

## SPRINT 03

### Nivell 1 - Exercici 1

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.

### 1.- DISSENYA I CREAR LA TAULA CREDIT CARDS

El codi de la targeta ha de ser clau primària per poder ser identificat com a únic i no serà mai un valor null. S'ha de poder saber a quin client pertany la targeta de crèdit i quina transacció de venda a realitzat.

The screenshot displays a database management interface with a left sidebar showing a schema tree and a main window with SQL code and its execution output.

**SQL Code:**

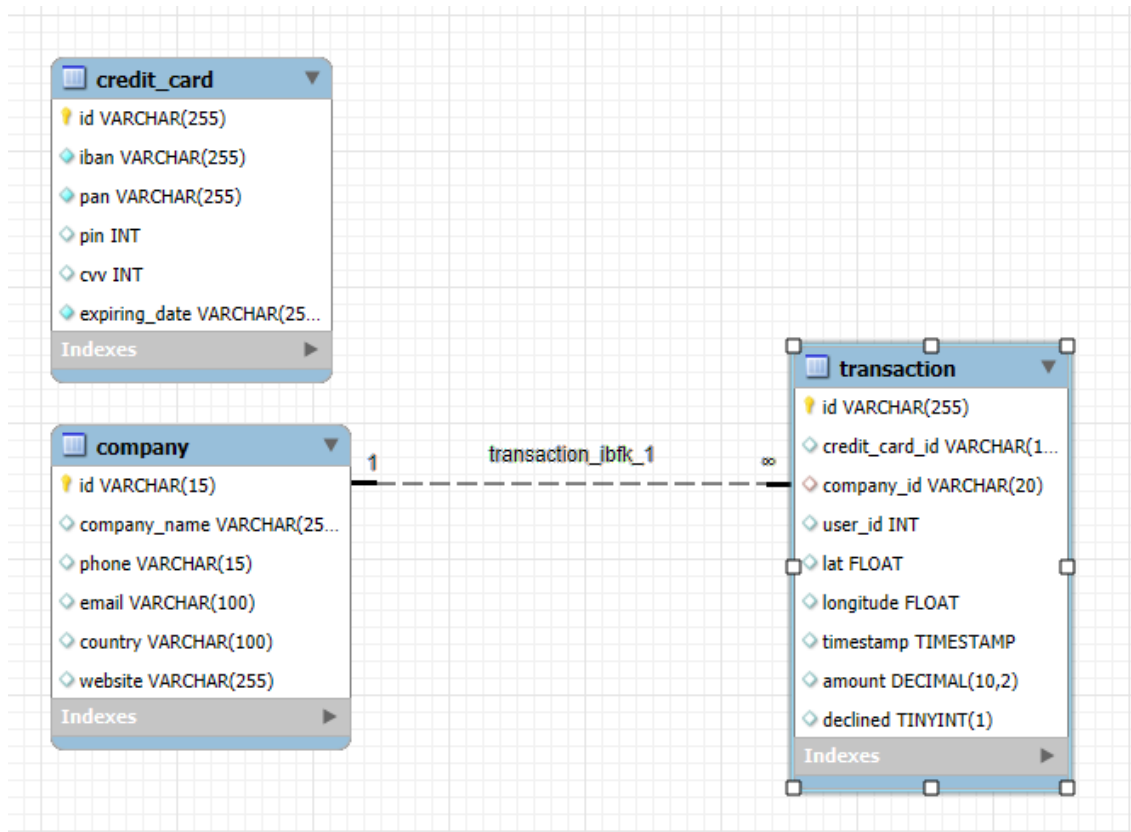
```

58 DROP TABLE IF EXISTS credit_card;
59 CREATE TABLE credit_card (
60     id VARCHAR(255) NOT NULL PRIMARY KEY,
61     iban VARCHAR(255) NOT NULL,
62     pan VARCHAR(255) NOT NULL,
63     pin INT,
64     cvv INT,
65     expiring_date NVARCHAR(255) NOT NULL )
66 ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
67
68 -- subimos datos a credit_card
69 -- revisamos diseño de la base de datos
70
71 -- CREAR clau foranea de credit_card en la taula transaction .
72 ALTER TABLE transaction
73 ADD CONSTRAINT fk_transaction_card
74 FOREIGN KEY (credit_card_id)
75 REFERENCES credit_card(id )
76 ON UPDATE CASCADE;
77
78
79
80 -- Modifiquem les taules companyies i transactions per que es puguin relacionar amb credit_card
81
82 ALTER TABLE no
83 DROP FOREIGN KEY nombre_del_constraint_de_la_fk;
84
85 ALTER TABLE company
86 RENAME COLUMN id TO company_id;
87
88
89

```

**Output:**

#	Time	Action	Message
105122	13:16:57	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccs-9578', 'XX991539646456110567870254', '3999808823061411', ...	1 row(s) affected
105123	13:16:57	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccs-9679', 'XX296393091587170202131236', '9690060468678689', ...	1 row(s) affected
105124	13:16:57	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccs-9580', 'XX781258889851950806677358', '5541182364498931', ...	1 row(s) affected
105125	13:16:57	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccs-9581', 'XX915670516405388124398147', '2624305470167630', ...	1 row(s) affected

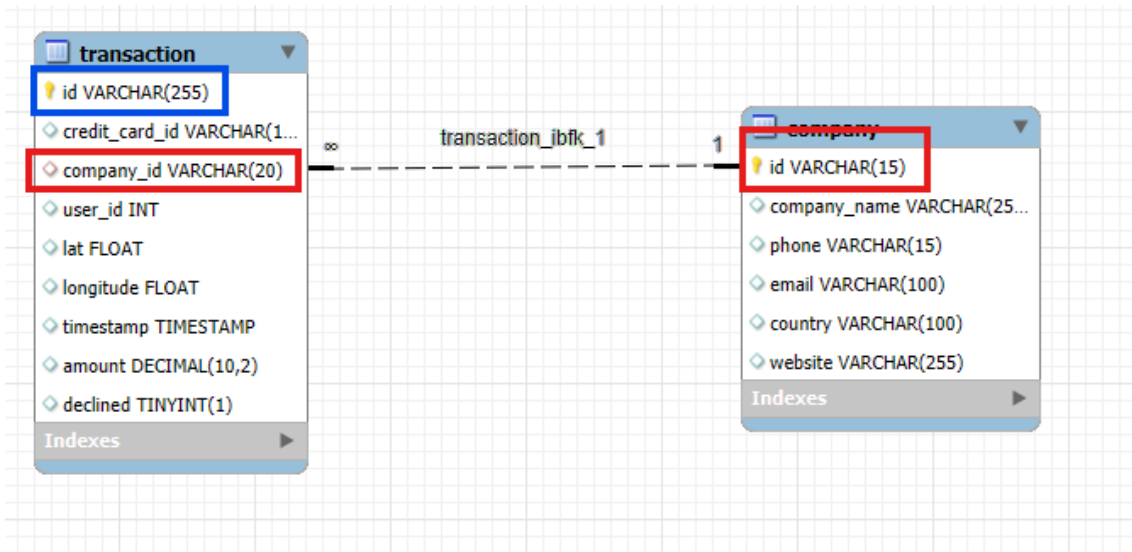


## 2.- CREACIÓ DE RELACIONS AMB LES TAULES JA EXISTENTS

### 2.1.- Taules existents i com es relacionen.

Existeixen ja a la base de dades les següents taules, Transaction i company. Aquestes dues taules les haurem de relacionar amb la nova taula, crèdit cards.

Es important tenir en compte que el camp ID d'ambdues taules són les seves respectives claus primàries i que transaction la definirem com a taula de fets. Això significa que a la taula transaction hi hauran d'existir les claus foranies cap a company i cap a crèdit cards de forma que puguin relacionar-se en un model d'estrella. Ara existeix la relació amb company, veure rectangle vermell. Ens manca doncs la relació amb credit\_cards.



## 2.2.- Definir claus primàries i foranies.

Per la taula credit\_card es crearà una clau primària al camp card\_id. Es crea la clau forana de la taula credit\_card, i serà el camp credit\_id.

**Navigator**

Filter objects

- SP\_Clear\_MarketPlace\_BDD
- Functions
- sys
- transactions
  - Tables could not be fetched
  - company
    - Columns
      - id
      - company\_name
      - phone
      - email
      - country
      - website
    - Indexes
    - ForeignKeys
    - Triggers
    - transaction
      - Columns
        - id
        - credit\_card\_id
        - company\_id

Administration **Schemas**

Information

Schema: **transactions**

Top Panel: T1A\_Sprint 03-AnsCPichot\_v1" | detose\_introducir\_sprint3\_credit

Don't Limit

```

58 * DROP TABLE IF EXISTS credit_card;
59 * CREATE TABLE credit_card (
60     id VARCHAR(255) NOT NULL PRIMARY KEY,
61     iban VARCHAR(255) NOT NULL,
62     pan VARCHAR(255) NOT NULL,
63     pin INT,
64     cvv INT,
65     expiring_date NVARCHAR(255) NOT NULL )
66 ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;
67
68 -- subimos datos a credit_card
69 -- revisamos diseño de la base de datos
70
71 -- CREAMOS la foreign de credit_card en la tabla transaction .
72 * ALTER TABLE transaction
73     ADD CONSTRAINT fk_transaction_card
74     FOREIGN KEY (credit_card_id)
75     REFERENCES credit_card(id )
76     ON UPDATE CASCADE;
77
78
79
80 -- Modificamos los tablas companies y transactions per que es puguin relacionar amb credit_card
81
82 * ALTER TABLE no
83     DROP FOREIGN KEY nombre_del_constraint_de_la_fk;
84
85 * ALTER TABLE company
86     RENAME COLUMN id to company_id;
87
88
89

```

Output:

#	Time	Action	Message
105123	13:16:57	INSERT INTO credit_card (id,iban,pan,pin,cvv,expiring_date) VALUES (Cc5-9579,'XXC296393091587170202131236','9690060468678689'...	1 row(s) affected
105124	13:16:57	INSERT INTO credit_card (id,iban,pan,pin,cvv,expiring_date) VALUES (Cc5-9580,'XXC781258888951950806677358','5541182364498931'...	1 row(s) affected
105125	13:16:57	INSERT INTO credit_card (id,iban,pan,pin,cvv,expiring_date) VALUES (Cc5-9581,'XX0315670516405388124388147','26243054701676307'...	1 row(s) affected
105126	13:21:19	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_card FOREIGN KEY (credit_card_id) REFERENCES credit_card(id) ON UPDATE...	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings: 0

### 3.- PUJAR LES DADES A LA TAULA CREDIT\_CARDS

Tenim un fitxer amb les dades que haurem d'insertar que ja ens indiquen quins camps haurem de tenir a la taula i són: id, iban, pan, pin, cvv, expiring\_date.

El camp expiring date és un format del tipus 10/30/22 això és MM/DD/YY però MYSQL treballa com a YYYY-MM-DD. Hi ha dues opcions :

- 1.- Pujar les dades en format varchar, cadena de caràcters.
- 2.- Modificar el camp expiring date perquè quedi com a data i més endavant puguem fer operacions amb aquest camps, com calcular dies que falten per que la data de la targeta expiri i poder enviar un mail a facturació o al client avisant de la propera caducitat. En aquest segon cas les passes a realitzar serien

1. Pujar les dades en format varchar, cadena de caràcters.
2. Modificar el camp expiring\_date i convertir-lo a data.
3. Modificar la taula i deixar-la com a data MM-DD-YYYY format standard de mysql
4. Quan fem seleccions escollir el format que li agradi més al client.

```

45 -- =====
46 -- preparar camps dates de la taula credit_card. No es demana explícitament.
47 -- =====
48
49 • SELECT expiring_date
50 FROM transactions.credit_card cc;
51
52 -- USE TRANSACTIONS;
53 • UPDATE credit_card cc -- Step. 3 Update values from type varchar to date so we can use them
54 SET expiring_date = STR_TO_DATE(expiring_date, '%m/%d/%Y');
55
56 • SELECT expiring_date
57 FROM transactions.credit_card cc;
58
59 • ALTER TABLE credit_card -- Nota no és possible utilitzar un aliàs amb alter table. -- Setp 4.- Prepare table for data analysis.
60 MODIFY expiring_date DATE;
61
62 • SELECT expiring_date, DATE_FORMAT(cc.expiring_date, '%m/%d/%Y') AS exp_date -- Set 5.- make sure it works properly
63 FROM transactions.credit_card cc;
64
65

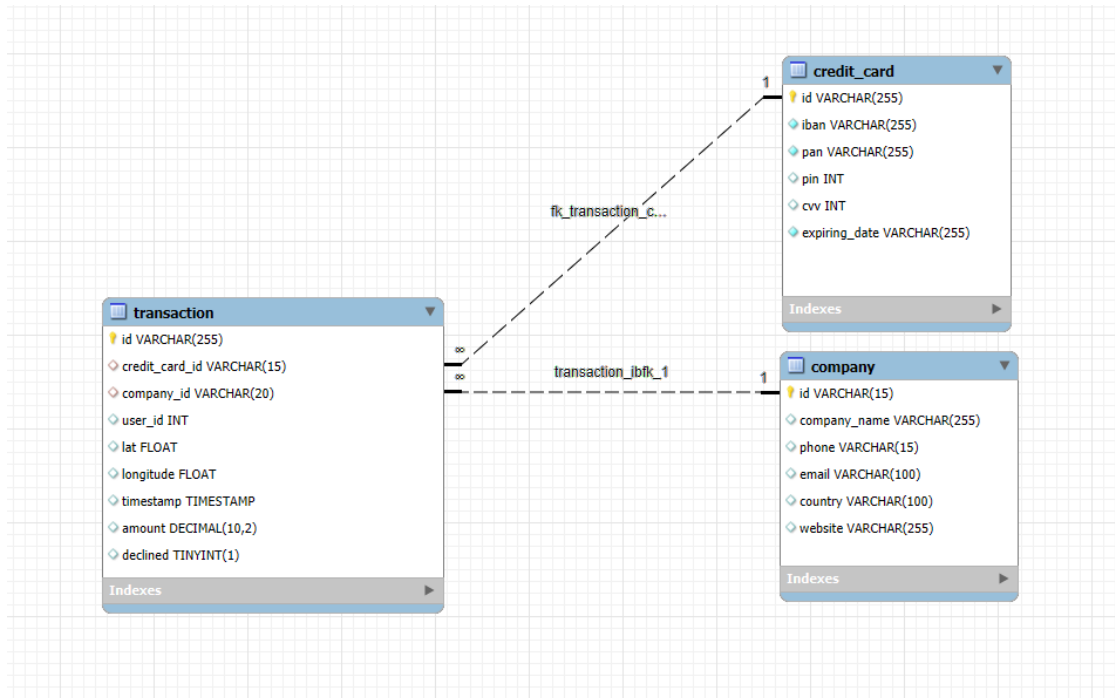
```

expiring_date	exp_date
2025-09-27	09/27/2025
2028-12-28	12/28/2028
2026-11-26	11/26/2026
2027-07-27	07/27/2027
2026-04-25	04/25/2026

#	Time	Action	Message
3	10:13:26	SELECT expiring_date FROM transactions.credit_card cc	5000 row(s) returned
4	10:13:26	ALTER TABLE credit_card -- Nota no és possible utilitzar un aliàs amb alter table. -- Setp 4.- Prepare table for data analysis. MODIFY expiring_date...	5000 row(s) affected Records: 5000 Duplicates: 0 Warnings: 0
5	10:13:26	SELECT expiring_date, DATE_FORMAT(cc.expiring_date, '%m/%d/%Y') AS exp_date-- Set 5.- make sure it works properly FROM transactions.cr...	5000 row(s) returned

## 4.- MOSTRAR EL DISSENY FINAL DE LA BASE DE DADES.



## Nivell 1 - Exercici 2

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.

Navigator: Schemas

- Filter objects
- biblioteca
- classmodels
- dorita
- hospitals
- sakila
- sys
- transactions
  - Tables
    - company
    - credit\_card
    - transaction
  - Views
  - Stored Procedures
  - Functions
- world

Query 1: ITA\_Sprint 03-AnaC Pichot | datos introducir sprint3 user

```

69
70 USE TRANSACTIONS;
71 -- Step 1.- Check if id exists
72 SELECT
73   cc.iban AS OLD
74 FROM credit_card cc
75  WHERE cc.id='CcU-2938';
76
77 -- Step 2.- Update value.
78 UPDATE transactions.credit_card cc
79 SET cc.iban = 'TR323456312213576817699999'
80  WHERE cc.id='CcU-2938' AND cc.iban='TR301950312213576817638661' ;
81
  
```

Result Grid

NEW
TR323456312213576817699999

Administration Schemas

Schema: transactions

credit\_card 4 credit\_card 5 x

Output

Action Output

#	Time	Action	Message
1	13:50:24	USE TRANSACTIONS	0 row(s) affected
2	13:50:24	SELECT cc.iban AS OLD FROM credit_card cc WHERE cc.id='CcU-2938'	1 row(s) returned
3	13:50:24	UPDATE transactions.credit_card cc SET cc.iban = 'TR323456312213576817699999' WHERE cc.id='CcU-2938' AND cc.iban='TR301950312213576817638661' ;	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
4	13:50:24	SELECT cc.iban AS NEW FROM credit_card cc WHERE cc.id='CcU-2938'	1 row(s) returned

### Nivell 1 - Exercici 3

En la taula "transaction" ingressa un nou usuari 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
```

- 1.- Primer comprovem si el camps que hem d'insertar són claus foranes de les altres taules. En aquest cas , crèdit\_card\_id i company\_id han d'existir a les seves respectives taules, sinó no podrem inserta les dades.
- 2.- Si les taules relacionades tenen les dades ja podrem inserta els valors a la taula transaccions.

**Schemas**

- biblioteca
- classicmodels
- dorita
  - Tables
  - Views
  - Stored Procedures
  - Functions
- hospitalites
- marketplace
- sakila
- sys
- transactions
  - Tables could not be fetched
    - company
    - credit\_card
    - transaction
  - Views could not be fetched
  - Stored Procedures could not be fetched
  - Functions could not be fetched
- world

**Administration Schemas Information:**

**Table: credit\_card**

Columns:	
id	varchar(255) PK
iban	varchar(255)
pan	varchar(255)
pin	int
cvv	int
expiring_date	date

```

-- SQL File 5: credit_card

FROM company com
WHERE ID='b-9999';

INSERT INTO company -- make sure the company exists company id is a foreign key.
(id ,company_name,phone,email,country,website)
VALUES ('b-9999','name_unknown','phone_unknown','email_unknown','country_unknown','website_unknown');

SELECT id -- ID MUST EXIST IN CREDIT CARD
FROM credit_card Car
WHERE id = 'Ccu-9999';

INSERT INTO credit_card
(id ,iban , pan , pin , cvv ,expiring_date )
VALUES ('CcU-9999',9999999999,9999999999999999,NULL,NULL,'2025-10-08');

INSERT INTO transaction -- Insert data. Timestamp unknown so we insert it as default value, null.
(id ,credit_card_id,company_id, user_id,lat,longitude,timestamp,amount,declined)
VALUES ('188B1D1D-5B23-A76C-55EF-C56BE4A99D0','CcU-9999','b-9999',9999,829.999,-117.999,NULL,111.11,0);

SELECT * -- CHECK RESULT
FROM credit_card Car
WHERE id = 'CcU-9999';
    
```

**Result Grid**

id	iban	pan	pin	cvv	expiring_date
CcU-9999	999999999999	9999999999999999			2025-10-08

credit\_card 13 x company 14

**Action Output**

#	Time	Action	Message
1	10:39:51	INSERT INTO credit_card (id ,iban , pan , pin , cvv ,expiring_date ) VALUES (CcU-9999',9999999999,9999999999999999,NULL,NULL,'2025-10-	1 row(s) affected
2	10:40:05	INSERT INTO transaction - Insert data. Timestamp unknown so we insert it as default value, null. (id ,credit_card_id,company_id,user_id,lat,longitu...	1 row(s) affected
3	10:40:15	SELECT * --CHECK RESULT FROM credit_card Car WHERE id = 'CcU-9999'	1 row(s) returned
4	10:40:15	SELECT * --CHECK RESULT FROM company.com WHERE ID='b-9999'	1 row(s) returned

## Nivell 1 Exercici 4

Des de recursos humans et solliciten eliminar la columna "pan" de la taula credit\_card. Recorda mostrar el canvi realitzat.

The screenshot shows the SQL Developer interface. The left pane displays the 'SCHEMAS' tree with 'transactions' expanded. The main pane shows a query window with the following SQL script:

```

185 cc.pan AS OLD
186 FROM credit_card cc;
187
188 USE TRANSACTIONS;
189 -- Step 2 .- Delete Column
190 ALTER TABLE credit_card
191 DROP COLUMN pan;
192
193 -- Step 3 .- Check it works.
194 SELECT
195 cc.pan AS NEW
196 FROM credit_card cc;
197
198 /* =====Nivell 2=====
199 Exercici 1
200 Elimina de la taula transaction el registre amb ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD de la base de dades.
201 Exercici 2
202 La secció de màrqueting desitja tenir accés a informació específica per a realitzar anàlisi i estratègies efectives.
203 S'ha sol·licitat crear una vista que proporcioni detalls clau sobre les companyies i les seves transaccions.
204 Serà necessària que creis una vista anomenada VistaMarketing que contingui la següent informació:
  
```

The right pane shows the 'Output' window with the following results:

#	Time	Action	Message
1	14:03:32	USE TRANSACTIONS	0 row(s) affected
2	14:03:32	SELECT cc.pan AS OLD FROM credit_card cc	5000 row(s) returned
3	14:03:46	USE TRANSACTIONS	0 row(s) affected
4	14:03:46	ALTER TABLE credit_card DROP COLUMN pan	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
5	14:03:46	SELECT cc.pan AS NEW FROM credit_card cc	Error Code: 1054. Unknown column 'cc.pan' in 'field list'

## Nivell 2 - Exercici 1

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

The screenshot shows the SQL Developer interface. The left pane displays the 'SCHEMAS' tree with 'transactions' expanded. The main pane shows a query window with the following SQL script:

```

123 S'ha sol·licitat crear una vista que proporcioni detalls clau sobre les companyies i les seves transaccions.
124 Serà necessària que creis una vista anomenada VistaMarketing que contingui la següent informació:
125 Nom de la companyia. Telèfon de contacte. País de residència. Mitjana de compra realitzat per cada companyia. Presenta la vista cr
126 Exercici 3
127 Filtra la vista VistaMarketing per a mostrar només les companyies que tenen el seu país de residència en "Germany" */
128
129
130 USE TRANSACTIONS;
131
132 -- Nivell 2 Exercici 1
133 -- Elimina de la taula transaction el registre amb ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD de la base de dades.
134
135
136 -- SELECT
137 -- *
138 DELETE
139 FROM transaction tt
140 WHERE ID='000447FE-B650-4DCF-85DE-C7ED0EE1CAAD';
141
142 -- Nivell 2 Exercici 2 Serà necessària que creis una vista anomenada VistaMarketing que contingui la següent informació:
  
```

The right pane shows the 'Output' window with the following results:

#	Time	Action	Message
1	14:04:39	USE TRANSACTIONS	0 row(s) affected
2	14:04:39	DELETE FROM transaction tt WHERE ID='000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'	1 row(s) affected

## Nivell 2 - 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.

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'SCHEMAS' pane shows the 'transactions' schema selected. The 'Query 1' pane contains the following SQL code:

```

141
142 -- Nivell 2 Exercici 2 Serà necessària que creïs una vista anomenada VistaMarketing que contingui la següent informació:
143 -- Nom de la companyia. Telèfon de contacte. País de residència. Mitjana de compra realitzat per cada companyia. Presenta la v:
144 USE TRANSACTIONS;
145
146 CREATE VIEW VistaMarketing AS
147 SELECT
148     co.company_name AS Companyia,
149     co.phone AS Telèfon,
150     co.country AS País,
151     ROUND(AVG(tt.amount), 2) AS PromigCompra
152 FROM company co
153     INNER JOIN transaction tt
154     ON co.id=tt.company_id
155 GROUP BY
156     tt.company_id
157 ORDER BY
158     PromigCompra DESC;
159
160 -- Nivell 2 Exercici 3. Filtra la vista VistaMarketing per a mostrar només les companyies que tenen el seu país de residència
  
```

The 'Output' pane shows the execution results:

#	Time	Action	Message
1	14:05:14	USE TRANSACTIONS	0 row(s) affected
2	14:05:14	CREATE VIEW VistaMarketing AS SELECT co.company_name AS Companyia, co.phone AS Telèfon, co.country AS País, ROUND(AVG(...	0 row(s) affected

## Nivell 2 - Exercici 3

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

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'SCHEMAS' pane shows the 'vistamarketing' view selected. The 'Query 1' pane contains the following SQL code:

```

163 CREATE OR REPLACE VIEW VistaMarketing AS
164 SELECT
165     co.company_name AS Companyia,
166     co.phone AS Telèfon,
167     co.country AS País,
168     ROUND(AVG(tt.amount), 2) AS PromigCompra
169 FROM company co
170     INNER JOIN transaction tt
171     ON co.id=tt.company_id
172 WHERE co.country="Germany"
173 GROUP BY
174     tt.company_id
175 ORDER BY
176     PromigCompra DESC;
177
178 /* =====NIVELL 3 =====
179 Exercici 1
180 La setmana vinent tindràs una nova reunió amb els gerents de màrqueting. Un company del teu equip va realitzar modificacions e
181 Et demana que l'ajudis a deixar els comandos executats per a obtenir el següent diagrama:
182
183 Exercici 2
184 L'empresa també us demana crear una vista anomenada "InformeTecnico" que contingui la següent informació:
185
186 ID de la transacció
187 Nom de l'interactuant
  
```

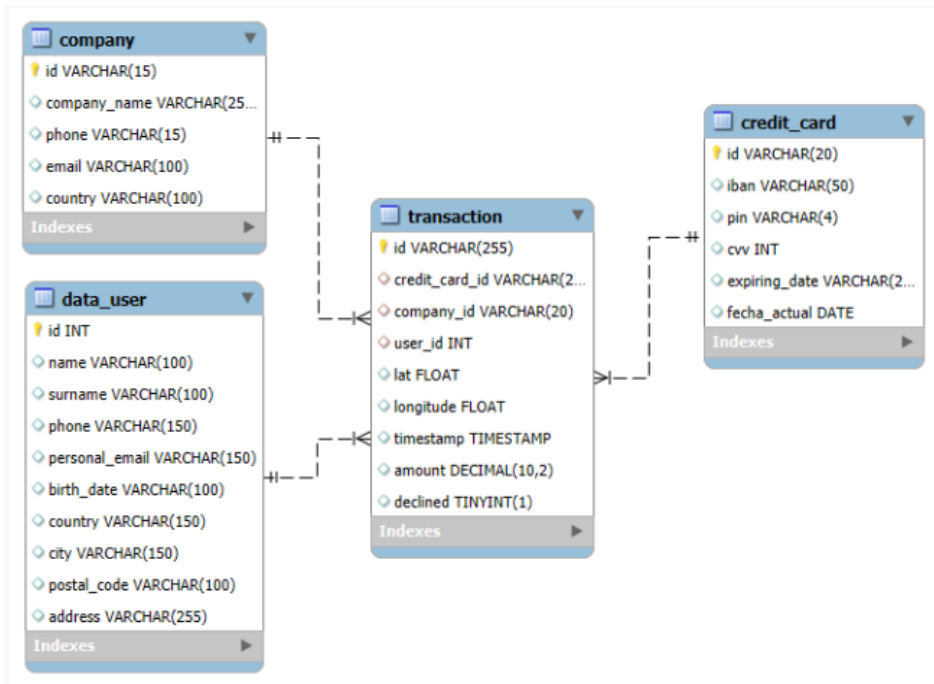
The 'Output' pane shows the execution results:

#	Time	Action	Message
1	14:13:44	USE TRANSACTIONS	0 row(s) affected
2	14:13:44	CREATE OR REPLACE VIEW VistaMarketing AS SELECT co.company_name AS Companyia, co.phone AS Telèfon, co.country AS País,...	0 row(s) affected



## Nivell 3 – Exercici 1

La setmana vinent tindrà 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:



1.- Creació d'una nova taula anomenada user.

2.- Insertar dades a la nova taula utilitzant l'script donat a l'exercici.

Navigator

SCHEMAS

Filter objects

biblioteca

classicmodels

donita

Tables

Views

Stored Procedures

Functions

hospitales

marketplace

sakila

sys

transactions

Tables could not be fetched

company

credit\_card

transaction

Views could not be fetched

Stored Procedures could not be fetched

Functions could not be fetched

world

Administration

Schemas

Information

Schema: transactions

Resum sql

ITA\_Sprint 03-AnaCPichot

ITA\_Sprint 03-AnaCPichot\_v1\*

SQL File 5\*

credit\_card

datos introducir sprint3 user

225 DROP TABLE user;

226 CREATE TABLE IF NOT EXISTS user (

227 id INT PRIMARY KEY,

228 name VARCHAR(100),

229 surname VARCHAR(100),

230 phone VARCHAR(150),

231 email VARCHAR(150),

232 birth\_date VARCHAR(100),

233 country VARCHAR(150),

234 city VARCHAR(150),

235 postal\_code VARCHAR(100),

236 address VARCHAR(255)

237 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

238

239 -- INSERT VALUES FROM FILE

Output

Action Output

#	Time	Action	Message
4982	11:07:29	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (	"4892", "Eeatkh", "Ayyimaoj... 1 row(s) affected
4983	11:07:29	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (	"4896", "Tufedm", "Jownbul... 1 row(s) affected
4984	11:07:29	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (	"4899", "Vpxdvdy", "Wrbxrofs... 1 row(s) affected
4985	11:07:29	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (	"4901", "Wevsom", "Xnrzm... 1 row(s) affected
4986	11:07:29	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (	"4906", "Qhsaiv", "Kylbjewd... 1 row(s) affected
4987	11:07:29	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (	"4914", "Bysfr", "Ihzumypa"... 1 row(s) affected

### 3.- Insertar l'usuari 9999 perquè serà clau forana.

The screenshot shows a SQL IDE interface with a script editor on the right and a schema browser on the left. The script editor contains SQL commands to insert a new user (id 9999) into the 'user' table. The output window shows the execution results, including several error messages related to foreign key constraints and ambiguous column references.

```

239 -- INSERT VALUES FROM FILE
240
241 -- AFEGERIR USUARI 99999 A LA TAULA USERS JA QUE SERÀ FOREIGN KEY
242 *
243 FROM user
244 where id = '9999';
245
246 -- INSERT USER
247 *
248 INSERT INTO user
249 (id, name, surname, phone, email, birth_date, country, city, postal_code, address)
250 VALUES (9999,NULL, NULL,NULL, NULL, NULL, NULL, NULL, NULL, NULL);
  
```

**Output:**

#	Time	Action	Message
1	11:19:02	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_user FOREIGN KEY (user_id) REFERENCES user(id) ON UPDATE CASCADE	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails
2	11:21:11	SELECT * FROM user INNER JOIN transaction tra ON user_id = tra.user_id WHERE user_id NOT IN ( SELECT* FROM transaction tra )	Error Code: 1054. Unknown column 'user_id' in 'IN/ALL/ANY subquery'
3	11:21:25	SELECT * FROM user us	5001 row(s) returned
4	11:21:56	SELECT * FROM user usu INNER JOIN transaction tra ON usu.id = tra.user_id WHERE user_id NOT IN ( SELECT* FROM transaction tra )	Error Code: 1241. Operand should contain 1 column(s)
5	11:22:10	SELECT id FROM user usu INNER JOIN transaction tra ON usu.id = tra.user_id WHERE user_id NOT IN ( SELECT* FROM transaction tra )	Error Code: 1052. Column 'id' in field list is ambiguous
6	11:22:20	SELECT user_id FROM user usu INNER JOIN transaction tra ON usu.id = tra.user_id WHERE user_id NOT IN ( SELECT* FROM transaction tra )	Error Code: 1241. Operand should contain 1 column(s)
7	11:22:36	SELECT user_id FROM user usu INNER JOIN transaction tra ON usu.id = tra.user_id	99999 row(s) returned
8	11:22:51	SELECT user_id FROM user usu INNER JOIN transaction tra ON usu.id = tra.user_id ORDER BY 1 DESC	99999 row(s) returned
9	11:23:08	SELECT DISTINCT user_id FROM user usu INNER JOIN transaction tra ON usu.id = tra.user_id ORDER BY 1 DESC	5000 row(s) returned
10	11:23:24	SELECT * FROM user where id = '99999'	1 row(s) returned
11	11:23:39	SELECT DISTINCT user_id FROM user usu INNER JOIN transaction tra ON usu.id = tra.user_id ORDER BY 1 DESC	5000 row(s) returned
12	11:24:54	SELECT * FROM user where id = '9999'	0 row(s) returned
13	11:25:25	SELECT * FROM user where id = '99999'	1 row(s) returned
14	11:25:31	DELETE FROM user where id = '99999'	1 row(s) affected
15	11:25:43	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (9999,NULL, NULL,NULL, NULL, NULL, NULL, NULL, NULL, NULL)	1 row(s) affected

### 4.- Connectar aquesta nova taula user amb la base de dades. Utilitzem la clau forana id i la connectem a la taula de fets transaction.

The screenshot shows a SQL IDE interface with a script editor on the right and a schema browser on the left. The script editor contains SQL commands to create a foreign key constraint between the 'transaction' table and the 'user' table. The output window shows the execution results, including a message indicating that 100,000 rows were affected.

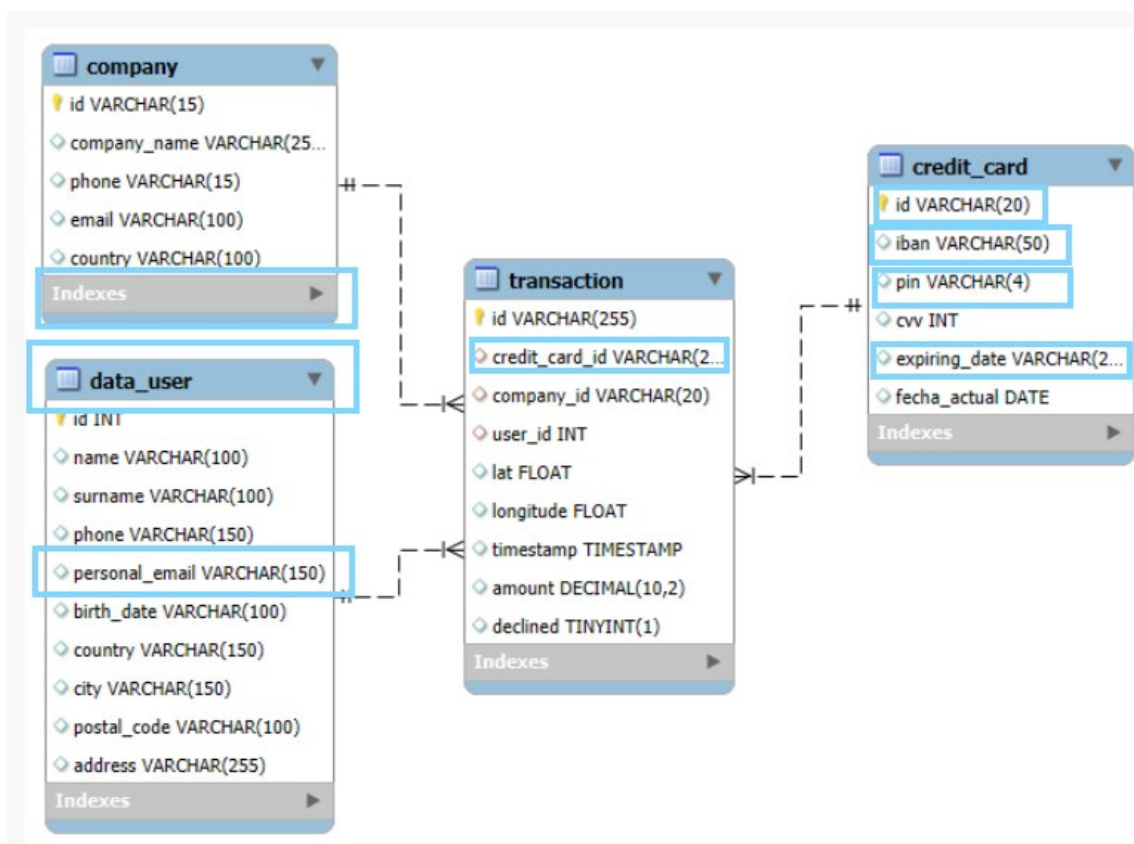
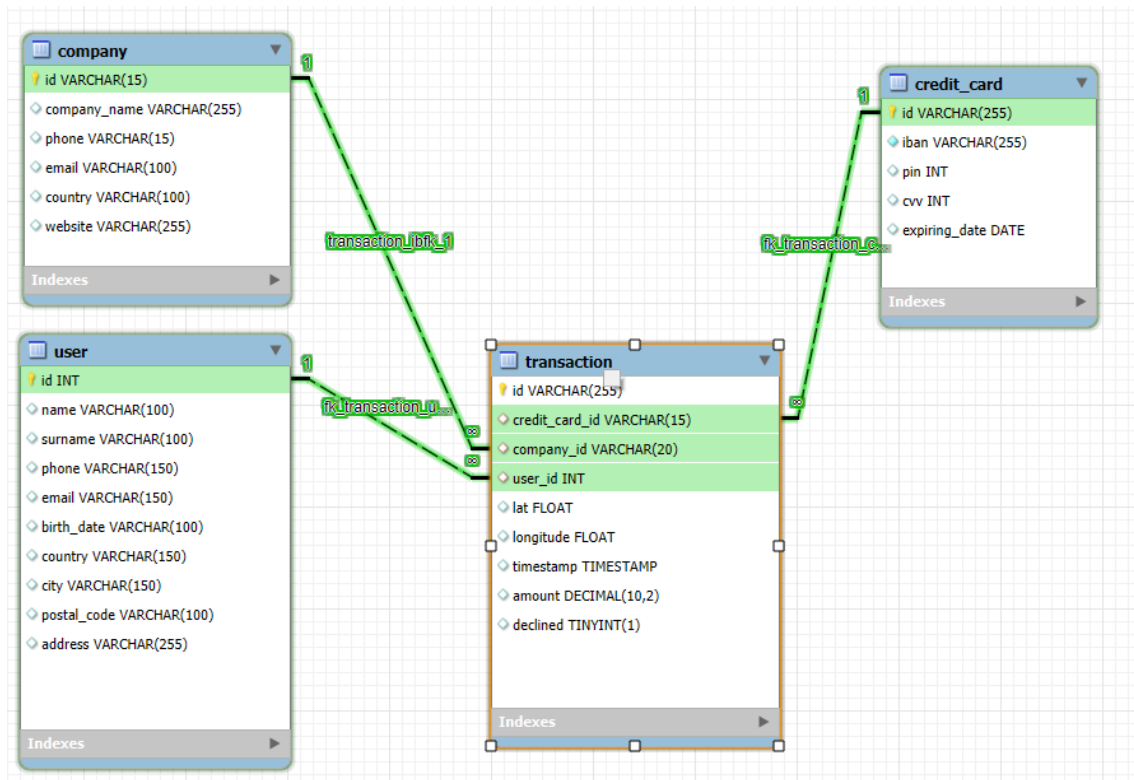
```

248 (id, name, surname, phone, email, birth_date, country, city, postal_code, address)
249 VALUES (9999,NULL, NULL,NULL, NULL, NULL, NULL, NULL, NULL, NULL);
250
251
252 -- CREAM clau foranea de credit_card en la taula transaction .
253 *
254 ALTER TABLE transaction
255 ADD CONSTRAINT fk_transaction_user
256 FOREIGN KEY (user_id)
257 REFERENCES user(id)
258 ON UPDATE CASCADE;
259
260 -- Modificacions realitzades en la taula user
261 -- Canviar el nom de la taula
262 *
263 ALTER TABLE transactions.user
264 RENAME TO transactions.data_user;
265
266 -- 1.- Modificar el datatype de les columnes.
267
268 -- renombra columna EMAIL
269 *
270 ALTER TABLE transactions.data_user
  
```

**Output:**

#	Time	Action	Message
1	11:27:58	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_user FOREIGN KEY (user_id) REFERENCES user(id) ON UPDATE CASCADE	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings: 0

## 5.- Revisem com tenim el disseny de la base de dades.



## 5.- Realitzar les modificacions a les taules necessàries per deixar-les com ens demanen.

Navigator: Schemas, Filter objects, Tables, Views, Stored Procedures, Functions, hospitales, marketplace, sakila, sys, transactions. Tables could not be fetched: company, credit\_card, transaction. Columns: id, credit\_card\_id, company\_id, user\_id, lat, longitude, timestamp, amount.

```

257 ON UPDATE CASCADE;
258
259 -- Et demana que l'ajudis a deixar els comandos executats per a obtenir el següent diagrama:
260
261 -- RENAME A TABLE from user to data_user
262 ALTER TABLE user
263 RENAME TO data_user;
264
265 -- RENAME A COLUMN from email to personal_email
266 ALTER TABLE data_user
267 RENAME COLUMN email TO personal_email;
268
269 -- DELETE A COLUMN
270 ALTER TABLE company
271 DROP COLUMN website;
272
273 -- Add a column
274 ALTER TABLE credit_card
275 ADD fecha_actual DATE DEFAULT(CURRENT_DATE);
276
277 -- CHANGE DATA TYPES
278 ALTER TABLE transaction
279 MODIFY credit_card_id VARCHAR(20)
280 CHARACTER SET utf8mb4
281 COLLATE utf8mb4_general_ci;
282
283 ALTER TABLE credit_card
284 MODIFY id VARCHAR(20)
285 CHARACTER SET utf8mb4
286 COLLATE utf8mb4_general_ci;
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
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Output: Action Output

#	Time	Action	Message
1	12:15:03	ALTER TABLE user RENAME TO data_user	0 row(s) affected
2	12:15:03	ALTER TABLE data_user RENAME COLUMN email TO personal_email	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
3	12:15:03	ALTER TABLE company DROP COLUMN website	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
4	12:15:03	ALTER TABLE credit_card ADD fecha_actual DATE DEFAULT(CURRENT_DATE)	5001 row(s) affected Records: 5001 Duplicates: 0 Warnings: 0

```

275 ADD fecha_actual DATE DEFAULT(CURRENT_DATE);
276
277 -- CHANGE DATA TYPES
278 -- we want to change a data type of a foreign key so we need to delete it before.
279 ALTER TABLE transaction
280 DROP FOREIGN KEY fk_transaction_card;
281
282 ALTER TABLE transaction
283 MODIFY credit_card_id VARCHAR(20)
284 CHARACTER SET utf8mb4
285 COLLATE utf8mb4_general_ci;
286
287 ALTER TABLE credit_card
288 MODIFY id VARCHAR(20)
289 CHARACTER SET utf8mb4
290 COLLATE utf8mb4_general_ci;
291
292 -- WE CREATE OUR FOREIGN KEY AGAIN
293 ALTER TABLE transaction
294 ADD CONSTRAINT fk_transaction_card
295 FOREIGN KEY (credit_card_id)
296 REFERENCES credit_card(id)
297 ON UPDATE CASCADE;
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Navigator: Schemas, Filter objects, Tables, Views, Stored Procedures, Functions, hospitales, marketplace, sakila, sys, transactions. Tables could not be fetched: company, credit\_card, transaction. Columns: id, credit\_card\_id, company\_id, user\_id, lat, longitude, timestamp, amount.

Column: credit\_card\_id  
Collation: utf8mb4\_0900\_ai\_ci  
Definition: credit\_card\_id varchar(15)

Output: Action Output

#	Time	Action	Message
1	12:22:39	ALTER TABLE transaction DROP FOREIGN KEY fk_transaction_card	0
2	12:22:44	ALTER TABLE transaction MODIFY credit_card_id VARCHAR(20) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci	1
3	12:22:50	ALTER TABLE credit_card MODIFY id VARCHAR(20) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci	5
4	12:22:57	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_card FOREIGN KEY (credit_card_id) REFERENCES credit_card(id) ON UPDATE CASCADE	1

Navigator

Filter objects

- Tables
- Views
- Stored Procedures
- Functions
- hospitales
- marketplace
- sakila
- sys
- transactions
  - Tables could not be fetched
  - company
  - credit\_card
  - transaction
  - Columns
    - id
    - credit\_card\_id
    - company\_id
    - user\_id
    - lat
    - longitude
    - timestamp
    - amount

Administration Schemas

Information

Column: **credit\_card\_id**

Collation: utf8mb4\_0900\_ai\_ci

Definition: **credit\_card\_id** varchar(15)

```

293 • ALTER TABLE transaction
294   ADD CONSTRAINT fk_transaction_card
295   FOREIGN KEY (credit_card_id)
296   REFERENCES credit_card(id)
297   ON UPDATE CASCADE;
298
299 • ALTER TABLE credit_card
300   MODIFY iban VARCHAR(50)
301   CHARACTER SET utf8mb4
302   COLLATE utf8mb4_general_ci;
303
304 • ALTER TABLE credit_card
305   MODIFY pin VARCHAR(4)
306   CHARACTER SET utf8mb4
307   COLLATE utf8mb4_general_ci;
308
309 • ALTER TABLE credit_card
310   MODIFY expiring_date VARCHAR(20)
311   CHARACTER SET utf8mb4
312   COLLATE utf8mb4_general_ci;
313
314 -- Exercici 2
315 -- L'empresa també us demana crear una vista anomenada "InformeTecnico" que contingui la següent
316

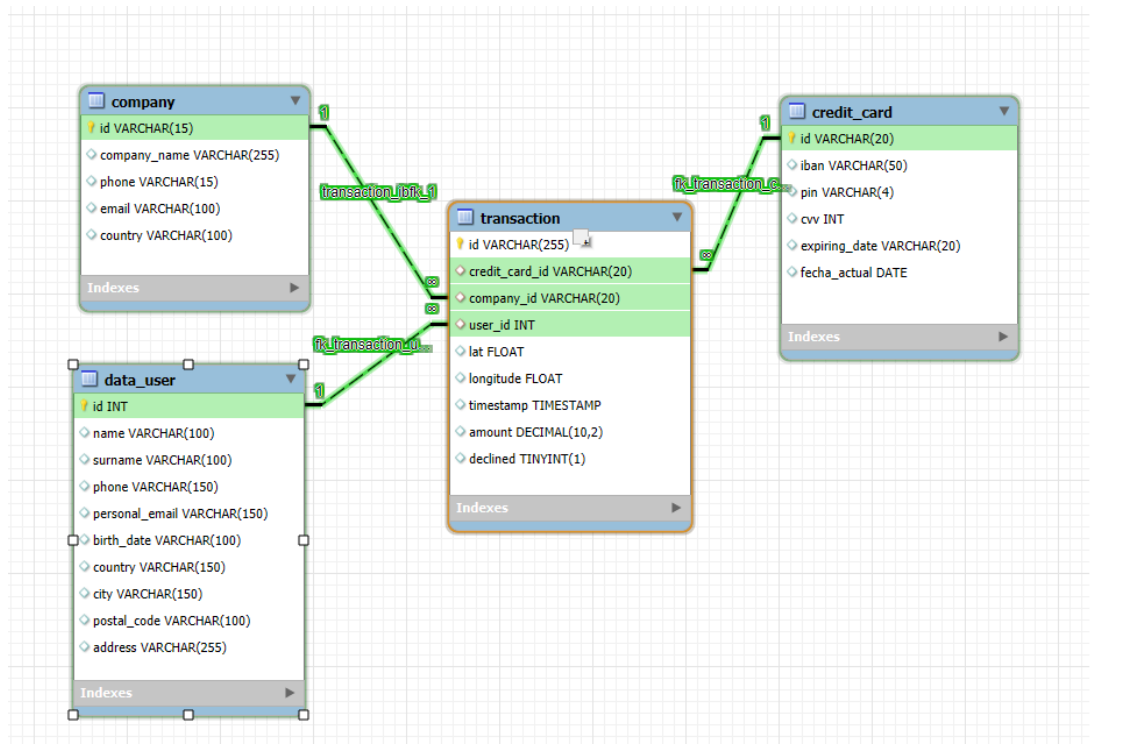
```

Output

Action Output

#	Time	Action
1	12:23:45	ALTER TABLE credit_card MODIFY iban VARCHAR(50) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci
2	12:23:48	ALTER TABLE credit_card MODIFY pin VARCHAR(4) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci
3	12:23:55	ALTER TABLE credit_card MODIFY expiring_date VARCHAR(20) CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci

6.- El resultat final és el següent model.



## Nivell 3 – Exercici 2

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

1. ID de la transacció
2. Nom de l'usuari/ària
3. Cognom de l'usuari/ària
4. IBAN de la targeta de crèdit usada.
5. 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.

The screenshot shows a SQL IDE interface with a Navigator on the left, a Query Editor in the center, and a Results/Output pane at the bottom.

**Navigator:** The 'transactions' schema is expanded, showing tables like 'company', 'credit\_card', 'data\_user', and 'transaction'. The 'Views' section shows the 'InformeTecnico' view.

**Query 1:** The SQL query is as follows:

```

262 uu.name AS Nom_Usuari,
263 uu.surname AS Cognom_Usuari,
264 cc.iban AS Num_Compte,
265 co.company_name AS Nom_Companyia
266 FROM transactions.transaction tt -- des de transactions
267 INNER JOIN transactions.data_user uu -- busco usuarios
268 ON tt.user_id=uu.id
269 INNER JOIN transactions.company co -- busco companyies
270 ON tt.company_id=co.id
271 INNER JOIN transactions.credit_card cc -- busco iban
272 ON tt.credit_card_id=cc.id;
273
274
275 SELECT
276 *
277 FROM InformeTecnico;

```

**Result Grid:** The results are displayed in a table with the following columns: ID\_Transacció, Nom\_Usuari, Cognom\_Usuari, Num\_Compte, and Nom\_Companyia.

ID_Transacció	Nom_Usuari	Cognom_Usuari	Num_Compte	Nom_Companyia
034EB351-669A-4B50-8F15-EA70B6DE1748	Jndiens	Rcfor	XX7826930491423553609370	Rutrum Non Inc.
033575C9-0E04-453E-905B-D8F4CCA18E4E	Jndiens	Rcfor	XX7826930491423553609370	Mus Aenean Eget Foundation
02DF083C-E622-42E0-675E-15BA39A27594	Jndiens	Rcfor	XX7826930491423553609370	Ac Fermentum Incorporated
025D3738-86D4-4F17-8F86-5F18659A6E24	Siga	Chvgapbuv	XX5559590368835304645299	Donec Fringilla PC
0488F7E9-5F5B-41AA-9FC3-77BE08439F9C	Naelh	Smqjyl	XX2035182877195191627307	Non Ante LLP

**Output:** The output shows the execution of the query, including the creation of the view and the selection of data.

#	Time	Action	Message
1	14:52:57	USE TRANSACTIONS	0 row(s) affected
2	14:52:57	CREATE VIEW InformeTecnico AS SELECT tt.id AS ID_Transacció, uu.name AS Nom_Usuari, uu.surname AS Cognom_Usuari, ...	0 row(s) affected
3	14:52:57	SELECT * FROM InformeTecnico	99999 row(s) returned