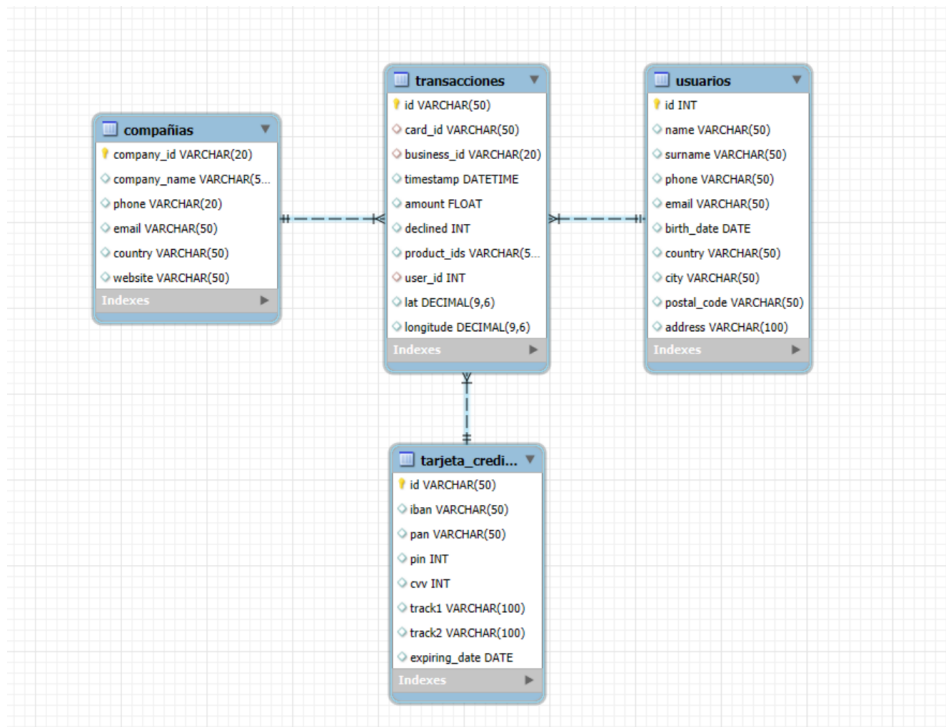


NIVEL 1.

Diagrama en Estrella resultante:



Ejercicio 1.

```

1404  -- Ejercicio 1. Realitza una subconsulta que mostri tots els usuaris amb més de 30 transaccions utilitzant almenys 2 taules.
1405
1406  • SELECT usuarios.id, usuarios.name, usuarios.surname
1407  FROM usuarios
1408  JOIN transacciones
1409  ON usuarios.id = transacciones.user_id
1410  GROUP BY usuarios.id, usuarios.name, usuarios.surname
1411  HAVING COUNT(transacciones.id) > 30;
1412

```

Result Grid			
Filter Rows:			
Export:			
Wrap Cell Content:			
id	name	surname	
92	Lynn	Riddle	
267	Ocean	Nelson	
272	Hedwig	Gilbert	
275	Kenyon	Hartman	

Ejercicio 2.

SPRINT 4* SQL File 6*

Limit to 200 rows

```

1  -- Exercici 2. Mostra la mitjana d'amount per IBAN de les targetes de crèdit a la companyia Donec Ltd, utilitza almenys 2 taules.
2  • SELECT AVG (amount) as monto_promedio, iban
3  FROM transacciones
4  JOIN tarjeta_credito
5  ON transacciones.card_id = tarjeta_credito.id
6  JOIN compañías
7  ON transacciones.business_id = compañías.company_id
8  WHERE company_name = 'Donec Ltd'
9  GROUP BY iban ;

```

Result Grid | Filter Rows: | Exports: | Wrap Cell Content: [1A](#)

monto_promedio	iban
203.71499252319336	PT87806228135092429456346

NIVEL 2.

Ejercicio 1.

```

1  -- Nivel 2
2  -- Crea una nova taula que reflecteixi l'estat de les targetes de crèdit basat en si les últimes tres transaccions van ser declinades i ge
3  -- Exercici 1
4  -- Quantes targetes estan actives?
5
6  -- Insertar datos en la tabla 'tarjeta_estado'
7  • INSERT INTO tarjeta_estado (card_id, estado)
8  SELECT
9  tarjeta_transaccion.card_id,
10 CASE
11 -- Contar el número de transacciones declinadas entre las últimas tres
12 WHEN SUM(CASE WHEN tarjeta_transaccion.declined = 1 THEN 1 ELSE 0 END) >= 3 THEN 'Inactiva'
13 ELSE 'Activa'
14 END AS estado_actual_de_tarjeta
15 FROM (
16 SELECT
17 transacciones.card_id,
18 transacciones.declined,
19 -- Asignar un número de fila a cada transacción por tarjeta, ordenado por la marca de tiempo de forma descendente
20 ROW_NUMBER() OVER (PARTITION BY transacciones.card_id ORDER BY transacciones.timestamp DESC) AS numero_de_fila
21 FROM
22 transacciones
23 ) AS tarjeta_transaccion
24 WHERE tarjeta_transaccion.numero_de_fila <= 3 -- Considerar solo las últimas tres transacciones
25 GROUP BY tarjeta_transaccion.card_id;
26

```

Tabla resultante:

The image shows two screenshots of the SQL Developer interface. The top screenshot displays a query in the 'SQL File 7*' window. The query is:

```
1 SELECT*
2 FROM tarjeta_estado;
```

Below the query, the 'Result Grid' shows a table with two columns: 'card_id' and 'estado'. The table contains 15 rows of data, all with the state 'Activa'.

card_id	estado
CcU-3337	Activa
CcU-3344	Activa
CcU-3351	Activa
CcU-3358	Activa
CcU-3365	Activa
CcU-3372	Activa
CcU-3379	Activa
CcU-3386	Activa
CcU-3393	Activa
CcU-3400	Activa
CcU-3407	Activa
CcU-3414	Activa
CcU-3421	Activa
CcU-3428	Activa
CcU-3435	Activa

The bottom screenshot shows a different query in the 'SQL File 7*' window. The query is:

```
1 -- Tarjetas que están activas
2 SELECT COUNT(*) AS numero_de_tarjetas_activas
3 FROM tarjeta_estado
4 WHERE estado = 'Activa';
```

Below the query, the 'Result Grid' shows a table with one column: 'numero_de_tarjetas_activas'. The table contains one row with the value 275.

numero_de_tarjetas_activas
275