



ESCUELA POLITÉCNICA NACIONAL

FACULTAD: ESFOT

PERÍODO ACADÉMICO: 2024 A

ASIGNATURA: BASES DE DATOS

DOCENTE: MSc. Yadira Franco

PARALELO: GR1

NOMBRE DEL ESTUDIANTE (S): Catucuamba Ariel – Valenzuela Isaac

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

```
1 • create table clientes(  
2     id_cliente int primary key auto_increment not null,  
3     nombre varchar(50),  
4     email varchar(50)  
5 );
```

Cuentas

id_cuenta,id_cliente,saldo

```

create table cuentas(
  id_cuenta int primary key auto_increment not null,
  saldo decimal(8,2),
  id_cliente int,
  foreign key (id_cliente) references clientes(id_cliente)
);

```

Transacciones

id_transaccion, id_cuenta, monto, fecha

```

create table transacciones(
  id_transaccion int auto_increment primary key not null,
  monto decimal(8,2),
  fecha timestamp default current_timestamp,
  id_cuenta int,
  foreign key (id_cuenta) references cuentas(id_cuenta)
);

```

Relaciones:

1. Cada cliente puede tener múltiples cuentas, pero cada cuenta pertenece a un único cliente.

```
foreign key (id_cliente) references clientes(id_cliente)
```

2. Cada cuenta puede tener múltiples transacciones, pero cada transacción está asociada a una sola cuenta.

```
foreign key (id_cuenta) references cuentas(id_cuenta)
```

3. Insertar (20 registro)

```
insert into clientes(nombre,email) values ('Ariel Catucuamba', 'a@gmail.com'),
('David Diaz', 'b@gmail.com'),('Carlos Hernández', 'c@gmail.com'),
('Mishell Melba', 'd@gmail.com'),('Luis Martínez', 'e@gmail.com'),
('Jimena Díaz', 'f@gmail.com'),('Juan Perez', 'g@gmail.com'),
('María García', 'h@gmail.com'),('Carlos Hernández', 'i@gmail.com'),
('Fausto Catucuamba', 'j@gmail.com'),('Luis Martínez', 'k@gmail.com'),
('Elena Rodríguez', 'l@gmail.com'),('Juan Perez', 'm@gmail.com'),
('María García', 'n@gmail.com'),('Carlos Hernández', 'o@gmail.com'),
('Ana López', 'p@gmail.com'),('Luis Martínez', 'q@gmail.com'),
('Elena Rodríguez', 'r@gmail.com'),('Sofía Sánchez', 's@gmail.com'),
('Miguel Torres', 't@gmail.com');

insert into Cuentas (saldo, id_cliente) values(1000.00, 1),
(1500.50, 1),(2000.75, 2),(2500.25, 3),(3000.00, 4),(3500.75, 5),(4000.50, 6),(4500.25, 7),(5000.00, 8),
(5500.75, 9),(6000.50, 10),(6500.25, 11),(7000.00, 12),(7500.00, 10),(8000.50, 11),(8500.75, 12),(9000.25, 13),
(9500.00, 14),(10000.75, 15),(10500.50, 16);

insert into Transacciones (monto, id_cuenta) values(100.00, 1),(200.50, 2),(300.75, 3),(400.25, 4),
(500.00, 5),(600.75, 6),(700.50, 7),(800.25, 8),(900.00, 9),(1000.75, 10),
(1100.50, 11),(1200.25, 12),(1300.00, 13),(1400.75, 14),(1500.50, 15),(1600.25, 16),(1700.00, 17),(1800.75, 18),
(1900.50, 19),(2000.25, 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.

```
45 • select * from transacciones;
46 -- @ Selecciona el id_cuenta y el saldo de todas las cuentas.
47 • select id_cuenta , saldo from cuentas;
48
```

Result Grid		Filter Rows:	Edit:	Export/Import:	Wrap Cell Content:
id_cuenta	saldo				
10	5500.75				
11	6000.50				
12	6500.25				
13	7000.00				
14	7500.00				
15	8000.50				
16	8500.75				
17	9000.25				
18	9500.00				
19	10000.75				
20	10500.50				

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

```

48 -- @ Filtra las cuentas que tienen un saldo mayor a 4000.
49 • select * from cuentas where saldo>4000;
50 --

```

Result Grid			
Filter Rows:			
	id_cuenta	saldo	id_cliente
▶	7	4000.50	6
	8	4500.25	7
	9	5000.00	8
	10	5500.75	9
	11	6000.50	10
	12	6500.25	11
	13	7000.00	12
	14	7500.00	10
	15	8000.50	11
	16	8500.75	12
	17	9000.25	13
	18	9500.00	14
	19	10000.75	15

- Seleccionar las cuentas con saldo menor a 2000

```

50 -- @ Seleccionar las cuentas con saldo menor a 2000
51 • select * from cuentas where saldo<2000;

```

Result Grid			
Filter Rows:			
	id_cuenta	saldo	id_cliente
▶	1	1000.00	1
	2	1500.50	1
*	NULL	NULL	NULL

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

```

52 -- @ Mostrar los nombres de los clientes y la longitud de sus nombres
53 • select nombre, length(nombre) as logintud_de_nombres from clientes;

```

Result Grid		
Filter Rows:		
	nombre	logintud_de_nombres
▶	Ariel Catucucamba	16
	David Diaz	11
	Carlos Hernández	17
	Mishell Melba	13
	Luis Martínez	14
	Jimena Diaz	11
	Juan Perez	10
	María García	14
	Carlos Hernández	17
	Fausto Catucucamba	17
	Luis Martínez	14
	Elena Rodríguez	16
	Juan Perez	10

- Seleccionar las cuentas con saldo entre 1000 y 3000

```

54 -- @ Seleccionar las cuentas con saldo entre 1000 y 3000
55 • select * from cuentas where saldo >=1000 and saldo<=3000;
56
57 -- otra forma es con between
58 • select * from cuentas where saldo between 1000 and 3000;

```

Result Grid			
Filter Rows:			
Edit: Export/Import: Wrap Cell Content: FA			
id_cuenta	saldo	id_cliente	
1	1000.00	1	
2	1500.50	1	
3	2000.75	2	
4	2500.25	3	
5	3000.00	4	
6	NULL	NULL	

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

```

59 -- @ Obtener una lista de todos los clientes y sus cuentas, mostrando los valores NULL
60 • select clientes.id_cliente, clientes.nombre, cuentas.saldo
61 from clientes left join cuentas on clientes.id_cliente = cuentas.id_cliente;

```

17	Luis Martínez	NULL
18	Elena Rodríguez	NULL
19	Sofía Sánchez	NULL
20	Miguel Torres	NULL

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

```

61 -- @ Concatenar el nombre del cliente y su email con un separador
62 • select concat(nombre, " ", email) as nombre_cliente from clientes;

```

Result Grid	
Filter Rows:	
Export: Wrap Cell Content: FA	
nombre_cliente	
Ariel Catucucamba a@gmail.com	
David Diaz b@gmail.com	
Carlos Hernández c@gmail.com	
Mishell Melba d@gmail.com	
Luis Martínez e@gmail.com	
Jimena Diaz f@gmail.com	
Juan Perez g@gmail.com	
María García h@gmail.com	
Carlos Hernández i@gmail.com	
Fausto Catucucamba j@gmail.com	
Luis Martínez k@gmail.com	
Elena Rodríguez l@gmail.com	
Juan Perez m@gmail.com	

- Extraer los últimos 3 caracteres del email de cada cliente

```
64 -- ❷ Extraer los últimos 3 caracteres del email de cada cliente
65 • select right(email,3) from clientes;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

right(email,3)
com
com
com
com
com
com
com
com
com
com
com
com
com
com

- Convertir el nombre del cliente a minúsculas

```
67 -- ❸ Convertir el nombre del cliente a minúsculas
68 • select lower(nombre) from clientes;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	lower(nombre)
▶	ariel catucuamba
	david diaz
	carlos hernández
	mishell melba
	luis martínez
	jimena diaz
	juan perez
	maría garcía
	carlos hernández
	fausto catucuamba
	luis martínez
	elena rodríguez
	juan perez

- Convertir el nombre del cliente a mayúsculas

```
70 -- ❹ Convertir el nombre del cliente a mayúsculas
71 • select upper(nombre) from clientes;
```

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

upper(nombre)
ARIEL CATUCUAMBA
DAVID DIAZ
CARLOS HERNÁNDEZ
MISHELL MELBA
LUIS MARTÍNEZ
JIMENA DIAZ
JUAN PEREZ
MARÍA GARCÍA
CARLOS HERNÁNDEZ
FAUSTO CATUCUAMBA
LUIS MARTÍNEZ
ELENA RODRÍGUEZ
JUAN PEREZ

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

```

72 -- @ Seleccionar las transacciones ordenadas por monto de mayor a menor, mostrando solo las primeras 10
73 • select * from transacciones order by monto desc limit 10;
74 --

```

id_transaccion	monto	fecha	id_cuenta
20	2000.25	2024-07-02 19:34:23	20
19	1900.50	2024-07-02 19:34:23	19
18	1800.75	2024-07-02 19:34:23	18
17	1700.00	2024-07-02 19:34:23	17
16	1600.25	2024-07-02 19:34:23	16
15	1500.50	2024-07-02 19:34:23	15
14	1400.75	2024-07-02 19:34:23	14
13	1300.00	2024-07-02 19:34:23	13
12	1200.25	2024-07-02 19:34:23	12
11	1100.50	2024-07-02 19:34:23	11
NULL	NULL	NULL	NULL

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

```

75 • select saldo*1.5 as saldo_interes from cuentas;

```

saldo_interes
1500.000
2250.750
3001.125
3750.375
4500.000
5251.125
6000.750
6750.375
7500.000
8251.125
9000.750
9750.375
10500.000

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

SQL Editor Interface showing a sequence of SQL queries and their results.

```

71 • select upper(nombre) from clientes;
72 -- @ Seleccionar las transacciones ordenadas por monto de mayor a menor, mostrando solo las primeras 10
73 • select * from transacciones order by monto desc limit 10;
74 -- @ Calcular el saldo de cada cuenta después de aplicar un interés del 5%
75 • select saldo*1.5 as saldo_interes from cuentas;
76
77 -- @ Reemplazar el dominio del email de los clientes: email, '@example.com', '@newdomain.com'
78 • update clientes set email = replace(email, '@example.com', '@newdomain.com')
79 where email like '%@example.com';
80 • select * from clientes;
81 -- cambiar modo seguro 0 u anular el modo seguro 1

```

Result Grid showing the updated client data:

id_cliente	nombre	email
1	Ariel Catucumbamba	a@newdomain.com
2	David Diaz	b@newdomain.com
3	Carlos Hernández	c@newdomain.com
4	Mishell Melba	d@newdomain.com
5	Luis Martinez	e@newdomain.com
6	Jimena Diaz	f@newdomain.com
7	Juan Perez	g@newdomain.com
8	Maria Garcia	h@newdomain.com
9	Carlos Hernández	i@newdomain.com
10	Fausto Catucumbamba	j@newdomain.com
11	Luis Martinez	k@newdomain.com
12	Elena Rodriguez	l@newdomain.com

clients 21 x

Ejercicio 2: Uso de Funciones de Cadena

- Selecciona el nombre de cada cliente.

```
83 -- [i] Selecciona el nombre de cada cliente.
84 • select nombre from clientes;
```

85

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

nombre
Ariel Catucuaamba
David Diaz
Carlos Hernández
Mishell Melba
Luis Martínez
Jimena Diaz
Juan Perez
María García
Carlos Hernández
Fausto Catucuaamba
Luis Martínez
Elena Rodríguez

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

```
85 -- [i] Añade una columna que muestre el email del cliente en el formato Correo: [email].
86 • select nombre,email from clientes;
```

87

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

nombre	email
Ariel Catucuaamba	a@newdomain.com
David Diaz	b@newdomain.com
Carlos Hernández	c@newdomain.com
Mishell Melba	d@newdomain.com
Luis Martínez	e@newdomain.com
Jimena Diaz	f@newdomain.com
Juan Perez	g@newdomain.com
María García	h@newdomain.com
Carlos Hernández	i@newdomain.com
Fausto Catucuaamba	j@newdomain.com
Luis Martínez	k@newdomain.com
Elena Rodríguez	l@newdomain.com

- Mostrar los emails de los clientes en minúsculas

```
87 -- [i] Mostrar los emails de los clientes en minúsculas
88 • select lower(email) from clientes;
```

89

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

lower(email)
a@newdomain.com
b@newdomain.com
c@newdomain.com
d@newdomain.com
e@newdomain.com
f@newdomain.com
g@newdomain.com
h@newdomain.com
i@newdomain.com
j@newdomain.com
k@newdomain.com
l@newdomain.com

- Mostrar el nombre del cliente en mayúsculas


```

89 -- @ Mostrar el nombre del cliente en mayúsculas
90 • select upper(email) from clientes;
91

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

upper(email)
A@NEWDOMAIN.COM
B@NEWDOMAIN.COM
C@NEWDOMAIN.COM
D@NEWDOMAIN.COM
E@NEWDOMAIN.COM
F@NEWDOMAIN.COM
G@NEWDOMAIN.COM
H@NEWDOMAIN.COM
I@NEWDOMAIN.COM
J@NEWDOMAIN.COM
K@NEWDOMAIN.COM
L@NEWDOMAIN.COM

Ejercicio 3: Agregación de Datos

- Selecciona el id_cuenta.

```

91 -- @ Selecciona el id_cuenta.
92 • select id_cuenta from cuentas;
93

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: [IA](#)

id_cuenta
1
2
3
4
5
6
7
8
9
10
11
14

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

```

93 -- @ Calcula el total de las transacciones realizadas en cada cuenta.
94 • select c.id_cuenta, COUNT(t.id_transaccion) as total_transacciones, SUM(t.monto) as monto_total
95 from cuentas c left join transacciones t on c.id_cuenta = t.id_cuenta
96 group by c.id_cuenta;
97

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

id_cuenta	total_transacciones	monto_total
1	1	100.00
2	1	200.50
3	1	300.75
4	1	400.25
5	1	500.00
6	1	600.75
7	1	700.50
8	1	800.25
9	1	900.00
10	1	1000.75
11	1	1100.50
12	1	1200.25

- Calcular el saldo total de todas las cuentas

```

98  --  Calcular el saldo total de todas las cuentas
99  •  select sum(saldo) as total_saldo from cuentas;
100

```

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

total_saldo
115007.25

- Calcular el monto total de todas las transacciones

```

101  --  Calcular el monto total de todas las transacciones
102  •  select sum(monto) as monto_transacciones from transacciones;
103

```

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

monto_transacciones
21007.50

- Calcular el monto promedio de las transacciones

```

104  --  Calcular el monto promedio de las transacciones
105  •  select avg(monto) as promedio_transacciones from transacciones;
106

```

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

promedio_transacciones
1050.375000

Ejercicio 4: Funciones de Fecha:

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

```





107  --  Selecciona el id_transaccion y la fecha de todas las transacciones.
108  •  select id_transaccion, fecha from transacciones;
109





```

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

id_transaccion	fecha
3	2024-07-02 19:34:23
4	2024-07-02 19:34:23
5	2024-07-02 19:34:23
6	2024-07-02 19:34:23
7	2024-07-02 19:34:23
8	2024-07-02 19:34:23
9	2024-07-02 19:34:23
10	2024-07-02 19:34:23
11	2024-07-02 19:34:23
12	2024-07-02 19:34:23
13	2024-07-02 19:34:23
14	2024-07-02 19:34:23
15	2024-07-02 19:34:23
16	2024-07-02 19:34:23
17	2024-07-02 19:34:23
18	2024-07-02 19:34:23
19	2024-07-02 19:34:23
20	2024-07-02 19:34:23

- Extraer el año de la fecha de cada transacción

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

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

```

116 --  Extraer el día de la semana de la fecha de cada transacción
117 • select fecha,day(fecha) as dia_fecha from transacciones;

```

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

	fecha	dia_fecha
▶	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2
	2024-07-02 19:34:23	2

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

```

119 --  Filtra las transacciones que se realizaron en el año 2024.
120 • select * from transacciones where year(fecha)=2024;

```

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

	id_transaccion	monto	fecha	id_cuenta
▶	1	100.00	2024-07-02 19:34:23	1
	2	200.50	2024-07-02 19:34:23	2
	3	300.75	2024-07-02 19:34:23	3
	4	400.25	2024-07-02 19:34:23	4
	5	500.00	2024-07-02 19:34:23	5
	6	600.75	2024-07-02 19:34:23	6
	7	700.50	2024-07-02 19:34:23	7
	8	800.25	2024-07-02 19:34:23	8
	9	900.00	2024-07-02 19:34:23	9
	10	1000.75	2024-07-02 19:34:23	10
	11	1100.50	2024-07-02 19:34:23	11
	12	1200.25	2024-07-02 19:34:23	12
	13	1300.00	2024-07-02 19:34:23	13
	14	1400.75	2024-07-02 19:34:23	14
	15	1500.50	2024-07-02 19:34:23	15
	16	1600.25	2024-07-02 19:34:23	16
	17	1700.00	2024-07-02 19:34:23	17
	18	1800.75	2024-07-02 19:34:23	18
	19	1900.50	2024-07-02 19:34:23	19

- Seleccionar las transacciones realizadas en 2023.

```

121 -- @ Seleccionar las transacciones realizadas en 2023.
122 • alter table transacciones add column fecha2 date;
123 • select * from transacciones;
124 • insert into Transacciones (monto, id_cuenta, fecha2) values(100.00, 1, "2023-12-11");
125 • select * from transacciones where year(fecha2)=2023;

```

Result Grid

	id_transaccion	monto	fecha	id_cuenta	fecha2
▶	21	100.00	2024-07-02 21:16:10	1	2023-12-11
•		NULL	NULL	NULL	NULL

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

```

127 -- @ Seleccionar las transacciones realizadas en el mes de junio de cualquier año
128 • insert into Transacciones (monto, id_cuenta, fecha2) values(100.00, 1, "2023-6-11");
129 • select * from transacciones where month(fecha2)=6;

```

Result Grid

	id_transaccion	monto	fecha	id_cuenta	fecha2
▶	22	100.00	2024-07-02 21:17:43	1	2023-06-11
•		NULL	NULL	NULL	NULL

- Seleccionar las transacciones realizadas el 15 de cada mes

```

130 -- @ Seleccionar las transacciones realizadas el 15 de cada mes
131 • insert into Transacciones (monto, id_cuenta, fecha2) values(100.00, 1, "2023-6-15");
132 • select fecha2 from transacciones where day(fecha2)=15;

```

Result Grid

	fecha2
▶	2023-06-15

Ejercicio 5: Agrupación y Contar Resultados

Objetivo: Contar el número de transacciones por cuenta.

- Selecciona el id_cuenta.

```

134 -- @ Selecciona el id_cuenta.
135 • select id_cuenta from cuentas;

```

id_cuenta
1
2
3
4
5
6
7
8
9
10
11
14
12
15
13
16
17
18
19

- Cuenta el número de transacciones realizadas en cada cuenta.

```

137 -- @ Cuenta el número de transacciones realizadas en cada cuenta.
138 • select id_cuenta, COUNT(*) AS numero_transacciones
139 from transacciones
140 group by id_cuenta;
141

```

id_cuenta	numero_transacciones
1	4
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
18	1
19	1

- Calcular el saldo promedio de todas las cuentas.

```

142 -- @ Calcular el saldo promedio de todas las cuentas.
143 • select avg(saldo) from cuentas;

```

avg(saldo)
5750.362500

Ejercicio 6: Filtración con Condiciones

- Seleccionar las transacciones con un monto negativo

No hay montos negativos

```
145 --  Seleccionar las transacciones con un monto negativo
146 • select * from transacciones where monto<0;
```

Result Grid

	id_transaccion	monto	fecha	id_cuenta	fecha2
*	NULL	NULL	NULL	NULL	NULL

- Seleccionar las transacciones con un monto positivo

```
48 --  Seleccionar las transacciones con un monto positivo
49 • select * from transacciones where monto>=0;
```

Result Grid

	id_transaccion	monto	fecha	id_cuenta
1	1	100.00	2024-07-02 19:34:23	1
2	2	200.50	2024-07-02 19:34:23	2
3	3	300.75	2024-07-02 19:34:23	3
4	4	400.25	2024-07-02 19:34:23	4
5	5	500.00	2024-07-02 19:34:23	5
6	6	600.75	2024-07-02 19:34:23	6
7	7	700.50	2024-07-02 19:34:23	7
8	8	800.25	2024-07-02 19:34:23	8
9	9	900.00	2024-07-02 19:34:23	9
10	10	1000.75	2024-07-02 19:34:23	10
11	11	1100.50	2024-07-02 19:34:23	11
12	12	1200.25	2024-07-02 19:34:23	12
13	13	1300.00	2024-07-02 19:34:23	13
14	14	1400.75	2024-07-02 19:34:23	14
15	15	1500.50	2024-07-02 19:34:23	15
16	16	1600.25	2024-07-02 19:34:23	16
17	17	1700.00	2024-07-02 19:34:23	17
18	18	1800.75	2024-07-02 19:34:23	18
19	19	1900.50	2024-07-02 19:34:23	19

Ejercicio 7: Ordenamiento

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

```
--
52  -- Seleccionar todas las cuentas ordenadas por saldo de mayor a menor:
53  • select * from cuentas order by saldo desc;
```

id_cuenta	saldo	id_cliente
20	10500.50	16
19	10000.75	15
18	9500.00	14
17	9000.25	13
16	8500.75	12
15	8000.50	11
14	7500.00	10
13	7000.00	12
12	6500.25	11
11	6000.50	10
10	5500.75	9
9	5000.00	8
8	4500.25	7
7	4000.50	6
6	3500.75	5
5	3000.00	4
4	2500.25	3
3	2000.75	2
2	1500.50	1

- Seleccionar todos los clientes ordenados alfabéticamente por nombre

```
155  -- ■ Seleccionar todos los clientes ordenados alfabéticamente por nombre
156  • select * from clientes order by nombre asc;
```

id_cliente	nombre	email
16	Ana López	p@newdomain.com
1	Ariel Catucuamba	a@newdomain.com
3	Carlos Hernández	c@newdomain.com
9	Carlos Hernández	i@newdomain.com
15	Carlos Hernández	o@newdomain.com
2	David Diaz	b@newdomain.com
12	Elena Rodríguez	l@newdomain.com
18	Elena Rodríguez	r@newdomain.com
10	Fausto Catucuamba	j@newdomain.com
6	Jimena Diaz	f@newdomain.com
7	Juan Perez	g@newdomain.com
13	Juan Perez	m@newdomain.com
5	Luis Martínez	e@newdomain.com
11	Luis Martínez	k@newdomain.com
17	Luis Martínez	q@newdomain.com
8	María García	h@newdomain.com
14	María García	n@newdomain.com
20	Miguel Torres	t@newdomain.com
4	Michell Melha	d@newdomain.com

- Seleccionar las transacciones realizadas en 2023


```

158 -- @ Seleccionar las transacciones realizadas en 2023
159 • select * from transacciones where year(fecha2)=2023;

```

Result Grid					
Filter Rows: <input type="text"/>					
Edit:					
Export/Import:					
Wrap Cell Content:					
	id_transaccion	monto	fecha	id_cuenta	fecha2
▶	24	100.00	2024-07-02 21:33:02	1	2023-12-11
•	NULL	NULL	NULL	NULL	NULL

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

```

61 -- @ Seleccionar las cuentas ordenadas por saldo de menor a mayor, mostrando solo las primeras 5
62 • select * from cuentas order by saldo asc limit 5;

```

Result Grid		
Filter Rows: <input type="text"/>		
Edit:		
Export/Import:		
Wrap Cell Content:		
Fetch rows:		
	id_cuenta	saldo
1	1	1000.00
2	2	1500.50
3	3	2000.75
4	4	2500.25
5	5	3000.00
•	NULL	NULL

Ejercicio 8: Contar Resultados

- Contar el número total de clientes

```

164 -- @ Contar el número total de clientes
165 • select count(*) as total_clientes from clientes;

```

Result Grid	
Filter Rows: <input type="text"/>	
Export:	
Wrap Cell Content:	
	total_clientes
▶	20

- Contar el número total de cuentas

```

167 -- @ Contar el número total de cuentas
168 • select count(*) from cuentas;

```

Result Grid	
Filter Rows: <input type="text"/>	
Export:	
Wrap Cell Content:	
	count(*)
▶	20

- Contar el número total de transacciones

```
170 -- @ Contar el número total de transacciones
171 • select count(*) from transacciones;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	count(*)			
▶	24			

- Redondear el saldo de cada cuenta a dos decimales

```
173 -- @ Redondear el saldo de cada cuenta a dos decimales
174 • select round(saldo,2) from cuentas;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	round(saldo,2)			
▶	1000.00			
	1500.50			
	2000.75			
	2500.25			
	3000.00			
	3500.75			
	4000.50			
	4500.25			
	5000.00			
	5500.75			
	6000.50			
	6500.25			
	7000.00			
	7500.00			
	8000.50			
	8500.75			
	9000.25			
	9500.00			
	10000.75			

Ejercicio 9: Máximos y Mínimos

- Encontrar la cuenta con el saldo más alto

```
176 -- @ Encontrar la cuenta con el saldo más alto
177 • select max(saldo) from cuentas;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	max(saldo)			
▶	10500.50			

- Encontrar la cuenta con el saldo más bajo

178

179 -- @ Encontrar la cuenta con el saldo más bajo

180 • select min(saldo) from cuentas;

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	min(saldo)			
▶	1000.00			

- Encontrar la transacción con el monto más alto

182 -- @ Encontrar la transacción con el monto más alto

183 • select max(monto) as montoMaximo from transacciones;

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	montoMaximo			
▶	2000.25			

- Encontrar la transacción con el monto más bajo

185 -- @ Encontrar la transacción con el monto más bajo

186 • select min(monto) from transacciones;

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	min(monto)			
▶	100.00			

```
187 -- Verificamos que sean el monto minimo y maximo
188 • select monto from transacciones order by monto asc;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
monto			
200.50			
300.75			
400.25			
500.00			
600.75			
700.50			
800.25			
900.00			
1000.75			
1100.50			
1200.25			
1300.00			
1400.75			
1500.50			
1600.25			
1700.00			
1800.75			
1900.50			
2000.25			

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,