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.

Mètode 1

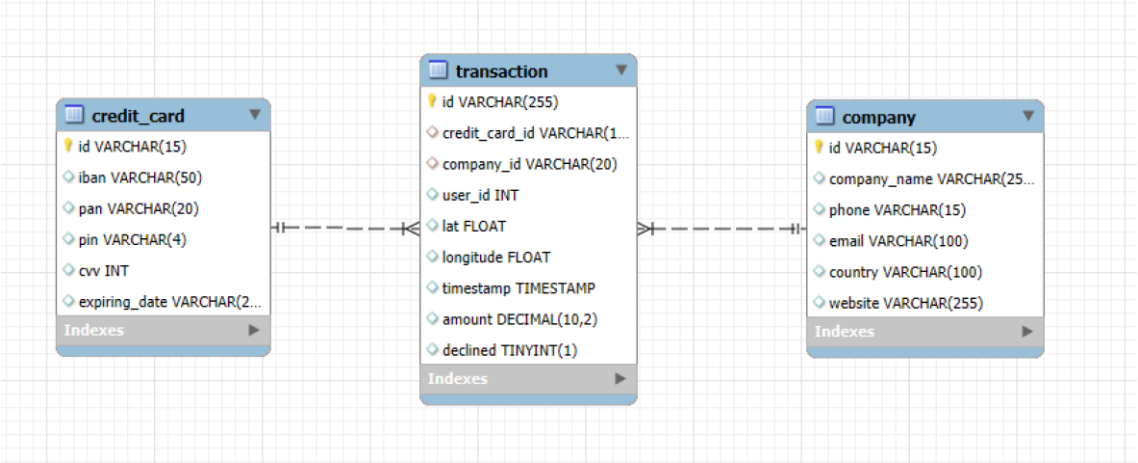
```
-- Eliminamos la base de datos transactions
DROP DATABASE transactions;

-- Creamos la base de datos por orden: primero company, segundo credit_card y por último transaction
CREATE DATABASE IF NOT EXISTS transactions;
USE transactions;

-- Creamos la tabla company
CREATE TABLE IF NOT EXISTS company (
    id VARCHAR(15) PRIMARY KEY,
    company_name VARCHAR(255),
    phone VARCHAR(15),
    email VARCHAR(100),
    country VARCHAR(100),
    website VARCHAR(255)
);

-- Creamos la tabla credit card
-- ejemplo campos id:'CcU-2938', iban:'TR301950312213576817638661', pan:'5424465566813633', pin:'3257', cvv:'984', expiring_date:'10/30/22'
CREATE TABLE IF NOT EXISTS credit_card (
    id VARCHAR(15) PRIMARY KEY,
    iban VARCHAR(50),
    pan VARCHAR(20),
    pin VARCHAR(4),
    cvv INT,
    expiring_date VARCHAR(20)
);

    -- Creamos la tabla transaction
CREATE TABLE IF NOT EXISTS transaction (
    id VARCHAR(255) PRIMARY KEY,
    credit_card_id VARCHAR(15),
    company_id VARCHAR(20),
    user_id INT REFERENCES user(id),
    lat FLOAT,
    longitude FLOAT,
    timestamp TIMESTAMP,
    amount DECIMAL(10, 2),
    declined BOOLEAN,
    FOREIGN KEY (company_id) REFERENCES company(id),
    FOREIGN KEY (credit_card_id) REFERENCES credit_card(id)
);
```

```
Output
Action Output
#  Time      Action                                                                                                  Message
7  10:03:23  DROP DATABASE `transactions`                                                                           4 row(s) affected
8  10:06:27  CREATE DATABASE IF NOT EXISTS transactions                                                            1 row(s) affected
9  10:06:27  USE transactions                                                                                      0 row(s) affected
10 10:06:27  CREATE TABLE IF NOT EXISTS company (  id VARCHAR(15) PRIMARY KEY,  comp...                            0 row(s) affected
11 10:06:27  CREATE TABLE IF NOT EXISTS credit_card (  id VARCHAR(15) PRIMARY KEY,  iba...                         0 row(s) affected
12 10:06:27  CREATE TABLE IF NOT EXISTS transaction (  id VARCHAR(255) PRIMARY KEY,  cr...                         0 row(s) affected
```

Primer de tot, fem un esborrat de la taula de dades sencera amb drop database.

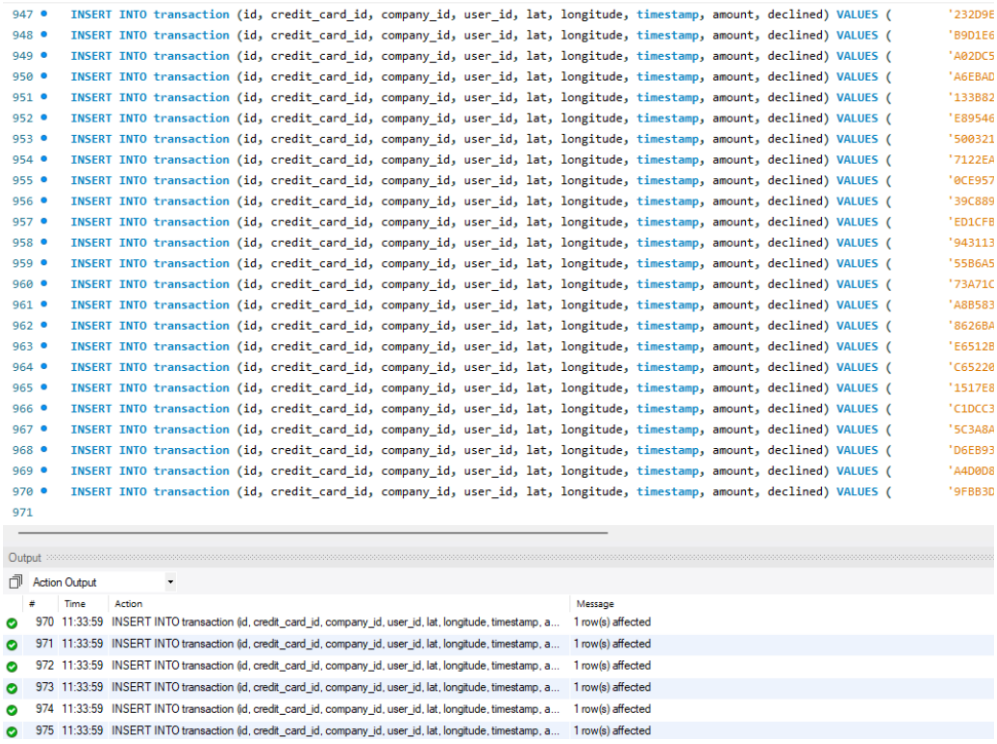
Després fem de nou la creació de la taula company amb la mateixa query del sprint 2, posteriorment creem la taula credit_card, on 'id' serà la PRIMARY KEY i amb els atributs 'iban', 'pan', 'pin', 'cvv' i 'expiring_date'. Els tipus d'aquestes dades les agafem del diagrama que es mostra al exercici 1 del nivell 3. Amb l'excepció de 'pan' que escollim que sigui VARCHAR(20).

Per últim creem la taula transaction modificant només una línia que fem que credit_card_id sigui una FOREIGN KEY i la referenciem a la taula credit_card(id), que hem creat just abans com a PRIMARY.



Aquí es mostra el diagrama un cop creades les taules i veiem els diferents atributs, tipus, multiplicitat i tipus de relació. En aquest cas, tant credit_card com company tenen una relació 1:N amb transaction. És a dir, una transaction només té una única company i crèdit card, però una mateixa companyia o crèdit card pot estar a múltiples transaction.

Després procedim a fer els insert into dels registres del sprint 2, primer omplim la taula company (sprint 2), després crèdit card (sprint 3) i per últim transaction (sprint 2).



MÈTODE 2:

Seguim amb la database del sprint2, fem la creació de la taula credit_card:

```sql
CREATE TABLE IF NOT EXISTS credit_card (
        id VARCHAR(15) PRIMARY KEY,
        iban VARCHAR(50),
        pan VARCHAR(20),
        pin VARCHAR(4),
        cvv INT,
        expiring_date VARCHAR(20)
    );
```

Output

Action Output

#	Time	Action	Message
1	13:03:32	CREATE TABLE IF NOT EXISTS credit_card (id VARCHAR(15) PRIMARY KEY, iban ...	0 row(s) affected

Ara omplim amb els INSERT INTO de dades_introduir_credit.

```sql
-- Insertamos datos de credit_card
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-2938', 'TR3019503122135768176386 61', '5424465566813633', '3257', '9
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-2945', 'DO26854763748537475216568689', '5142423821948828', '9080',
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-2952', 'BG45IVQL52710525608255', '4556 453 55 5287', '4598', '438',
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-2959', 'CR72424772443358415 35', '372461377349375', '3583', '667', '
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-2966', 'BG72LKTQ15627628377363', '448566 886747 7265', '4900', '130
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-2973', 'PT878062281350924294563 46', '544 58654 54343 384', '8760',
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-2980', 'DE392418818830862771 36', '402400 7145845969', '5075', '596'
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-2987', 'GE896814348377487818 13', '3763 747687 76666', '2298', '797'
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-2994', 'BH627144283680667652 94', '344283273252593', '7545', '595',
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3001', 'CY490874266547745812668321 10', '511722 924833 2244', '9562'
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3008', 'LU507216693616119230', '4485744464433884', '1856', '740', '
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3015', 'PS119398216295715968342456821', '3784 662233 17389', '3246'
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3022', 'GT916951628505569774231218 57', '5164 1379 4842 3951', '5610
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3029', 'AZ623174139824414181237397 46', '3429 279566 77631', '9708',
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3036', 'AZ393360029258428658439419 94', '3768 451556 48766', '2232',
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3043', 'TN648814331051485217953 5', '455676 6437463635', '5969', '19
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3050', 'FR516774436917583683185 4477', '4024007123722', '4834', '126
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3057', 'LU931822574697545215', '3484 621767 21237', '6805', '848',
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3064', 'PS146965545449253377627273133', '3467 732741 26810', '3865'
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3071', 'NO8923814763512', '3464 789562 23352', '6625', '661', '12/2
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ( 'CcU-3078', 'IS025127145884623279548733', '4539 322 74 2377', '9405', '7
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
267	13:07:02	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4800', 'Sl...	1 row(s) affected	0.000 sec
268	13:07:02	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4807', 'L...	1 row(s) affected	0.000 sec
269	13:07:02	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4814', 'M...	1 row(s) affected	0.000 sec
270	13:07:02	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4821', 'L...	1 row(s) affected	0.000 sec
271	13:07:03	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4828', 'B...	1 row(s) affected	0.000 sec
272	13:07:03	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4835', 'P...	1 row(s) affected	0.015 sec
273	13:07:03	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4842', 'S...	1 row(s) affected	0.016 sec
274	13:07:03	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4849', 'S...	1 row(s) affected	0.016 sec
275	13:07:03	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4856', 'T...	1 row(s) affected	0.000 sec

Fem la visualització de totes les entrades que hem afegit.

```
1 •  select * from credit_card;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	iban	pan	pin	cvv	expiring_date
CcU-2938	TR301950312213576817638661	5424465566813633	3257	984	10/30/22
CcU-2945	DO268547637485374752165686 89	5142423821948828	9080	887	08/24/23
CcU-2952	BG45IVQL52710525608255	4556 453 55 5287	4598	438	06/29/21
CcU-2959	CR7242477244335841535	372461377349375	3583	667	02/24/23
CcU-2966	BG72LKTQ15627628377363	448566 886747 7265	4900	130	10/29/24
CcU-2973	PT8780622813509242945 6346	544 58654 54343 384	8760	887	01/30/25
CcU-2980	DE39241881883086277136	402400 7145845969	5075	596	07/24/22
CcU-2987	GE89681434837748781813	3763 747687 76666	2298	797	10/31/23
CcU-2994	BH62714428368066765294	344283273252593	7545	595	02/28/22
CcU-3001	CY490874266547745812668 32110	511722 924833 2244	9562	867	09/16/22
CcU-3008	LU507216693616119230	4485744464433884	1856	740	04/05/25
CcU-3015	PS119398216295715968342456821	3784 662233 17389	3246	822	01/31/22
CcU-3022	GT916951628505569774231218 57	5164 1379 4842 3951	5610	342	04/25/25
CcU-3029	AZ623174139824414181237397 46	3429 279566 77631	9708	505	09/02/23
CcU-3036	AZ393360029258428658439419 94	3768 451556 48766	2232	565	10/27/25
CcU-3043	TN648814331051485217953 5	455676 6437463635	5969	196	06/07/25
CcU-3050	FR516774436917583683185447 7	4024007123722	4834	126	10/09/23
CcU-3057	LU931822574697545215	3484 621767 21237	6805	848	09/14/25
CcU-3064	PS146965545449253377627273 133	3467 732741 26810	3865	498	06/03/25
CcU-3071	NO8923814763512	3464 789562 23352	6625	661	12/20/23

credit_card 4

Output

Action Output

#	Time	Action	Message
1	13:24:08	select * from credit_card LIMIT 0, 1000	275 row(s) returned

Hem afegit 275 files.

Ara fem el ALTER TABLE de la taula transaction per tenir credit_card_id com a FOREIGN KEY.

Primer ens assegurem que no hagi cap credit_card_id de la taula transaction que no estigui a la taula crèdit_card.

```
1 •  SELECT DISTINCT credit_card_id
2    FROM transaction
3    WHERE credit_card_id NOT IN (SELECT id FROM credit_card);
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

credit_card_id

Output

Action Output

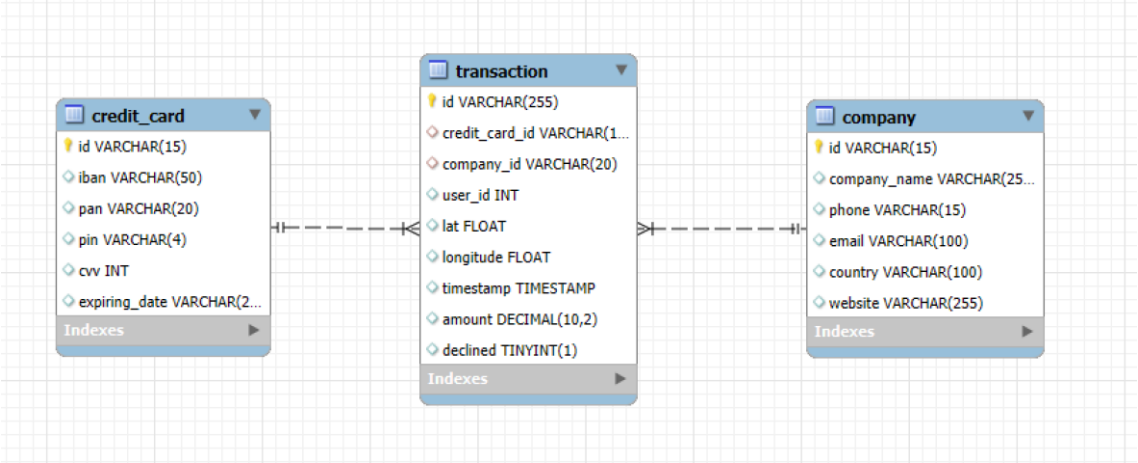
#	Time	Action	Message
1	13:09:26	SELECT DISTINCT credit_card_id FROM transaction WHERE credit_card_id NOT IN (SELEC...	0 row(s) returned

En efecte no n'hi ha cap. Ara fem el ALTER TABLE.

```
1 •  ALTER TABLE transaction
2    ADD CONSTRAINT fk_transaction_credit_card
3    FOREIGN KEY (credit_card_id) REFERENCES credit_card(id);
```

```
Output
Action Output
#  Time      Action                                                                                  Message
1  13:13:39  ALTER TABLE transaction  ADD CONSTRAINT fk_transaction_credit_card  FOREIGN KEY (cre...  587 row(s) affected Records: 587  Duplicates: 0  Warnings: 0
```

Per últim realitzem el diagrama.

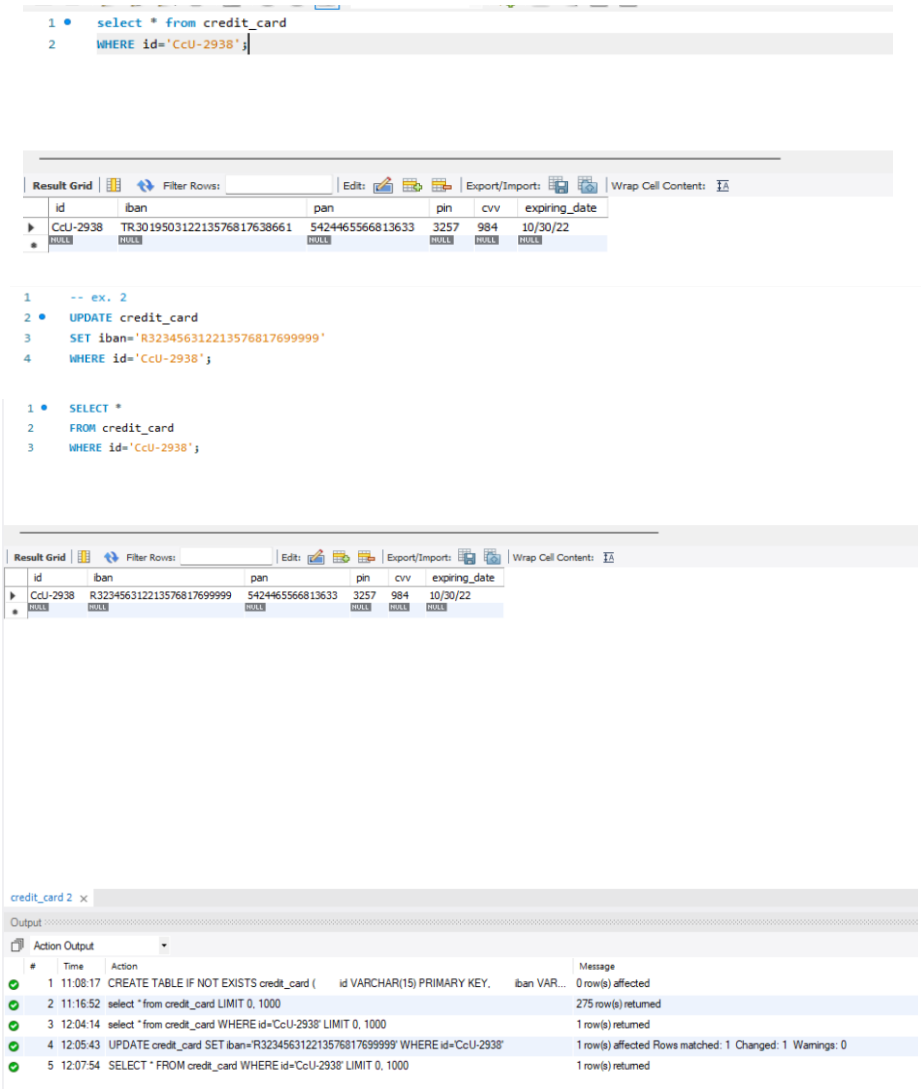


Ja està explicat en el mètode 1.

- Exercici 2

El departament de Recursos Humans ha identificat un error en el número de compte de l'usuari amb ID CcU-2938. La informació que ha de mostrar-se per a aquest registre és: R323456312213576817699999. Recorda mostrar que el canvi es va realitzar.

Ara mostrarem primer un select de totes les columnes de la taula crèdit_card amb l'id demanat ('CcU-2938') per veure el camp iban abans de la modificació. Després fem un Update de la taula crèdit_card on canviarem el camp iban amb SET al valor demanat per l'enunciat al registre on id coincideixi amb 'CcU-2938'. Per últim, mostrarem select de totes les columnes de la taula crèdit_card amb l'id demanat ('CcU-2938') per veure el camp iban després de la modificació.



Queda comprovat que s'ha fet el Update del camp IBAN.

- 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

```sql
INSERT INTO company (id) VALUES ('b-9999');
INSERT INTO credit_card(id) VALUES ('CcU-9999');
INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES (
'108B1D1D-5B23-A76C-55EF-C568E49A99DD',
'CcU-9999',
'b-9999',
'9999',
'829.999',
'-117.999',
'111.11',
'0');

SELECT *
FROM transaction
WHERE id='108B1D1D-5B23-A76C-55EF-C568E49A99DD';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

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

transaction 5

Output

Action Output

#	Time	Action	Message
1	13:31:57	INSERT INTO company (id) VALUES ('b-9999')	1 row(s) affected
2	13:31:57	INSERT INTO credit_card(id) VALUES ('CcU-9999')	1 row(s) affected
3	13:31:57	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, decli...	1 row(s) affected
4	13:31:57	SELECT * FROM transaction WHERE id='108B1D1D-5B23-A76C-55EF-C568E49A99DD' LIMIT...	1 row(s) returned

Hem fet la inserció primer de la companyia b-9999 i després de crèdit_card amb id CcU-9999, tots els demés camps NULL. Aquests s'han de fer primer ja que són FOREIGN KEYS de la taula TRANSACTION. Per últim fem el INSERT INTO de transaction amb les dades de l'enunciat i fem un SELECT d'aquest id per visualitzar les dades introduïdes.

- Exercici 4

Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_*card. Recorda mostrar el canvi realitzat.

```
325     -- ex. 4
326 •   ALTER TABLE credit_card
327     DROP COLUMN pan;
328
329 •   SELECT *
330     FROM credit_card;
331
332
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

id	iban	pin	cvv	expiring_date
CcU-2938	TR301950312213576817638661	3257	984	10/30/22
CcU-2945	DO268547637485374752165686 89	9080	887	08/24/23
CcU-2952	BG45IVQL52710525608255	4598	438	06/29/21
CcU-2959	CR7242477244335841535	3583	667	02/24/23
CcU-2966	BG72LKTQ15627628377363	4900	130	10/29/24
CcU-2973	PT87806228135092429456346	8760	887	01/30/25
CcU-2980	DE39241881883086277136	5075	596	07/24/22
CcU-2987	GE89681434837748781813	2298	797	10/31/23
CcU-2994	BH62714428368066765294	7545	595	02/28/22
CcU-3001	CY490874266547745812668321 10	9562	867	09/16/22
CcU-3008	LU507216693616119230	1856	740	04/05/25
CcU-3015	PS119398216295715968342456821	3246	822	01/31/22
CcU-3022	GT91695162850556977423121857	5610	342	04/25/25
CcU-3029	AZ623174139824414181237397 46	9708	505	09/02/23
CcU-3036	AZ393360029258428658439419 94	2232	565	10/27/25
CcU-3043	TN648814331051485217953 5	5969	196	06/07/25
CcU-3050	FR516774436917583683185447 7	4834	126	10/09/23

credit_card 1 ×

Output

Action Output

#	Time	Action	Message
1	13:36:38	ALTER TABLE credit_card DROP COLUMN pan	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
2	13:36:57	SELECT * FROM credit_card LIMIT 0, 1000	276 row(s) returned

Fem un ALTER TABLE per eliminar (DROP COLUMN) la columna 'pan'. Després visualitzem tota la taula i comprovem que ja no apareix aquesta columna.

Nivell 2

Exercici 1

Elimina de la taula transaction el registre amb ID 02C6201E-D90A-1859-B4EE-88D2986D3B02 de la base de dades.

Es fa l'eliminació del registre indicat amb la comanda DELETE i un WHERE per localitzar el registre que té aquell id. Com podem veure només afecta a 1 fila.

```
1 •   DELETE FROM transaction
2     WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02';
```

Output

Action Output

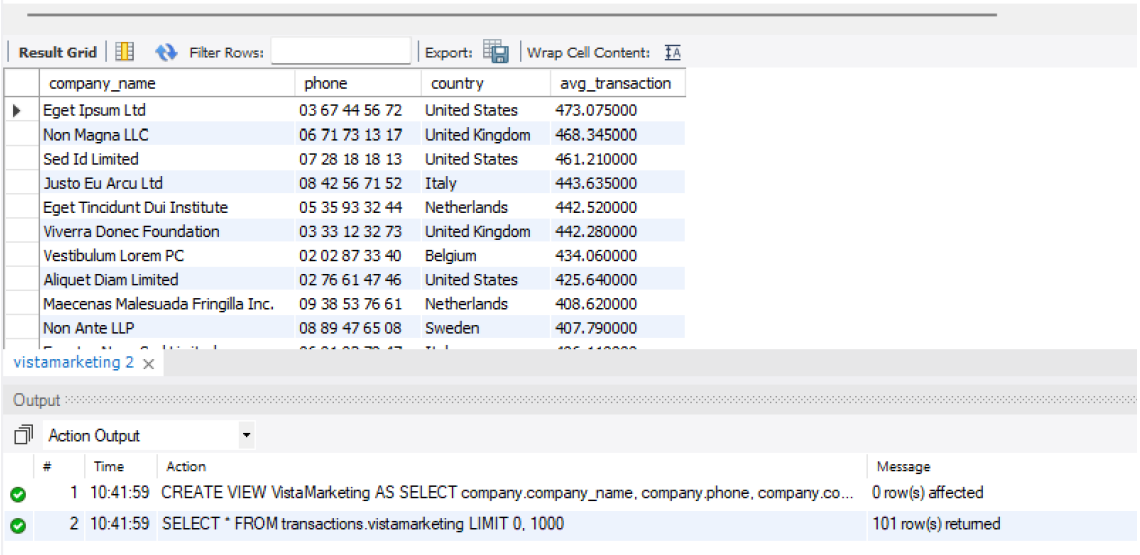
#	Time	Action	Message
1	10:31:30	DELETE FROM transaction WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02'	1 row(s) affected

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.

Es procedeix a crear la vista indicada a l'enunciat. Aquesta tindrà el nom de VistaMarketing i contindrà les columnes, nom de la companyia, telèfon, país i mitjana de compra per companyia tenint en compte tant transaccions declined com no declined. Haurem de combinar dades amb un JOIN de les taules transaction i company. Amb ORDER BY AVG(amount) DESC, obtindrem la vista ordenada de major a menor mitjana de compra que haurem agrupat per totes les columnes que agafem amb el SELECT sense tenir en compte el AVG. Després la visualitzarem amb la query SELECT * FROM VistaMarketing.

```sql
CREATE VIEW VistaMarketing AS
SELECT company.company_name, company.phone, company.country, AVG(transaction.amount) AS avg_transaction
FROM company
JOIN transaction ON transaction.company_id = company.id
GROUP BY company.company_name, company.phone, company.country
ORDER BY avg_transaction DESC;

SELECT * FROM transactions.vistamarketing;
```



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

company_name	phone	country	avg_transaction
Eget Ipsum Ltd	03 67 44 56 72	United States	473.075000
Non Magna LLC	06 71 73 13 17	United Kingdom	468.345000
Sed Id Limited	07 28 18 18 13	United States	461.210000
Justo Eu Arcu Ltd	08 42 56 71 52	Italy	443.635000
Eget Tincidunt Dui Institute	05 35 93 32 44	Netherlands	442.520000
Viverra Donec Foundation	03 33 12 32 73	United Kingdom	442.280000
Vestibulum Lorem PC	02 02 87 33 40	Belgium	434.060000
Aliquet Diam Limited	02 76 61 47 46	United States	425.640000
Maecenas Malesuada Fringilla Inc.	09 38 53 76 61	Netherlands	408.620000
Non Ante LLP	08 89 47 65 08	Sweden	407.790000

vistamarketing 2

Output

Action Output

#	Time	Action	Message
1	10:41:59	CREATE VIEW VistaMarketing AS SELECT company.company_name, company.phone, company.co...	0 row(s) affected
2	10:41:59	SELECT * FROM transactions.vistamarketing LIMIT 0, 1000	101 row(s) returned

Exercici 3

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

Fem un select de totes les columnes de la vista que acabem de crear (VistaMarketing) amb un filtre on només surt el país coincident amb Alemanya.

```
1 •  SELECT * FROM transactions.vistamarketing WHERE country='Germany';
2
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

company_name	phone	country	avg_transaction
Aliquam PC	01 45 73 52 16	Germany	385.265000
Ac Industries	09 34 65 40 60	Germany	289.645000
Rutrum Non Inc.	02 66 31 61 09	Germany	266.900000
Nunc Interdum Incorporated	05 18 15 48 13	Germany	244.025238
Augue Foundation	06 88 43 15 63	Germany	240.800000
Ac Fermentum Incorporated	06 85 56 52 33	Germany	206.465000
Auctor Mauris Corp.	05 62 87 14 41	Germany	184.310000
Convallis In Incorporated	06 66 57 29 50	Germany	156.730000

vistamarketing 2

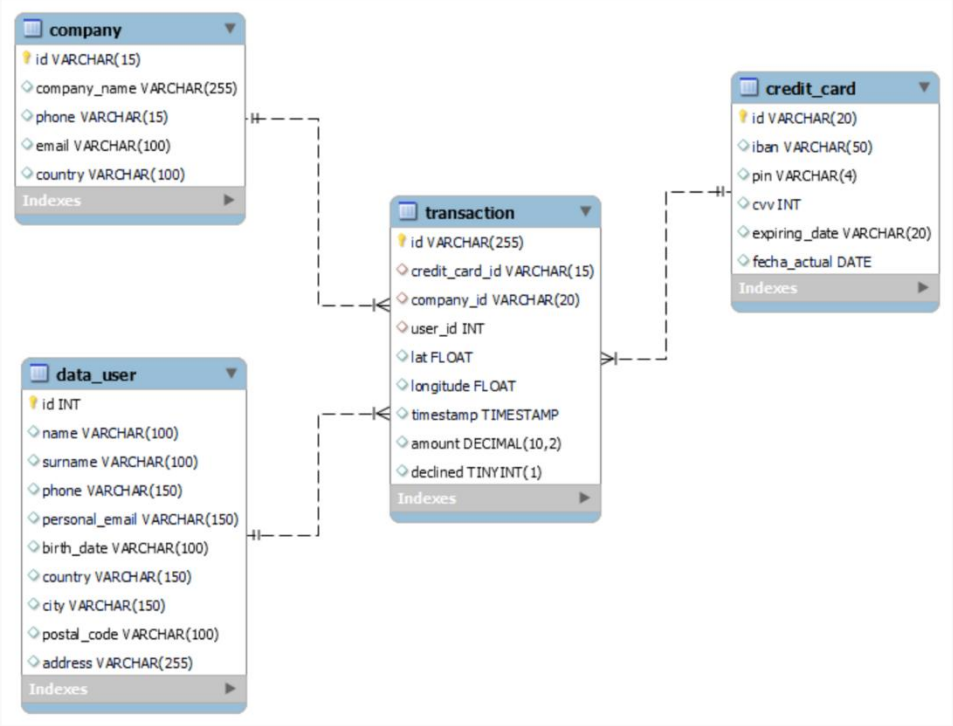
Output

Action Output

#	Time	Action	Message
1	10:44:05	CREATE VIEW VistaMarketing AS SELECT company.company_name, company.phone, ...	0 row(s) affected
2	10:44:05	SELECT * FROM transactions.vistamarketing LIMIT 0, 1000	101 row(s) returned
3	10:51:05	SELECT * FROM transactions.vistamarketing WHERE country='Germany' LIMIT 0, 1000	8 row(s) returned

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:



Farem servir ALTER TABLE per fer les modificacions de les taules I després mostrarem el diagrama. Primer fem la creació de l'esquema de la taula data_user, després la omplim (inserts) i per últim arreglem les relacions de les taules per tal de deixar el diagrama com es vol. A l'hora de carregar les dades es realitza un SET foreign_key_checks = 0 i al acabar d'introduir-les es canvia per 1.

```
1     CREATE INDEX idx_user_id ON transaction(user_id);
2
3  •  CREATE TABLE IF NOT EXISTS user (
4             id INT PRIMARY KEY,
5             name VARCHAR(100),
6             surname VARCHAR(100),
7             phone VARCHAR(150),
8             email VARCHAR(150),
9             birth_date VARCHAR(100),
10            country VARCHAR(150),
11            city VARCHAR(150),
12            postal_code VARCHAR(100),
13            address VARCHAR(255),
14            FOREIGN KEY(id) REFERENCES transaction(user_id)
15        );
16
17
```

Output

Action Output

#	Time	Action	Message
4	11:00:00	CREATE INDEX idx_user_id ON transaction(user_id)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
5	11:00:00	CREATE TABLE IF NOT EXISTS user (id INT PRIMARY KEY, name VARCHAR(100), ...	0 row(s) affected

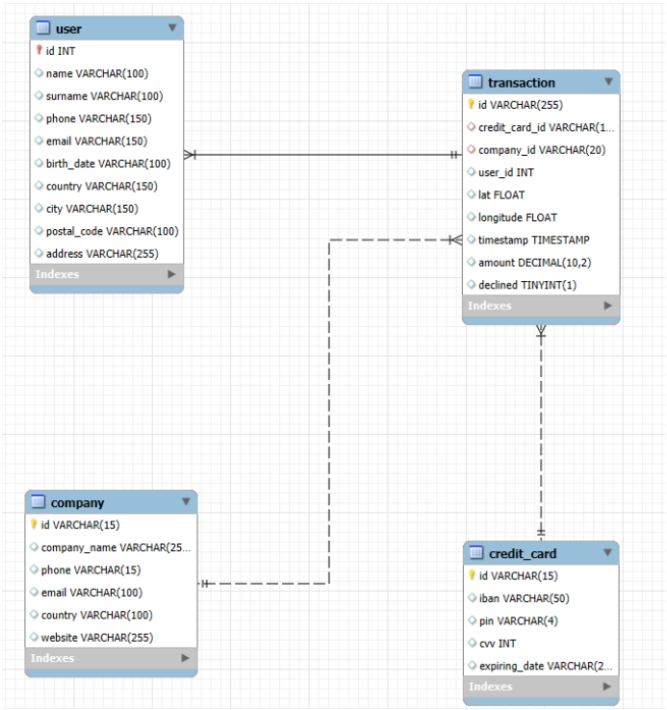
Juan Cano Pradas SPRINT 3 IT Academy

```
1  SET foreign_key_checks = 0;
2
3  -- Insertamos datos de user
4  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "1", "Zeu
5  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "2", "Gar
6  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "3", "Cia
7  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "4", "How
8  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "5", "Hay
9  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "6", "Joe
10 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "7", "Raf
11 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "8", "Nis
12 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "9", "Man
13 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (      "10", "Ro
14 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (      "11", "Jo
15 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (      "12", "Be
16 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (      "13", "Al
17 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (      "14", "Sa
18 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (      "15", "No
19 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (      "16", "Er
20 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (      "17", "Br
21 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (      "18", "Ru
```

Output

Action Output

#	Time	Action	Message
6	11:00:39	SET foreign_key_checks = 0	0 row(s) affected
7	11:00:39	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address...	1 row(s) affected
8	11:00:39	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address...	1 row(s) affected
9	11:00:39	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address...	1 row(s) affected
10	11:00:39	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address...	1 row(s) affected
11	11:00:39	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address...	1 row(s) affected
12	11:00:39	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address...	1 row(s) affected
13	11:00:39	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address...	1 row(s) affected
14	11:00:39	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address...	1 row(s) affected
15	11:00:39	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address...	1 row(s) affected



Juan Cano Pradas SPRINT 3 IT Academy

```sql
SELECT DISTINCT user_id
FROM transaction
WHERE user_id NOT IN (SELECT id from user);
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

user_id
9999

transaction 3

Output

Action Output

#	Time	Action	Message
1	11:09:06	SELECT DISTINCT user_id FROM transaction WHERE user_id NOT IN (SELECT id from user) LIMIT ...	1 row(s) returned

```sql
SELECT DISTINCT user_id
FROM transaction
WHERE user_id NOT IN (SELECT id from user);

INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "9999", "Pepito", "de los Palotes"

ALTER TABLE transaction
ADD CONSTRAINT fk_transaction_user
FOREIGN KEY (user_id) REFERENCES user(id);
```

Output

Action Output

#	Time	Action	Message	D
1	11:09:06	SELECT DISTINCT user_id FROM transaction WHERE user_id NOT IN (SELECT id from user) LIMIT ...	1 row(s) returned	0.
2	11:11:58	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) ...	1 row(s) affected	0.
3	11:12:21	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_user FOREIGN KEY (user_id) REFER...	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0	0.

```
376 •  SELECT DISTINCT user_id
377    FROM transaction
378    WHERE user_id NOT IN (SELECT id from user);
379
380 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "9999", "Pepito"
381
382 •  ALTER TABLE transaction
383    ADD CONSTRAINT fk_transaction_user
384    FOREIGN KEY (user_id) REFERENCES user(id);
385
386 •  ALTER TABLE user
387    DROP CONSTRAINT user_ibfk_1;
388
389    -- ex 2
390
391
392
```

Output

Action Output

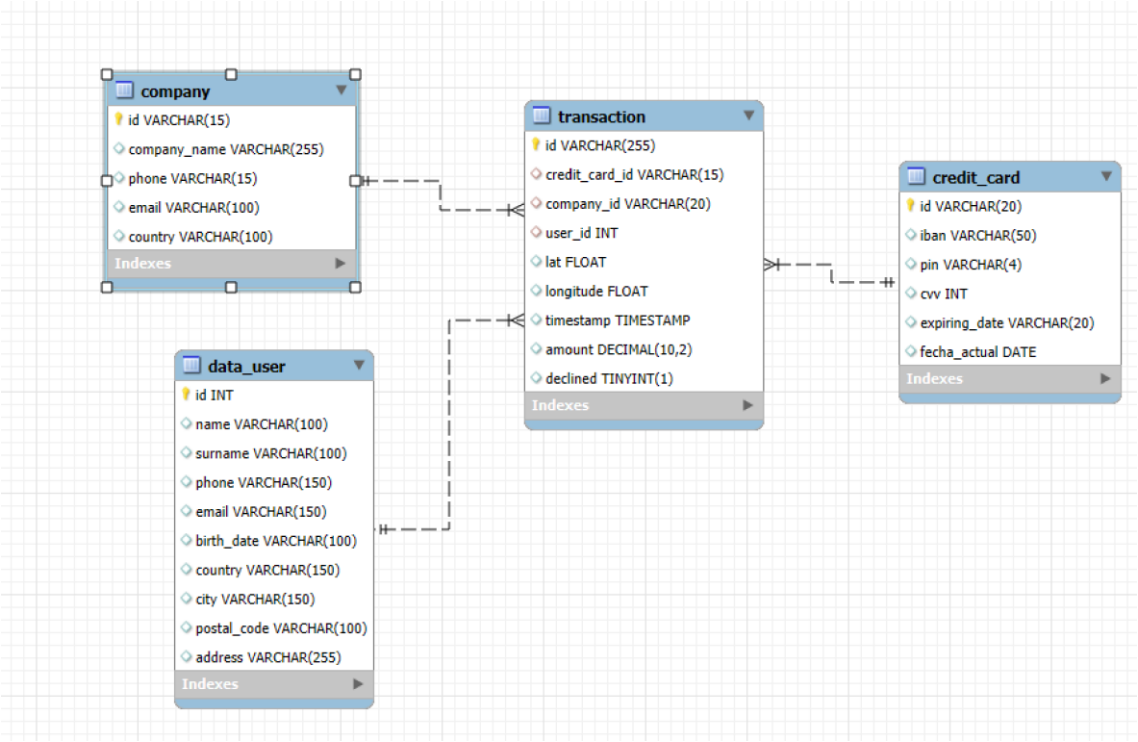
#	Time	Action	Message
1	11:41:48	ALTER TABLE user DROP CONSTRAINT user_ibfk_1	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

```
1  •  ALTER TABLE company
2     DROP COLUMN website;
3
4  •  ALTER TABLE user
5     RENAME TO data_user;
6
7  •  ALTER TABLE credit_card
8     ADD COLUMN fecha_actual DATE;
9
10 •  ALTER TABLE credit_card
11    MODIFY COLUMN id VARCHAR(20);
```

Output

Action Output

#	Time	Action	Message
1	11:41:48	ALTER TABLE user DROP CONSTRAINT user_ibfk_1	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
2	11:53:12	ALTER TABLE company DROP COLUMN website	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
3	11:53:12	ALTER TABLE user RENAME TO data_user	0 row(s) affected
4	11:53:12	ALTER TABLE credit_card ADD COLUMN fecha_actual DATE	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
5	11:53:13	ALTER TABLE credit_card MODIFY COLUMN id VARCHAR(20)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0



Aquí es mostra l'esquema que resulta després d'aplicar totes les modificacions.

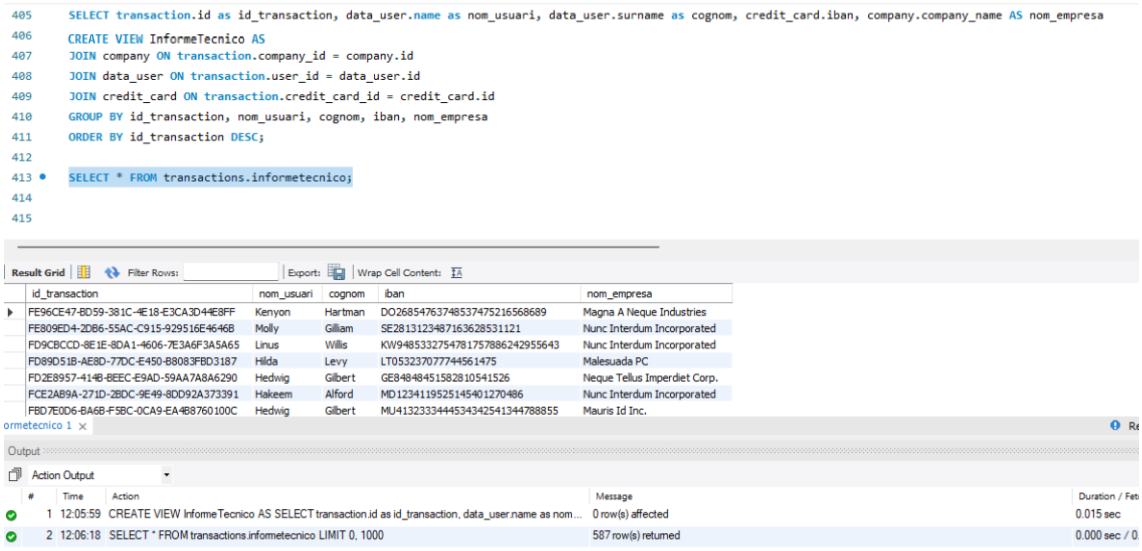
Exercici 2

L'empresa també et sol·licita 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.
- Assegura't d'incloure informació rellevant de totes dues taules i utilitza àlies per a canviar de nom columnes segons sigui necessari.

Mostra els resultats de la vista, ordena els resultats de manera descendent en funció de la variable ID de transaction.

Aquí realitzem la creació de la vista InformeTecnico a partir de 3 JOINS de les taules company, data_user, crèdit_card amb la taula TRANSACTION i en tots els casos amb els id corresponents a cadascuna de les taules (company, data_user, crèdit_card) que es troben com a foreign key a la taula transaction. Per últim fem l'agrupació per id_transaction, nom_usuari, cognom, iban i nom_empresa; i fem l'ordenació de major a menor segons el camp id_transaction (en aquest cas al ser un valor alfanumèric les files s'ordenen des de Z-A i 9-0). Finalment es fa la visualització de la vista per mostrar el seu contingut.



```
405    SELECT transaction.id as id_transaction, data_user.name as nom_usuari, data_user.surname as cognom, credit_card.iban, company.company_name AS nom_empresa
406    CREATE VIEW InformeTecnico AS
407    JOIN company ON transaction.company_id = company.id
408    JOIN data_user ON transaction.user_id = data_user.id
409    JOIN credit_card ON transaction.credit_card_id = credit_card.id
410    GROUP BY id_transaction, nom_usuari, cognom, iban, nom_empresa
411    ORDER BY id_transaction DESC;
412
413 •  SELECT * FROM transactions.informetecnico;
414
415
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

id_transaction	nom_usuari	cognom	iban	nom_empresa
FE96CE47-BD59-381C-4E18-E3CA3D44E8FF	Kenyon	Hartman	DO26854763748537475216568689	Magna A Neque Industries
FE809ED4-2DB6-55AC-C915-929516E4646B	Molly	Gilliam	SE2813123487163628531121	Nunc Interdum Incorporated
FD9CBCCD-8E1E-8DA1-4606-7E3A6F3A5A65	Linus	Willis	KW9485332754781757886242955643	Nunc Interdum Incorporated
FD89D51B-AE8D-77DC-E450-B8083FBD3187	Hilda	Levy	LT053237077744561475	Malesuada PC
FD2E8957-414B-BEEC-E9AD-59AA7A8A6290	Hedwig	Gilbert	GE84848451582810541526	Neque Tellus Imperdiet Corp.
FCE2AB9A-271D-2BDC-9E49-8DD92A373391	Hakeem	Alford	MD1234119525145401270486	Nunc Interdum Incorporated
FBD7E0D6-BA6B-F5BC-0CA9-EA4B8760100C	Hedwig	Gilbert	MU4132333444534342541344788855	Mauris Id Inc.

ormetecnico 1 ×

Output

Action Output

#	Time	Action	Message	Duration / Fet
1	12:05:59	CREATE VIEW InformeTecnico AS SELECT transaction.id as id_transaction, data_user.name as nom ...	0 row(s) affected	0.015 sec
2	12:06:18	SELECT * FROM transactions.informetecnico LIMIT 0, 1000	587 row(s) returned	0.000 sec / 0