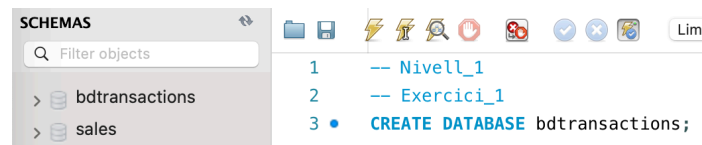


Nivell 1

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 puguem realitzar les següents consultes:

Primer creem una nova base de dades a través de codi:

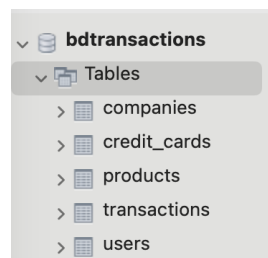


Una vegada observats tots els arxius csv, dissenyarem una base de dades amb un esquema d'estrella. La consulta que utilitzarem per crear les taules és aquesta:

```
5 • CREATE TABLE companies (  
6     company_id VARCHAR(15) PRIMARY KEY NOT NULL,  
7     company_name VARCHAR(255),  
8     phone VARCHAR(15),  
9     email VARCHAR(100),  
10    country VARCHAR(100),  
11    website VARCHAR(255)  
12 );  
13 • CREATE TABLE users (  
14     id INT PRIMARY KEY NOT NULL,  
15     name VARCHAR(255),  
16     surname VARCHAR(255),  
17     phone VARCHAR(20),  
18     email VARCHAR(255),  
19     birth_date VARCHAR(20),  
20     country VARCHAR(255),  
21     city VARCHAR(255),  
22     postal_code VARCHAR(50),  
23     address VARCHAR(255)  
24 );  
25 • CREATE TABLE credit_cards (  
26     id VARCHAR(15) PRIMARY KEY NOT NULL,  
27     user_id INT,  
28     iban VARCHAR(100),  
29     pan VARCHAR(100),  
30     pin VARCHAR(100),  
31     cvv VARCHAR(15),  
32     track1 VARCHAR(255),  
33     track2 VARCHAR(255),  
34     expiring_date VARCHAR(25)  
35 );  
38 • CREATE TABLE products (  
39     id INT auto_increment PRIMARY KEY,  
40     product_name VARCHAR(100),  
41     price DECIMAL(10,2),  
42     colour VARCHAR(50),  
43     weight DECIMAL(10,2),  
44     warehouse_id VARCHAR(10)  
45 );  
46 • CREATE TABLE transactions (  
47     id VARCHAR(255) PRIMARY KEY,  
48     card_id VARCHAR(15),  
49     business_id VARCHAR(15),  
50     timestamp TIMESTAMP,  
51     amount DECIMAL(10, 2),  
52     declined BOOLEAN,  
53     product_ids INT,  
54     user_id INT,  
55     lat FLOAT,  
56     longitude FLOAT,  
57     FOREIGN KEY (card_id) REFERENCES credit_cards(id),  
58     FOREIGN KEY (business_id) REFERENCES companies(company_id),  
59     FOREIGN KEY (product_ids) REFERENCES products(id),  
60     FOREIGN KEY (user_id) REFERENCES users(id)  
61 );
```

Com es pot observar en la consulta hem creat cinc taules: “transactions” és la taula de fets i la resta són taules de dimensions. La taula “users” agrupa els arxius csv de les tres regions: “users_ca”, “users_uk” i “users_usa”.

Una vegada creades les taules amb les columnes específiques, fem les definicions explícites de les claus foranes. Amb això també ens assegurem que en el moment de relacionar, ambdues columnes, per exemple ‘products.id’ i ‘transactions.products_ids’, tinguin la mateixa estructura de dades, en aquest cas INT.



He intentat fer la importació mitjançant comandament però no he pogut, ni modificant la configuració des d'Administration -- Options File -- secure-file-priv.

```
62 • SELECT @@secure_file_priv;
63 --
64 • LOAD DATA INFILE '/Users/maria/workspace/data-analytics/S4/companies.csv'
65 INTO TABLE companies
66 FIELDS TERMINATED BY ','
67 LINES TERMINATED BY '\r\n'
68 IGNORE 1 ROWS;
```

100% 21:55

Action Output

	Time	Action	Response	Duration / Fetch Time
7	16:18:35	SHOW VARIABLES LIKE "secure_file_priv"	1 row(s) returned	0.0014 sec / 0.00001...
8	16:21:36	LOAD DATA INFILE '/Users/maria/workspace/data-analytics/S4/companies.csv' INTO TABLE...	Error Code: 1290. The MySQL server is running with t...	0.00036 sec

Pel que he fet la importació de la següent manera. Des de l'opció “Import records from an external file” per tal d'introduir totes les dades:

62

-- Insertem les dades

63

•

SELECT * FROM bdtransactions.companies;

100%

55:57

Result Grid

Filter Rows:

Search

Edit:

Export/Import:

company_id	company_name	phone	email	country	website	
NULL	NULL	NULL	NULL	NULL	NULL	

Table Data Import

Configure Import Settings

Detected file format: csv

Encoding:

<input checked="" type="checkbox"/>	Source Column	Dest Column
<input checked="" type="checkbox"/>	company_id	<input type="text" value="company_id"/>
<input checked="" type="checkbox"/>	company_name	<input type="text" value="company_name"/>
<input checked="" type="checkbox"/>	phone	<input type="text" value="phone"/>
<input checked="" type="checkbox"/>	email	<input type="text" value="email"/>
<input checked="" type="checkbox"/>	country	<input type="text" value="country"/>

company_id	company_n...	phone	email	country	website
b-2222	Ac Ferme...	06 85 56...	donec.po...	Germany	https://ins...
b-2226	Magna A...	04 14 44...	risus.don...	Australia	https://wh...
b-2230	Fusce Co...	08 14 97...	risus@pr...	United St...	https://pi...

Una vegada executats tots els passos, ens apareix que s'han agregat 100 registres i en fer la consulta de nou, observem que les dades s'han introduït correctament:

```
62 -- Insertem les dades
63 • SELECT * FROM bdtransactions.companies;
```

100% 40:63

Result Grid Filter Rows: Edit: Export/Import:

	company_id	company_name	phone	email	country	website
	b-2222	Ac Fermentum Incorporated	06 85 56 52 33	donec.portitor.tellus@yahoo.net	Germany	https://instagram.com/site
	b-2226	Magna A Neque Industries	04 14 44 64 62	risus.donec.nibh@icloud.org	Australia	https://whatsapp.com/group/9
	b-2230	Fusce Corp.	08 14 97 58 85	risus@protonmail.edu	United States	https://pinterest.com/sub/cars
	b-2234	Convallis In Incorporated	06 66 57 29 50	mauris.ut@aol.couk	Germany	https://cnn.com/user/110
	b-2238	Ante Jaculis Nec Foundation	08 23 04 99 53	sed.dictum.proin@outlook.ca	New Zealand	https://netflix.com/settings

Repetim aquesta acció amb la resta de taules, tenint en compte que la taula “users” agrupa tres csv diferents.

En arribar a l'arxiu “users_ca” ens indica un problema relacionat amb la codificació de caràcters:

Configure Import Settings

Detected file format: csv

Encoding:

Error

Unhandled exception: 'ascii' codec can't decode byte 0xc3 in position 733: ordinal not in range(128)

Check the log for more details.

Anem a revisar el csv i ràpidament detectem que el problema ve d'un registre que s'ha escrit amb accent "Québec", caràcter que no està en el rang ASCII estàndard.

edu, Dec 2, 1988", Canada, Murdochville, S7L 6E0, Ap #880-03
.couk, "Jan 6, 1982", Canada, Baddeck, K3X 6Z5, 441-8969 Rhon
tlook.ca, "Oct 15, 1994", Canada, Maple Creek, Y2C 9E6, "517-
olor@google.ca, "May 7, 1983", Canada, Dieppe, E7S 4P8, Ap #9
utlook.net, "Nov 17, 2000", Canada, Quebec City, B4K 0J6, 858
m@google.edu, "Sep 22, 1981", Canada, Rae-Edzo, 20Y 8L2, Ap #
eh 23, 1993" Canada Annapolis Royal S4Y 8V5 Ap #983-6042

Modifiquem el csv per i desem. Una vegada fet aquest canvi els arxius s'importen correctament.

D'altra banda, a l'hora d'importar els registres de la taula "products" també ens retorna un error en els valors de la columna 'price' ja que aquests contenen el símbol '\$':

The following tasks will now be performed. Please monitor the execution.

- ✓ Prepare Import
- ✓ Import data file

Finished performing tasks. Click [Next >] to continue.

Message Log

```
Starting...
Prepare Import...
Prepare Import done
Import data file....
- Prepare Import
- Begin Import
- Row import failed with error: ("Incorrect decimal value: '$161.11' for column 'price' at row 1", 1366)
- Row import failed with error: ("Incorrect decimal value: '$9.24' for column 'price' at row 1", 1366)
- Row import failed with error: ("Incorrect decimal value: '$171.13' for column 'price' at row 1", 1366)
- Row import failed with error: ("Incorrect decimal value: '$71.89' for column 'price' at row 1", 1366)
- Row import failed with error: ("Incorrect decimal value: '$171.22' for column 'price' at row 1", 1366)
- Row import failed with error: ("Incorrect decimal value: '$136.60' for column 'price' at row 1", 1366)
```

Per solucionar-ho, anem a l'arxiu csv "products" i fem un search i replace de tots els símbols '\$' i desem. Tornem a importar i s'executa sense errors.

La última taula que ens queda per importar les dades és “transactions” i de nou ens apareixen errors:

[illegible]

Optem per modificar l'esquema de la taula a format cadena perquè accepti els valors, abans però hem d'eliminar temporalment la restricció de clau externa:

```
67 -- Modifiquem el tipus de dades de la columna 'products_ids'
68 • START TRANSACTION;
69 • ALTER TABLE transactions DROP FOREIGN KEY transactions_ibfk_3;
70 • ALTER TABLE products
71   MODIFY COLUMN id VARCHAR(255);
72 • ALTER TABLE transactions
73   MODIFY COLUMN product_ids VARCHAR(255);
74 • ALTER TABLE transactions ADD CONSTRAINT FOREIGN KEY (product_ids) REFERENCES products(id);
--
```

Un cop modificat tornem a provar d'introduir les dades però tan sols es bolquen 110 registres dels 587, ja que la columna 'product_ids' conté més d'un codi de producte associat separat per comes. Així que optem per crear una taula d'unió que relacioni les transaccions amb els productes. Anomenem la nova taula de relació "transactions_products":

```
78 -- Creem la taula de relació
79 • CREATE TABLE transactions_products (
80     transaction_product_id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
81     transaction_id VARCHAR(255),
82     product_id VARCHAR(255),
83     FOREIGN KEY (transaction_id) REFERENCES transactions(id),
84     FOREIGN KEY (product_id) REFERENCES products(id)
85 );
```

Com que ja tenim creada la taula de relació "transactions_products", podem eliminar la columna 'products_ids' de la taula "transactions" i inserir les dades, que s'han creat mitjançant un script de python.

```
86 -- Eliminem la restricció de clau i la columna product_ids de la taula transactions
87 • ALTER TABLE transactions DROP FOREIGN KEY transactions_ibfk_3;
88 • ALTER TABLE transactions DROP COLUMN product_ids;
89 -- Insertem les dades
90 • ALTER TABLE transactions_products DROP FOREIGN KEY transactions_products_ibfk_1;
91 • SELECT * FROM bdtransactions.transactions;
```

Aquest script el que fa és extreure l'id de "transactions" i els id de "products" de cada fila i imprimeix una instrucció SQL INSERT per cada combinació de id de transactions i id de products, per inserir-los a la taula de relació "transactions_products".

```
script.py > ...
1  import csv
2
3  # Open the CSV file and iterate through each row
4  with open('transactions.csv', 'r') as file:
5      reader = csv.DictReader(file, delimiter=',')
6      for row in reader:
7          transaction_id = row['id']
8          product_ids = row['product_ids'].split(',')
9
10         # Insert a record into transactions_products for each product ID
11         for product_id in product_ids:
12             # Perform the insertion
13             print(f"INSERT INTO transactions_products (transaction_id, product_id) VALUES ('{transaction_id}', '{product_id.strip()}');")
14
```

```

1  -- Inserir les dades de transactions_products
2  • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('108B1D1D-5B23-A76C-55EF-C568E49A05DD', '59');
3  • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('7DC26247-20EC-53FE-E555-B6C2E55CA5D5', '71');
4  • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('7DC26247-20EC-53FE-E555-B6C2E55CA5D5', '41');
5  • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('72997E96-DC2C-A4D7-7C24-66C302F8AE5A', '97');
6  • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('72997E96-DC2C-A4D7-7C24-66C302F8AE5A', '41');
7  • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('72997E96-DC2C-A4D7-7C24-66C302F8AE5A', '3');
8  • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('AB069F53-965E-A2A8-CE06-CA8C4FD92501', '11');
9  • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('AB069F53-965E-A2A8-CE06-CA8C4FD92501', '13');
10 • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('AB069F53-965E-A2A8-CE06-CA8C4FD92501', '61');
11 • INSERT INTO transactions_products (transaction_id, product_id) VALUES ('AB069F53-965E-A2A8-CE06-CA8C4FD92501', '29');

```

Fem import les dades de “transactions”:

Import Results

File /Users/maria/Downloads/transactions.csv was imported in 2.190 s

Table bdtransactions.transactions has been used

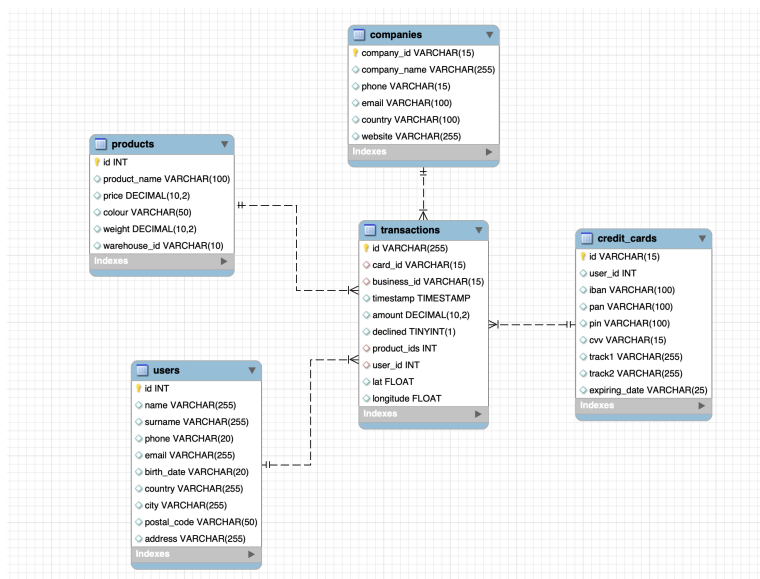
587 records imported

I la taula de relació queda de la següent manera:

```
92 • SELECT * FROM bdtransactions.transactions_products;
```

transaction_product_id	transaction_id	product_id
1	108B1D1D-5B23-A76C-55EF-C568E49A05DD	59
2	7DC26247-20EC-53FE-E555-B6C2E55CA5D5	71
3	7DC26247-20EC-53FE-E555-B6C2E55CA5D5	41
4	72997E96-DC2C-A4D7-7C24-66C302F8AE5A	97
5	72997E96-DC2C-A4D7-7C24-66C302F8AE5A	41
6	72997E96-DC2C-A4D7-7C24-66C302F8AE5A	3
7	AB069F53-965E-A2A8-CE06-CA8C4FD92501	11
8	AB069F53-965E-A2A8-CE06-CA8C4FD92501	13
9	AB069F53-965E-A2A8-CE06-CA8C4FD92501	61
10	AB069F53-965E-A2A8-CE06-CA8C4FD92501	29

L'esquema final de la base de dades queda de la següent manera:



- Exercici 1

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

The screenshot shows a database query editor with the following SQL code:

```
1 -- Exercici_1
2 -- Mostrem tots els usuaris amb més de 30 transaccions
3 • SELECT u.id, u.name AS user_name, COUNT(*) AS total_transactions
4 FROM users u
5 JOIN transactions t ON t.user_id = u.id
6 GROUP BY u.id, u.name
7 HAVING total_transactions > 30
8 ORDER BY total_transactions DESC;
```

The results are displayed in a table with the following data:

id	user_name	total_transactions
272	Hedwig	76
267	Ocean	52
275	Kenyon	48
92	Lynn	39

The bottom of the screenshot shows the 'Action Output' section with the following details:

Time	Action	Response	Duration / Fetch Time
17:09:51	SELECT u.id, u.name AS user_name, COUNT(*) AS total_transactions FROM users u JOIN transactions t ON t.user_id = u.id GROUP BY u.id, u.name HAVING total_transactions > 30 ORDER BY total_transactions DESC;	4 row(s) returned	0.0013 sec / 0.00000...

- Exercici 2

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

The screenshot shows a database query editor with the following SQL code:

```
1 -- Exercici_2
2 -- Mostrem la mitjana d'amount per IBAN de les targetes de crèdit a la companyia Donec Ltd
3 • SELECT cc.iban, ROUND(AVG(t.amount), 2) AS average_amount
4 FROM transactions t
5 JOIN credit_cards cc ON t.card_id = cc.id
6 WHERE t.business_id IN (SELECT companies.company_id FROM companies WHERE company_name = 'Donec Ltd')
7 GROUP BY cc.iban;
```

The results are displayed in a table with the following data:

iban	average_amount
PT87806228135092429456346	203.72

The bottom of the screenshot shows the 'Action Output' section with the following details:

Time	Action	Response	Duration / Fetch Time
17:24:31	SELECT cc.iban, ROUND(AVG(t.amount), 2) AS average_amount FROM transactions t JOIN credit_cards cc ON t.card_id = cc.id WHERE t.business_id IN (SELECT companies.company_id FROM companies WHERE company_name = 'Donec Ltd') GROUP BY cc.iban;	1 row(s) returned	0.00059 sec / 0.000...

Nivell 2

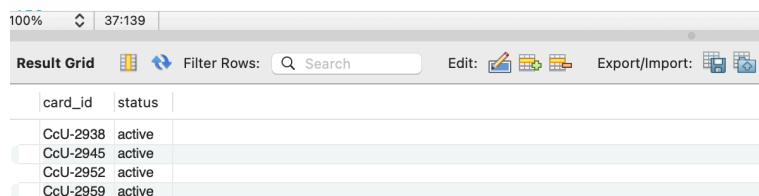
Crea una nova taula que reflecteixi l'estat de les targetes de crèdit basat en si les últimes tres transaccions van ser declinades i genera la següent consulta:

En primer lloc creem la nova taula que anomenem "credit_card_status":

```
137 -- Creem la nova taula amb l'estat de les targetes de crèdit
138 • CREATE TABLE credit_card_status (
139     card_id VARCHAR(15) PRIMARY KEY,
140     status VARCHAR(25)
141 );
```

A continuació, definim com estat actiu o inactiu en funció de si les tres últimes transaccions han sigut declinades per cada targeta, i després insertem els resultats a la taula "credit_card_status":

```
142 -- Calculem l'estat de les targetes de crèdit
143 • INSERT INTO credit_card_status (card_id, status)
144     SELECT card_id,
145            CASE
146                WHEN COUNT(*) = 3 AND SUM(declined) = 3 THEN 'inactive'
147                ELSE 'active'
148            END AS status
149     FROM (
150         SELECT card_id, declined
151         FROM transactions
152         ORDER BY card_id, timestamp DESC
153     ) AS sorted_transactions
154     GROUP BY card_id;
155 -- Mostrem la taula credit_card_status
156 • SELECT * FROM bdtransactions.credit_card_status;
157
```



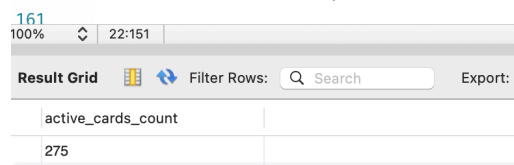
card_id	status
CcU-2938	active
CcU-2945	active
CcU-2952	active
CcU-2959	active

- Exercici 1

Quantes targetes estan actives?

Totes les targetes estan actives. Un total de 275.

```
157 -- Exercici_1
158 • SELECT COUNT(*) AS active_cards_count
159     FROM credit_card_status
160     WHERE status = 'active';
161
```



active_cards_count
275

Nivell 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:

Anteriorment, ja hem creat la taula “transactions_products” que ens permet unir les dades de transaccions amb les de productes.

- Exercici 1

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

Realitzem la consulta que ens mostra quantes vegades s'ha venut cada producte. Recordem que hi ha un total de 1457 transaccions:

```
32 -- Nivell_3
33 • SELECT * FROM bdtransactions.transactions_products;
34 -- Exercici_1
35 • SELECT product_id, COUNT(*) AS total_sales
36 FROM transactions_products
37 GROUP BY product_id
38 ORDER BY total_sales DESC;
39
40
41
```

The screenshot shows a database management interface. At the top, there's a status bar with '100%', '14:34', and '1 error found'. Below it is a 'Result Grid' section with a search bar and an 'Export' button. The main area displays a table with two columns: 'product_id' and 'total_sales'. The table contains 15 rows of data. To the right of the table is a sidebar with icons for 'Result Grid', 'Form Editor', 'Field Types', and 'Query Stats'. Below the table, there's a 'Read Only' button. At the bottom, there's an 'Action Output' section showing a log of actions. The log shows two actions: one at 17:38:38 and another at 17:38:38, both with a duration of 0.00084 sec.

product_id	total_sales
23	68
67	68
79	66
2	65
43	65
47	62
1	61
17	61
97	61
13	60
53	58
61	57
83	57
7	54
71	54

Time	Action	Response	Duration / Fetch Time
17:38:38	SELECT product_id, COUNT(*) AS total_sales FROM transactions_products GROUP BY product_id	26 row(s) returned	0.00084 sec / 0.000...
17:38:38	SELECT product_id, COUNT(*) AS total_sales FROM transactions_products GROUP BY product_id	26 row(s) returned	0.00084 sec / 0.000...