal Barranco', 'Jr. Asia 3000'),
(22, 'Sucursal Chorrillos', 'Psj. África 3100'),
(24, 'Sucursal Villa El Salvador', 'Av. Oceanía 3200'),
(26, 'Sucursal San Borja', 'Av. Inca 3300'),
(28, 'Sucursal Jesús María', 'Calle Misti 3400'),
(30, 'Sucursal Magdalena', 'Jr. Túpac Amaru 3500'),
(32, 'Sucursal Breña', 'Psj. Pachacútec 3600'),
(34, 'Sucursal Los Olivos', 'Av. Huayna Cápac 3700'),
(36, 'Sucursal Rímac', 'Av. Siempre Viva 101'),
(38, 'Sucursal Pueblo Libre', 'Av. Libertad 111'),
(40, 'Sucursal El Agustino', 'Av. América 666'),
(42, 'Sucursal La Victoria', 'Av. La Marina 800'),
(44, 'Sucursal Lince', 'Psj. Loreto 1000'),
(46, 'Sucursal Independencia', 'Psj. África 3100'),
(48, 'Sucursal San Juan', 'Psj. Canadá 2600'),
(50, 'Sucursal Ventanilla', 'Calle Falsa 123'),
(2, 'Sucursal SMP', 'Jr. Cielo 100');

```

➤ Tabla **producto_sucursal**:

```

--Tabla producto_sucursal
TRUNCATE TABLE producto_sucursal;
SELECT * FROM producto_sucursal;
INSERT INTO producto_sucursal (id_producto, id_sucursal, stock) VALUES
(1, 1, 45),
(2, 2, 30),
(3, 3, 76),
(4, 4, 12),
(5, 5, 59),
(6, 6, 80),
(7, 7, 22),
(8, 8, 15),
(9, 9, 50),

```

(10, 10, 90),
(11, 11, 33),
(12, 12, 67),
(13, 13, 18),
(14, 14, 41),
(15, 15, 29),
(16, 16, 60),
(17, 17, 55),
(18, 18, 38),
(19, 19, 74),
(20, 20, 85),
(21, 21, 47),
(22, 22, 69),
(23, 23, 26),
(24, 24, 93),
(25, 25, 34),
(26, 26, 11),
(27, 27, 72),
(28, 28, 44),
(29, 29, 57),
(30, 30, 20),
(31, 31, 79),
(32, 32, 48),
(33, 33, 66),
(34, 34, 27),
(35, 35, 54),
(36, 36, 39),
(37, 37, 83),
(38, 38, 31),
(39, 39, 61),
(40, 40, 23),
(41, 41, 78),
(42, 42, 49),
(43, 43, 62),
(44, 44, 25),
(45, 45, 53),
(46, 46, 37),
(47, 47, 84),
(48, 48, 32),
(49, 49, 65),
(50, 50, 21),
(1, 51, 58),
(2, 52, 16),
(3, 53, 73),
(4, 54, 42),
(5, 55, 35),
(6, 56, 68),
(7, 57, 24),
(8, 58, 51),

(9, 59, 77),
(10, 60, 19),
(11, 61, 36),
(12, 62, 52),
(13, 63, 13),
(14, 64, 70),
(15, 65, 28),
(16, 66, 64),
(17, 67, 40),
(18, 68, 81),
(19, 69, 22),
(20, 70, 56),
(21, 71, 14),
(22, 72, 75),
(23, 73, 34),
(24, 74, 63),
(25, 75, 27),
(26, 76, 48),
(27, 77, 17),
(28, 78, 60),
(29, 79, 39),
(30, 80, 53),
(31, 81, 21),
(32, 82, 69),
(33, 83, 26),
(34, 84, 72),
(35, 85, 15),
(36, 86, 50),
(37, 87, 33),
(38, 88, 76),
(39, 89, 44),
(40, 90, 62),
(41, 91, 19),
(42, 92, 58),
(43, 93, 24),
(44, 94, 67),
(45, 95, 35),
(46, 96, 79),
(47, 97, 23),
(48, 98, 55),
(49, 99, 12),
(50, 100, 70),
(1, 101, 29),
(2, 102, 65),
(3, 103, 41),
(4, 104, 53),
(5, 105, 18),
(6, 106, 74),
(7, 107, 30),

(8, 108, 62),
(9, 109, 14),
(10, 110, 47),
(11, 111, 21),
(12, 112, 56),
(13, 113, 34),
(14, 114, 68),
(15, 115, 25),
(16, 116, 79),
(17, 117, 38),
(18, 118, 50),
(19, 119, 27),
(20, 120, 61),
(21, 121, 17),
(22, 122, 43),
(23, 123, 75),
(24, 124, 29),
(25, 125, 54),
(26, 126, 33),
(27, 127, 69),
(28, 128, 22),
(29, 129, 58),
(30, 130, 14),
(31, 131, 65),
(32, 132, 37),
(33, 133, 49),
(34, 134, 25),
(35, 135, 71),
(36, 136, 30),
(37, 137, 56),
(38, 138, 22),
(39, 139, 63),
(40, 140, 19),
(41, 141, 54),
(42, 142, 33),
(43, 143, 67),
(44, 144, 29),
(45, 145, 71),
(46, 146, 40),
(47, 147, 50),
(48, 148, 18),
(49, 149, 62),
(50, 150, 27);

7) Anexos

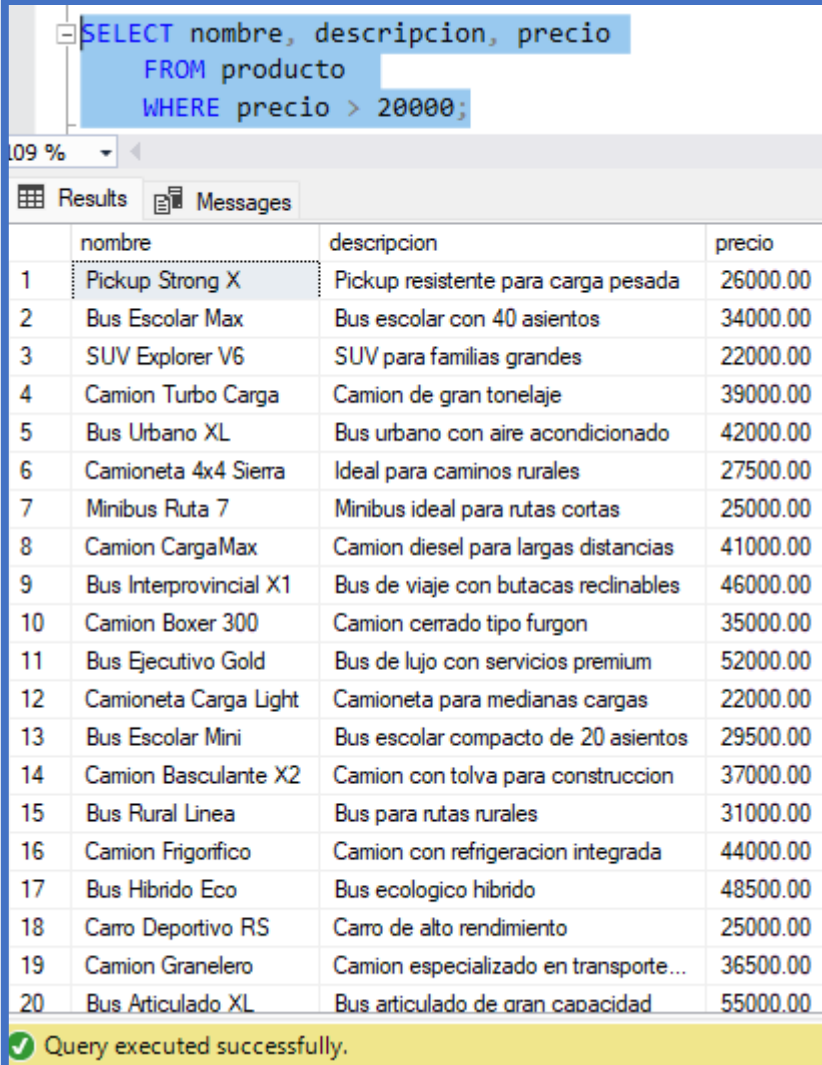
❖ Ejercicios

--LISTA DE EJERCICIOS

Ejercicio N°1

Obtener el nombre y descripción de los productos cuyo precio es mayor a 20,000.

```
SELECT*FROM producto;  
  
SELECT nombre, descripcion, precio  
FROM producto  
WHERE precio > 20000;
```



The screenshot shows a database query interface. At the top, the SQL query is entered in a text box: `SELECT nombre, descripcion, precio FROM producto WHERE precio > 20000;`. Below the query box, there are tabs for 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with 20 rows of data. The table has four columns: 'nombre', 'descripcion', and 'precio'. The 'precio' column is highlighted in blue. At the bottom of the interface, a green checkmark icon and the text 'Query executed successfully.' are visible.

	nombre	descripcion	precio
1	Pickup Strong X	Pickup resistente para carga pesada	26000.00
2	Bus Escolar Max	Bus escolar con 40 asientos	34000.00
3	SUV Explorer V6	SUV para familias grandes	22000.00
4	Camion Turbo Carga	Camion de gran tonelaje	39000.00
5	Bus Urbano XL	Bus urbano con aire acondicionado	42000.00
6	Camioneta 4x4 Sierra	Ideal para caminos rurales	27500.00
7	Minibus Ruta 7	Minibus ideal para rutas cortas	25000.00
8	Camion CargaMax	Camion diesel para largas distancias	41000.00
9	Bus Interprovincial X1	Bus de viaje con butacas reclinables	46000.00
10	Camion Boxer 300	Camion cerrado tipo furgon	35000.00
11	Bus Ejecutivo Gold	Bus de lujo con servicios premium	52000.00
12	Camioneta Carga Light	Camioneta para medianas cargas	22000.00
13	Bus Escolar Mini	Bus escolar compacto de 20 asientos	29500.00
14	Camion Basculante X2	Camion con tolva para construccion	37000.00
15	Bus Rural Linea	Bus para rutas rurales	31000.00
16	Camion Frigorifico	Camion con refrigeracion integrada	44000.00
17	Bus Hibrido Eco	Bus ecologico hibrido	48500.00
18	Carro Deportivo RS	Carro de alto rendimiento	25000.00
19	Camion Granelero	Camion especializado en transporte...	36500.00
20	Bus Articulado XL	Bus articulado de gran capacidad	55000.00

✓ Query executed successfully.

Ejercicio N°2

Mostrar los créditos con monto mayor a 5000.

```
SELECT * FROM credito
WHERE monto > 5000;
```

--Ejercicio N°2
/* Mostrar los créditos con monto mayor a 50,000. */
SELECT * FROM credito
WHERE monto > 5000;

109 %

Results Messages

	id_credito	id_cuenta	id_producto	monto	interes	fecha_inicio	fecha_vencimiento
1	1	1	1	17000.00	6.00	2023-07-24	2027-07-17
2	2	2	2	10000.00	5.50	2024-05-23	2027-05-14
3	3	3	3	9000.00	6.50	2022-05-10	2025-05-04
4	4	4	4	15000.00	6.00	2022-09-13	2026-09-06
5	5	5	5	20000.00	6.00	2023-08-19	2027-08-12
6	6	6	6	10000.00	4.50	2023-04-23	2026-04-16
7	7	7	7	11000.00	4.50	2022-02-26	2026-02-19
8	8	8	8	17000.00	4.50	2023-03-03	2026-02-24
9	9	9	9	10000.00	6.00	2022-01-09	2026-01-02
10	10	10	10	20000.00	4.50	2022-09-08	2025-09-02
11	11	11	11	17000.00	5.50	2022-01-14	2025-01-08
12	12	12	12	15000.00	6.00	2022-01-12	2025-01-06
13	13	13	13	10000.00	6.00	2023-10-21	2026-10-14
14	14	14	14	9000.00	4.50	2023-03-18	2027-03-11
15	15	15	15	11000.00	6.00	2022-02-15	2026-02-08
16	16	16	16	17000.00	5.50	2022-07-08	2025-07-02
17	17	17	17	20000.00	5.00	2022-03-03	2025-02-25
18	18	18	18	11000.00	6.00	2023-06-23	2026-06-16
19	19	19	19	12000.00	5.00	2024-03-14	2027-03-07
20	20	20	20	12000.00	5.00	2022-02-21	2026-02-14

✓ Query executed successfully.

Ejercicio N°3

Listar todos los créditos con el interés aplicado mayor a 4%.

```
SELECT * FROM credito
WHERE interes > 4;
```

--Ejercicio N°3
/* Listar todos los créditos con el interés aplicado mayor a 4%.*/
SELECT * FROM credito
WHERE interes > 4;

09 %

Results Messages

	id_credito	id_cuenta	id_producto	monto	interes	fecha_inicio	fecha_vencimiento
1	1	1	1	17000.00	6.00	2023-07-24	2027-07-17
2	2	2	2	10000.00	5.50	2024-05-23	2027-05-14
3	3	3	3	9000.00	6.50	2022-05-10	2025-05-04
4	4	4	4	15000.00	6.00	2022-09-13	2026-09-06
5	5	5	5	20000.00	6.00	2023-08-19	2027-08-12
6	6	6	6	10000.00	4.50	2023-04-23	2026-04-16
7	7	7	7	11000.00	4.50	2022-02-26	2026-02-19
8	8	8	8	17000.00	4.50	2023-03-03	2026-02-24
9	9	9	9	10000.00	6.00	2022-01-09	2026-01-02
10	10	10	10	20000.00	4.50	2022-09-08	2025-09-02
11	11	11	11	17000.00	5.50	2022-01-14	2025-01-08
12	12	12	12	15000.00	6.00	2022-01-12	2025-01-06
13	13	13	13	10000.00	6.00	2023-10-21	2026-10-14
14	14	14	14	9000.00	4.50	2023-03-18	2027-03-11
15	15	15	15	11000.00	6.00	2022-02-15	2026-02-08
16	16	16	16	17000.00	5.50	2022-07-08	2025-07-02
17	17	17	17	20000.00	5.00	2022-03-03	2025-02-25
18	18	18	18	11000.00	6.00	2023-06-23	2026-06-16
19	19	19	19	12000.00	5.00	2024-03-14	2027-03-07
20	20	20	20	12000.00	5.00	2022-02-21	2026-02-14

Query executed successfully.

Ejercicio N°4

Listar todas las garantías asociadas a un crédito específico (por ejemplo, id_credito=20).

```
SELECT*FROM garantia_credito;  
SELECT g.*  
FROM garantia g  
JOIN garantia_credito gc ON g.id_garantia = gc.id_garantia  
WHERE gc.id_credito = 20;
```

--Ejercicio N°4
/* Listar todas las garantías asociadas a un crédito específico (por ejemplo, id_credito=20).*/
SELECT*FROM garantia_credito;
SELECT g.*
FROM garantia g
JOIN garantia_credito gc ON g.id_garantia = gc.id_garantia
WHERE gc.id_credito = 20;

Results Messages

	id_credito	id_garantia	fecha_asignacion	valor_asegurado
1	1	1	2024-01-15	10500.50
2	2	2	2024-02-10	15000.00
3	3	3	2024-03-05	17500.75
4	4	4	2024-04-20	9800.00
5	5	5	2024-05-15	13450.25
6	6	6	2024-06-12	19000.00
7	7	7	2024-07-08	14250.00
8	8	8	2024-08-23	16000.80
9	9	9	2024-09-10	8000.00
10	10	10	2024-10-02	12000.00
11	11	11	2024-10-25	17000.00
12	12	12	2024-11-15	18500.00
13	13	13	2024-12-05	15000.00

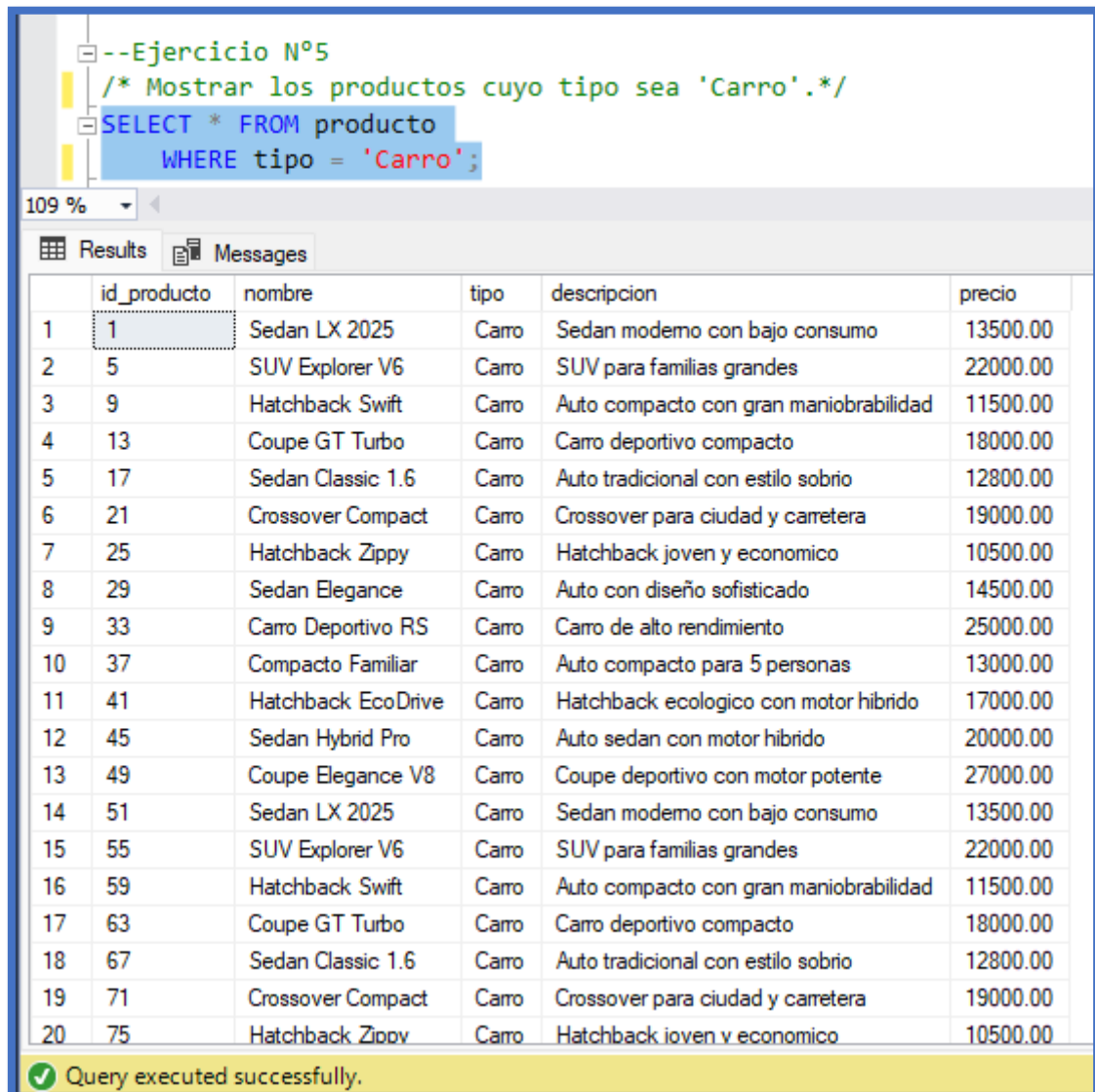
	id_garantia	tipo	descripcion	valor
1	20	Garantia Judicial	Respaldo en proceso legal	12000.00

Query executed successfully. DESKTOP-

Ejercicio N°5

Mostrar los productos cuyo tipo sea 'Carro'.

```
SELECT * FROM producto
WHERE tipo = 'Carro';
```



The screenshot shows a database query execution window. The query is: `--Ejercicio N°5`
`/* Mostrar los productos cuyo tipo sea 'Carro'.*/`
`SELECT * FROM producto`
`WHERE tipo = 'Carro';`

The results are displayed in a table with the following columns: `id_producto`, `nombre`, `tipo`, `descripcion`, and `precio`. The table contains 20 rows of data, all of which are cars (`tipo = 'Carro'`).

	id_producto	nombre	tipo	descripcion	precio
1	1	Sedan LX 2025	Carro	Sedan moderno con bajo consumo	13500.00
2	5	SUV Explorer V6	Carro	SUV para familias grandes	22000.00
3	9	Hatchback Swift	Carro	Auto compacto con gran maniobrabilidad	11500.00
4	13	Coupe GT Turbo	Carro	Carro deportivo compacto	18000.00
5	17	Sedan Classic 1.6	Carro	Auto tradicional con estilo sobrio	12800.00
6	21	Crossover Compact	Carro	Crossover para ciudad y carretera	19000.00
7	25	Hatchback Zippy	Carro	Hatchback joven y economico	10500.00
8	29	Sedan Elegance	Carro	Auto con diseño sofisticado	14500.00
9	33	Carro Deportivo RS	Carro	Carro de alto rendimiento	25000.00
10	37	Compacto Familiar	Carro	Auto compacto para 5 personas	13000.00
11	41	Hatchback EcoDrive	Carro	Hatchback ecologico con motor hibrido	17000.00
12	45	Sedan Hybrid Pro	Carro	Auto sedan con motor hibrido	20000.00
13	49	Coupe Elegance V8	Carro	Coupe deportivo con motor potente	27000.00
14	51	Sedan LX 2025	Carro	Sedan moderno con bajo consumo	13500.00
15	55	SUV Explorer V6	Carro	SUV para familias grandes	22000.00
16	59	Hatchback Swift	Carro	Auto compacto con gran maniobrabilidad	11500.00
17	63	Coupe GT Turbo	Carro	Carro deportivo compacto	18000.00
18	67	Sedan Classic 1.6	Carro	Auto tradicional con estilo sobrio	12800.00
19	71	Crossover Compact	Carro	Crossover para ciudad y carretera	19000.00
20	75	Hatchback Zippy	Carro	Hatchback joven y economico	10500.00

Query executed successfully.

Ejercicio N°6

Obtener los créditos otorgados después de una fecha específica (ejemplo: '2023-01-01').

```
SELECT * FROM credito
WHERE fecha_inicio > '2024-01-01';
```

--Ejercicio N°6
/* Obtener los créditos otorgados después de una fecha específica (ejemplo: '2023-01-01').*/
SELECT * FROM credito
WHERE fecha_inicio > '2024-01-01';

109 %

ResultsMessages

	id_credito	id_cuenta	id_producto	monto	interes	fecha_inicio	fecha_vencimiento
1	2	2	2	10000.00	5.50	2024-05-23	2027-05-14
2	19	19	19	12000.00	5.00	2024-03-14	2027-03-07
3	26	26	26	9000.00	6.50	2024-06-15	2027-06-08
4	30	30	30	9000.00	6.00	2024-07-23	2028-07-15
5	32	32	32	15000.00	4.50	2024-04-06	2027-03-30
6	42	42	42	12000.00	4.50	2024-03-16	2028-03-08
7	44	44	44	17000.00	6.00	2024-02-13	2028-02-06
8	48	48	48	11000.00	6.50	2024-02-01	2027-01-25
9	50	50	50	11000.00	5.50	2024-06-01	2028-05-25
10	51	51	1	18000.00	6.50	2024-01-10	2028-01-03
11	57	57	7	9000.00	4.50	2024-02-17	2027-02-10
12	62	62	12	18000.00	5.00	2024-04-27	2027-04-20
13	66	66	16	10000.00	5.50	2024-05-06	2027-04-29
14	74	74	24	10000.00	6.00	2024-01-26	2028-01-19
15	77	77	27	9000.00	4.50	2024-03-03	2027-02-24
16	81	81	31	9500.00	6.00	2024-06-21	2028-06-13
17	88	88	38	12000.00	4.50	2024-02-10	2028-02-03
18	93	93	43	20000.00	5.00	2024-05-01	2027-04-24
19	110	110	10	15000.00	5.00	2024-01-30	NULL
20	118	118	18	10000.00	4.50	2024-06-11	2027-06-04

Query executed successfully. DESKTOP

Ejercicio N°7

Mostrar los créditos que tienen asociado al menos una garantía.

```
SELECT DISTINCT c.*  
FROM credito c  
JOIN garantia_credito gc ON c.id_credito = gc.id_credito  
ORDER BY fecha_inicio DESC;
```

--Ejercicio N°7
/* Mostrar los créditos que tienen asociado al menos una garantía.*/

```
SELECT DISTINCT c.*  
FROM credito c  
JOIN garantia_credito gc ON c.id_credito = gc.id_credito  
ORDER BY fecha_inicio DESC;
```

109 %

Results Messages

	id_credito	id_cuenta	id_producto	monto	interes	fecha_inicio	fecha_vencimiento
1	30	30	30	9000.00	6.00	2024-07-23	2028-07-15
2	81	81	31	9500.00	6.00	2024-06-21	2028-06-13
3	26	26	26	9000.00	6.50	2024-06-15	2027-06-08
4	118	118	18	10000.00	4.50	2024-06-11	2027-06-04
5	50	50	50	11000.00	5.50	2024-06-01	2028-05-25
6	2	2	2	10000.00	5.50	2024-05-23	2027-05-14
7	123	123	23	18000.00	6.00	2024-05-13	2028-05-06
8	66	66	16	10000.00	5.50	2024-05-06	2027-04-29
9	93	93	43	20000.00	5.00	2024-05-01	2027-04-24
10	62	62	12	18000.00	5.00	2024-04-27	2027-04-20
11	32	32	32	15000.00	4.50	2024-04-06	2027-03-30
12	139	139	39	17000.00	6.00	2024-04-04	2028-03-28
13	42	42	42	12000.00	4.50	2024-03-16	2028-03-08
14	19	19	19	12000.00	5.00	2024-03-14	2027-03-07
15	77	77	27	9000.00	4.50	2024-03-03	2027-02-24
16	57	57	7	9000.00	4.50	2024-02-17	2027-02-10
17	44	44	44	17000.00	6.00	2024-02-13	2028-02-06
18	88	88	38	12000.00	4.50	2024-02-10	2028-02-03
19	131	131	31	15000.00	4.50	2024-02-02	2027-01-26
20	48	48	48	11000.00	6.50	2024-02-01	2027-01-25

✓ Query executed successfully.

Ejercicio N°8

Mostrar los clientes que han solicitado créditos.

```
SELECT DISTINCT cliente.*  
FROM cliente  
JOIN credito ON cliente.id_cliente = credito.id_credito;
```

--Ejercicio N°8
/* Mostrar los clientes que han solicitado créditos.*/
SELECT DISTINCT cliente.*
FROM cliente
JOIN credito ON cliente.id_cliente = credito.id_credito;

109 %

Results Messages

	id_cliente	nombre	apellido	email	telefono
1	1	Diego	Gomez	dgomez@unmsm.edu.pe	984621098
2	2	Lucia	Garcia	lucia.g@unmsm.edu.pe	984126350
3	3	Ana	Perez	ana.perez@gmail.com	923187645
4	4	Carlos	Lopez	clopez@hotmail.com	949112287
5	5	Valeria	Morales	valeria.m@gmail.com	987612530
6	6	Pedro	Soto	pedro.soto@unmsm.edu.pe	957489621
7	7	Flor	Mendoza	flor.m@unmsm.edu.pe	913476281
8	8	Luis	Vargas	l.vargas@unmsm.edu.pe	937416205
9	9	Gabriela	Silva	g.silva@gmail.com	917438612
10	10	Carmen	Delgado	carmen.d@gmail.com	974562013
11	11	Ricardo	Silva	ricardo.silva@gmail.com	983214875
12	12	Jorge	Cruz	jorge.c@gmail.com	961875230
13	13	Pedro	Lopez	pedro.lopez@hotmail.com	918736402
14	14	Lucia	Vargas	lucia.vargas@gmail.com	989764321
15	15	Raul	Cruz	raulc@hotmail.com	999283615
16	16	Juan	Lopez	juanlopez@unmsm.edu.pe	963142578
17	17	Gabriela	Chavez	gabriela.ch@unmsm.edu.pe	976431209
18	18	Carmen	Cruz	carmen.cruz@hotmail.com	981432672
19	19	Jorge	Salazar	jorge.salazar@unmsm.edu.pe	952348713
20	20	Diego	Cruz	diego.cruz@hotmail.com	918726354

✓ Query executed successfully.

Ejercicio N°9

Listar créditos con su monto, interés y la garantía asociada (nombre y descripción de la garantía).

```
SELECT c.id_credito, c.monto, c.interes, g.tipo AS garantia_tipo,
g.descripcion AS garantia_descripcion
FROM credito c
JOIN garantia_credito gc ON c.id_credito = gc.id_credito
JOIN garantia g ON gc.id_garantia = g.id_garantia
ORDER BY interes ASC;
```

--Ejercicio N°9
/* Listar créditos con su monto, interés y la garantía asociada (nombre y descripción de la garantía).*/

```
SELECT c.id_credito, c.monto, c.interes, g.tipo AS garantia_tipo, g.descripcion AS garantia_descripcion
FROM credito c
JOIN garantia_credito gc ON c.id_credito = gc.id_credito
JOIN garantia g ON gc.id_garantia = g.id_garantia
ORDER BY interes ASC;
```

109 %

Results Messages

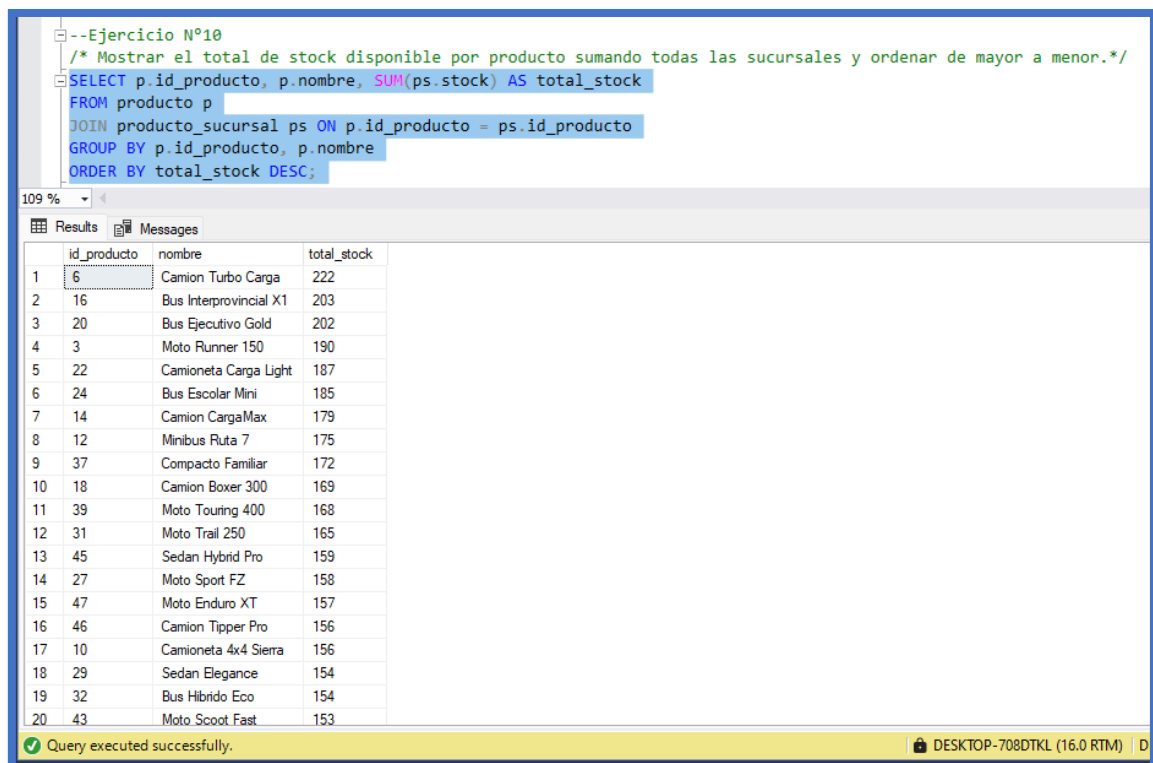
	id_credito	monto	interes	garantia_tipo	garantia_descripcion
1	6	10000.00	4.50	Fianza	Garantia personal
2	7	11000.00	4.50	Carta Fianza	Garantia bancaria
3	8	17000.00	4.50	Garantia Bancaria	Respaldo financiero
4	10	20000.00	4.50	Hipoteca Rural	Terreno agricola como garantia
5	14	9000.00	4.50	Seguro de Garantia	Seguro contra incumplimiento
6	21	9000.00	4.50	Hipoteca	Casa como respaldo
7	22	17000.00	4.50	Prenda Vehicular	Auto usado como garantia
8	24	12000.00	4.50	Seguro de Desgravamen	Cubre deuda en fallecimiento
9	35	10000.00	4.50	Garantia Solidaria	Responsabilidad conjunta
10	36	9000.00	4.50	Hipoteca Comercial	Inmueble comercial en garantia
11	37	15000.00	4.50	Prenda Agraria	Equipo agricola como garantia
12	32	15000.00	4.50	Caucion	Garantia para cumplimiento de contrato
13	42	12000.00	4.50	Prenda Vehicular	Auto usado como garantia
14	56	15000.00	4.50	Carta Fianza	Garantia bancaria
15	57	9000.00	4.50	Garantia Bancaria	Respaldo financiero
16	65	9500.00	4.50	Hipoteca Comercial	Inmueble comercial en garantia
17	76	15000.00	4.50	Carta Fianza	Garantia bancaria
18	77	9000.00	4.50	Garantia Bancaria	Respaldo financiero
19	72	9500.00	4.50	Aval	Tercero responsable solidario
20	84	12000.00	4.50	Garantia Solidaria	Resoonsabilidad coniunta

Query executed successfully. DESKTOP-708DTKL (16.0

Ejercicio N°10

Mostrar el total de stock disponible por producto sumando todas las sucursales y ordenar de mayor a menor.

```
SELECT p.id_producto, p.nombre, SUM(ps.stock) AS total_stock
FROM producto p
JOIN producto_sucursal ps ON p.id_producto = ps.id_producto
GROUP BY p.id_producto, p.nombre
ORDER BY total_stock DESC;
```



The screenshot shows a SQL query execution window. The query is as follows:

```
--Ejercicio N°10
/* Mostrar el total de stock disponible por producto sumando todas las sucursales y ordenar de mayor a menor.*/
SELECT p.id_producto, p.nombre, SUM(ps.stock) AS total_stock
FROM producto p
JOIN producto_sucursal ps ON p.id_producto = ps.id_producto
GROUP BY p.id_producto, p.nombre
ORDER BY total_stock DESC;
```

The results are displayed in a table with the following columns: id_producto, nombre, and total_stock. The results are ordered by total_stock in descending order.

	id_producto	nombre	total_stock
1	6	Camion Turbo Carga	222
2	16	Bus Interprovincial X1	203
3	20	Bus Ejecutivo Gold	202
4	3	Moto Runner 150	190
5	22	Camioneta Carga Light	187
6	24	Bus Escolar Mini	185
7	14	Camion CargaMax	179
8	12	Minibus Ruta 7	175
9	37	Compacto Familiar	172
10	18	Camion Boxer 300	169
11	39	Moto Touring 400	168
12	31	Moto Trail 250	165
13	45	Sedan Hybrid Pro	159
14	27	Moto Sport FZ	158
15	47	Moto Enduro XT	157
16	46	Camion Tipper Pro	156
17	10	Camioneta 4x4 Sierra	156
18	29	Sedan Elegance	154
19	32	Bus Hibrido Eco	154
20	43	Moto Scoot Fast	153

Query executed successfully. DESKTOP-708DTKL (16.0 RTM) | D

Ejercicio N°11

Obtener el monto total de créditos otorgados por cada entidad financiera.

```
SELECT ef.nombre, SUM(c.monto) AS total_credito
FROM entidad_financiera ef
JOIN credito c ON ef.id_entidadFinanciera = c.id_credito
GROUP BY ef.nombre;
```

--Ejercicio N°11
/* Obtener el monto total de créditos otorgados por cada entidad financiera.*/

```
SELECT ef.nombre, SUM(c.monto) AS total_credito
FROM entidad_financiera ef
JOIN credito c ON ef.id_entidadFinanciera = c.id_credito
GROUP BY ef.nombre;
```

09 %

Results Messages

	nombre	total_credito
1	Banco Andino	116500.00
2	Banco del Pueblo	74000.00
3	Banco Emprende	102000.00
4	Banco Familiar	70000.00
5	Banco Uno	97000.00
6	Caja Arequipa	73000.00
7	Caja Cusco	87000.00
8	Caja Lima	79000.00
9	Caja Metropolitana	87000.00
10	Credito Seguro	86500.00
11	Financiera ABC	85000.00
12	Financiera Capital	76000.00
13	Financiera Esperanza	128000.00
14	MicroBanco	73000.00
15	Microfinanzas Andinas	102500.00

Query executed successfully.

Ejercicio N°12

Mostrar los productos que tienen 20 unidades de stock porsucursal.

```
SELECT p.id_producto, p.nombre, p.descripcion, COALESCE(ps.stock,20) AS stock
FROM producto p
LEFT JOIN producto_sucursal ps ON p.id_producto = ps.id_producto
WHERE ps.stock IS NULL OR ps.stock = 20;
```

```
--Ejercicio N°12
/* Mostrar los productos que tienen 20 unidades de stock porsucursal.*/
SELECT p.id_producto, p.nombre, p.descripcion, COALESCE(ps.stock,20) AS stock
FROM producto p
LEFT JOIN producto_sucursal ps ON p.id_producto = ps.id_producto
WHERE ps.stock IS NULL OR ps.stock = 20;
```

109 %

Results Messages

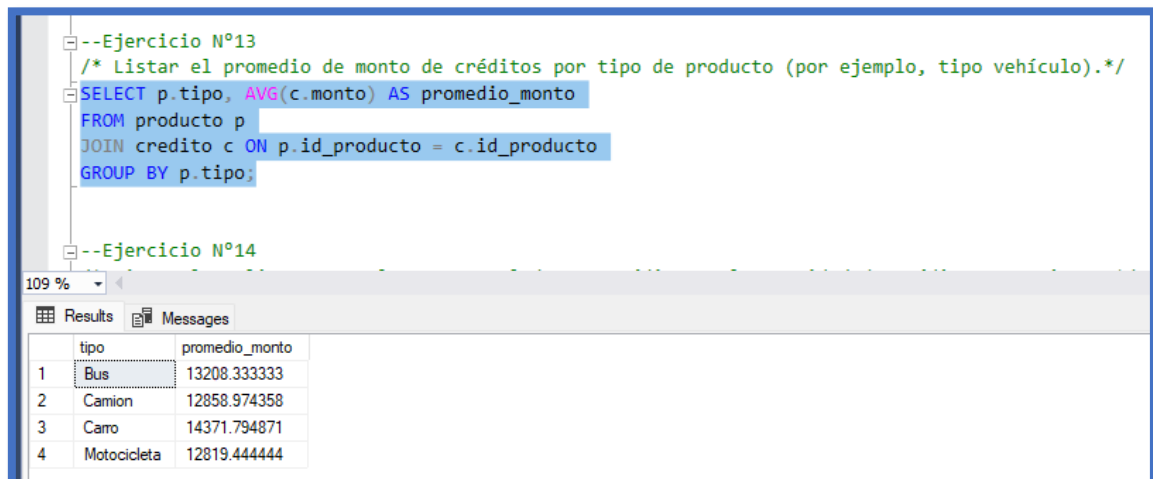
	id_producto	nombre	descripcion	stock
1	30	Camion Frigorifico	Camion con refrigeracion integrada	20
2	51	Sedan LX 2025	Sedan moderno con bajo consumo	20
3	52	Pickup Strong X	Pickup resistente para carga pesada	20
4	53	Moto Runner 150	Moto ligera ideal para ciudad	20
5	54	Bus Escolar Max	Bus escolar con 40 asientos	20
6	55	SUV Explorer V6	SUV para familias grandes	20
7	56	Camion Turbo Carga	Camion de gran tonelaje	20
8	57	Scooter EcoLite	Scooter electrico urbano	20
9	58	Bus Urbano XL	Bus urbano con aire acondicionado	20
10	59	Hatchback Swift	Auto compacto con gran maniobrabilidad	20
11	60	Camioneta 4x4 Sierra	Ideal para caminos rurales	20
12	61	Moto CrossX 200	Motocicleta de motocross resistente	20
13	62	Minibus Ruta 7	Minibus ideal para rutas cortas	20
14	63	Coupe GT Turbo	Carro deportivo compacto	20
15	64	Camion CargaMax	Camion diesel para largas distancias	20
16	65	Moto Urbana RZ	Moto economica para uso diario	20
17	66	Bus Interprovincial X1	Bus de viaje con butacas reclinables	20
18	67	Sedan Classic 1.6	Auto tradicional con estilo sobrio	20
19	68	Camion Boxer 300	Camion cerrado tipo furgon	20
20	69	Moto Retro 125	Motocicleta con diseño clasico	20

Query executed successfully.

Ejercicio N°13

Listar el promedio de monto de créditos por tipo de producto (por ejemplo, tipo vehículo).

```
SELECT p.tipo, AVG(c.monto) AS promedio_monto
FROM producto p
JOIN credito c ON p.id_producto = c.id_producto
GROUP BY p.tipo;
```



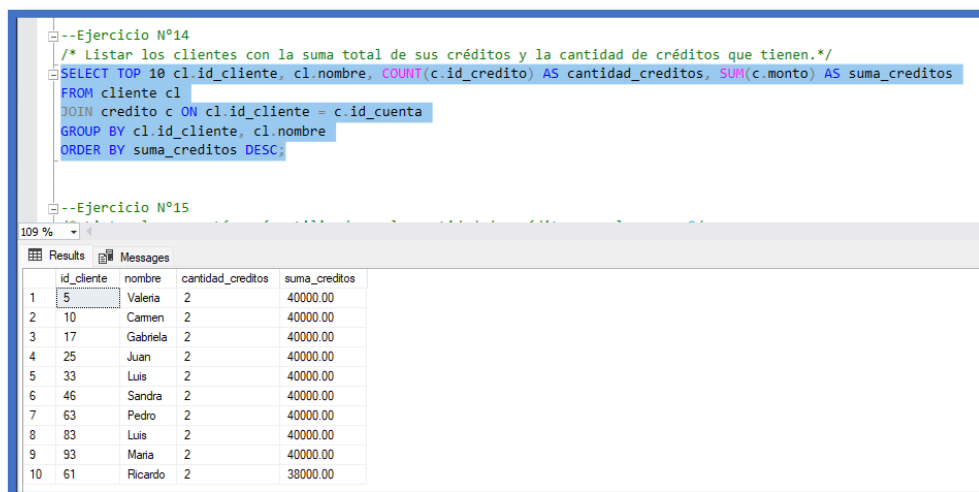
```
--Ejercicio N°13
/* Listar el promedio de monto de créditos por tipo de producto (por ejemplo, tipo vehículo).*/
SELECT p.tipo, AVG(c.monto) AS promedio_monto
FROM producto p
JOIN credito c ON p.id_producto = c.id_producto
GROUP BY p.tipo;
```

	tipo	promedio_monto
1	Bus	13208.333333
2	Camion	12858.974358
3	Carro	14371.794871
4	Motocicleta	12819.444444

Ejercicio N°14

Listar los clientes con la suma total de sus créditos y la cantidad de créditos que tienen.

```
SELECT TOP 10 cl.id_cliente, cl.nombre, COUNT(c.id_credito) AS
cantidad_creditos, SUM(c.monto) AS suma_creditos
FROM cliente cl
JOIN credito c ON cl.id_cliente = c.id_cuenta
GROUP BY cl.id_cliente, cl.nombre
ORDER BY suma_creditos DESC;
```



```
--Ejercicio N°14
/* Listar los clientes con la suma total de sus créditos y la cantidad de créditos que tienen.*/
SELECT TOP 10 cl.id_cliente, cl.nombre, COUNT(c.id_credito) AS cantidad_creditos, SUM(c.monto) AS suma_creditos
FROM cliente cl
JOIN credito c ON cl.id_cliente = c.id_cuenta
GROUP BY cl.id_cliente, cl.nombre
ORDER BY suma_creditos DESC;
```

	id_cliente	nombre	cantidad_creditos	suma_creditos
1	5	Valeria	2	40000.00
2	10	Carmen	2	40000.00
3	17	Gabriela	2	40000.00
4	25	Juan	2	40000.00
5	33	Luis	2	40000.00
6	46	Sandra	2	40000.00
7	63	Pedro	2	40000.00
8	83	Luis	2	40000.00
9	93	Maria	2	40000.00
10	61	Ricardo	2	38000.00

Ejercicio N°15

Listar las garantías más utilizadas y la cantidad de créditos que las usan.

```
SELECT g.tipo, COUNT(gc.id_credito) AS cantidad_usos
FROM garantia g
JOIN garantia_credito gc ON g.id_garantia = gc.id_garantia
GROUP BY g.tipo
ORDER BY cantidad_usos DESC;
```

--Ejercicio N°15
/* Listar las garantías más utilizadas y la cantidad de créditos que las usan.*/

```
SELECT g.tipo, COUNT(gc.id_credito) AS cantidad_usos
FROM garantia g
JOIN garantia_credito gc ON g.id_garantia = gc.id_garantia
GROUP BY g.tipo
ORDER BY cantidad_usos DESC;
```

--Ejercicio N°16

109 %

Results Messages

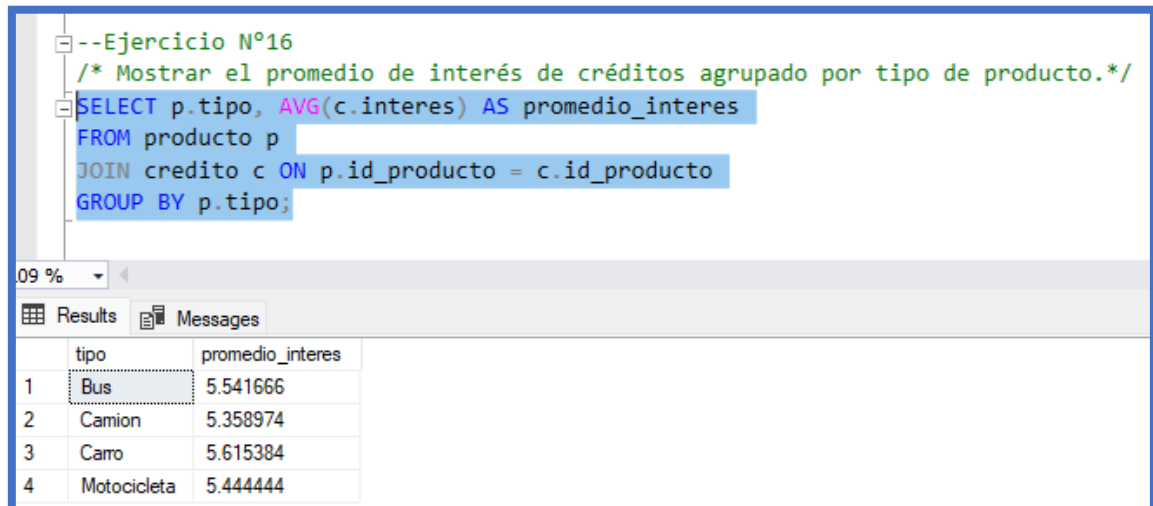
	tipo	cantidad_usos
1	Prenda Vehicular	10
2	Seguro de Desgravamen	9
3	Prenda Mercantil	9
4	Hipoteca	9
5	Aval	9
6	Carta Fianza	9
7	Deposito en garantia	9
8	Fianza	9
9	Garantia Bancaria	9
10	Hipoteca Rural	8
11	Prenda Agraria	6
12	Prenda Industrial	6
13	Hipoteca Comercial	6
14	Seguro de Garantia	6
15	Garantia Fiduciaria	6
16	Garantia Judicial	6
17	Garantia Patronal	6
18	Garantia Solidaria	6
19	Caucion	6
20	Aval Personal	6

Query executed successfully.

Ejercicio N°16

Mostrar el promedio de interés de créditos agrupado por tipo de producto.

```
SELECT p.tipo, AVG(c.interes) AS promedio_interes
FROM producto p
JOIN credito c ON p.id_producto = c.id_producto
GROUP BY p.tipo;
```



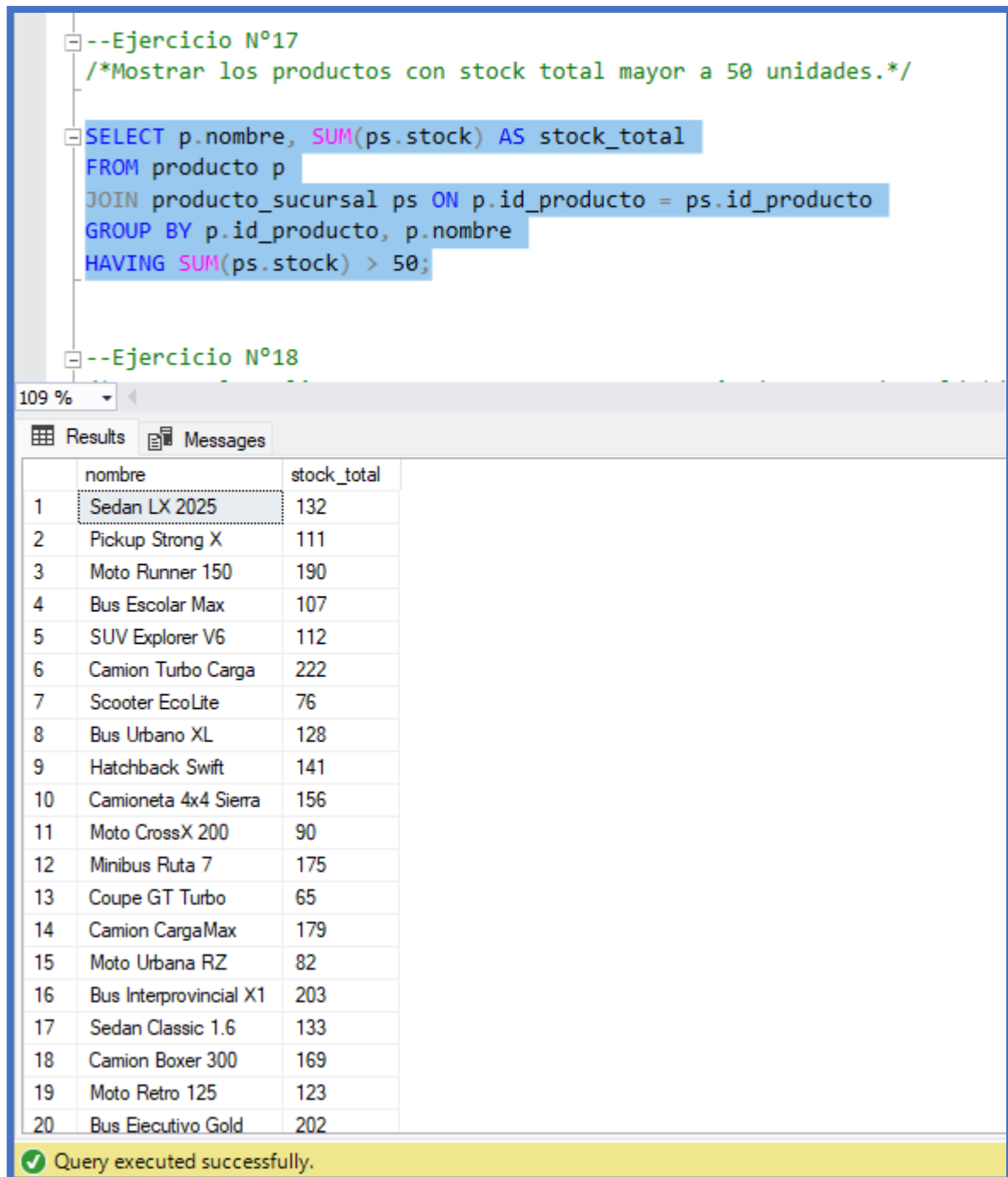
```
--Ejercicio N°16
/* Mostrar el promedio de interés de créditos agrupado por tipo de producto.*/
SELECT p.tipo, AVG(c.interes) AS promedio_interes
FROM producto p
JOIN credito c ON p.id_producto = c.id_producto
GROUP BY p.tipo;
```

	tipo	promedio_interes
1	Bus	5.541666
2	Camion	5.358974
3	Carro	5.615384
4	Motocicleta	5.444444

Ejercicio N°17

Mostrar los productos con stock total mayor a 50 unidades.

```
SELECT p.nombre, SUM(ps.stock) AS stock_total
FROM producto p
JOIN producto_sucursal ps ON p.id_producto = ps.id_producto
GROUP BY p.id_producto, p.nombre
HAVING SUM(ps.stock) > 50;
```



```
--Ejercicio N°17
/*Mostrar los productos con stock total mayor a 50 unidades.*/

SELECT p.nombre, SUM(ps.stock) AS stock_total
FROM producto p
JOIN producto_sucursal ps ON p.id_producto = ps.id_producto
GROUP BY p.id_producto, p.nombre
HAVING SUM(ps.stock) > 50;

--Ejercicio N°18
```

109 %

Results Messages

	nombre	stock_total
1	Sedan LX 2025	132
2	Pickup Strong X	111
3	Moto Runner 150	190
4	Bus Escolar Max	107
5	SUV Explorer V6	112
6	Camion Turbo Carga	222
7	Scooter EcoLite	76
8	Bus Urbano XL	128
9	Hatchback Swift	141
10	Camioneta 4x4 Sierra	156
11	Moto CrossX 200	90
12	Minibus Ruta 7	175
13	Coupe GT Turbo	65
14	Camion CargaMax	179
15	Moto Urbana RZ	82
16	Bus Interprovincial X1	203
17	Sedan Classic 1.6	133
18	Camion Boxer 300	169
19	Moto Retro 125	123
20	Bus Ejecutivo Gold	202

✓ Query executed successfully.

Ejercicio N°18

Mostrar los clientes con cuentas que tengan más de 2,000 de saldo

```
SELECT c.nombre, c.apellido, cu.saldo
FROM cliente c
JOIN cuenta cu ON c.id_cliente = cu.id_cliente
WHERE cu.saldo > 2000;
```

--Ejercicio N°18
/*Mostrar los clientes con cuentas que tengan más de 2,000 de saldo*/

```
SELECT c.nombre, c.apellido, cu.saldo
FROM cliente c
JOIN cuenta cu ON c.id_cliente = cu.id_cliente
WHERE cu.saldo > 2000;
```

109 %

Results Messages

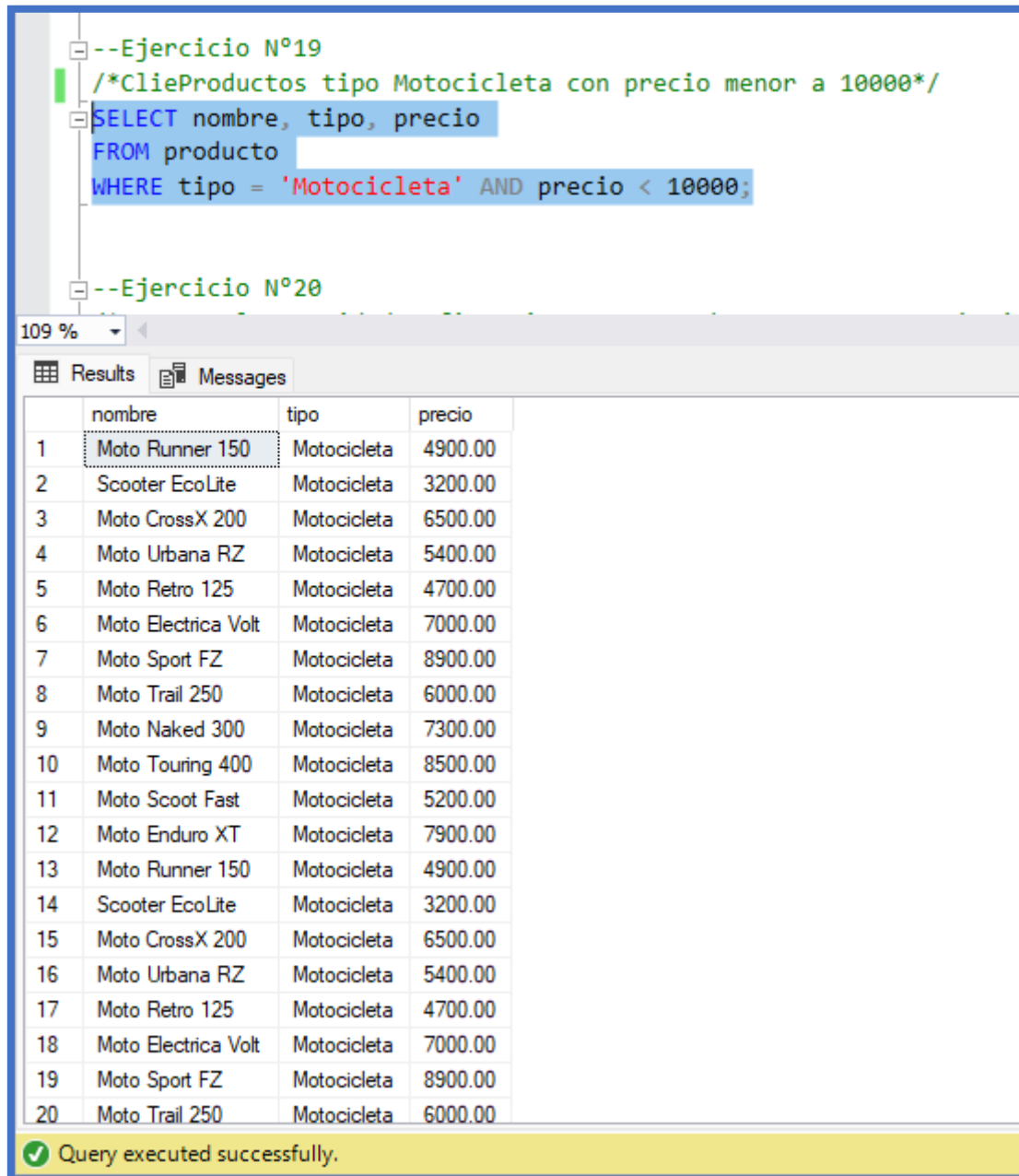
	nombre	apellido	saldo
1	Diego	Gomez	3200.00
2	Ana	Perez	12000.00
3	Valeria	Morales	4520.90
4	Gabriela	Silva	6700.00
5	Raul	Cruz	7800.50
6	Jorge	Salazar	6800.00
7	Maria	Soto	3200.75
8	Ricardo	Torres	4000.90
9	Diego	Salazar	2700.00
10	Victor	Garcia	7200.00
11	Ana	Rojas	2400.00
12	Alfredo	Chavez	2300.00
13	Diego	Gomez	2900.00
14	Ana	Perez	8200.00
15	Valeria	Morales	2520.90
16	Gabriela	Silva	7700.00
17	Raul	Cruz	8800.50
18	Jorge	Salazar	7800.00
19	Maria	Soto	4200.75
20	Ricardo	Torres	5000.90

✓ Query executed successfully.

Ejercicio N°19

ClieProductos tipo Motocicleta con precio menor a 10000

```
SELECT nombre, tipo, precio
FROM producto
WHERE tipo = 'Motocicleta' AND precio < 10000;
```



```
--Ejercicio N°19
/*ClieProductos tipo Motocicleta con precio menor a 10000*/
SELECT nombre, tipo, precio
FROM producto
WHERE tipo = 'Motocicleta' AND precio < 10000;

--Ejercicio N°20
```

109 %

Results Messages

	nombre	tipo	precio
1	Moto Runner 150	Motocicleta	4900.00
2	Scooter EcoLite	Motocicleta	3200.00
3	Moto CrossX 200	Motocicleta	6500.00
4	Moto Urbana RZ	Motocicleta	5400.00
5	Moto Retro 125	Motocicleta	4700.00
6	Moto Electrica Volt	Motocicleta	7000.00
7	Moto Sport FZ	Motocicleta	8900.00
8	Moto Trail 250	Motocicleta	6000.00
9	Moto Naked 300	Motocicleta	7300.00
10	Moto Touring 400	Motocicleta	8500.00
11	Moto Scoot Fast	Motocicleta	5200.00
12	Moto Enduro XT	Motocicleta	7900.00
13	Moto Runner 150	Motocicleta	4900.00
14	Scooter EcoLite	Motocicleta	3200.00
15	Moto CrossX 200	Motocicleta	6500.00
16	Moto Urbana RZ	Motocicleta	5400.00
17	Moto Retro 125	Motocicleta	4700.00
18	Moto Electrica Volt	Motocicleta	7000.00
19	Moto Sport FZ	Motocicleta	8900.00
20	Moto Trail 250	Motocicleta	6000.00

✓ Query executed successfully.

Ejercicio N°20

Mostrar las entidades financieras con nombre que contenga 'Caja'

```
SELECT nombre, direccion
FROM entidad_financiera
WHERE nombre LIKE '%Caja%';
```

The screenshot shows a database query execution window. At the top, the query is displayed in a text editor with syntax highlighting. Below the query, there are tabs for 'Results' and 'Messages'. The 'Results' tab is active, showing a table with two columns: 'nombre' and 'direccion'. The table contains 20 rows of data, all of which include 'Caja' in the name. A status bar at the bottom indicates that the query was executed successfully.

```
--Ejercicio N°20
/*Mostrar las entidades financieras con nombre que contenga 'Caja'*/
SELECT nombre, direccion
FROM entidad_financiera
WHERE nombre LIKE '%Caja%';
```

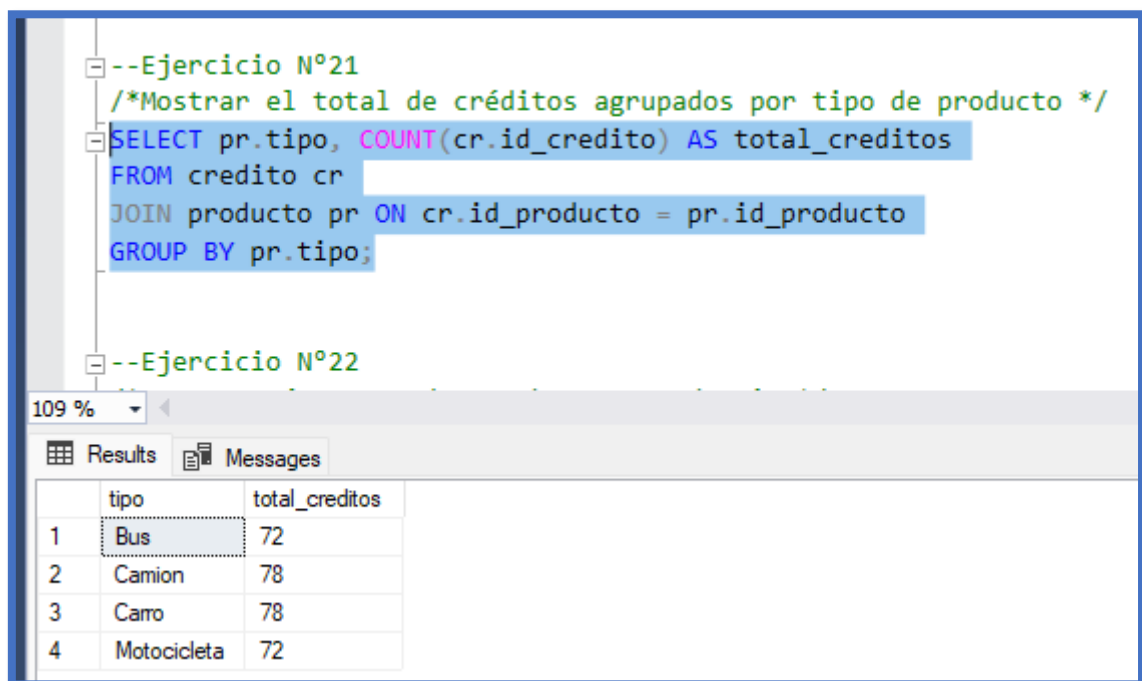
	nombre	direccion
1	Caja Lima	Av. La Marina 200
2	Caja Arequipa	Av. Cultura 900
3	Caja Metropolitana	Calle Bolívar 720
4	Caja Lima	Av. Libertad 800
5	Caja Arequipa	Calle Piura 556
6	Caja Metropolitana	Av. Salaverry 300
7	Caja Lima	Av. Panamericana 1420
8	Caja Arequipa	Jr. Miraflores 345
9	Caja Metropolitana	Av. Andes 222
10	Caja Cusco	Av. Universitaria 1050
11	Caja Cusco	Av. La Marina 200
12	Caja Cusco	Calle Piura 556
13	Caja Cusco	Av. Salaverry 300
14	Caja Lima	Av. La Marina 200
15	Caja Arequipa	Av. Cultura 900
16	Caja Metropolitana	Calle Bolívar 720
17	Caja Lima	Av. Libertad 800
18	Caja Arequipa	Calle Piura 556
19	Caja Metropolitana	Av. Salaverry 300
20	Caja Lima	Av. Panamericana 1420

✓ Query executed successfully.

Ejercicio N°21

Mostrar el total de créditos agrupados por tipo de producto

```
SELECT pr.tipo, COUNT(cr.id_credito) AS total_credits
FROM credito cr
JOIN producto pr ON cr.id_producto = pr.id_producto
GROUP BY pr.tipo;
```



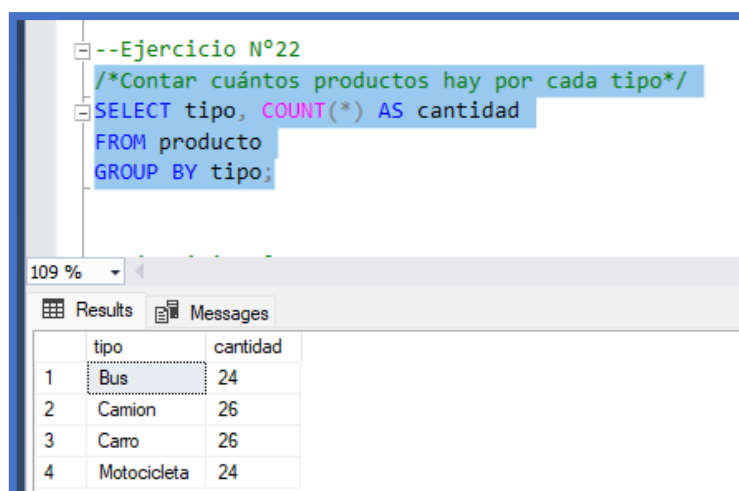
```
--Ejercicio N°21
/*Mostrar el total de créditos agrupados por tipo de producto */
SELECT pr.tipo, COUNT(cr.id_credito) AS total_credits
FROM credito cr
JOIN producto pr ON cr.id_producto = pr.id_producto
GROUP BY pr.tipo;
```

	tipo	total_credits
1	Bus	72
2	Camion	78
3	Carro	78
4	Motocicleta	72

Ejercicio N°22

Contar cuántos productos hay por cada tipo

```
SELECT tipo, COUNT(*) AS cantidad
FROM producto
GROUP BY tipo;
```



```
--Ejercicio N°22
/*Contar cuántos productos hay por cada tipo*/
SELECT tipo, COUNT(*) AS cantidad
FROM producto
GROUP BY tipo;
```

	tipo	cantidad
1	Bus	24
2	Camion	26
3	Carro	26
4	Motocicleta	24

Ejercicio N°23

Listar los correos electrónicos de clientes que tienen cuenta en Banco Uno

```
SELECT cli.nombre, cli.apellido, cli.email
FROM cliente cli
JOIN cuenta cu ON cli.id_cliente = cu.id_cliente
JOIN entidad_financiera ef ON cu.id_entidadFinanciera =
ef.id_entidadFinanciera
WHERE ef.nombre = 'Banco Uno';
```

--Ejercicio N°23
/*Listar los correos electrónicos de clientes que tienen cuenta en Banco Uno*/

```
SELECT cli.nombre, cli.apellido, cli.email
FROM cliente cli
JOIN cuenta cu ON cli.id_cliente = cu.id_cliente
JOIN entidad_financiera ef ON cu.id_entidadFinanciera = ef.id_entidadFinanciera
WHERE ef.nombre = 'Banco Uno';
```

09 %

Results Messages

	nombre	apellido	email
1	Diego	Gomez	dgomez@unmsm.edu.pe
2	Ricardo	Silva	ricardo.silva@gmail.com
3	Valeria	Ramos	valeria.ramos@hotmail.com
4	Camen	Delgado	camen.d@gmail.com
5	Diego	Cruz	diego.cruz@hotmail.com
6	Pedro	Rojas	p.rojas@gmail.com
7	Valeria	Morales	valeria.m@gmail.com
8	Camen	Delgado	camen.d@gmail.com
9	Juan	Silva	juan.silva@gmail.com
10	Diego	Gomez	dgomez@unmsm.edu.pe
11	Ricardo	Silva	ricardo.silva@gmail.com
12	Valeria	Ramos	valeria.ramos@hotmail.com
13	Camen	Delgado	camen.d@gmail.com
14	Diego	Cruz	diego.cruz@hotmail.com
15	Pedro	Rojas	p.rojas@gmail.com
16	Valeria	Morales	valeria.m@gmail.com
17	Camen	Delgado	camen.d@gmail.com
18	Juan	Silva	juan.silva@gmail.com

✓ Query executed successfully.