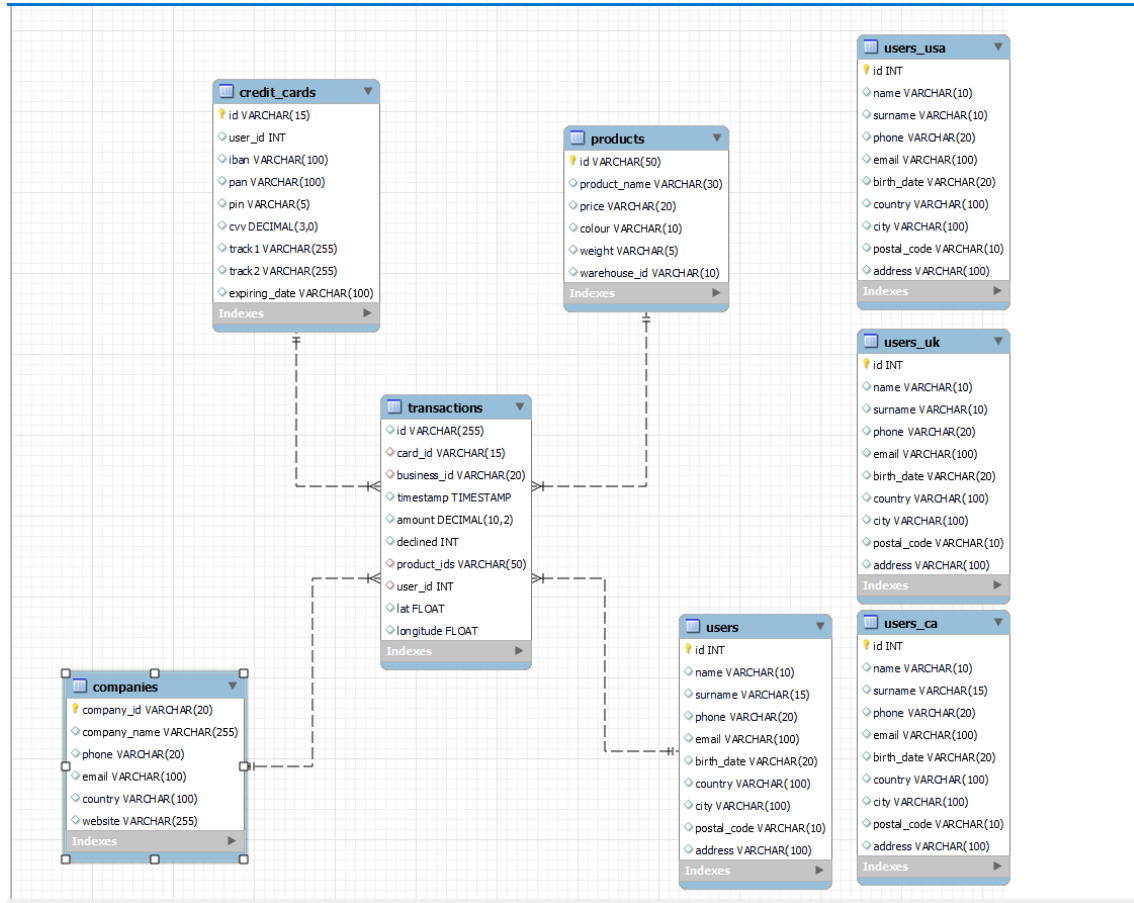


Descàrrega els arxius CSV, estudia'ls i dissenya una base de dades amb un esquema d'estrella que contingui, almenys 4 taules de les quals puguis realitzar les següents consultes:



En la imagen está el diagrama de estrella hice las siguientes relaciones.

- Credit_cards. Ids = t.card_id
- Companies.Company_id= t.business_id
- Products.id = t.product_ids
- Users.id = user_id

Para poder facilitar la relación con los users, cree la tabla users uniendo todos los datos de users_ca, users_usa y users_uk ya que sus ids son nominales consecutivas y no repetidas.

```
-- Creamos la tabla transactions
CREATE TABLE IF NOT EXISTS transactions (
    id VARCHAR(255),
    card_id VARCHAR(15) references credit_cards(id),
    business_id varchar (20) references companies(company_id),
    timestamp TIMESTAMP,
    amount decimal (10,2),
    declined int,
    product_ids varchar(50) references products(id),
    user_id INT references users(id),
    lat float,
    longitude float,
    FOREIGN KEY (card_id) REFERENCES credit_cards(id),
    FOREIGN KEY (business_id) REFERENCES companies(company_id),
    FOREIGN KEY (product_ids) REFERENCES products(id),
    foreign key (user_id) references users(id)
);

87 -- Creamos la tabla transactions
88 CREATE TABLE IF NOT EXISTS transactions (
89     id VARCHAR(255),
90     card_id VARCHAR(15) references credit_cards(id),
91     business_id varchar (20) references companies(company_id),
92     timestamp TIMESTAMP,
93     amount decimal (10,2),
94     declined int,
95     product_ids varchar(50) references products(id),
96     user_id INT references users(id),
97     lat float,
98     longitude float,
99     FOREIGN KEY (card_id) REFERENCES credit_cards(id),
100     FOREIGN KEY (business_id) REFERENCES companies(company_id),
101     FOREIGN KEY (product_ids) REFERENCES products(id),
102     foreign key (user_id) references users(id)
103 );
104
105 alter table transactions drop constraint transactions_ibfk_4; #drop constraint para poder hacer la relacion entre users y la tabla de transactions
106 Alter table transactions add foreign key (user_id) references users(id);
107 CREATE INDEX transactions_ibfk_4 on users(id); #create index para la tabla de transactions y añadir los datos de "datos_insertar_sprint_04"
108 alter table transactions drop constraint transactions_ibfk_3; #drop constraint en caso de que haya problemas entre products y transactions
109 alter table transactions modify product_ids varchar(50); #modificación hecha posteriori en transactions(product_ids y products(id), valor original INT.
110 alter table products modify id varchar(50);
```

He tenido que hacer varias alteraciones de tablas ya que me saltaban errores de constraint, de primary y foreign keys y dejo constancia de la modificación de products(id) y transactions(product_ids).

Recomiendo cargar las tablas de companies, credit_cards, products y users_ca, users_usa y users_uk, después insertar los datos de cada una de las tablas y entonces crear la tabla users y cargar la tabla de transactions.

****** Nota post corrección, podemos evitar todas las alteraciones de tabla que se ven en la imagen superior utilizando los siguientes códigos ya introducidos en el script.

```
SET FOREIGN_KEY_CHECKS=0;
```

```
SET SQL_SAFE_UPDATES = 0;
```

Utilizarlos antes de insertar los datos en las tablas.

Para insertar los datos tuve que utilizar dos conversores diferentes.

<https://konbert.com/convert/csv/to/tsv> que me daba la conversión más precisa directamente de archivo a insert into de mysql y <https://www.convertcsv.com/> que me ayudó a limpiar primero el archivo de transactions antes de pasarlo por www.konbert.com

Como uso el ordenador del aula no podía añadir los archivos mediante query.

```
LOAD DATA INFILE 'C:\ProgramData\MySQL\MySQL Server 8.3\Uploads\copia excels/transactions.csv'
INTO TABLE transactions
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n'
IGNORE 1 ROWS;
```

```
"11:37:07      LOAD DATA INFILE 'C:\ProgramData\MySQL\MySQL Server 8.3\Uploads\copia
excels/transactions.csv' INTO TABLE transactions FIELDS TERMINATED BY ',' ENCLOSED BY '"'
LINES TERMINATED BY '\n' IGNORE 1 ROWS
Error Code: 1290. The MySQL server is running with the --secure-file-priv option so it cannot
execute this statement 0.000 sec"
```

En el código subido a github saldrán con # para que no sean interpretados cuando se utilice el script.

Un problema añadido que tuve fue que en mi ordenador privado tampoco pude subir los archivos. Me decía que tenía que utilizar la carpeta designada por mysqlbench. Utilice la siguiente función para averiguar que carpeta era.

```
# SHOW VARIABLES LIKE "secure_file_priv";
```

Aunque me dio el resultado y pude acceder a la carpeta, no me dejó subir los archivos. Por eso utilice los conversores.

Álvaro Míguez

Sprint 04

- Exercici 1

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

Usando las tablas transactions y users podemos sacar nombre y apellidos y hacer en el user_id de transactions para ver cuántas veces figuran en la tabla de transactions y así saber el número de pedidos. He añadido, además, la suma total de sus pedidos y el id de usuario.

```
8      #ex 1
9
10
11 •   SELECT users.name, users.surname, users.id, count(transactions.user_id) as ct, sum(amount)
12     FROM transactions
13     INNER JOIN users on users.id = user_id
14     GROUP BY users.name, users.surname, users.id, transactions.user_id
15     HAVING ct >= 30
16     order by ct desc;
17
18 •   select count(user_id) as total_orders, sum(amount), users.name, users.surname, users.id
19     from transactions
20     INNER JOIN users on users.id = user_id
21     GROUP BY users.name, users.surname, users.id, transactions.user_id
22     having count(transactions.user_id) >= 30;
23
```

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

	name	surname	id	ct	sum(amount)
▶	Hedwig	Gilbert	272	76	18351.30
	Ocean	Nelson	267	52	13052.24
	Kenyon	Hartman	275	48	12011.56
	Lynn	Riddle	92	39	11451.57

Para ver la 2ª fórmula en acción y ver que dan los mismos resultados

```
8      #ex 1
9
10
11 •   SELECT users.name, users.surname, users.id, count(transactions.user_id) as ct, sum(amount)
12     FROM transactions
13     INNER JOIN users on users.id = user_id
14     GROUP BY users.name, users.surname, users.id, transactions.user_id
15     HAVING ct >= 30
16     order by ct desc;
17
18 •   select count(user_id) as total_orders, sum(amount), users.name, users.surname, users.id
19     from transactions
20     INNER JOIN users on users.id = user_id
21     GROUP BY users.name, users.surname, users.id, transactions.user_id
22     having count(transactions.user_id) >= 30;
23
```

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

	total_orders	sum(amount)	name	surname	id
	48	12011.56	Kenyon	Hartman	275
	76	18351.30	Hedwig	Gilbert	272
	52	13052.24	Ocean	Nelson	267
	39	11451.57	Lynn	Riddle	92

Álvaro Míguez

Sprint 04

- Exercici 2

Mostra la mitjana de la suma de transaccions per IBAN de les targetes de crèdit en la companyia Donec Ltd. utilitzant almenys 2 taules.

Usando las tablas, transactions, users, credit_cards y companies podemos encontrar la compañía de Donec Ltd, en credit_cards la información del iban y de users el id de la tarjeta y juntándolo con transacciones para ver los pedidos y su cantidad para hacer la media (avg), como ya tenemos los join entre las tablas podemos buscar el iban a través del Company_name.

```
25 #ex 2
26
27 • SELECT iban, AVG(amount), transactions.card_id, transactions.user_id, companies.company_name FROM transactions
28 JOIN credit_cards ON transactions.card_id = credit_cards.id
29 JOIN companies ON transactions.business_id = company_id
30 JOIN users ON transactions.user_id = users.id
31 WHERE company_name = "Donec Ltd"
32 GROUP BY iban, card_id, user_id, company_name;
```

iban	AVG(amount)	card_id	user_id	company_name
PT87806228135092429456346	203.715000	CcU-2973	275	Donec Ltd

Nivel 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?

Juntando la información de credit_cards, users y transaction, podemos contar las veces que se les ha declinado una compra. En este caso solo hay 3 usuarios que tendrían las tarjetas desactivadas. Los usuarios 272, 275 y 267.

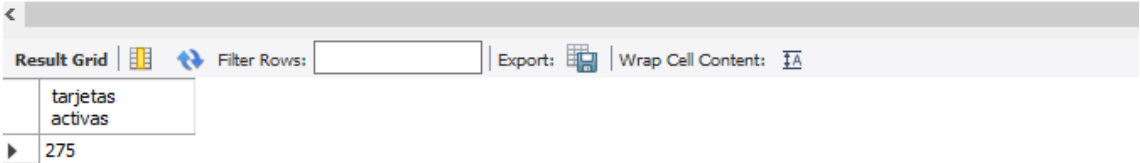
```
34 #ex 2.1
35 * select sum(declined) as declined_total, transactions.user_id, if(sum(declined) >= 3, "inactive", "active") as activacion from transactions
36 join credit_cards on transactions.card_id = credit_cards.id
37 join users on transactions.user_id = users.id
38 group by user_id
39 order by declined_total desc;
```

declined_total	user_id	activacion
3	272	inactive
3	275	inactive
3	267	inactive
2	274	active
2	273	active
2	271	active
2	270	active
2	269	active
2	268	active
0	92	active
0	91	active
0	90	active
0	89	active
0	88	active
0	87	active
0	86	active
0	85	active
0	84	active
0	83	active
0	82	active
0	81	active
0	80	active
0	96	active
0	95	active
0	94	active
0	93	active
0	136	active
0	135	active
0	134	active

Post Corrección Edu Carnero.

Con Edu revisamos el ejercicio y me explicó que lo había planteado mal, ya que ni había creado una nueva tabla ni había buscado los estados de tarjetas basados únicamente en las últimas tres transacciones, sino que había buscado 3 pagos declinados o más en total. Después del feedback procedí a cambiarlo.

```
41 • Create temporary table tarjeta (with ct as(
42   select *, row_number() over (partition by card_id order by timestamp desc) as seq
43   from transactions)
44   select card_id,
45         if (SUM(declined) >= 3, 'inactiva', 'activa') as activa
46   from ct
47   where seq <= 3
48   group by card_id);
49
50 • SELECT COUNT(activa) AS 'tarjetas activas'
51   FROM tarjeta
52   WHERE activa = 'activa';
53
54
```



tarjetas activas
275

Primero creamos la tabla temporal tarjeta, así no me afecta al diagrama de estrella. Seleccionamos toda la tabla de transactions y usamos el partition by para agregar la información de card_id ordenándolo por timestamp para poder revisar el tiempo de las compras y ver cuantos declined se han hecho recientemente por usuario de tarjeta.

Usamos el if sobre la suma de declined para saber si está inactiva o activa, más de 3 = inactiva, menos de 3 = activa y lo contamos dentro de la tabla.

Por último, ejecutamos un count y así podemos ver cuantas están activas. Nos da que los 275 usuarios ninguno tiene 3 compras consecutivas declinadas recientes.

Nivel 3

Crea una taula amb la qual puguem unir les dades del nou arxiu products.csv amb la base de dades creada, tenint en compte que des de transaction tens product_ids. Genera la següent consulta:

Ejercicio 1

Necessitem conèixer el nombre de vegades que s'ha venut cada producte.

Haciendo el count en product_ids de transactions podemos contar el número de veces que se han pedido. Juntándolo con la tabla de products, podemos ver el Id de producto y su nombre.

Álvaro Míguez

Sprint 04

```
41 #ex 3.1
42 • select product_name, count(product_ids) as totalorders, products.id as ItemId from transactions
43 join products on product_ids = products.id
44 group by product_name, itemid
45 order by totalorders desc;
```

product_name	totalorders	ItemId
Tully maester Tarly	7	29
skywalker ewok	7	5
kingsblood Littlefinger the	7	53
riverlands north	6	23
Direwolf Littlefinger	6	37
duel	6	43
Direwolf Stannis	5	59
Direwolf riverlands the	5	79
duel tourney	5	83
skywalker ewok	5	89
Direwolf Stannis	4	1
skywalker ewok sith	4	17
dooku solo	4	19
duel tourney Lannister	4	3
Winterfell Lannister	4	61
Winterfell	4	67
north of Casterly	4	7
Karstark Dorne	3	11
palpatine chewbacca	3	13

```
41 #ex 3.1
42 • select product_name, count(product_ids) as totalorders, products.id as ItemId from transactions
43 join products on product_ids = products.id
44 group by product_name, itemid
45 order by totalorders desc;
46
47 • select * from transactions
48 join products on product_ids = products.id
49 where product_ids = "47";
50
51 • select * from products
52 where product_name = 'direwolf stannis';
53
54
55
56
```

Para poder hacer otras pruebas de conteo, buscaba un id de producto concreto como se puede ver en la 2ª formula. O buscaba un nombre para ver si se correspondía.

Post corrección Edu Carnero.

Edu me explicó que al tener los products id con comas, mysql workbench no me lo diferencia como ids diferentes si no como una id única con comas. De manera que el ejercicio anterior está mal planteado. Aunque la formula en si funciona, no responde al ejercicio correctamente. He intentado separar los valores de product_id por comas, pero la tabla no termina de funcionar correctamente ya que, aunque consigo la separación, en listas de 2 y 3 valores me repite los valores para terminar de rellenar la lista en lugar de darme null. Dejo foto para aclaración.

Álvaro Míguez
Sprint 04

```
65 • select
66     product_ids,
67     substring_index(substring_index(product_ids, ',', 1), ',', 1) as product_id_1,
68     if(substring_index(substring_index(product_ids, ',', 2), ',', -1) = product_ids, null,
69     substring_index(substring_index(product_ids, ',', 2), ',', -1)) as product_id_2,
70     if(substring_index(substring_index(product_ids, ',', -2), ',', 1) = product_ids, null,
71     substring_index(substring_index(product_ids, ',', -2), ',', 1)) as product_id_3,
72     if (substring_index(substring_index(product_ids, ',', -1), ',', -1) = product_ids, null,
73     substring_index(substring_index(product_ids, ',', -1), ',', -1)) as product_ids_4
74     from new_transactions;
75
```

Result Grid					
Filter Rows:		Export:		Wrap Cell Content:	
	product_ids	product_id_1	product_id_2	product_id_3	product_ids_4
▶	59	59	NULL	NULL	NULL
	71, 41	71	41	71	41
	97, 41, 3	97	41	41	3
	11, 13, 61, 29	11	13	61	29
	47, 37, 11, 1	47	37	11	1
	23, 19, 71	23	19	19	71
	59, 13, 23	59	13	13	23
	47, 67, 31, 5	47	67	31	5
	17, 13, 73	17	13	13	73
	47	47	NULL	NULL	NULL
	1	1	NULL	NULL	NULL
	61, 13, 7, 71	61	13	7	71
	59	59	NULL	NULL	NULL
	41, 83	41	83	41	83

Ejemplo claro sería la 2ª fila “71,41” me da de resultados “71, 41, 71, 41” cuando me tendría que dar “71, 41, null, null”.