ESCUELA POLITÉCNICA NACIONAL

FACULTAD: ESFOT

PERÍODO ACADÉMICO: 2024 A **ASIGNATURA:** BASES DE DATOS

DOCENTE: MSc. Yadira Franco **PARALELO**: GR1

NOMBRE DEL ESTUDIANTE (S): Carlos Pérez y Mateo Torres **NOTA:**

TEMA: **CONSULTAS BÁSICAS**

TIEMPO:

INSTRUCCIONES:

PROBLEMA A ESTUDIAR Y RESOLVER

Estamos diseñando una base de datos para un banco "Los afortunados". Este banco necesita una base de datos para gestionar a sus clientes, las cuentas bancarias de esos clientes y las transacciones que se realizan en esas cuentas.

1. Crear y relacionar tres tablas: Clientes, Cuentas y Transacciones.

Requisitos TABLA:

Cliente

id_cliente, nombre, email

Cuentas

id_cuenta,id_cliente,saldo

Transacciones

id_transaccion, id_cuenta, monto, fecha

Relaciones:

1. Cada cliente puede tener múltiples cuentas, pero cada cuenta pertenece a un único cliente.
2. Cada cuenta puede tener múltiples transacciones, pero cada transacción está asociada a una sola cuenta.

```sql
create table clientes (
id_cliente int not null primary key auto_increment,
nombre varchar (50) not null,
email varchar (50) not null unique
);

create table cuentas (
id_cuenta int not null primary key auto_increment,
saldo decimal (5, 2) not null,
id_cliente int not null,
foreign key (id_cliente) references clientes (id_cliente)
);

create table transacciones (
id_transaccion int not null primary key auto_increment,
monto decimal (5,2) not null,
fecha date not null default current_timestamp,
id_cuenta int not null,
foreign key (id_cuenta) references cuentas (id_cuenta)
);
```

3. Insertar (20 registro)

```sql
-- Registros clientes
INSERT INTO clientes (nombre, email) VALUES
('Juan Pérez', 'juan.perez@example.com'),
('María López', 'maria.lopez@example.com'),
('Carlos García', 'carlos.garcia@example.com'),
('Ana Fernández', 'ana.fernandez@example.com'),
('Luis Martínez', 'luis.martinez@example.com'),
('Marta Gómez', 'marta.gomez@example.com'),
('José Sánchez', 'jose.sanchez@example.com'),
('Laura Díaz', 'laura.diaz@example.com'),
('Pedro Hernández', 'pedro.hernandez@example.com'),
('Elena Ruiz', 'elena.ruiz@example.com'),
('Javier Ramos', 'javier.ramos@example.com'),
('Carmen Morales', 'carmen.morales@example.com'),
('Raúl Ortiz', 'raul.ortiz@example.com'),
('Lucía Medina', 'lucia.medina@example.com'),
('Diego Romero', 'diego.romero@example.com'),
('Sara Alonso', 'sara.alonso@example.com'),
('Miguel Torres', 'miguel.torres@example.com'),
('Isabel Castro', 'isabel.castro@example.com'),
('Antonio Gil', 'antonio.gil@example.com'),
('Nuria Soto', 'nuria.soto@example.com');
```

```sql
-- Registros cuentas
INSERT INTO cuentas (saldo, id_cliente) VALUES
(1500.50, 1),
(2300.75, 2),
(1800.00, 3),
(500.25, 4),
(3200.60, 5),
(2100.00, 6),
(750.90, 7),
(940.40, 8),
(1200.00, 9),
(1330.80, 10),
(2450.00, 11),
(760.00, 12),
(1800.25, 13),
(2200.50, 14),
(1500.00, 15),
(2300.00, 16),
(1800.00, 17),
(1300.50, 18),
(2100.00, 19),
(1500.00, 20);
```

```sql
-- Registros transacciones
INSERT INTO transacciones (monto, fecha, id_cuenta) VALUES
(100.50, '2024-07-01', 1),
(200.75, '2024-07-02', 2),
(150.00, '2024-07-03', 3),
(50.25, '2024-07-04', 4),
(300.60, '2024-07-05', 5),
(100.00, '2024-07-06', 6),
(75.90, '2024-07-07', 7),
(90.40, '2024-07-08', 8),
(120.00, '2024-07-09', 9),
(130.80, '2024-07-10', 10),
(245.00, '2024-07-11', 11),
(76.00, '2024-07-12', 12),
(180.25, '2024-07-13', 13),
(220.50, '2024-07-14', 14),
(150.00, '2024-07-15', 15),
(230.00, '2024-07-16', 16),
(180.00, '2024-07-17', 17),
(130.50, '2024-07-18', 18),
(210.00, '2024-07-19', 19),
(150.00, '2024-07-20', 20);
```

Ejercicios Propuestos

Selección y Filtración Básica

CREAR EN GRUPO DE DOS ESTUDIANTES

Ejercicio 1: Selección y Filtración Básica

- Selecciona el id_cuenta y el saldo de todas las cuentas.

```
-- Saldo e id de cuentas
select saldo, id_cuenta from cuentas;
```

Result Grid | Filter Rows: | Edit:

saldo	id_cuenta
1500.50	1
2300.75	2
1800.00	3
500.25	4
3200.60	5
2100.00	6
750.90	7
940.40	8
1200.00	9
1330.80	10
2450.00	11
760.00	12
1800.25	13
2200.50	14
1500.00	15
2300.00	16
1800.00	17

- Filtra las cuentas que tienen un saldo mayor a 4000.

```
-- Saldo mayor a 4000
select saldo from cuentas where saldo > 4000;
```

Result Grid | Filter Rows: | Export: | Wrap C

saldo

```
98 •  select saldo from cuentas where saldo > 2000;
99
```

Result Grid | Filter Rows: | Export: | Wrap

saldo
2300.75
3200.60
2100.00
2450.00
2200.50
2300.00
2100.00

- Seleccionar las cuentas con saldo menor a 2000

```
100    -- Saldo menor a 2000
101 •  select saldo from cuentas where saldo < 2000;
102
```

Result Grid | Filter Rows: | Export: | Wrap

saldo
1500.50
1800.00
500.25
750.90
940.40
1200.00
1330.80
760.00
1800.25
1500.00
1800.00
1300.50
1500.00

- Mostrar los nombres de los clientes y la longitud de sus nombres

```
-- Mostrar los nombres de los clientes y la longitud de sus nombres
SELECT nombre, LENGTH(nombre) AS longitud_nombre FROM clientes;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombre	longitud_nombre
Juan Pérez	11
María López	13
Carlos García	14
Ana Fernández	14
Luis Martínez	14
Marta Gómez	12
José Sánchez	14
Laura Díaz	11
Pedro Hernández	16
Elena Ruiz	10
Javier Ramos	12
Carmen Morales	14
Raúl Ortiz	11
Lucía Medina	13
Diego Romero	12
Sara Alonso	11
Miguel Torres	13

- Seleccionar las cuentas con saldo entre 1000 y 3000

```
-- Seleccionar las cuentas con saldo entre 1000 y 3000
SELECT * FROM cuentas WHERE saldo BETWEEN 1000 AND 3000;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

id_cuenta	saldo	id_cliente
1	1500.50	1
2	2300.75	2
3	1800.00	3
6	2100.00	6
9	1200.00	9
10	1330.80	10
11	2450.00	11
13	1800.25	13
14	2200.50	14
15	1500.00	15
16	2300.00	16
17	1800.00	17
18	1300.50	18
19	2100.00	19
20	1500.00	20

- Obtener una lista de todos los clientes y sus cuentas, mostrando los valores NULL

```
-- Obtener una lista de todos los clientes y sus cuentas, mostrando los valores NULL
SELECT c.nombre, cu.id_cuenta, cu.saldo FROM clientes c LEFT JOIN cuentas cu ON c.id_cliente = cu.id_cliente;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombre	id_cuenta	saldo
Juan Pérez	1	1500.50
María López	2	2300.75
Carlos García	3	1800.00
Ana Fernández	4	500.25
Luis Martínez	5	3200.60
Marta Gómez	6	2100.00
José Sánchez	7	750.90
Laura Díaz	8	940.40
Pedro Hernández	9	1200.00
Elena Ruiz	10	1330.80
Javier Ramos	11	2450.00
Carmen Morales	12	760.00
Raúl Ortiz	13	1800.25
Lucía Medina	14	2200.50
Diego Romero	15	1500.00
Sara Alonso	16	2300.00
Miguel Torres	17	1800.00

- Concatenar el nombre del cliente y su email con un separador

```
-- Concatenar el nombre del cliente y su email con un separador
SELECT CONCAT(nombre, ' / ', email) AS nombre_email FROM clientes;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombre_email
Juan Pérez / juan.perez@example.com
María López / maria.lopez@example.com
Carlos García / carlos.garcia@example.com
Ana Fernández / ana.fernandez@example.com
Luis Martínez / luis.martinez@example.com
Marta Gómez / marta.gomez@example.com
José Sánchez / jose.sanchez@example.com
Laura Díaz / laura.diaz@example.com
Pedro Hernández / pedro.hernandez@example....
Elena Ruiz / elena.ruiz@example.com
Javier Ramos / javier.ramos@example.com
Carmen Morales / carmen.morales@example.com
Raúl Ortiz / raul.ortiz@example.com
Lucía Medina / lucia.medina@example.com
Diego Romero / diego.romero@example.com
Sara Alonso / sara.alonso@example.com
Miguel Torres / miguel.torres@example.com

- Extraer los últimos 3 caracteres del email de cada cliente

```
-- Extraer los últimos 3 caracteres del email de cada cliente
SELECT SUBSTRING(email, -3) AS ultimos_tres_caracteres_email FROM clientes;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ultimos_tres_caracteres_email
com
com
com
com
com
com
com
com
com
com
com
com
com
com
com
com

- Convertir el nombre del cliente a minúsculas

```
-- Convertir el nombre del cliente a minúsculas
SELECT LOWER(nombre) AS nombre_minusculas FROM clientes;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombre_minusculas
juan pérez
maría lópez
carlos garcía
ana fernández
luis martínez
marta gómez
josé sánchez
laura díaz
pedro hernández
elena ruiz
javier ramos
carmen morales
raúl ortiz
lucía medina
diego romero
sara alonso

- Convertir el nombre del cliente a mayúsculas

```
-- Convertir el nombre del cliente a minúsculas
SELECT UPPER(nombre) AS nombre_minusculas FROM clientes;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombre_minusculas
JUAN PÉREZ
MARÍA LÓPEZ
CARLOS GARCÍA
ANA FERNÁNDEZ
LUIS MARTÍNEZ
MARTA GÓMEZ
JOSÉ SÁNCHEZ
LAURA DÍAZ
PEDRO HERNÁNDEZ
ELENA RUIZ
JAVIER RAMOS
CARMEN MORALES
RAÚL ORTIZ
LUCÍA MEDINA
DIEGO ROMERO
SARA ALONSO
MIGUEL TORRES

- Seleccionar las transacciones ordenadas por monto de mayor a menor, mostrando solo las primeras 10

```
-- Seleccionar las transacciones ordenadas por monto de mayor a menor, mostrando solo las primeras 10
SELECT * FROM transacciones ORDER BY monto DESC LIMIT 10;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows:

id_transaccion	monto	fecha	id_cuenta
5	300.600	2024-07-05	5
11	245.000	2024-07-11	11
16	230.000	2024-07-16	16
14	220.500	2024-07-14	14
19	210.000	2024-07-19	19
2	200.750	2024-07-02	2
13	180.250	2024-07-13	13
17	180.000	2024-07-17	17
3	150.000	2024-07-03	3
15	150.000	2024-07-15	15
NULL	NULL	NULL	NULL

- Calcular el saldo de cada cuenta después de aplicar un interés del 5%

```sql
-- Calcular el saldo de cada cuenta después de aplicar un interés del 5%
SELECT id_cuenta, saldo, saldo * 1.05 AS saldo_con_interes FROM cuentas;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id_cuenta	saldo	saldo_con_interes
4	500.25	525.2625
5	3200.60	3360.6300
6	2100.00	2205.0000
7	750.90	788.4450
8	940.40	987.4200
9	1200.00	1260.0000
10	1330.80	1397.3400
11	2450.00	2572.5000
12	760.00	798.0000
13	1800.25	1890.2625
14	2200.50	2310.5250
15	1500.00	1575.0000
16	2300.00	2415.0000
17	1800.00	1890.0000
18	1300.50	1365.5250
19	2100.00	2205.0000

- Reemplazar el dominio del email de los clientes: email, '@example.com', '@newdomain.com'

```sql
-- Reemplazar el dominio del email de los clientes: email, '@example.com', '@newdomain.com'
SELECT email, REPLACE(email, '@example.com', '@newdomain.com') AS nuevo_email FROM clientes;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

email	nuevo_email
ana.fernandez@example.com	ana.fernandez@newdomain.com
antonio.gil@example.com	antonio.gil@newdomain.com
carlos.garcia@example.com	carlos.garcia@newdomain.com
carmen.morales@example.com	carmen.morales@newdomain.com
diego.romero@example.com	diego.romero@newdomain.com
elena.ruiz@example.com	elena.ruiz@newdomain.com
isabel.castro@example.com	isabel.castro@newdomain.com
javier.ramos@example.com	javier.ramos@newdomain.com
jose.sanchez@example.com	jose.sanchez@newdomain.com
juan.perez@example.com	juan.perez@newdomain.com
laura.diaz@example.com	laura.diaz@newdomain.com
lucia.medina@example.com	lucia.medina@newdomain.com
luis.martinez@example.com	luis.martinez@newdomain.com
maria.lopez@example.com	maria.lopez@newdomain.com
marta.gomez@example.com	marta.gomez@newdomain.com
miguel.torres@example.com	miguel.torres@newdomain.com
nuria.soto@example.com	nuria.soto@newdomain.com

Ejercicio 2: Uso de Funciones de Cadena

- Selecciona el nombre de cada cliente.

```
133    -- Selecciona el nombre de cada cliente.
134    SELECT nombre FROM clientes;
135
```

Result Grid | Filter Rows: | Export: | Wrap Cell

nombre
Juan Pérez
María López
Carlos García
Ana Fernández
Luis Martínez
Marta Gómez
José Sánchez
Laura Díaz
Pedro Hernández
Elena Ruiz
Javier Ramos
Carmen Morales
Raúl Ortiz
Lucía Medina
Diego Romero
Sara Alonso
Miguel Torres

- Añade una columna que muestre el email del cliente en el formato Correo: [email].

```
136    --  Añade una columna que muestre el email del cliente en el formato Correo: [email].
137 •  SELECT nombre, CONCAT('Correo: ', email) AS formato_email FROM clientes;
138
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombre	formato_email
Juan Pérez	Correo: juan.perez@example.com
María López	Correo: maria.lopez@example.com
Carlos García	Correo: carlos.garcia@example.com
Ana Fernández	Correo: ana.fernandez@example.com
Luis Martínez	Correo: luis.martinez@example.com
Marta Gómez	Correo: marta.gomez@example.com
José Sánchez	Correo: jose.sanchez@example.com
Laura Díaz	Correo: laura.diaz@example.com
Pedro Hernández	Correo: pedro.hernandez@example.com
Elena Ruiz	Correo: elena.ruiz@example.com
Javier Ramos	Correo: javier.ramos@example.com
Carmen Morales	Correo: carmen.morales@example.com
Raúl Ortiz	Correo: raul.ortiz@example.com
Lucía Medina	Correo: lucia.medina@example.com
Diego Romero	Correo: diego.romero@example.com
Sara Alonso	Correo: sara.alonso@example.com
Miguel Torres	Correo: miguel.torres@example.com

- Mostrar los emails de los clientes en minúsculas

```
139    -- Mostrar los emails de los clientes en minúsculas
140 •  SELECT LOWER(email) AS email_minusculas FROM clientes;
141
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content

email_minusculas
ana.fernandez@example.com
antonio.gil@example.com
carlos.garcia@example.com
carmen.morales@example.com
diego.romero@example.com
elena.ruiz@example.com
isabel.castro@example.com
javier.ramos@example.com
jose.sanchez@example.com
juan.perez@example.com
laura.diaz@example.com
lucia.medina@example.com
luis.martinez@example.com
maria.lopez@example.com
marta.gomez@example.com
miguel.torres@example.com
nuria.soto@example.com

- Mostrar el nombre del cliente en mayúsculas

```
142    -- Mostrar el nombre del cliente en mayúsculas
143    SELECT UPPER(nombre) AS nombre_mayusculas FROM clientes;
144
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombre_mayusculas
JUAN PÉREZ
MARÍA LÓPEZ
CARLOS GARCÍA
ANA FERNÁNDEZ
LUIS MARTÍNEZ
MARTA GÓMEZ
JOSÉ SÁNCHEZ
LAURA DÍAZ
PEDRO HERNÁNDEZ
ELENA RUIZ
JAVIER RAMOS
CARMEN MORALES
RAÚL ORTIZ
LUCÍA MEDINA
DIEGO ROMERO
SARA ALONSO
MIGUEL TORRES

Ejercicio 3: Agregación de Datos

- Selecciona el id_cuenta.

```sql
-- Selecciona el id_cuenta.
SELECT id_cuenta FROM cuentas;
```

Result Grid | Filter Rows:

id_cuenta
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16

- Calcula el total de las transacciones realizadas en cada cuenta.

```sql
-- Calcula el total de las transacciones realizadas en cada cuenta.
SELECT id_cuenta, SUM(monto) AS total_transacciones FROM transacciones GROUP BY id_cuenta;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id_cuenta	total_transacciones
1	100.500
2	200.750
3	150.000
4	50.250
5	300.600
6	100.000
7	75.900
8	90.400
9	120.000
10	130.800
11	245.000
12	76.000
13	180.250
14	220.500
15	150.000
16	230.000

- Calcular el saldo total de todas las cuentas

```
-- Calcular el saldo total de todas las cuentas
SELECT SUM(saldo) AS saldo_total FROM cuentas;
```

Result Grid | Filter Rows: | Export: | Wrap C

saldo_total
33335.45

- Calcular el monto total de todas las transacciones

```
-- Calcular el monto total de todas las transacciones
SELECT SUM(monto) AS monto_total_transacciones FROM transacciones;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

monto_total_transacciones
3091.450

- Calcular el monto promedio de las transacciones

```
-- Calcular el monto promedio de las transacciones
SELECT AVG(monto) AS monto_promedio FROM transacciones;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

monto_promedio
154.5725000

Ejercicio 4: Funciones de Fecha:

- Selecciona el id_transaccion y la fecha de todas las transacciones.

```
-- Selecciona el id_transaccion y la fecha de todas las transacciones.
SELECT id_transaccion, fecha FROM transacciones;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap C

id_transaccion	fecha
1	2024-07-01
2	2024-07-02
3	2024-07-03
4	2024-07-04
5	2024-07-05
6	2024-07-06
7	2024-07-07
8	2024-07-08
9	2024-07-09
10	2024-07-10
11	2024-07-11
12	2024-07-12
13	2024-07-13
14	2024-07-14
15	2024-07-15
16	2024-07-16
17	2024-07-17

- Extraer el año de la fecha de cada transacción

```
-- Extraer el año de la fecha de cada transacción
SELECT id_transaccion, YEAR(fecha) AS año FROM transacciones;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id_transaccion	año
1	2024
2	2024
3	2024
4	2024
5	2024
6	2024
7	2024
8	2024
9	2024
10	2024
11	2024
12	2024
13	2024
14	2024
15	2024
16	2024
17	2024

- Extraer el mes de la fecha de cada transacción

```
-- Extraer el mes de la fecha de cada transacción
SELECT id_transaccion, MONTH(fecha) AS mes FROM transacciones;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id_transaccion	mes
1	7
2	7
3	7
4	7
5	7
6	7
7	7
8	7
9	7
10	7
11	7
12	7
13	7
14	7
15	7
16	7
17	7

- Extraer el día de la semana de la fecha de cada transacción

```
-- Extraer el día de la semana de la fecha de cada transacción
SELECT id_transaccion, DAYOFWEEK(fecha) AS dia_semana FROM transacciones;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id_transaccion	dia_semana
1	2
2	3
3	4
4	5
5	6
6	7
7	1
8	2
9	3
10	4
11	5
12	6
13	7
14	1
15	2
16	3
17	4

- Filtra las transacciones que se realizaron en el año 2024.

```
-- Filtra las transacciones que se realizaron en el año 2024.
SELECT * FROM transacciones WHERE YEAR(fecha) = 2024;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

id_transaccion	monto	fecha	id_cuenta
1	100.500	2024-07-01	1
2	200.750	2024-07-02	2
3	150.000	2024-07-03	3
4	50.250	2024-07-04	4
5	300.600	2024-07-05	5
6	100.000	2024-07-06	6
7	75.900	2024-07-07	7
8	90.400	2024-07-08	8
9	120.000	2024-07-09	9
10	130.800	2024-07-10	10
11	245.000	2024-07-11	11
12	76.000	2024-07-12	12
13	180.250	2024-07-13	13
14	220.500	2024-07-14	14
15	150.000	2024-07-15	15
16	230.000	2024-07-16	16
17	180.000	2024-07-17	17

- Seleccionar las transacciones realizadas en 2023.

```
-- Filtra las transacciones que se realizaron en el año 2023.
update transacciones set fecha = '2023-06-02' where id_transaccion = 1;
update transacciones set fecha = '2023-07-02' where id_transaccion = 2;
update transacciones set fecha = '2023-08-02' where id_transaccion = 3;

SELECT * FROM transacciones WHERE YEAR(fecha) = 2023;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap

id_transaccion	monto	fecha	id_cuenta
1	100.500	2023-06-02	1
2	200.750	2023-07-02	2
3	150.000	2023-08-02	3
NULL	NULL	NULL	NULL

- Seleccionar las transacciones realizadas en el mes de junio de cualquier año

```
-- Seleccionar las transacciones realizadas en el mes de junio de cualquier año
SELECT * FROM transacciones WHERE MONTH(fecha) = 6;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_transaccion	monto	fecha	id_cuenta
1	100.500	2023-06-02	1
NULL	NULL	NULL	NULL

- Seleccionar las transacciones realizadas el 15 de cada mes

```
-- Seleccionar las transacciones realizadas el 15 de cada mes
SELECT * FROM transacciones WHERE DAY(fecha) = 15;
```

Result Grid | Filter Rows: | Edit: | Export/Import:

id_transaccion	monto	fecha	id_cuenta
15	150.000	2024-07-15	15
NULL	NULL	NULL	NULL

Ejercicio 5: Agrupación y Contar Resultados

Objetivo: Contar el número de transacciones por cuenta.

- Selecciona el id_cuenta.

```
-- Selecciona el id_cuenta.
SELECT id_cuenta FROM cuentas;
```

Result Grid | Filter Rows:

id_cuenta
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16

- Cuenta el número de transacciones realizadas en cada cuenta.

```
-- Cuenta el número de transacciones realizadas en cada cuenta.
SELECT id_cuenta, COUNT(*) AS numero_transacciones FROM transacciones GROUP BY id_cuenta;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id_cuenta	numero_transacciones
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	1
14	1
15	1
16	1
17	1

- Calcular el saldo promedio de todas las cuentas.

```
-- Calcular el saldo promedio de todas las cuentas.
SELECT avg(saldo) as saldoTotal from cuentas;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Con

saldoTotal
1666.772500

Resets all sorted columns

Ejercicio 6: Filtración con Condiciones

- Seleccionar las transacciones con un monto negativo

```
194    -- Seleccionar las transacciones con un monto negativo
195 •  SELECT id_transaccion, monto as MontoNegativo from transacciones where monto<0;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content

id_transaccion	MontoNegativo
NULL	NULL

- Seleccionar las transacciones con un monto positivo

```
196    -- Seleccionar las transacciones con un monto positivo
197 •  SELECT id_transaccion, monto as MontoPositivo from transacciones where monto>0;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Conten

id_transaccion	MontoPositivo
1	100.500
2	200.750
3	150.000
4	50.250
5	300.600
6	100.000
7	75.900
8	90.400
9	120.000
10	130.800
11	245.000
12	76.000

Ejercicio 7: Ordenamiento

- Seleccionar todas las cuentas ordenadas por saldo de mayor a menor

```
198    -- Seleccionar todas las cuentas ordenadas por saldo de mayor a menor
199 •  SELECT id_cuenta, saldo from cuentas order by saldo desc;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | W

id_cuenta	saldo
5	3200.60
11	2450.00
2	2300.75
16	2300.00
14	2200.50
6	2100.00
19	2100.00
13	1800.25
3	1800.00
17	1800.00
1	1500.50
15	1500.00

- Seleccionar todos los clientes ordenados alfabéticamente por nombre

```
198    -- Seleccionar todos los clientes ordenados alfabéticamente por nombre
199 •  SELECT nombre from clientes order by nombre asc;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

nombre
Ana Fernández
Antonio Gil
Carlos García
Carmen Morales
Diego Romero
Elena Ruiz
Isabel Castro
Javier Ramos
José Sánchez
Juan Pérez
Laura Díaz
Lucía Medina

- Seleccionar las transacciones realizadas en 2023

```
-- Seleccionar las transacciones realizadas en 2023
SELECT id_transaccion, fecha as Transacciones2023 from transacciones where year(fecha) = 2023 order by fecha asc;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_transaccion	Transacciones2023
1	2023-06-02
2	2023-07-02
3	2023-08-02
NULL	NULL

- Seleccionar las cuentas ordenadas por saldo de menor a mayor, mostrando solo las primeras 5

```
-- Seleccionar las cuentas ordenadas por saldo de menor a mayor, mostrando solo las primeras 5
SELECT id_cuenta, saldo as saldoMenor from cuentas order by saldo asc limit 5;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows:

id_cuenta	saldoMenor
4	500.25
7	750.90
12	760.00
8	940.40
9	1200.00

Ejercicio 8: Contar Resultados

- Contar el número total de clientes

```
-- Contar el número total de clientes
SELECT count(*) as numeroClientes from clientes;
```

Result Grid | Filter Rows: | Export: | Wrap C

numeroClientes
20

- Contar el número total de cuentas

```
SELECT count(*) as numeroCuentas from cuentas;
```

Result Grid | Filter Rows: | Export: | Wrap

numeroCuentas
20

- Contar el número total de transacciones

```
-- Contar el número total de transacciones
SELECT count(*) as numeroTransacciones from transacciones;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

numeroTransacciones
20

- Redondear el saldo de cada cuenta a dos decimales

```
-- Redondear el saldo de cada cuenta a dos decimales
SELECT round(saldo,2) as saldoRedondeado from cuentas;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content

saldoRedondeado
1500.50
2300.75
1800.00
500.25
3200.60
2100.00
750.90
940.40
1200.00
1330.80
2450.00
760.00

Ejercicio 9: Máximos y Mínimos

- Encontrar la cuenta con el saldo más alto

```
-- Encontrar la cuenta con el saldo más alto
SELECT id_cuenta, saldo as saldoAlto from cuentas where saldo = (select max(saldo) from cuentas);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_cuenta	saldoAlto
5	3200.60

- Encontrar la cuenta con el saldo más bajo

```
-- Encontrar la cuenta con el saldo más bajo
SELECT id_cuenta, saldo as saldoBajo from cuentas where saldo = (select min(saldo) from cuentas);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_cuenta	saldoBajo
4	500.25

- Encontrar la transacción con el monto más alto

```
-- Encontrar la transacción con el monto más alto
SELECT id_transaccion, monto as montoAlto from transacciones where monto = (select max(monto) from transacciones);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_transaccion	montoAlto
5	300.600

- Encontrar la transacción con el monto más bajo

```
-- Encontrar la transacción con el monto más bajo
SELECT id_transaccion, monto as montoBajo from transacciones where monto = (select min(monto) from transacciones);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id_transaccion	montoBajo
4	50.250

Enlace de GitHub: https://github.com/Mateor13/Consultas.git

PRESENTACIÓN

- PRESENTAR EN PDF
- SUBIR A GIT HUB
- Y SUBIR EL SQL-QUERY

PRACTICAR POR FAVOR PARA LA SIGUIENTE CLASE UN TEST DE CONOCIMIENTO, TEORICO Y PRACTICO.