

# Nivell 1

preparació i creació de la bbdd:

Primer de tot obro els documents CSV amb notepad per donar un cop d'ull al tipus de dades que tenen.

Copio els títols a un Excel i em faig una estructura general dels arxius per poder-ne crear les taules: (copy-paste dels headers, text-to-columns, i copy – transpose).

RESUM CSV:						
			SEPARADOR ;			
Companies	Credit_Cards	Products	Transactions	Users_CA	Users_UK	Users_USA
company_id	id	id	id	id	id	id
company_name	user_id	product_name	card_id	name	name	name
phone	iban	price	business_id	surname	surname	surname
email	pan	colour	timestamp	phone	phone	phone
country	pin	weight	amount	email	email	email
website	cvv	warehouse_id	declined	birth_date	birth_date	birth_date
	track1		product_ids	country	country	country
	track2		user_id	city	city	city
	expiring_date		lat	postal_code	postal_code	postal_code
			longitude	address	address	address

Primer creem l'esquema:

```
1 • CREATE DATABASE transactions;
```

```
3 10:18:58 CREATE DATABASE transactions 1 row(s) affected
```

Un cop creat l'esquema, creem les taules:

```
CREATE TABLE companies
(
  company_id VARCHAR(45),
  company_name VARCHAR(45),
  phone VARCHAR(45),
  email VARCHAR(45),
  country VARCHAR(45),
  website VARCHAR(45));

CREATE TABLE products (
  id VARCHAR(45),
  product_name VARCHAR(45),
  price VARCHAR(45),
  colour VARCHAR(45),
  weight VARCHAR(45),
  warehouse_id VARCHAR(45));

CREATE TABLE transactions (
  id VARCHAR(45),
  card_id VARCHAR(45),
  business_id VARCHAR(45),
  timestamp VARCHAR(45),
  amount VARCHAR(45),
  declined VARCHAR(45),
  product_ids VARCHAR(45),
  user_id VARCHAR(45),
  lat VARCHAR(45),
  longitude VARCHAR(45));

CREATE TABLE users (
  id VARCHAR(45),
  name VARCHAR(45),
  surname VARCHAR(45),
  phone VARCHAR(45),
  email VARCHAR(45),
  birth_date VARCHAR(45),
  country VARCHAR(45),
  city VARCHAR(45),
  postal_code VARCHAR(45),
  address VARCHAR(100));

CREATE TABLE credit_cards (
  id VARCHAR(45),
  user_id VARCHAR(45),
  iban VARCHAR(45),
  pan VARCHAR(45),
  pin VARCHAR(45),
  cvv VARCHAR(45),
  track1 VARCHAR(100),
  track2 VARCHAR(100),
  expiring_date VARCHAR(45));
```

✓	4	10:28:22	CREATE TABLE companies (company_id VARCHAR(45), company_name VARCHAR(45), pri...	0 row(s) affected
✓	5	10:28:25	CREATE TABLE products (id VARCHAR(45), product_name VARCHAR(45), pri...	0 row(s) affected
✓	6	10:28:28	CREATE TABLE transactions (id VARCHAR(45), card_id VARCHAR(45), busin...	0 row(s) affected
✓	7	10:40:13	CREATE TABLE users (id VARCHAR(45), name VARCHAR(45), sumame VAR...	0 row(s) affected
✓	8	10:40:17	CREATE TABLE credit_cards (id VARCHAR(45), user_id VARCHAR(45), iban V...	0 row(s) affected

Com que els tres arxius d'usuaris tenen les mateixes columnes, decideixo crear una sola taula d'usuaris, ja que el país també es mostra a la columna country, i hi carregaré tots tres arxius.

Importo el contingut dels arxius CSV.

Al primer intent rebo l'error "The MySQL server is running with the --secure-file-priv option"

Demano al programa que em digui quina és la carpeta segura:

```
SHOW VARIABLES LIKE 'secure_file_priv';
```

	Variable_name	Value	
▶	secure_file_priv	C:\ProgramData\MySQL\MySQL Server 8.0\Uploads\	
✓	8	11:47:00 SHOW VARIABLES LIKE 'secure_file_priv'	1 row(s) returned

Usaré aquesta URL per desar-hi els arxius CSV i pujar-los des d'aquí.

SQL File 7\* x

Don't Limit

```

1 • LOAD DATA
2 INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\companies.csv'
3 INTO TABLE companies
4 FIELDS TERMINATED BY ','
5 IGNORE 1 ROWS;

```

✓ 15 11:58:56 LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\... 275 row(s) affected Records: 275 Deleted: 0 Skipped: 0 Warnings: 0

A la taula transactions especifico que el separador és punt i coma:

```

LOAD DATA
INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\transactions.csv'
INTO TABLE transactions
FIELDS TERMINATED BY ';'
IGNORE 1 ROWS;

```

✓	15	11:58:56	LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\credit_card...	2
✓	16	12:03:43	LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\products.c...	1
✓	17	12:04:41	LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\transaction...	5

Per pujar el fitxer de users, especifico que hi ha cadenes de dades (en aquest cas les dates) que estan entre dobles cometes. (em torno a trobar que l'adreça és massa llarga, canvio la longitud del camp a varchar(100) i em segueix donant error a la línia 4, que sembla que no està interpretant bé el final de línia.

Veig a notepad que els salts de línia són CR LF

```
ldressCR LF
"0",Canada,Rigolet,V6T 6M7,154-5415 Auctor St.CR LF
nada,Murdochville,S7E 6E0,Ap #880-6372 Ultrices. St.CR LF
Canada,Baddeck,K3X 6Z5,441-8969 Rhoncus RoadCR LF
94",Canada,Maple Creek,Y2C 9E6,"517-6759 Ut, Av."CR LF
7, 1983",Canada,Dieppe,E7S 4P8,Ap #916-8051 A St.CR LF
2000",Canada,Québec City,B4K 0J6,8588 Massa. AveCR LF
', 1981",Canada,Rae-Edzo,20Y 8L2,Ap #636-8055 Egestas St.CR LF
Annapolis Royal,S4Y 8V5,Ap #983-6042 Amet StreetCR LF
c 14, 1991",Canada,Cambridge Bay,93Z 5S5,Ap #388-8542 Est St.CR LF
,Ottawa,A1S 9W6,601-6142 Etiam St.CR LF
Canada,Cumberland,S5Y 2L8,7285 Sed St.CR LF
1986",Canada,Fort Smith,V3G 8B3,4756 Tempor Rd.CR LF
1983",Canada,Mission,ROV 9R2,"P.O. Box 207, 6843 Imperdiet Avenue"CR LF
```

```
ALTER TABLE users
MODIFY COLUMN address VARCHAR(100);
```

19 12:08:39 ALTER TABLE users MODIFY COLUMN address VARCHAR(100) 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

Indico salt de línia en l'estandard de Windows (CR LF = \r\n)

Importo els 3 arxius de clients a la mateixa taula:

```
LOAD DATA
INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\users_usa.csv'
INTO TABLE users
FIELDS TERMINATED BY ','
OPTIONALLY ENCLOSED BY '"'
LINES TERMINATED BY '\r\n'
IGNORE 1 ROWS;
```

I comprovo que el nombre de fileres de la taula correspon a la suma dels 3 arxius:

```
29 13:07:40 LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\users_ca.csv...' 75 row(s) affected Records: 75 Deleted: 0 Skipped: 0 W
30 13:21:37 LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\users_UK.cs...' 50 row(s) affected Records: 50 Deleted: 0 Skipped: 0 W
31 13:22:03 LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\users_usa.cs...' 150 row(s) affected Records: 150 Deleted: 0 Skipped: 0
32 13:22:11 SELECT * FROM transactsp4.users 275 row(s) returned
```

Un cop tinc carregades les dades dels fitxers CSV, estableixo les primary keys i les relacions entre taules:

```
#establim FK I RELACIONS ENTRE TAULES:
ALTER TABLE companies
ADD CONSTRAINT PK_company PRIMARY KEY (company_id);

ALTER TABLE Products
ADD CONSTRAINT PK_prod PRIMARY KEY (id);

ALTER TABLE users
ADD CONSTRAINT PK_user PRIMARY KEY (id);

ALTER TABLE transactions
ADD CONSTRAINT PK_trans PRIMARY KEY(id);
```

Quan intento establir la foreign key de transactions a products, em dona error perquè hi ha productes que no són a la taula de productes. Veig que a transactions hi ha registres que tenen més d'una id de producte.

```
SELECT product_ids
FROM transactions
WHERE product_ids NOT IN (SELECT id FROM products);
```

	product_ids
▶	71, 1, 19
	47, 97, 43
	47, 67, 31, 5
	89, 83, 79
	43, 31
	47, 23
	67, 7
	29, 41, 11

Per resoldre-ho, creo una nova taula per incloure-hi els detalls de cada transacció:

```
CREATE TABLE transaction_details (
  detail_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  transact_id VARCHAR(45),
  product_ids VARCHAR(25)
);
```

Primer faig un cte recurrent que separi els elements de la columna product\_ids en elements individuals, mantenint-los relacionats amb la columna id de transaction.

Un cop fet el split, introdueixo els resultats dins de la taula transaction\_details.

```
21 10:56:20 INSERT INTO transaction_details (transact_id, product_ids) SELECT id, part FRO... 1457 row(s) affected Records: 1457 Duplicates: 0 Warnings: 0
24 10:57:06 WITH RECURSIVE split_cte AS ( SELECT id, CASE WHEN ... 1457 row(s) returned
```

Faig un join de les dues taules per comprovar que de cada transacció on hi havia més d'un producte ha creat una filera per producte:

```
SELECT transactions.id, transactions.product_ids, transaction_details.product_ids
FROM transactions
JOIN transaction_details
ON transactions.id = transaction_details.transact_id
WHERE length(transactions.product_ids)>3
ORDER BY id;
```

id	product_ids	product_ids
063FBA79-99EC-66FB-29F7-25726D1764A5	47, 67, 31, 5	47
063FBA79-99EC-66FB-29F7-25726D1764A5	47, 67, 31, 5	31
063FBA79-99EC-66FB-29F7-25726D1764A5	47, 67, 31, 5	67
063FBA79-99EC-66FB-29F7-25726D1764A5	47, 67, 31, 5	5
0668296C-CDB9-A883-76BC-2E4C44F8C8AE	89, 83, 79	89
0668296C-CDB9-A883-76BC-2E4C44F8C8AE	89, 83, 79	79
0668296C-CDB9-A883-76BC-2E4C44F8C8AE	89, 83, 79	83
06CD9AA5-9B42-D684-DDDD-A5E394FEBA99	43, 31	43
06CD9AA5-9B42-D684-DDDD-A5E394FEBA99	43, 31	31
07A46D48-31A3-7E87-65B9-0DA902AD109F	47, 23	47
07A46D48-31A3-7E87-65B9-0DA902AD109F	47, 23	23
09DE92CE-6F27-2BB7-13B5-9385B2B3B8E2	67, 7	67

També he comprovat l'opció `find_in_set` i em dona el mateix resultat:

```
CREATE TABLE IF NOT EXISTS orders2 (
  transaction_id VARCHAR(100) NOT NULL,
  product_id INT NOT NULL);

INSERT INTO orders2 (transaction_id, product_id)
SELECT transactions.id as transaction_id, products.id as product_id
FROM transactions
JOIN products ON FIND_IN_SET(products.id, REPLACE (transactions.product_ids, " ", "")) > 0;
```

94 11:58:45 INSERT INTO orders2 (transaction\_id, product\_id) SELECT transactions.id as tran... 1457 row(s) affected Records: 1457 Duplicates: 0 Warnings: 0

Un cop tinc la informació de productes en una taula separada, ja puc crear les relacions entre les taules:

```
ALTER TABLE transactions
ADD CONSTRAINT FK_user FOREIGN KEY(user_id) REFERENCES users(id);
```

81 12:49:49 ALTER TABLE transactions ADD CONSTRAINT FK\_user FOREIGN KEY(user... 587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0

```
ALTER TABLE transactions
ADD CONSTRAINT FK_company FOREIGN KEY(business_id) REFERENCES companies(company_id);
```

85 13:11:21 ALTER TABLE transactions ADD CONSTRAINT FK\_company FOREIGN KEY(... 587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0

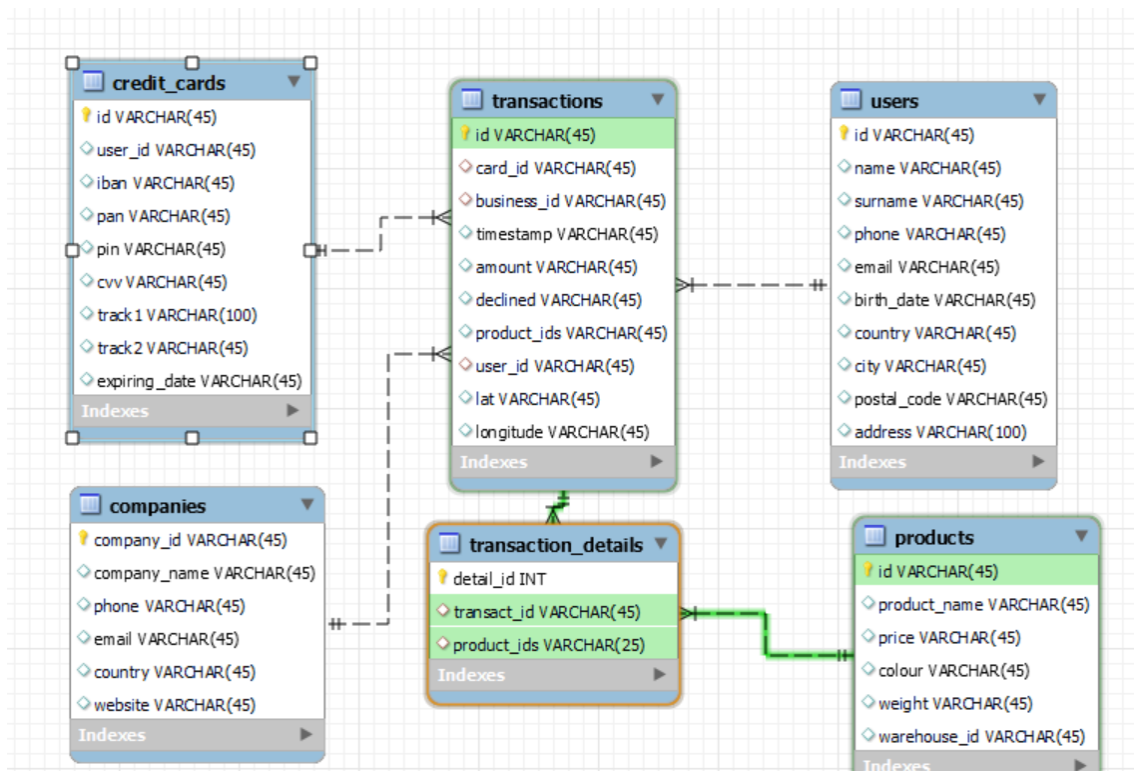
```
ALTER TABLE transaction_details
ADD CONSTRAINT FK_transact FOREIGN KEY(transact_id) REFERENCES transactions(id);
```

89 13:17:09 ALTER TABLE transaction\_details ADD CONSTRAINT FK\_transact FOREIGN... 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

```
ALTER TABLE transaction_details
ADD CONSTRAINT FK_product FOREIGN KEY(product_ids) REFERENCES products(id);
```

91 13:19:00 ALTER TABLE transaction\_details ADD CONSTRAINT FK\_product FOREIGN ... 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

Comprovo que les relacions estan ben creades:



## N1- Exercici 1

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

Primer de tot, i per assegurar la consistència en totes les queries, creo una vista de les transaccions que estan actives (així evito haver de filtrar declined a cada consulta.)

```

CREATE VIEW Transact AS(
  SELECT *
  FROM transactions
  WHERE declined = 0);
  
```

6 08:10:29 CREATE VIEW Transact AS( SELECT \* FROM transactions WHERE declined... 0 row(s) affected

Faig un join de la taula users i la view Transact, demano que compti transaccions agrupades per usuari, i que filtri les que són superiors a 30.

```

SELECT user_id, users.name, users.surname, COUNT(transact.ID) as num_trans
FROM transact
JOIN users
ON transact.user_id = users.id
GROUP BY user_id
HAVING num_trans > 30
;
  
```



	user_id	name	surname	num_trans
▶	92	Lynn	Riddle	39
	267	Ocean	Nelson	39
	272	Hedwig	Gilbert	38

7 08:12:49 SELECT user\_id, users.name, users.surname, COUNT(transact.ID) as num\_tran... 3 row(s) returned

Poso nom d'usuari i num de transaccions perquè el resultat sigui més clar, tot i que l'enunciat només demana llista d'usuaris.

## N1-Exercici 2

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

Faig la selecció de l'iban i la mitja per tarja.

Començo pel where pq em filtri només la companyia Donec Ltd, l'inserto transact perquè em filtri les transaccions de la companyia Donec Ltd, i finalment faig la selecció de l'iban i la mitja de quantitats agrupant per iban i arrodonint a 2 dígits.

```
SELECT iban, ROUND(AVG(amount),2) AS mitja_per_tarja
FROM credit_cards
JOIN transact
ON transact.card_id = credit_cards.id
WHERE transact.business_id = (SELECT company_id
FROM companies
WHERE company_name = "Donec Ltd")
GROUP BY iban;
```

	iban	mitja_per_tarja
▶	PT87806228135092429456346	42.82

✓ 1 19:17:26 SELECT iban, ROUND(AVG(amount),2) AS mitja\_per\_tarja FROM credit\_cards J... 1 row(s) returned

## 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:

### N2-Exercici 1

Quantes targetes estan actives?

Primer creo la taula amb les columnes "targeta" i "estat"

```
CREATE TABLE Estat_Targetes (
    targeta VARCHAR(45),
    estat VARCHAR(20)
);
```

✓ 15 19:43:41 CREATE TABLE Estat\_Targetes ( targeta VARCHAR(45), estat VARCHAR... 0 row(s) affected

Per introduir la informació sobre les targetes a la taula que he creat:

Primer faig una subconsulta interna combinada amb window function Row\_number perquè agrupi les transaccions per targeta i les ordeni per data.

El select principal amb case, ens indicarà si una targeta està caducada quan la suma de declined sigui de tres. (posant un límit de 3 línies i separant per targeta).

```
INSERT INTO Estat_Targetes (targeta, estat)
SELECT
    card_id,
    CASE
        WHEN SUM(declined) = 3 THEN 'Bloquejada'
        ELSE 'Activa'
    END AS estat
FROM (
    SELECT
        card_id,
        declined,
        ROW_NUMBER() OVER (PARTITION BY card_id ORDER BY timestamp DESC) AS fila
    FROM transactions
) ultimes_transact
WHERE fila <= 3
```

18 19:44:15 INSERT INTO Estat\_Targetes (targeta, estat) SELECT card\_id, CASE ... 275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0

Compto les targetes actives amb un select count de targetes on l'estat sigui activa:

```
SELECT COUNT(*)
FROM estat_targetes
WHERE estat = "Activa";
```

COUNT(*)
275

12 11:28:33 SELECT COUNT(\*) FROM estat\_targetes WHERE estat = "Activa" 1 row(s) returned

## Nivell 3

### N3-Exercici 1

(la taula ja l'he creada al primer nivell)

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

```
SELECT product_ids, COUNT(transact_id) AS recompte_vendes
FROM transaction_details
GROUP BY product_ids
```



	product_ids	recompte_vendes
▶	1	39
	11	30
	13	38
	17	35
	18	36

14 11:33:45 SELECT product\_ids, COUNT(transact\_id) AS recompte\_vendes FROM transa... 52 row(s) returned