

SPRINT 3: Manipulació de taules

Elaborat per:

Vanesa Pérez Ramírez

Data: 24/09/2025

SPRINT 3: Manipulació de taules	1
Nivell 1	3
Exercici 1	3
a) La teva tasca és dissenyar i crear una taula anomenada "credit_card" que emmagatzemi detalls crucials sobre les targetes de crèdit. La nova taula ha de ser capaç d'identificar de manera única cada targeta i establir una relació adequada amb les altres dues taules ("transaction" i "company"). Després de crear la taula serà necessari que ingressis la informació del document denominat "dades_introduir_credit". Recorda mostrar el diagrama i realitzar una breu descripció d'aquest.	3
b) Recorda mostrar el diagrama i realitzar una breu descripció d'aquest.	7
Exercici 2	7
a) El departament de Recursos Humans ha identificat un error en el número de compte associat a la targeta de crèdit amb ID CcU-2938. La informació que ha de mostrar-se per a aquest registre és: TR323456312213576817699999. Recorda mostrar que el canvi es va realitzar.	7
Exercici 3	8
Exercici 4	12
Nivell 2	13
Exercici 1	13
Exercici 2	15
Exercici 3	16
Nivell 3	16
Exercici 1	16
Exercici 2	24

Nivell 1

Exercici 1

- a) La teva tasca és dissenyar i crear una taula anomenada "credit_card" que emmagatzemi detalls crucials sobre les targetes de crèdit. La nova taula ha de ser capaç d'identificar de manera única cada targeta i establir una relació adequada amb les altres dues taules ("transaction" i "company"). Després de crear la taula serà necessari que ingressis la informació del document denominat "dades_introduir_credit". Recorda mostrar el diagrama i realitzar una breu descripció d'aquest.

Se crea la nueva tabla con las columnas, las columnas se conocen abriendo el archivo con los datos a pegar en la nueva tabla, en él se hace referencia a las columnas

en los que se van a insertar los datos. Las columnas son: id, iban, pan, pin, cvv, expiring_date

Leyendo el código que se va a ejecutar vemos que los datos que se van a insertar son: ('CcU-4233', 'GI10TNGB218311472843723', '349284173298327', '7910', '380', '12/03/23')

el primero es un código de número letras y simbolo de una longitud de 8 pondré VARCHAR(10) y corresponde a la columna id que es la que utilizaremos como PK de la tabla

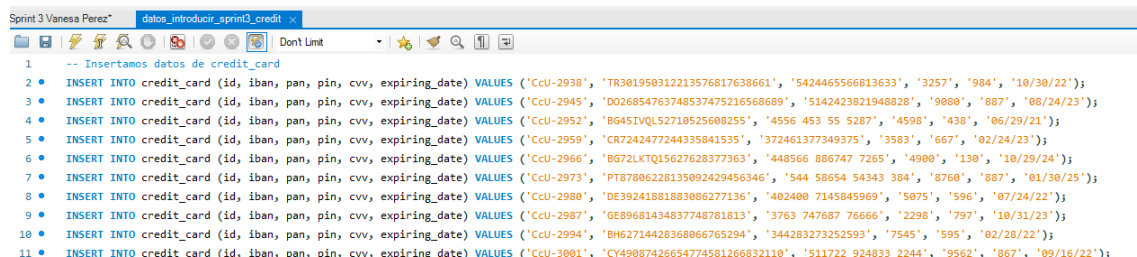
El segundo valor es el número IBAN de la targeta que dispone de letras y números y la longitud es variable con lo que le asignaré un tipo de variable VARCHAR (100)

La tercera columna (PAN) tiene valores numéricos y espacios con lo que el tipo de variable la determinaré VARCHAR (100).

La cuarta columna(PIN) son números de 4 dígitos con lo que le asignaré SMALLINT (4) como especificación de tipo de variable.

La quinta columna (cvv) es un número de 3 dígitos con lo que le asignaré SAMLLINT (3)

La sexta columna (expiring_date) es una fecha.



```

1  -- Insertamos datos de credit_card
2  INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-2938', 'TR301950312213576817638661', '5424465566813633', '3257', '984', '10/30/22');
3  INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-2945', 'D026854763748537475216568689', '5142423821948828', '9080', '887', '08/24/23');
4  INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-2952', 'B045IVQL52710525608255', '4556 453 55 5287', '4598', '438', '06/29/21');
5  INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-2959', 'CR7242477244335841535', '372461377349375', '3583', '667', '02/24/23');
6  INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-2966', 'B672LKTQ15627628377363', '448566 886747 7265', '4900', '130', '10/29/24');
7  INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-2973', 'PT87806228135092429456346', '544 58654 54343 364', '8760', '887', '01/30/25');
8  INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-2980', 'DE39241881883086277136', '402400 7145845969', '5075', '596', '07/24/22');
9  INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-2987', 'GE89681434837748781813', '3763 747687 76666', '2298', '797', '10/31/23');
10 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-2994', 'BH62714428368066765294', '344283273252593', '7545', '595', '02/28/22');
11 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-3001', 'CV49007426654774581266832110', '511722 924833 2244', '9562', '867', '09/16/22');
  
```

Se ejecuta el código de creación de la tabla y Se revisa el contenido de la tabla creada

The screenshot shows the SQL Developer interface. The top pane contains SQL code for creating a table and inserting data. The bottom pane shows the execution results.

```
9  • USE transactions;
10 # Se crea la nueva tabla con las columnas, las columnas se conocen abriendo el archivo con los datos a pegar en la nueva tabla, en él se hace referencia a las columnas
11 # en los que se van a insertar los datos. Las columnas son: id, iban, pan, pin, cvv, expiring_date
12 # Leyendo el código que se va a ejecutar vemos que los datos que se van a insertar son: ('Ccu-4233', 'G10TNG8218311472843723', '349284173298327', '7910', '380', '12/03/23')
13 # el primero es un código de número letras y símbolo de una longitud de 8 pondré VARCHAR(10) y corresponde a la columna id que es la que utilizaremos como PK de la tabla
14 # El segundo valor es el número IBAN de la tarjeta que dispone de letras y números y la longitud es variable con lo que le asignaré un tipo de variable VARCHAR (100)
15 # La tercera columna (PAN) tiene valores numéricos y espacios con lo que el tipo de variable la determinará VARCHAR (100).
16 # La cuarta columna (PIN) son números de 4 dígitos con lo que le asignaré SMALLINT (4) como especificación de tipo de variable.
17 # La quinta columna (cvv) es un número de 3 dígitos con lo que le asignaré SMALLINT (3)
18 # La sexta columna (expiring_date) es una fecha.
19
20 • CREATE TABLE IF NOT EXISTS credit_card (
21     id VARCHAR(15) NOT NULL PRIMARY KEY,
22     iban VARCHAR(100) NOT NULL,
23     pan VARCHAR(100) NOT NULL,
24     pin SMALLINT(4) NOT NULL,
25     cvv SMALLINT (3) NOT NULL,
26     expiring_date VARCHAR(8) NOT NULL
27 );
28
29 #Se revisa el contenido de la tabla creada
30 • describe credit_card;
31 • SELECT * FROM credit_card;
```

The bottom pane shows the execution results. The first row shows the table structure: id VARCHAR(15), iban VARCHAR(100), pan VARCHAR(100), pin SMALLINT(4), cvv SMALLINT(3), expiring_date VARCHAR(8). The second row shows the execution of the SELECT statement, returning 0 rows.

Es selecciona l'arxiu amb el codi per introduir les dades i s'executa.

The screenshot shows the SQL Developer interface. The top pane contains SQL code for inserting data into the credit_card table. The bottom pane shows the execution results.

```
1  -- Insertamos datos de credit_card
2  • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2938', 'TR301950312213576817638661', '5424465566813633', '3257', '984', '10/30/22');
3  • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2945', 'D026854763748537475216568689', '5142423821948828', '9080', '887', '08/24/23');
4  • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2952', 'B6451VQL52710525608255', '4556 453 55 5287', '4598', '438', '06/29/21');
5  • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2959', 'CR7242477244335841535', '372461377349375', '3583', '667', '02/24/23');
6  • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2966', 'BG72LKTQ15627628377363', '448566 886747 7265', '4900', '130', '10/29/24');
7  • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2973', 'PT87806228135092429456346', '544 58654 54343 384', '8760', '887', '01/30/25');
8  • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2980', 'DE39241881883086277136', '402400 7145845969', '5075', '596', '07/24/22');
9  • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2987', 'GE98681434837748781813', '3763 747687 76666', '2298', '797', '10/31/23');
10 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2994', 'BH62714428368066765294', '344283273252593', '7545', '595', '02/28/22');
11 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3001', 'CV49087426654774581266832110', '511722 924833 2244', '9562', '867', '09/16/22');
12 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3008', 'LU507216693616119230', '4485744464433884', '1856', '740', '04/05/25');
13 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3015', 'PS119398216295715968342456821', '3784 662233 17389', '3246', '822', '01/31/22');
14 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3022', 'GT91695162850556977423121857', '5164 1379 4842 3951', '5610', '342', '04/25/25');
15 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3029', 'AZ62317413982441418123739746', '3429 279566 77631', '9708', '505', '09/02/23');
16 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3036', 'AZ39336002925842865843941994', '3768 451556 48766', '2232', '565', '10/27/25');
17 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3043', 'TN6488143310514852179535', '455676 6437463635', '5969', '196', '06/07/25');
18 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3050', 'FR5167744369175836831854477', '4024007123722', '4834', '126', '10/09/23');
19 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3057', 'LU931822574697545215', '3484 621767 21237', '6805', '848', '09/14/25');
20 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3064', 'PS146965545449253377627273133', '3467 732741 26810', '3865', '498', '06/03/25');
21 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3071', 'NO8923814763512', '3464 789562 23352', '6625', '661', '12/20/23');
22 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3078', 'IS025127145884623279548733', '4539 322 74 2377', '9405', '720', '03/08/23');
23 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3085', 'BE63114723972437', '5266 3346 1135 1687', '7241', '413', '05/10/23');
24 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3092', 'RO6515001166122125447487', '3488 754223 46253', '9417', '594', '12/19/22');
25 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3099', 'PT26105275356823705537218', '448 55418 98863 789', '5612', '564', '01/22/23');
26 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3106', 'AT684251637751136592', '349547146395283', '9733', '209', '01/27/24');
27 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3113', 'IE26LCGT47732137572752', '341834822877471', '9011', '287', '06/12/21');
28 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3120', 'RS72655766666166237144', '527646 533375 6577', '7658', '265', '01/16/21');
29 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3127', 'PT83533461438644342816864', '4716 443 46 4368', '8038', '924', '01/16/23');
30 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3134', 'BG23MYJQ52668951824779', '5146 3453 9766 2168', '7260', '935', '08/24/25');
31 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3141', 'CH4437804777669672438', '3775 626726 45261', '2923', '330', '05/11/24');
32 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3148', 'FI6261403224677114', '3733 238351 51810', '2326', '333', '09/28/21');
33 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3155', 'AD277720476327722080982', '4532263578421', '3015', '779', '01/12/22');
34 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3162', 'HU56074054826233730628233311', '455666 645685 4443', '5898', '603', '05/18/20');
35 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3169', 'AT658218806585843788', '5396 2876 7721 4764', '6102', '420', '06/25/24');
```

The bottom pane shows the execution results. The first row shows the execution of the INSERT statement, returning 1 row(s) affected. The second row shows the execution of the INSERT statement, returning 1 row(s) affected.

Compovem com s'han guardad les dades a la taula:

The screenshot shows the SQL Server Enterprise Manager interface. The top pane displays the SQL script for creating a table named `credit_card` and executing a `SELECT` query. The bottom pane shows the results of the `SELECT` query in a grid format.

SQL Script:

```

12 # Leyendo el código que se va a ejecutar vemos que los datos que se van a insertar son: ('CcU-4233', 'G10TNGB218311472843723', '349284173298327', '7918', '380', '12/03/23')
13 # el primero es un código de número letras y símbolo de una longitud de 8 pondré VARCHAR(18) y corresponde a la columna id que es la que utilizaremos como PK de la tabla
14 # El segundo valor es el número IBAN de la tarjeta que dispone de letras y números y la longitud es variable con lo que le asignaré un tipo de variable VARCHAR (100)
15 # La tercera columna (PAN) tiene valores numéricos y espacios con lo que el tipo de variable la determinaré VARCHAR (100).
16 # La cuarta columna(PIN) son números de 4 dígitos con lo que le asignaré SMALLINT (4) como especificación de tipo de variable.
17 # La quinta columna (cvv) es un número de 3 dígitos con lo que le asignaré SMALLINT (3)
18 # La sexta columna (expiring_date) es una fecha.
19
20 CREATE TABLE IF NOT EXISTS credit_card (
21     id VARCHAR(15) NOT NULL PRIMARY KEY,
22     iban VARCHAR(100) NOT NULL,
23     pan VARCHAR(100) NOT NULL,
24     pin SMALLINT(4) NOT NULL,
25     cvv SMALLINT (3) NOT NULL,
26     expiring_date VARCHAR(8) NOT NULL
27 );
28
29 #Se revisa el contenido de la tabla creada
30 describe credit_card;
31
32 #COMPROBAMOS LOS DATOS DE LA TABLA
33 SELECT * FROM credit_card;
34

```

Result Grid:

id	iban	pan	pin	cvv	expiring_date
CS-4914	XX6911026973612252125582	8253038594284233	2506	651	10/30/27
CS-4915	XX2085410839696163655898	5801667527002109	1639	583	09/30/28
CS-4916	XX2380542834047466200742	7790215440817581	9958	639	09/27/29
CS-4917	XX7164673801829257629462	9881350761258759	9678	929	02/28/26
CS-4918	XX6339277573588653925135	8063131715036462	7325	568	10/28/28
CS-4919	XX9867280447728460565874	9947491090999749	7206	964	09/28/29
CS-4920	XX1051142484356852894978	6051250274394388	3510	365	08/30/28
CS-4921	XX4712816319758314296351	7979918290876966	1893	588	04/25/27
CS-4922	XX502230098673855171794	4837600502277220	2973	140	06/29/28
CS-4923	XX708742780569041218601	4648538991118415	7941	309	02/28/27
CS-4924	XX4843053608843126477896	603161970084206	5633	700	11/26/26

Output:

#	Time	Action	Message	Duration / Fetch
5030	17:46:57	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcS-5581', 'XX915670516405388124398147', '26243054...	1 row(s) affected	0.031 sec
5031	17:50:11	SELECT * FROM credit_card	5000 row(s) returned	0.000 sec / 0.016 sec

Per veure quines relacions es poden establir entre la nova taula creada i les que ja existeixen mirem les columnes que tenen les taules transaction i company i veiem que la taula transaction té una columna (`crèdit_card_id`) que pot ser la foreign key que es relacioni amb la id de la taula `crèdit_card`. Les altres dues taules estan relacionades entre elles per tant si relacionem `credit_card` i `transaction` quedaran les tres taules relacionades:

The screenshot shows the SQL Server Enterprise Manager interface. The top pane displays the SQL script for showing columns from the `transaction` and `company` tables. The bottom pane shows the results of the `SHOW COLUMNS` queries in a grid format.

SQL Script:

```

32 #COMPROBAMOS LOS DATOS DE LA TABLA
33 SELECT * FROM credit_card;
34 # Para crear la relación con el resto de tablas miramos las columnas que hay en cada una de las otras dos tablas. Vemos que las otras dos tablas están relacionadas
35 #entre ellas y podemos observar que la única relación que se puede realizar es con la tabla transaction que dispone de la columna credit_card_id. Con lo que realizaremos una
36 #de la tabla transaction para que la columna credit_card_id sea la Foreign Key que la relaciona con la tabla credit_card
37 SHOW COLUMNS FROM transaction;
38 SHOW COLUMNS FROM company;

```

Result Grid:

Field	Type	Null	Key	Default	Extra
id	varchar(255)	NO	PRI		
credit_card_id	varchar(15)	YES			
company_id	varchar(20)	YES	MUL		
user_id	int	YES			
lat	float	YES			
longitude	float	YES			
timestamp	timestamp	YES			
amount	decimal(10,2)	YES			
declined	tinyint(1)	YES			

Output:

#	Time	Action	Message
5039	17:54:33	SHOW COLUMNS FROM transaction	9 row(s) returned
5040	17:57:48	SHOW COLUMNS FROM company	9 row(s) returned

SPRINT3

```
12 #COMPROBAMOS LOS DATOS DE LA TABLA
13 • SELECT * FROM credit_card;
14 # Para crear la relación con el resto de tablas miramos las columnas que hay en cada una de las otras dos tablas. Vemos que las otras dos tablas están relacionadas
15 #entre ellas y podemos observar que la única relación que se puede realizar es con la tabla transaction que dispone de la columna credit_card_id. Con lo que realizaremos una
16 #de la tabla transaction para que la columna credit_card_id sea la Foreign Key que la relaciona con la tabla credit_card
17 • SHOW COLUMNS FROM transaction;
18 • SHOW COLUMNS FROM company;
```

Field	Type	Null	Key	Default	Extra
id	varchar(15)	NO	PRI		
company_name	varchar(255)	YES			
phone	varchar(15)	YES			
email	varchar(100)	YES			
country	varchar(100)	YES			
website	varchar(255)	YES			

#	Time	Action	Message
5040	17:57:48	SHOW COLUMNS FROM transaction	9 row(s) returned
5041	18:02:20	SHOW COLUMNS FROM company	6 row(s) returned

Creem el codi per relacionar les dues taules mencionades amb ALTER TABLE:

He decidid permetre l'actualizació de la taula en cas de modificació a la taula mare però no permetre el que s'elimini la transacció en cas d'eliminar la tarja de la taula mare

```
40 • ALTER TABLE transaction
41 ADD CONSTRAINT fk_transaction_credit_card_id FOREIGN KEY (credit_card_id )
42 REFERENCES credit_card (id)
43 ON DELETE RESTRICT
44 ON UPDATE CASCADE;
```

#	Time	Action	Message	Duration / Fetch
5054	18:27:28	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_credit_card_id FOREIGN KEY credit_card_id REFERENCES credit...	Error Code: 1064 You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right ...	0.016 sec
5055	18:32:15	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_credit_card_id FOREIGN KEY (credit_card_id) REFERENCES credit...	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings: 0	7.215 sec

I comprovem el canvi a FK de credit_card_id a la taula transaction

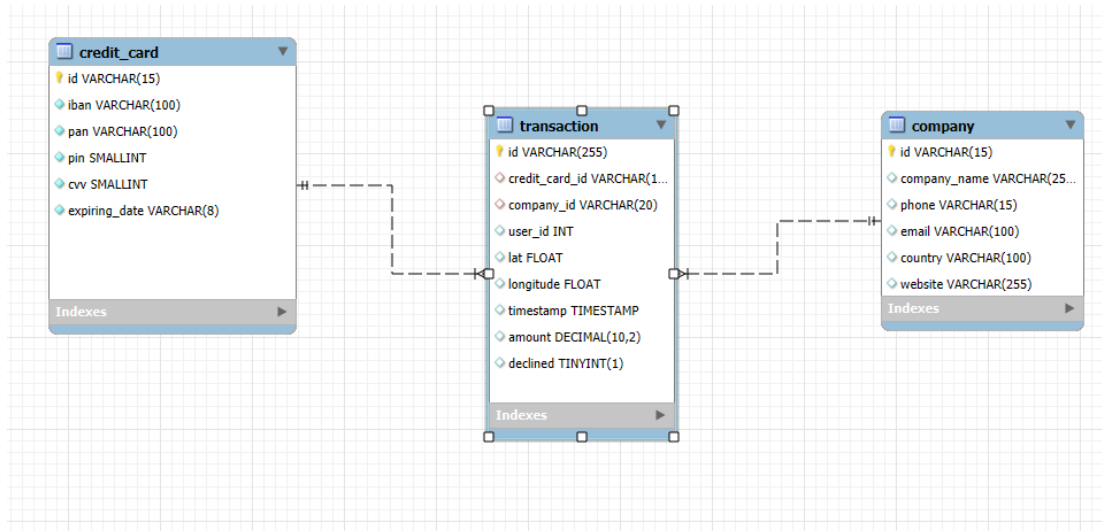
```
33 • SELECT * FROM credit_card;
34 # Para crear la relación con el resto de tablas miramos las columnas que hay en cada una de las otras dos tablas. Vemos que las otras dos tablas están relacionadas
35 #entre ellas y podemos observar que la única relación que se puede realizar es con la tabla transaction que dispone de la columna credit_card_id. Con lo que realizaremos una
36 #de la tabla transaction para que la columna credit_card_id sea la Foreign Key que la relaciona con la tabla credit_card
37 • SHOW COLUMNS FROM transaction;
38 • SHOW COLUMNS FROM company;
```

Field	Type	Null	Key	Default	Extra
id	varchar(255)	NO	PRI		
credit_card_id	varchar(15)	YES	MUL		
company_id	varchar(20)	YES	MUL		
user_id	int	YES			
lat	float	YES			
longitude	float	YES			
timestamp	timestamp	YES			
amount	decimal(10,2)	YES			
declined	tinyint(1)	YES			

#	Time	Action	Message
5055	18:32:15	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_credit_card_id FOREIGN KEY (credit_card_id) REFERENCES credit...	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings: 0
5056	18:33:06	SHOW COLUMNS FROM transaction	9 row(s) returned

b) Recorda mostrar el diagrama i realitzar una breu descripció d'aquest.

Finalment, es pot veure que s'a creat la taula credit_card a la base de dades transactions i que la seva Primary Key es la id. També s'observa que s'ha creat la relació amb la taula transaction de 1 a N.



Exercici 2

a) El departament de Recursos Humans ha identificat un error en el número de compte associat a la targeta de crèdit amb ID CcU-2938. La informació que ha de mostrar-se per a aquest registre és: TR323456312213576817699999. Recorda mostrar que el canvi es va realitzar.

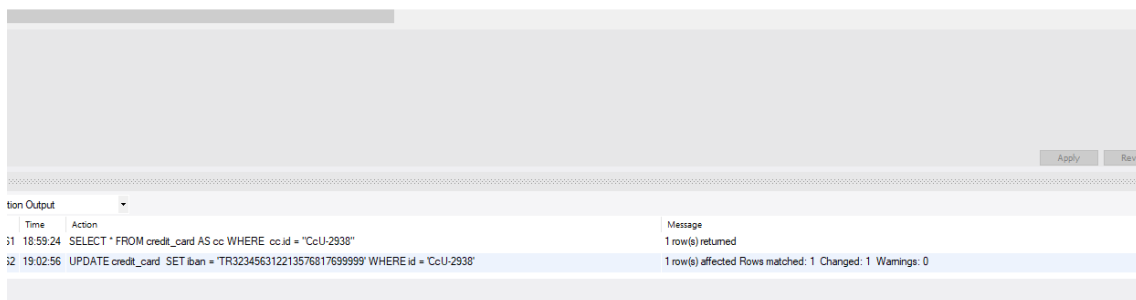
Per fer el canvi de valor, actualitzem la taula crèdit_card amb UPDATE posant la condició de que l'iban que es canviï sigui al que correspon a la fila del registre de la id CcU-2938:

```

#Exercici 2
# a) El departament de Recursos Humans ha identificat un error en el número de compte associat a la targeta de crèdit amb ID CcU-2938.
# La informació que ha de mostrar-se per a aquest registre és: TR323456312213576817699999. Recorda mostrar que el canvi es va realitzar.
# Para modificar un valor en una tabla se usa UPDATE,

UPDATE credit_card
SET iban = 'TR323456312213576817699999'
WHERE id = 'CcU-2938';

SELECT *
FROM credit_card
WHERE id = 'CcU-2938';
  
```



Després per mostrar el canvi fem la selecció de filtran per l'id que identifica el registre canviat:

The screenshot shows a database management interface. At the top, a SQL query is entered in a text area:

```
53
54 • SELECT *
55 FROM credit_card
56 WHERE id = 'CcU-2938';
```

Below the query, a 'Result Grid' displays the results of the query. The grid has columns: id, iban, pan, pin, cvv, and expiring_date. The data row shows:

id	iban	pan	pin	cvv	expiring_date
CcU-2938	TR323456312213576817699999	5424465566813633	3257	984	10/30/22

Below the result grid, there is an 'Output' section showing the execution log. It contains two entries:

#	Time	Action	Message
5062	19.02.56	UPDATE credit_card SET iban = 'TR323456312213576817699999' WHERE id = 'CcU-2938'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
5063	19.07.13	SELECT * FROM credit_card WHERE id = 'CcU-2938'	1 row(s) returned

Exercici 3

En la taula "transaction" ingressa una nova transacció amb la següent informació:

Id	108B1D1D-5B23-A76C-55EF-C568E49A99DD
credit_card_id	CcU-9999
company_id	b-9999
user_id	9999
lat	829.999
longitude	-117.999
amount	111.11
declined	0

Es realitza la inserció de la desdes a través de INSERT INTO però no es pot fer, el programa t'avisava de que hi ha una restricció per la clau forànea. Comproven si la tarja existeix a la taula de credit_card i comprovem que no existeix:

SPRINT3

```
57
58 #Exercici 3
59 #En la taula "transaction" ingressa una nova transacció amb la següent informació:
60 • INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined)
61   VALUES ('10881D10-5B23-A76C-55EF-C568E49A9900', 'CcU-9999', 'b-9999', 9999, 829.9999, -117.999, 111.11, 0);
62
63
64
65
```

Output

#	Time	Action	Message
5067	19:16:28	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES ('10881D10-5B23-A76C...	Error Code: 1452: Cannot add or update a child row: a foreign key constraint fails (transactions.transaction, CONSTRAINT transacti...

Comprovació d'existència de id a la crèdit_card:

```
62
63 • SELECT *
64   FROM credit_card
65  WHERE id = 'CcU-9999';
66
67
68
```

Result Grid

id	iban	pan	pin	cvv	expiring_date
NULL	NULL	NULL	NULL	NULL	NULL

credit_card 21 x

Output

#	Time	Action	Message
5068	19:17:38	SELECT * FROM credit_card WHERE id = 'CcU-9999'	0 row(s) returned
5069	19:20:38	SELECT * FROM credit_card WHERE id = 'CcU-9999'	0 row(s) returned

Durant la correcció s'ha dit que sí que s'ha d'afegir els id a les taules que ho necessiten per poder introduir les dades de la transacció encara que no disposem de la informació completa.

Per tant, he introduït a cada taula credit-card, company i en aquest moment data_user(perquè ja la tinc creada, les noves id.

Automatic disabled. Use manually (current card toggle at

```

68 WHERE id = 'Ccu-9999';
69 SELECT *
70 FROM company
71 WHERE id = 'b-9999';
72 SELECT *
73 FROM data_user
74 WHERE id = '9999';
75
76 INSERT INTO credit_card (id)
77 VALUES ('Ccu-9999');
78
79 INSERT INTO company (id)
80 VALUES ('b-9999');
81
82 INSERT INTO data_user (id)
83 VALUES ('9999');
84
85 show columns from credit_card;
86 ALTER TABLE MODIFY
87
88
89 -- ADD INFORMATION ABOUT ID CREDIT CARD AND ID COMPANY
90 -- Exercici 4
91 -- Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Recordar mostrar el canvi realitzat
92

```

Field	Type	Null	Key	Default	Extra
id	varchar(20)	NO	PRI		
iban	varchar(100)	YES			
pin	varchar(4)	YES			
cvv	int	YES			
expiring_date	varchar(8)	NO			
fecha_actual	date	YES		curdate()	DEFAULT_GENERATED

Result 14 x

Output

#	Time	Action	Message	Duration / Fetch
16	09:59:46	SELECT * FROM user WHERE id = b-9999	Error Code: 1146. Table 'transactions.user' doesn't exist	0.000 sec / 0.000 sec
17	09:59:58	SELECT * FROM data_user WHERE id = b-9999	0 row(s) returned	0.000 sec / 0.000 sec
18	10:10:56	INSERT INTO credit_card (id) VALUES (Ccu-9999)	Error Code: 1364. Field 'expiring_date' doesn't have a default value	0.000 sec / 0.000 sec
19	10:10:59	INSERT INTO company (id) VALUES (b-9999)	1 row(s) affected	0.000 sec / 0.000 sec

A l'insertar les dades de la taula crèdit_card dona un error ja que no tenim la columna expiring_date no permet null per tant hem de modificar la taula per a que ens accepti afegir un null.

```

94
95 show columns from credit_card;
96 ALTER TABLE credit_card
97 MODIFY expiring_date VARCHAR (8) NULL;
98
99
100 -- ADD INFORMATION ABOUT ID CREDIT CARD AND ID COMPANY
101 -- Exercici 4
102 -- Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Recordar mostrar el canvi realitzat
103

```

Field	Type	Null	Key	Default	Extra
id	varchar(20)	NO	PRI		
iban	varchar(100)	YES			
pin	varchar(4)	YES			
cvv	int	YES			
expiring_date	varchar(8)	YES			
fecha_actual	date	YES		curdate()	DEFAULT_GENERATED

Result 15 x

Output

#	Time	Action	Message	Duration / Fetch
24	10:15:06	show columns from credit_card	6 row(s) returned	0.000 sec / 0.000 sec
25	10:32:38	ALTER TABLE credit_card MODIFY expiring_date VARCHAR (8) NULL	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	1.532 sec
26	10:32:43	show columns from credit_card	6 row(s) returned	0.016 sec / 0.000 sec

Executo de nou la ordre per crear nou registre a l'id de crèdit_card :

SPRINT3

```
76 • INSERT INTO credit_card (id)
77   VALUES ('ccu-9999');
78
79 • INSERT INTO company (id)
80   VALUES ('b-9999');
81
82 • INSERT INTO data_user (id)
83   VALUES ('9999');
84
85 • show columns from credit_card;
86 • ALTER TABLE credit_card
87   MODIFY expiring_date VARCHAR (8) NULL;
88
89
90 -- ADD INFORMATION ABOUT ID CREDIT CARD AND ID COMPANY
91 -- Exercici 4
92 -- Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Recorda mostrar el canvi realitzat
93
94 -- Comproven la taula abans de la eliminació:
95
96 • SELECT *
97   FROM credit_card;
98 -- Eliminem columna
99 • ALTER TABLE credit_card
100   DROP COLUMN pan;
101
102 -- comprovem taula després de l'eliminació
103
104 • SELECT *
105   FROM credit_card;
```

Output			
Action Output			
#	Time	Action	Message
26	10:32:43	show columns from credit_card	6 row(s) returned
27	10:33:12	show columns from credit_card	6 row(s) returned
28	10:33:17	INSERT INTO credit_card (id) VALUES ('ccu-9999')	1 row(s) affected

Executo de nou la introducció de la transacció:

```
59 -- En la taula "transaction" ingressa una nova transacció amb la següent informació:
60 • INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined)
61   VALUES ('100B1D1D-5B23-A76C-55EF-C56BE48A9900', 'ccu-9999', 'b-9999', 9999, 829.9999, -117.999, 111.11, 0);
62
63 -- Comproven l'error, mirant si la id de la credit card existeix a credit_card.
64 -- Durant la correcció s'ha dit que sí que s'ha d'afegir els id a les taules que ho necessiten per poder introduir les dades de la transacció encara que que no disposem de la info
65
66 • SELECT *
67   FROM credit_card
68   WHERE id = 'ccu-9999';
69 • SELECT *
70   FROM company
71   WHERE id = 'b-9999';
72 • SELECT *
73   FROM data_user
74   WHERE id = '9999';
75
76 • INSERT INTO credit_card (id)
77   VALUES ('ccu-9999');
78
79 • INSERT INTO company (id)
80   VALUES ('b-9999');
81
82 • INSERT INTO data_user (id)
83   VALUES ('9999');
84
85 • show columns from credit_card;
86 • ALTER TABLE credit_card
87   MODIFY expiring_date VARCHAR (8) NULL;
88
89
90 -- ADD INFORMATION ABOUT ID CREDIT CARD AND ID COMPANY
91 -- Exercici 4
92 -- Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Recorda mostrar el canvi realitzat
93
94 -- Comproven la taula abans de la eliminació:
```

Output			
Action Output			
#	Time	Action	Message
27	10:33:12	show columns from credit_card	6 row(s) returned
28	10:33:17	INSERT INTO credit_card (id) VALUES ('ccu-9999')	1 row(s) affected
29	10:34:32	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES ('100B1D1D-5B23-A76C-5...	1 row(s) affected

Comprobo que la transacció nova apareix:

SPRINT3

The screenshot shows a database interface with a SQL query editor at the top. The query is:

```
89 • SELECT *
90 FROM transaction
91 WHERE id = '108B1D1D-5B23-A76C-55EF-C568E49A990D';
```

Below the query editor is a 'Result Grid' showing a single row of data:

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
108B1D1D-5B23...	CoU-9999	b-9999	9999	830	-117.999	111-11	0	

On the right side, there is a vertical toolbar with icons for 'Result Grid', 'Form Editor', 'Field Types', and 'Query Status'. Below the result grid, there is an 'Action Output' section showing the execution log:

#	Time	Action	Message
28	10:33:17	INSERT INTO credit_card (id) VALUES (CoU-9999)	1 row(s) affected
29	10:34:32	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES ('108B1D1D-5B23-A76C-5...	1 row(s) affected
30	10:37:01	SELECT * FROM transaction WHERE id = '108B1D1D-5B23-A76C-55EF-C568E49A990D'	1 row(s) returned

Exercici 4

Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Recordar mostrar el canvi realitzat

Taula abans de la eliminació de la columna pan:

The screenshot shows a database interface with a SQL query editor at the top. The query is:

```
70 • SELECT *
71 FROM credit_card;
72
```

Below the query editor is a 'Result Grid' showing a table with columns: id, iban, pan, pin, cvv, and expiring_date. The table contains 10 rows of data.

id	iban	pan	pin	cvv	expiring_date
CcS-4857	XX4857591835292505850771	2314242385113924	1819	467	09/27/25
CcS-4858	XX85817681137002436094025	6582720299715533	3964	817	12/28/28
CcS-4859	XX7826930491423553609370	8861684536289642	4983	277	11/26/26
CcS-4860	XX5559590368835304645299	2481155515498459	6876	661	07/27/27
CcS-4861	XX2035182877195191627307	1308930301149557	5710	398	04/25/26
CcS-4862	XX4774721462463645409758	6715617009807829	4042	174	11/27/26
CcS-4863	XX1476829664245046207111	3140879819451394	5969	449	12/27/29
CcS-4864	XX8380298893385731196159	5793672133649114	8481	139	02/28/26
CcS-4865	XX7085078596101025280599	5101552687251312	7847	903	11/25/28
CcS-4866	XX4792859188206596406839	8080768801072613	9271	961	02/28/25

Below the result grid, there is an 'Action Output' section showing the execution log:

#	Time	Action	Message
5069	19:20:38	SELECT * FROM credit_card WHERE id = 'CoU-9999'	0 row(s) returned
5070	19:23:49	SELECT * FROM credit_card	5000 row(s) returned

S'aplica la eliminació de la columna a través de ALTER TABLE i es fa la visualització de la taula crèdit_card per mostrar el canvi a la taula.

SPRINT3

```
75 • ALTER TABLE credit_card
76 DROP COLUMN pan;
77
78 • SELECT *
79 FROM credit_card;
```

id	iban	pin	cvv	expiring_date
CcS-4857	XX4857591835292505850771	1819	467	09/27/25
CcS-4858	XX8581768137002436094025	3964	817	12/28/28
CcS-4859	XX7626930491423553609370	4983	277	11/26/26
CcS-4860	XX5599990368835304645299	6876	661	07/27/27
CcS-4861	XX2035182877195191627307	5710	398	04/25/26
CcS-4862	XX4774721462463645409758	4042	174	11/27/26
CcS-4863	XX1476829664245046207111	5969	449	12/27/29
CcS-4864	XX8380298893385731196159	8481	139	02/28/26
CcS-4865	XX7085078596101025280599	7847	903	11/25/28
CcS-4866	XX4792859188206596406839	9271	961	02/28/25

credit_card 23 x

Output

Action Output

#	Time	Action	Message
5071	19:27:04	ALTER TABLE credit_card DROP COLUMN pan	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
5072	19:27:23	SELECT * FROM credit_card	5000 row(s) returned

Nivell 2

Exercici 1

Elimina de la taula transaction el registre amb ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD de la base de dades.

Abans de la eliminació, apareix a la taula transaction:

```
13 • SELECT *
14 FROM transaction
15 where id= '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD';
```

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
000447FE-B650-...	CcS-5019	b-2370	438	41.5972	12.2218	2016-12-21 20:07:18	155.63	0

transaction 24 x

Output

Action Output

#	Time	Action	Message
5073	19:32:01	SELECT * FROM transaction where id= '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'	1 row(s) returned
5074	19:32:20	SELECT * FROM transaction where id= '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'	1 row(s) returned

Apliquem un DELETE per eliminar el registre:

```
86 • DELETE FROM transaction
88 WHERE id= '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD';
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
5074	19:32:20	SELECT * FROM transaction where id= '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'	1 row(s) returned	0.000 sec / 0.000 s
5075	19:34:05	DELETE FROM transaction WHERE id= '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'	1 row(s) affected	0.032 sec

Si busquem el registre després de eliminar-lo ja no surt:

SPRINT3

83 • SELECT *

84 FROM transaction

85 WHERE id= '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD';

Result Grid

Filter Rows:

Edit

Export/Import

Wrap Cell Content

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	dedined

ransaction 25 x

Output

Apply

Revert

Action Output

#	Time	Action	Message
5075	19:34:05	DELETE FROM transaction WHERE id= '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'	1 row(s) affected
5076	19:35:13	SELECT * FROM transaction WHERE id= '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'	0 row(s) returned

Exercici 2

La secció de màrqueting desitja tenir accés a informació específica per a realitzar anàlisi i estratègies efectives. S'ha sol·licitat crear una vista que proporcioni detalls clau sobre les companyies i les seves transaccions. Serà necessària que creïs una vista anomenada VistaMarketing que contingui la següent informació: Nom de la companyia. Telèfon de contacte. País de residència. Mitjana de compra realitzat per cada companyia. Presenta la vista creada, ordenant les dades de major a menor mitjana de compra.

Per realitzar la consulta que finalment s'utilitzarà per crear la vista he decidit fer primer una CTE en la que creo una taula en la que he seleccionat les id de les companyies i la funció d'agregació de mitjana de l'amount de la taula transaction aplicant la condició de que les ventes declinades no apareguin als càlculs. Amb aquesta acció obtinc les mitjanes per cada id de les companyies.

Posteriorment, he creat el codi que aconseguirà les dades que es volen incloure. He fet una join entre la taula que surt de la CTE y la taula company.

Per últim, he creat la vista amb CREATE VIEW.

Aquest exercici es podia fer amb una subquery aplicada al FROM, però he decidit fer-ho amb cte per que va més ràpid i simplifica la consulta.

```

111 CREATE VIEW VistaMarketing AS(
112
113     WITH avg_purchases_company AS(
114         SELECT t.company_id, ROUND(AVG(t.amount),2) AS avg_purchase
115         FROM transaction t
116         WHERE declined = 0
117         GROUP BY t.company_id)
118     SELECT c.company_name, c.phone, c.country, av.avg_purchase
119     FROM avg_purchases_company av
120     JOIN company c
121     ON c.id = av.company_id
122     ORDER BY av.avg_purchase DESC
123 )
124
125
126
127
128
129

```

Output

Time	Action	Message	Duration / Fetch
114 10:44:32	WITH avg_purchases_company AS(SELECT t.company_id, ROUND(AVG(t.amount),2) AS avg_purchase FROM transaction t WHERE declined = 0 GROUP BY t.company_id)	100 rows returned	0.328 sec / 0.000 /
115 10:44:49	CREATE VIEW VistaMarketing AS(WITH avg_purchases_company AS(SELECT t.company_id, ROUND(AVG(t.amount),2) AS avg_purchase FROM transaction t WHERE declined = 0 GROUP BY t.company_id) SELECT c.company_name, c.phone, c.country, av.avg_purchase FROM avg_purchases_company av JOIN company c ON c.id = av.company_id ORDER BY av.avg_purchase DESC)	0 rows affected	0.082 sec /

La ejecución de la vista es la siguiente:

```

119 FROM avg_purchases_company av
120 JOIN company c
121 ON c.id = av.company_id
122 ORDER BY av.avg_purchase DESC
123
124
125
126
127
128
129

```

View: transactions.vistamarketing

company_name	phone	country	avg_purchase
Air-Powerman Incorporated	06 65 50 52 53	Germany	284.51
Predium Netque Corp.	07 77 40 55 28	Australia	275.58
Umsa Convidis Associates	06 51 24 77 04	United States	272.87
Alt Associates	09 96 61 03 65	New Zealand	272.74
Metus Vixit Associates	08 33 44 40 68	Australia	270.03
Aliquid Dami Limited	02 76 61 47 46	United States	269.28
Nec Luctus LLC	02 14 71 75 73	Norway	268.60
Neque Tulas Incorporated	04 43 10 34 25	Ireland	267.58
Oras Consulting	07 50 10 01 03	Belgium	267.38
Seid LLC	01 67 10 26 52	Belgium	266.61

Output

Time	Action	Message	Duration / Fetch
133 19:29:01	SELECT * FROM transactions.vistamarketing	100 rows returned	0.408 sec /
134 19:29:18	SELECT * FROM transactions.vistamarketing	100 rows returned	0.422 sec /

Exercici 3

Filtra la vista VistaMarketing per a mostrar només les companyies que tenen el seu país de residència en "Germany".

Cridem la vista i filtrem amb WHERE.

```

134 SELECT *
135 FROM vistamarketing
136 WHERE country = 'Germany'
137
138

```

company_name	phone	country	avg_purchase
Ac Fermentum Incorporated	06 85 56 52 33	Germany	284.91
Nunc Interdum Incorporated	05 18 15 46 13	Germany	259.32
Cornellia In Incorporated	06 66 57 29 50	Germany	257.69
Ac Industries	09 34 65 40 60	Germany	255.17
Rubrum Non Inc.	02 66 31 61 09	Germany	255.14
Auctor Mauris Corp.	05 62 87 14 41	Germany	254.68
Augue Foundation	06 88 43 15 63	Germany	253.56
Aliquam PC	01 45 73 52 16	Germany	252.96

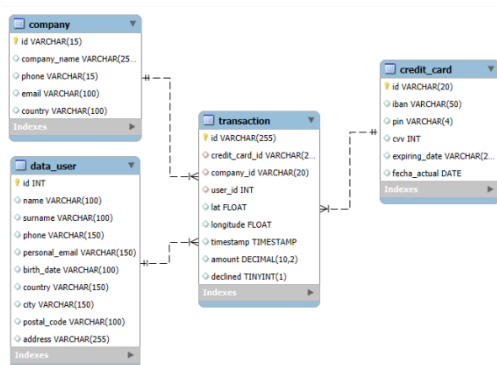
Output

#	Time	Action	Message	Duration / Fetch
137	19:12:38	SELECT * FROM vistamarketing	100 row(s) returned	0.360 sec / 0.000 sec
138	19:14:04	SELECT * FROM vistamarketing WHERE country = 'Germany'	8 row(s) returned	0.328 sec / 0.000 sec

Nivell 3

Exercici 1

La setmana vinent tindràs una nova reunió amb els gerents de màrqueting. Un company del teu equip va realitzar modificacions en la base de dades, però no recorda com les va realitzar. Et demana que l'ajudis a deixar els comandos executats per a obtenir el següent diagrama:



En aquesta activitat, és necessari que descriguis el "pas a pas" de les tasques realitzades. És important realitzar descripcions senzilles, simples i fàcils de comprendre. Per a realitzar aquesta activitat hauràs de treballar amb els arxius denominats "estructura_dades_user" i "dades_introduir_user"

Recorda continuar treballant sobre el model i les taules amb les quals ja has treballat fins ara.

Inicialment he carregat la taula users i les seves dades.

He mirat les característiques de les columnes y he vist la Primary Key id

The screenshot shows a database management tool interface. At the top, there's a toolbar with various icons and a dropdown menu set to 'Don't Limit'. Below the toolbar, a SQL query is entered: `SHOW COLUMNS FROM user;`. The main area displays the 'Result Grid' for this query, showing the structure of the 'user' table.

Field	Type	Null	Key	Default	Extra
id	char(10)	NO	PRI		
name	varchar(100)	YES			
surname	varchar(100)	YES			
phone	varchar(150)	YES			
email	varchar(150)	YES			
birth_date	varchar(100)	YES			
country	varchar(150)	YES			
city	varchar(150)	YES			
postal_code	varchar(100)	YES			
address	varchar(255)	YES			

Below the table structure, the 'Output' section shows the execution results of the query. It includes a table with columns: #, Time, Action, and Message.

#	Time	Action	Message
5139	19:22:48	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ("4999", "Ominoy..."	1 row(s) affected
5140	19:25:33	SHOW COLUMNS FROM user	10 row(s) returned

A la taula transaction hi és la columna t.user_id, per tant la utilitzarem com a clau forànea referenciada a la Primary key u.id i així relacionar les dues taules. Destacar que el tipus de dades no son les mateixes i les haurem de canviar

The screenshot shows the same database management tool interface. The SQL query is now: `SHOW COLUMNS FROM transaction;`. The 'Result Grid' displays the structure of the 'transaction' table.

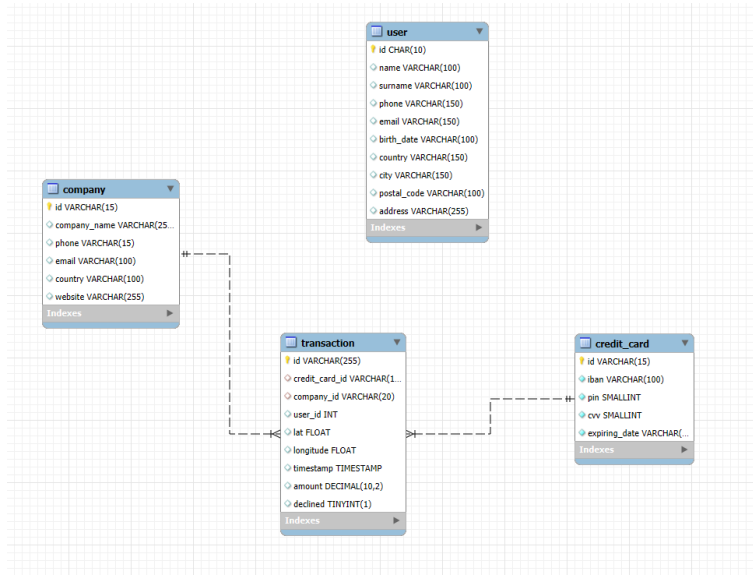
Field	Type	Null	Key	Default	Extra
id	varchar(255)	NO	PRI		
credit_card_id	varchar(15)	YES	MUL		
company_id	varchar(20)	YES	MUL		
user_id	int	YES			
lat	float	YES			
longitude	float	YES			
timestamp	timestamp	YES			
amount	decimal(10,2)	YES			
declined	tinyint(1)	YES			

Below the table structure, the 'Output' section shows the execution results of the query. It includes a table with columns: #, Time, Action, Message, and Duration.

#	Time	Action	Message	Duration
5143	19:30:17	SHOW COLUMNS FROM transaction	9 row(s) returned	0.000 s
5144	19:34:04	SHOW COLUMNS FROM transaction	9 row(s) returned	0.000 s

On the right side of the interface, there's a sidebar with various tool icons. A tooltip is visible over the 'Table Grid' icon, stating: 'Automatic con disabled. Use tl manually get l current caret p toggle auton'.

Busquem l'esquema actual o bé amb el teclat Ctrl+R o bé a través de Database –Reverse Engineer.



Comprovem les relacions que ja existeixen i veiem que el nom de la taula es user i a l'esquema indicat ha d'apareixer com data_user i falta la relació amb la taula transaction, així com que els tipus de dades de la id a la taula actual i a l'esquema son diferents, per tant haurem de modificar id a tipus de dades INT i així poder relacionar-les correctament.

Primer canviarem el nom de la taula amb RENAME TABLE:

```

135 FROM vistamarketing
136 WHERE country = 'Germany'
137
138
139
140 -- Nivell 3
141
142 -- Exercici 1
143 -- La setmana vinent tindrà una nova reunió amb els gerents de marketing. Un company del teu equip va realitzar modificacions en la base de dades,
144 -- però no records com les va realitzar. Et demana que l'ajudis a deixar els comandos executats per a obtenir el diagrama de relacions.
145
146 SHOW COLUMNS FROM user;
147 SHOW COLUMNS FROM transaction;
148
149 RENAME TABLE user TO data_user;
150
151
152
153
154
155
156
157
158
159
160
161
162
163

```

Output:

Time	Action	Message	Duration / Fetch
19:30:00	DESCRIBE TABLE transaction	1 row(s) returned	0.000 sec / 0.000 sec
19:40:31	RENAME TABLE user TO data_user	0 row(s) affected	0.140 sec

Cambiarem el tipus de dada de la columna id de la taula data_user amb ALTER TABLEMODIFY la veurem executant SHOW COLUMNS:

SPRINT3

```
151
152 • ALTER TABLE data_user
153   MODIFY id INT NOT NULL ;
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
```

#	Time	Action	Message	Duration / Fetch
10180	20:31:17	ALTER TABLE data_user MODIFY id INT NOT NULL	5000 row(s) affected Records: 5000 Duplicates: 0 Warnings: 0	1.829 sec

```
146 SHOW COLUMNS FROM user;
147 • SHOW COLUMNS FROM transaction;
148
149 • RENAME TABLE user TO data_user;
150 • SHOW COLUMNS FROM data_user;
151
152 • ALTER TABLE data_user
153   MODIFY id INT NOT NULL ;
154
155
156
157
158
159
160
161
162
163
164
```

Field	Type	Null	Key	Default	Extra
id	int	NO	PRI	NULL	
name	varchar(100)	YES		NULL	
surname	varchar(100)	YES		NULL	
phone	varchar(150)	YES		NULL	
email	varchar(150)	YES		NULL	
birth_date	varchar(100)	YES		NULL	
country	varchar(150)	YES		NULL	
idnu	varchar(100)	YES		NULL	

#	Time	Action	Message	Duration / Fetch
10182	20:32:19	SHOW COLUMNS FROM data_user	10 row(s) returned	0.015 sec
10183	20:32:39	SHOW COLUMNS FROM data_user	10 row(s) returned	0.000 sec

Y després afegim la relació a través de la FK i tornem a executar SHOW COLUMNS a on podem comprovar que ja està establerta la relació:

```
159
160 • ALTER TABLE transaction
161   ADD CONSTRAINT fk_transaction_data_user_id
162   FOREIGN KEY (user_id) REFERENCES data_user (id)
163   ON UPDATE CASCADE
164   ON DELETE SET NULL;
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
```

#	Time	Action	Message	Duration / Fetch
10190	20:37:50	SHOW COLUMNS FROM data_user	10 row(s) returned	0.000 sec / 0.1
10191	20:37:54	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_data_user_id FOREIGN KEY (user_id) REFERENCES data_user (id) ...	99999 row(s) affected Records: 99999 Duplicates: 0 Warnings: 0	11.515 sec

SPRINT3

```

159
160 • ALTER TABLE transaction
161   ADD CONSTRAINT fk_transaction_data_user_id
162   FOREIGN KEY (user_id) REFERENCES data_user (id)
163   ON UPDATE CASCADE
164   ON DELETE SET NULL;
165
166
167
168
169
170

```

Field	Type	Null	Key	Default	Extra
id	varchar(255)	NO	PRI		
credit_card_id	varchar(15)	YES	MUL		
company_id	varchar(20)	YES	MUL		
user_id	int	YES	MUL		
lat	float	YES			
longitude	float	YES			
timestamp	timestamp	YES			
amount	decimal(10,2)	YES			

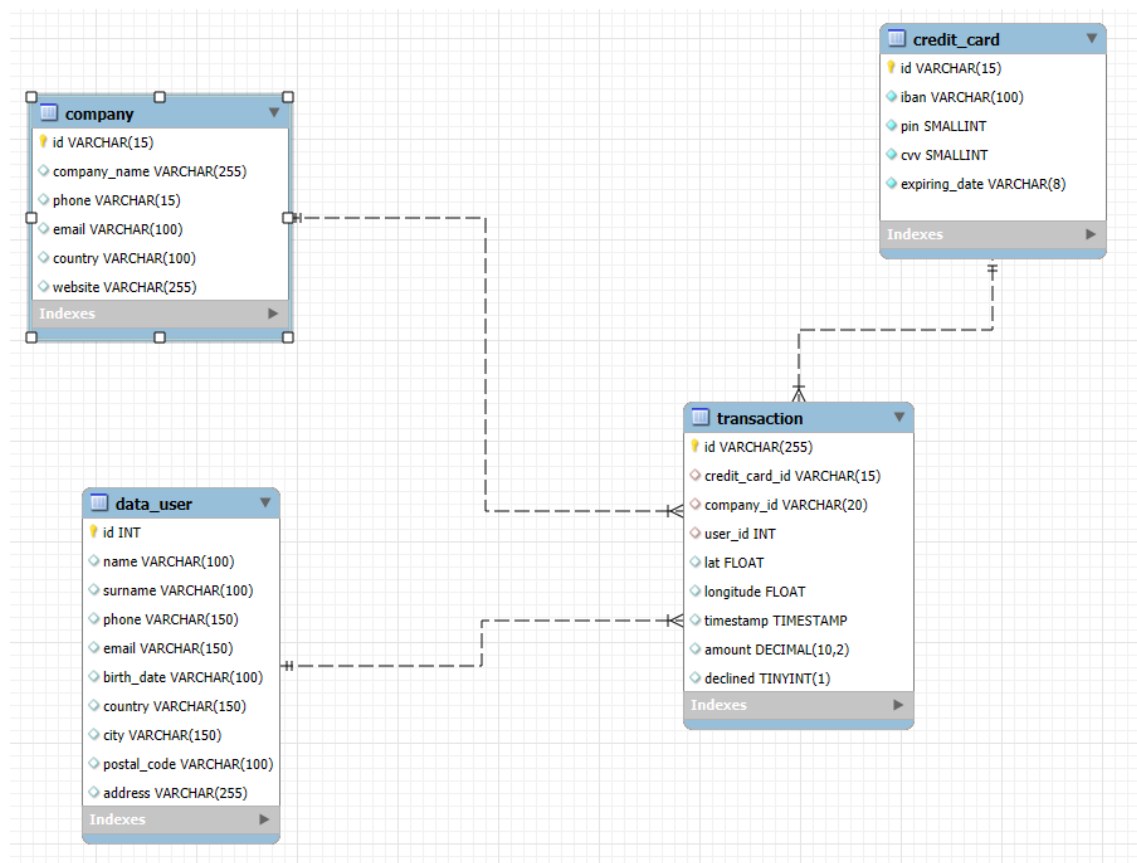
Result 34 x

Output

Action Output

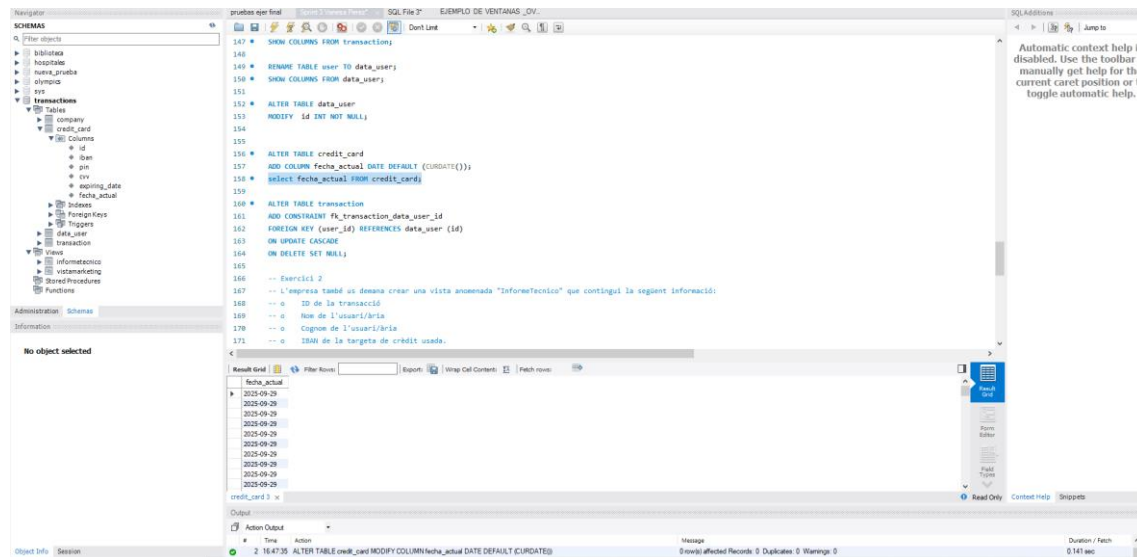
#	Time	Action	Message	Duration / Fetch
10191	20:37:54	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_data_user_id FOREIGN KEY (user_id) REFERENCES data_user (id) ...	99999 row(s) affected Records: 99999 Duplicates: 0 Warnings: 0	11.515 sec
10192	20:38:31	SHOW COLUMNS FROM transaction	9 row(s) returned	0.015 sec / 0.000 sec

Per últim comprovarem com ha quedat el diagrama:



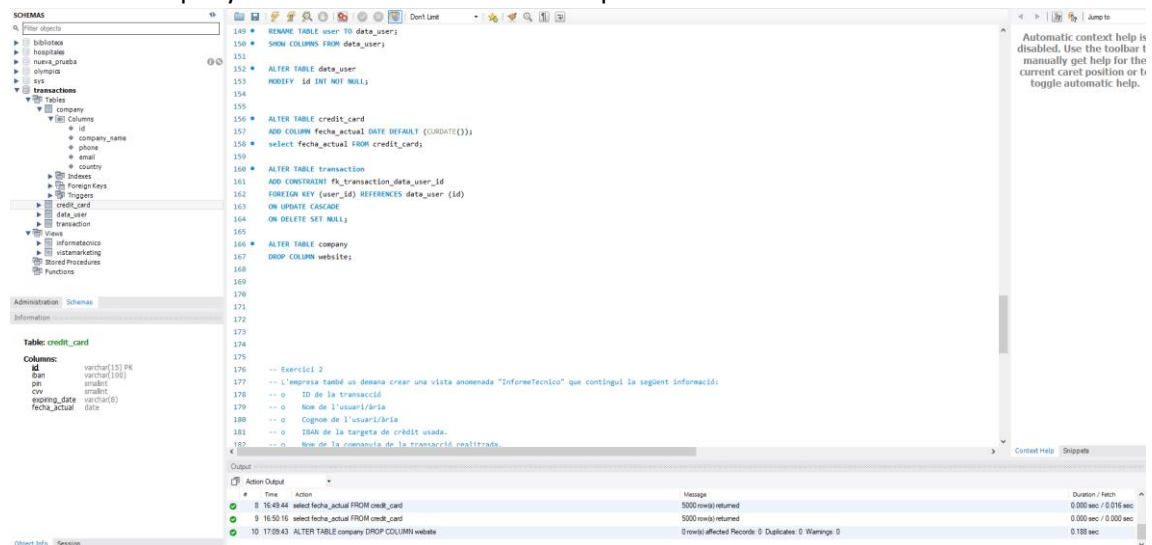
Veiem que encara cal afegir una columna nova a la taula **credit_card** que s'anomena **fecha_actual** y es del tipus data: ho fem amb **ALTER TABLE....ADD COLUMN**

SPRINT3



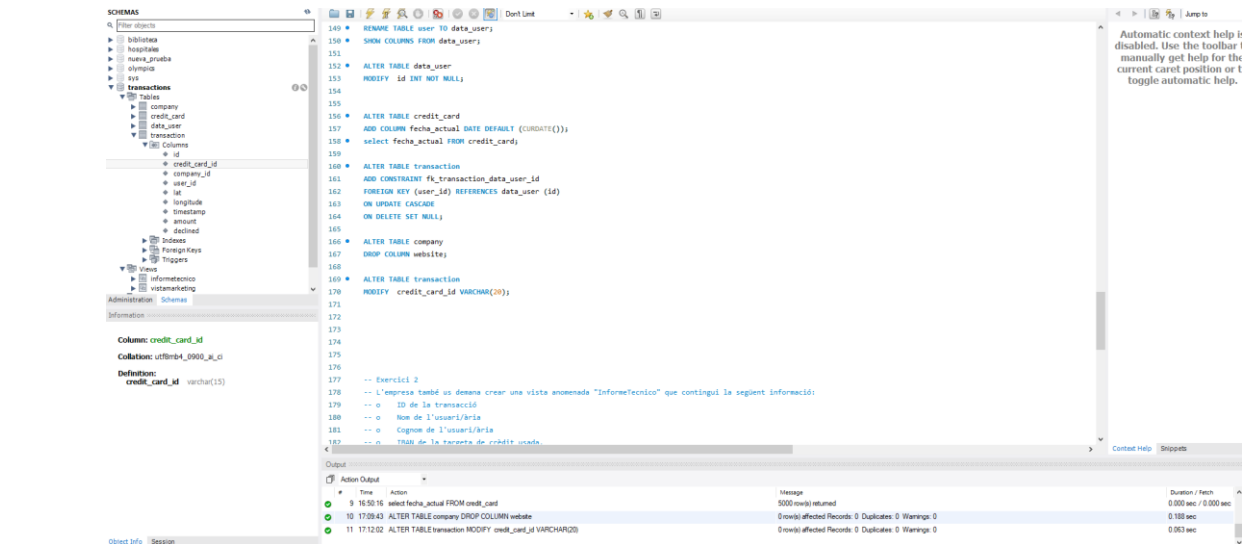
Al tornar a revisar els esquemes veiem que:

- A la taula company actual li hem de treure el camp de website.

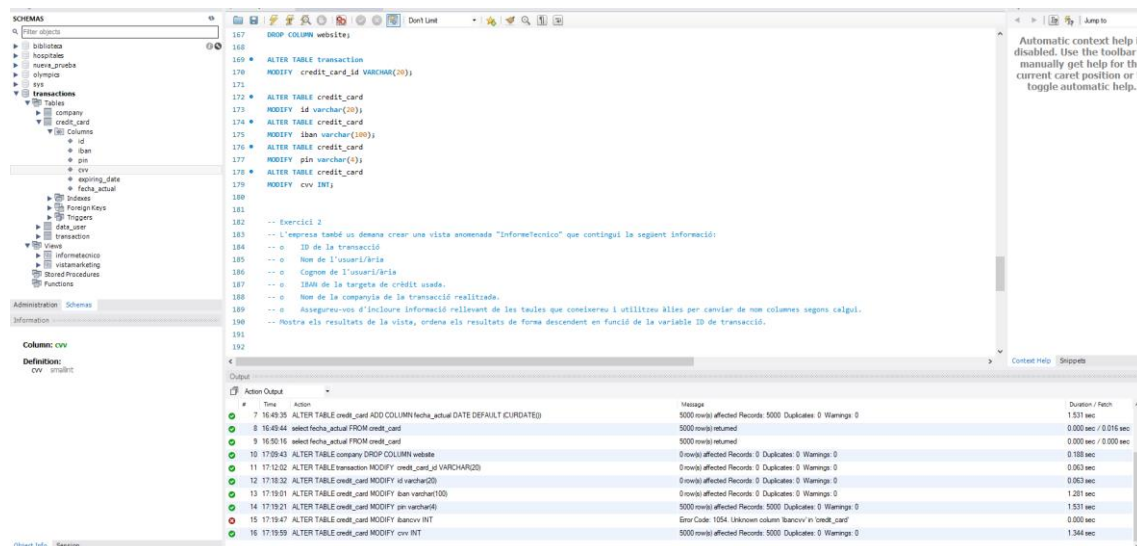


- A la taula transaction a la columna credit_card_id li he de canviar el tipus de lletra a VARCHAR(20).

SPRINT3



- A la taula credit_card s'han de canviar els tipus de lletra del següents camps:
 - id
 - iban
 - pin
 - ccv



A la taula data_user hem de canviar el nom del camp email pel de personal_email:

SPRINT3

The screenshot shows a database management interface. On the left, a schema diagram displays tables: olympics, sys, transactions, company, credit_card, data_user, and transaction. The 'data_user' table is highlighted, showing its columns: id, name, surname, phone, personal_mail, birth_date, country, city, postal_code, address, and a foreign key relationship with the 'transaction' table.

The center pane shows the output of SQL execution, including a list of actions and their results. The actions include selecting data from 'credit_card', dropping a column from 'company', and altering various tables and columns. The results show the number of rows affected, duplicates, and warnings for each action.

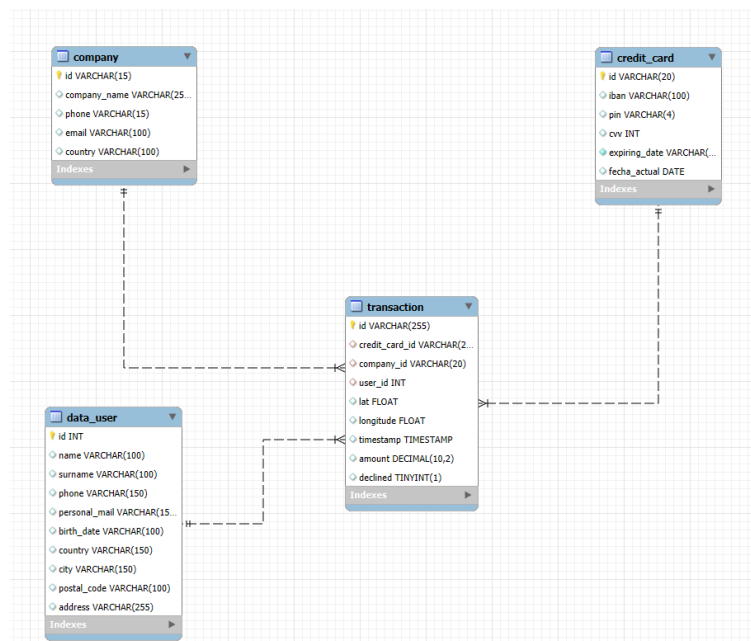
The right pane shows the definition of the 'data_user' table, including its columns and data types:

```

Table: data_user
Columns:
id INT PK
name varchar(100)
surname varchar(100)
phone varchar(150)
personal_mail varchar(150)
birth_date varchar(150)
country varchar(150)
city varchar(150)
postal_code varchar(100)
address varchar(255)

```

L'estat actual de l'esquema coincideix amb el que ens ha donat l'enunciat en quant a nom de taules, nom de camps, tipus de dades i relacions entre taules:



Exercici 2

L'empresa també us demana crear una vista anomenada "InformeTecnico" que contingui la següent informació:

- ID de la transacció
- Nom de l'usuari/ària
- Cognom de l'usuari/ària
- IBAN de la targeta de crèdit usada.
- Nom de la companyia de la transacció realitzada.
- Assegureu-vos d'incloure informació rellevant de les taules que coneixereu i utilitzeu àlies per canviar de nom columnes segons calgui.

Mostra els resultats de la vista, ordena els resultats de forma descendent en funció de la variable ID de transacció.

He creat una taula amb els camps que sol·liciten i he afegit diverses columnes d'anàlisis creades amb finestres per fer càlculs grups i no perdre la informació de cada transacció.

Amb aquesta taula he creat la finestra sol·licitada:

The screenshot shows a SQL IDE with a schema browser on the left, a central SQL editor, and an output window at the bottom. The schema browser shows a database named 'informatica' with several tables including 'transaction'. The SQL editor contains the following code:

```

224 * CREATE VIEW InformeTecnico AS(
225     WITH cte_user_transaction AS(
226         SELECT DATE(timestamp) as date, t.id, t.user_id, d.name, d.surname, cc.iban, c.company_name, amount
227         FROM company c
228             JOIN transaction t
229             ON c.id = t.company_id
230             JOIN credit_card cc
231             ON cc.id = t.credit_card_id
232             JOIN data_user d
233             ON d.id = t.user_id
234         WHERE declined = 0
235         ORDER BY t.id
236     )
237     SELECT date, id, user_id, name, surname, iban, company_name, amount,
238           SUM(amount) OVER (PARTITION BY user_id) AS total_purchases_user,
239           ROUND(AVG(amount) OVER (PARTITION BY user_id),2) AS average_purchase_user,
240           MAX(amount) OVER (PARTITION BY user_id) AS purchase_max_user,
241           MIN(amount) OVER (PARTITION BY user_id) AS purchase_min_user,
242           SUM(amount) OVER () AS total_purchases,
243           ROUND(AVG(amount) OVER ()),2) AS average_total_purchases,
244           ROUND((amount / (SUM(amount) OVER (PARTITION BY user_id))) * 100, 2) AS Perc_purchases_user_totales
245     FROM cte_user_transaction cut
246     ORDER BY id
247 );
248
249 * SELECT *
250 FROM InformeTecnico;
251
252 * CREATE INDEX ind_transaction_amount ON transaction(amount);
253 * CREATE INDEX ind_transaction_user_id ON transaction(user_id);
254
255
256
257
258
259
260
261
262

```

The output window at the bottom shows the execution results:

Time	Action	Message	Duration / Fetch
31 10:38:02	SELECT * FROM InformeTecnico	99763 rows returned	16.672 sec / 0.359 sec
32 10:42:45	DROP VIEW 'transaction'.InformeTecnico	0 rows affected	0.063 sec
33 10:42:54	CREATE VIEW InformeTecnico AS(WITH cte_user_transaction AS(SELECT DATE(timestamp) as date, t.id, t.user_id, d.name, d.surname, cc.iban, c.company_name, amount FROM company c JOIN transaction t ON c.id = t.company_id JOIN credit_card cc ON cc.id = t.credit_card_id JOIN data_user d ON d.id = t.user_id WHERE declined = 0 ORDER BY t.id) SELECT date, id, user_id, name, surname, iban, company_name, amount, SUM(amount) OVER (PARTITION BY user_id) AS total_purchases_user, ROUND(AVG(amount) OVER (PARTITION BY user_id),2) AS average_purchase_user, MAX(amount) OVER (PARTITION BY user_id) AS purchase_max_user, MIN(amount) OVER (PARTITION BY user_id) AS purchase_min_user, SUM(amount) OVER () AS total_purchases, ROUND(AVG(amount) OVER ()),2) AS average_total_purchases, ROUND((amount / (SUM(amount) OVER (PARTITION BY user_id))) * 100, 2) AS Perc_purchases_user_totales FROM cte_user_transaction cut ORDER BY id);	0 rows affected	0.062 sec

He cridat la view per comprovar que funciona. Com he arreglat a posterior de crear els index la vista, al pantallazo ja surt reduït però en realitat sense l'index triguem 10 segons més.

SPRINT3

Automatic context help disabled. Use the toolbar manually get help for current caret position or toggle automatic help

```

227 FROM company c
228 JOIN transaction t
229 ON c.id = t.company_id
230 JOIN credit_card cc
231 ON cc.id = t.credit_card_id
232 JOIN data_user d
233 ON d.id = t.user_id
234 WHERE declined = 0
235 ORDER BY t.id
236
237 SELECT date, id, user_id, name, surname, iban, company_name, amount,
238 ROUND(AVG(amount) OVER (PARTITION BY user_id) AS total_purchases_user,
239 MAX(amount) OVER (PARTITION BY user_id) AS purchase_max_user,
240 MIN(amount) OVER (PARTITION BY user_id) AS purchase_min_user,
241 SUM(amount) OVER () AS total_purchases,
242 ROUND(SUM(amount) OVER (PARTITION BY user_id) AS average_purchase_user,
243 FROM cte_user_transaction cut
244 ORDER BY id
245
246
247
248
249
250
251

```

date	id	user_id	name	surname	iban	company_name	amount	total_purchases_user	average_purchase_user	purchase_max_user	purchase_min_user
2024-08-28	00043448-3848...	4713	Conny	Purdon	X881796022284713133643889	Eggt Product Oil Institute	395.40	4986.48	262.49	812.11	14.18
2020-07-14	00045048-620E...	2118	Vishale	Umuny	X373322868262791235278991	Reque Telus Imperdet Corp.	326.01	4032.86	237.23	526.01	51.94
2017-09-04	000483C3-3C26...	2115	Unhuva	Umdunum	X3705022829713140880703252	Puace Corp.	161.60	4932.73	259.62	604.51	20.92
2017-01-05	0001544-F00E...	3038	Qpfyfa	Pmpfufqj	X0024626846308739323468	Kiaa Aenean Eget Foundation	148.91	5335.90	251.63	521.61	60.32
2025-09-23	0008A312-EDFE...	215	Kengen	Welson	R07604068334800208155	Alquam Jacula Lacus Corp.	294.59	7914.79	263.83	569.66	9.46
2023-12-31	0009A1S1-8BCF...	2938	Rneqrs	Jbvgpne	X098633577818763892658326	Lozem Ipsum Dolor Corp.	383.83	9526.49	269.39	611.80	13.14
2017-07-18	0002084-C45...	3902	Nemene	Duchtham	X373807454411531571218547	Aenean Luctus Vulputate Foundation	157.80	4528.65	238.35	452.69	20.92
2018-09-08	000A2DEC-C08E...	1886	Huzen	Machymyz	X3714741880105967529287700	Nulla Integer Vulputate Corp.	326.94	3806.99	271.93	507.98	79.53
2022-09-17	000A2B44-3414...	1385	Ghong	Gdufwel2	X314381394176393663185294	Acitor Hicurs Corp.	369.71	6211.26	259.64	602.16	40.43
2020-02-07	000A0879-347C...	42	Lacy	Branch	D62123121288627227	Quamod Neurs Institute	382.43	6918.72	388.28	611.98	31.52
2017-07-10	000A0D04-F06E...	3553	Evdhij	PfyaA886	X06076716020396121547880	Pede Cum Ltd	188.94	3454.91	181.84	514.64	9.46
2019-05-28	000B08F8-4V4E...	1459	Wlokoz	Tbfhosee	X04958011473946372918956	Alquam Jacula Lacus Corp.	96.22	7388.58	256.66	474.55	9.46
2024-07-15	00000312-C336...	864	Kyoff	Aulesteng	X329462384711398635063023	Dorac Fingula PC	93.13	6323.83	274.94	507.95	93.13

Informetecnico 19

Output

#	Time	Action	Message
32	10:42:45	DROP VIEW 'Informetecnico'	0 row(s) affected
33	10:42:54	CREATE VIEW Informetecnico AS WITH cte_user_transaction AS (SELECT DATE(timestamp) as date, t.id, t.user_id, t.name, t.surname, t.amount FROM Informetecnico)	0 row(s) affected
34	10:43:51	SELECT * FROM Informetecnico	93763 row(s) returned

Per reduir el temps d'execució he creat dos index:

```

281
282 • SELECT *
283 FROM Informetecnico;
284
285 • CREATE INDEX data_user_id ON data_user(id);
286 • CREATE INDEX indx_transaction_amount ON transaction(amount);
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318

```

Output

#	Time	Action	Message
43	11:44:55	CREATE INDEX data_user_id ON data_user(id)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
44	11:46:11	CREATE INDEX indx_transaction_amount ON transaction(amount)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

Comprovem la reducció de temps:

Query Text:

```

193 WITH (amount) OVER (PARTITION BY user_id ) AS compra_max_user,
194 MIN(amount) OVER (PARTITION BY user_id ) AS compra_min_user,
195 SUM(amount) OVER () as total_ventas,
196 AVG(amount) OVER () as media_total_ventas,
197 (amount/ (SUM(amount) OVER (PARTITION BY user_id))) * 100 AS Porc_compra_user_sobre_compras_totales
198 FROM cte_user_transaction cut
199 ORDER BY id
200 );
201
202 * SELECT *
203 FROM InformeTecnico;
204
205 * CREATE INDEX Indx_transaction_amount ON transaction(amount);
206 * CREATE INDEX Indx_transaction_user_id ON transaction(user_id);
207
208
209

```

Results Grid:

date	id	user_id	name	surname	ban	company_name	amount	total_compras_user	media_compra_user	compra_max_user	compra_min_user	tot
2024-08-28	00043440-2046...	4713	Germys	Fryceone	X389179022224713133661889	Eget Thosant Dai Institute	395.43	4086.48	262.446316	613.12	14.18	238
2020-07-14	0004506-452E...	2138	Violante	Ukynunly	XX715228809262791935275891	Neque Telus Imperdet Corp.	326.61	4032.86	237.227059	526.61	51.94	238
2017-09-04	000481C3-3C26...	2115	Unhuva	Ubdendome	XX78502282971314080782522	Fuice Corp.	181.60	4932.73	259.617368	604.51	20.52	238
2017-01-01	000514A4-9D0E...	3025	QyKfa	Hmupfelf	X381942630453687893321845	Rux Annem Eget Foundation	148.91	5538.90	251.61818	521.61	60.33	238
2023-09-23	0006A712-8DFF...	215	Kegan	Watson	RO7NDAPC683348380208155	Alquam Tacus Lacus Corp.	294.59	7914.79	263.826333	589.66	9.46	238
2023-12-11	0006A151-WDCF...	2928	Rmngz	Zfbcvne	X388633177818763823658136	Lozem Ipsum Dolor Corp.	283.63	8936.49	269.389308	813.80	13.14	238
2017-07-18	0006C4H4-4C45...	2962	Nemem	Bumtatem	XX7280745451153371218547	Amet Luctus Volutate Foundation	197.80	4526.65	238.330000	432.09	20.52	238
2018-09-08	000A2EC-CD6E...	1886	Huzen	Machymez	XX7147418801050675282790	Nulla Integer Volutate Corp.	339.94	3806.99	271.927857	587.98	79.53	238
2022-05-17	000A2554-1414...	1385	Ghong	Gashelz	XX14381394176393653185254	Auctor Maura Corp.	269.71	6231.26	259.638813	602.38	40.43	238
2020-05-07	000A8D79-3A72...	42	Lucy	Branch	DC521621138675727	Etiamd Neums Institute	362.43	8918.72	388.380000	611.98	31.52	238
2017-07-10	000AED-4F0E...	3553	Evdqj	Prlykadd	X38076736220239921547880	Fede Cum Ltd	188.94	3454.91	181.837368	514.64	9.46	238
2018-05-28	000C0F8-9A8E...	1459	Wlqoz	Tfblatene	XX49582114729463729168836	Alquam Tacus Lacus Corp.	96.22	7108.58	256.663571	494.56	9.46	238
2024-07-11	000D0F52-423B...	864	Kyeff	Alueregen	XX1464632847113662060223	Doreet Fregdia PC	92.13	6323.65	274.944304	367.95	92.13	238
2023-07-26	000DA8F-ABEA...	1823	Zngopt	Gvtyvsey	X380940388265634132291812	Justa Eu Anru Ltd	298.18	6011.87	222.661852	532.38	30.14	238
2018-06-01	000E4E7-8B4...	3023	Ruphal	Smakdchp	XX262616788779151720207	Eget Thosant Dai Institute	171.13	5564.49	196.627222	583.11	13.14	238
2018-05-30	0006A42-3C13...	2156	Prmuvus	Mceazfti	XX71281638679576175879821	Lacus Quisque Associates	105.03	4605.00	232.795000	589.75	13.14	238

Execution Log:

Time	Action	Message	Duration / Fetch
58 13:01:23	CREATE VIEW InformeTecnico AS: SELECT DATE@timestamp() as date, id, user_id, name, d.nur...	Error Code: 1050. Table 'InformeTecnico' already exists	0.000 sec
59 13:01:26	SELECT * FROM InformeTecnico	95762 row(s) returned	17.287 sec / 0.375 sec