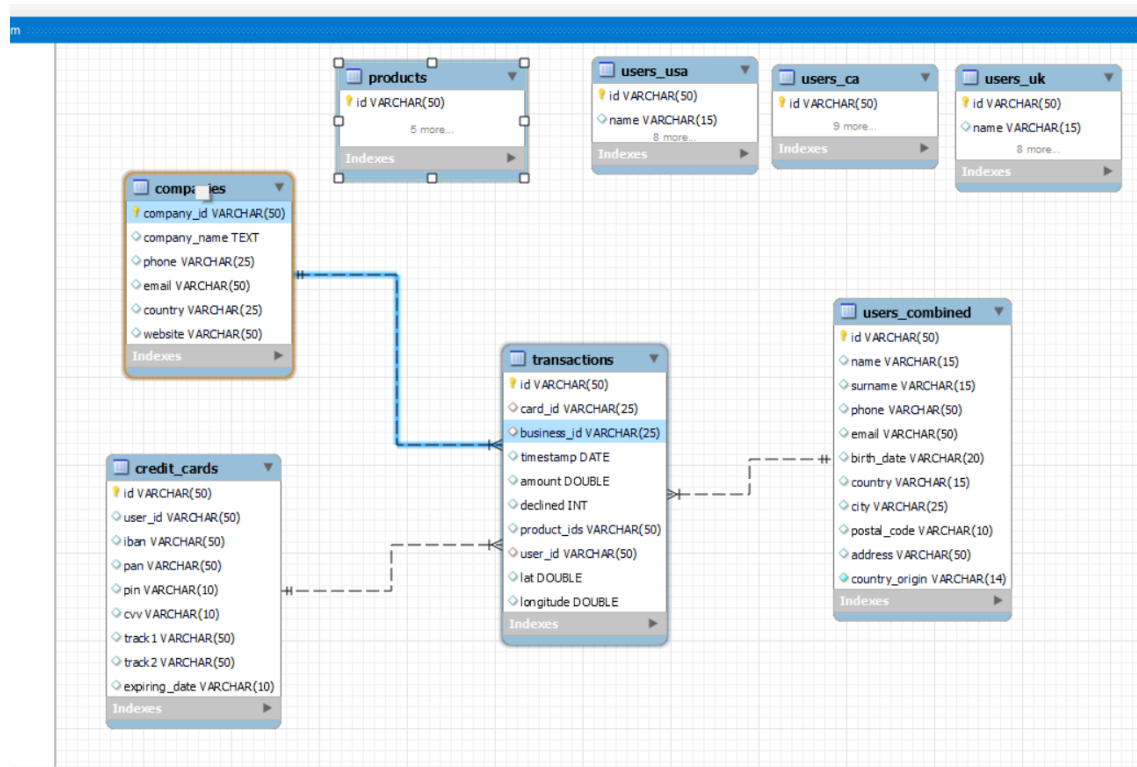


SPRINT 4:

Descàrrega els arxius CSV, estudia'ls i dissenya una base de dades amb un esquema d'estrella que contingui, almenys 4 taules de les quals puguis realitzar les següents consultes:



```

8  ##AÑADIMOS LAS PK Y FK DE LAS TABLAS:
9  #TABLA TRANSACTIONS:
10 • ALTER TABLE transactions
11   ADD PRIMARY KEY (id(50));
12
13 • ALTER TABLE COMPANIES
14   ADD PRIMARY KEY (company_id(50));
15
16 • ALTER TABLE CREDIT_CARDS
17   ADD PRIMARY KEY (id(50));
18
19 • ALTER TABLE PRODUCTS
20   ADD PRIMARY KEY (id);
21
22 • ALTER TABLE users_ca
23   ADD PRIMARY KEY (id);
24
25 • ALTER TABLE users_uk
26   ADD PRIMARY KEY (id);
27
28 • ALTER TABLE users_usa
29   ADD PRIMARY KEY (id);
30
31

```

```

12  #TABLA TRANSACTION CAMBIOS DE FORMATO DE LOS CAMPOS:

```

```

13 • ALTER TABLE transactions
14   MODIFY COLUMN id VARCHAR(50),
15   MODIFY COLUMN card_id VARCHAR(25),
16   MODIFY COLUMN business_id VARCHAR(25),
17   MODIFY COLUMN timestamp date,
18   MODIFY COLUMN product_ids VARCHAR(25);
19

```

```

10  #TABLA COMPANIES CAMBIOS DE FORMATO:

```

```

11 • ALTER TABLE companies
12   MODIFY COLUMN company_id VARCHAR(50),
13   MODIFY COLUMN phone VARCHAR(25),
14   MODIFY COLUMN email VARCHAR(50),
15   MODIFY COLUMN country VARCHAR(25),
16   MODIFY COLUMN website VARCHAR(50);
17

```

#TABLA credit_cards CAMBIOS DE FORMATO:

- ALTER TABLE credit_cards
MODIFY COLUMN user_id VARCHAR(50),
MODIFY COLUMN iban VARCHAR(50),
MODIFY COLUMN pan VARCHAR(50),
MODIFY COLUMN pin VARCHAR(10),
MODIFY COLUMN cvv VARCHAR(10),
MODIFY COLUMN track1 VARCHAR(50),
MODIFY COLUMN track2 VARCHAR(50),
MODIFY COLUMN expiring_date VARCHAR(10);

#TABLA PRODUCTS DE FORMATO:

- ALTER TABLE products
MODIFY COLUMN id VARCHAR(50),
MODIFY COLUMN product_name VARCHAR(50),
MODIFY COLUMN price VARCHAR(50),
MODIFY COLUMN colour VARCHAR(10),
MODIFY COLUMN weight VARCHAR(10),
MODIFY COLUMN warehouse_id VARCHAR(50);

```
MODIFY COLUMN surname VARCHAR(15),  
MODIFY COLUMN phone VARCHAR(50),  
MODIFY COLUMN email VARCHAR(50),  
MODIFY COLUMN birth_date VARCHAR(20),  
MODIFY COLUMN country VARCHAR(15),  
MODIFY COLUMN city VARCHAR(25),  
MODIFY COLUMN postal_code VARCHAR(10),  
MODIFY COLUMN address VARCHAR(50);
```

#TABLA users_usa DE FORMATO:

- ALTER TABLE users_usa
MODIFY COLUMN id VARCHAR(50),
MODIFY COLUMN name VARCHAR(15),
MODIFY COLUMN surname VARCHAR(15),
MODIFY COLUMN phone VARCHAR(50),
MODIFY COLUMN email VARCHAR(50),
MODIFY COLUMN birth_date VARCHAR(20),
MODIFY COLUMN country VARCHAR(15),
MODIFY COLUMN city VARCHAR(25),
MODIFY COLUMN postal_code VARCHAR(10),
MODIFY COLUMN address VARCHAR(50);

```

#combino la tablas de users para tener solo una
• CREATE TABLE users_combined AS
  SELECT *, 'Canada' AS country_origin FROM users_ca
  UNION ALL
  SELECT *, 'United Kingdom' AS country_origin FROM users_uk
  UNION ALL
  SELECT *, 'United States' AS country_origin FROM users_usa;

• ALTER TABLE users_combined
  ADD PRIMARY KEY (id);

# FK:

```

Exercici 1

Realitza una subconsulta que mostri tots els usuaris amb més de 30 transaccions utilitzant almenys 2 taules.

```

129
130 #Exercici 1
131 #Realitza una subconsulta que mostri tots els usuaris amb més de 30 transaccions utilitzant almenys 2 taules.
132
133 • SELECT u.id, u.name, u.email
134 FROM users_combined u
135 WHERE 30 < (
136     SELECT COUNT(*)
137     FROM transactions t
138     WHERE t.user_id = u.id
139 );
140
141

```

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

	id	name	email
▶	267	Ocean	aenean@yahoo.com
	272	Hedwig	sem.eget@idoud.edu
	275	Kenyon	convallis.ante.lectus@yahoo.com
	92	Lynn	vitae.aliquet@outlook.edu
*	NULL	NULL	NULL

users_combined 67 x

Output

Action Output

#	Time	Action	Message
1	12:43:58	SELECT u.id, u.name, u.email FROM users_combined u WHERE 30 < (SELECT COUNT(*) FROM t...	4 row(s) returned

LI DEMANO A LA TAULA USERS_COMBINED QUE EM SELECCIONI NOMÉS ALS USUARIS QUE HAGIN FET MÉS DE 30 TRANSACCIONS.

FAIG UNA SUBCONSULTA AMB LA TAULA TRANSACTIONS JA QUE LA INFO DEL ID DE LES TRANSACCIONS LA TINC EN AQUESTA TAULA.

- Exercici 2

Mostra la mitjana d'amount per IBAN de les targetes de crèdit a la companyia Donec Ltd, utilitza almenys 2 taules.

```
141
142  #- Exercici 2
143  #Mostra la mitjana d'amount per IBAN de les targetes de crèdit a la companyia Donec Ltd, utilitza almenys 2 taules.
144
145  • SELECT AVG(t.amount) as average_amount, c.iban
146     FROM transactions t
147     JOIN credit_cards c ON t.card_id = c.id
148     join companies co on t.business_id = co.company_id
149     WHERE co.company_name = 'Donec Ltd'
150     GROUP BY c.iban;
```

Result Grid

average_amount	iban
203.715	PT87806228135092429456346

result 70 x

Output

Action Output

#	Time	Action	Message
1	13:07:50	SELECT AVG(t.amount) as average_amount, c.iban FROM transactions t JOIN credit_cards c ON t.card_...	1 row(s) returned

JUNTO LOS CAMPOS DE AMOUNT (TAULA TRANSACTIONS) AMB IBAN DE LA TAULA (COMPANIES) AMB UN JOIN.

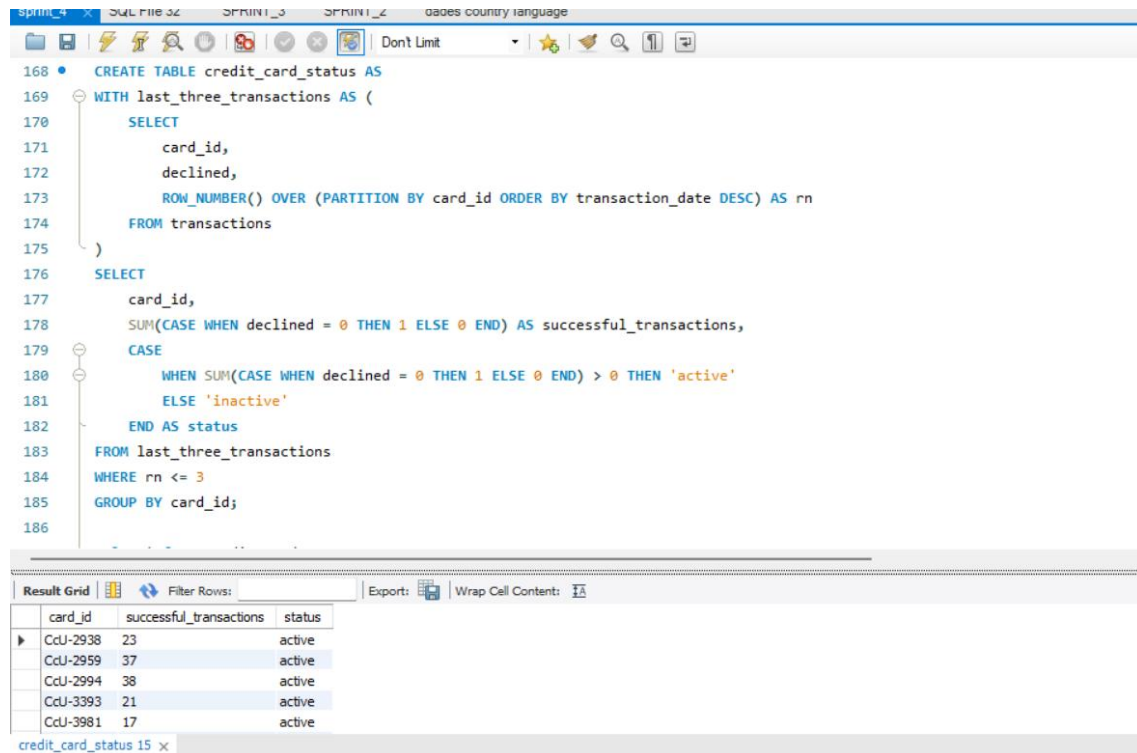
I LI DEMANO NOMÉS LA INFO DE LA EMPRESA DONEC LTD.

NIVELL 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 genera la següent consulta:

Exercici 1

Quantes targetes estan actives?



```
168 • CREATE TABLE credit_card_status AS
169 WITH last_three_transactions AS (
170     SELECT
171         card_id,
172         declined,
173         ROW_NUMBER() OVER (PARTITION BY card_id ORDER BY transaction_date DESC) AS rn
174     FROM transactions
175 )
176 SELECT
177     card_id,
178     SUM(CASE WHEN declined = 0 THEN 1 ELSE 0 END) AS successful_transactions,
179     CASE
180         WHEN SUM(CASE WHEN declined = 0 THEN 1 ELSE 0 END) > 0 THEN 'active'
181         ELSE 'inactive'
182     END AS status
183 FROM last_three_transactions
184 WHERE rn <= 3
185 GROUP BY card_id;
186
```

Result Grid

card_id	successful_transactions	status
CcU-2938	23	active
CcU-2959	37	active
CcU-2994	38	active
CcU-3393	21	active
CcU-3981	17	active

credit_card_status 15 x

CREO UNA NOVA TABLA: CREDIT_CARD_STATUS

He creat la taula credit_card_status seleccionant només les tres últimes transaccions de cada targeta utilitzant la funció ROW_NUMBER() per ordenar-les per data.

Només s'han processat aquestes tres transaccions per comptar quantes van ser exitoses (declined = 0) amb una funció SUM(CASE ...)

L'estat de la targeta es determina com 'active' si almenys una de les tres transaccions va ser exitosa, o 'inactive' si totes van ser declinades

Els resultats es van agrupar per card_id per obtenir un registre únic per targeta. Això permet que l'estat reflecteixi exclusivament les tres últimes transaccions.