

Tasca S3.01. Manipulació de taules

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.

He començat creant la taula "credit_card":

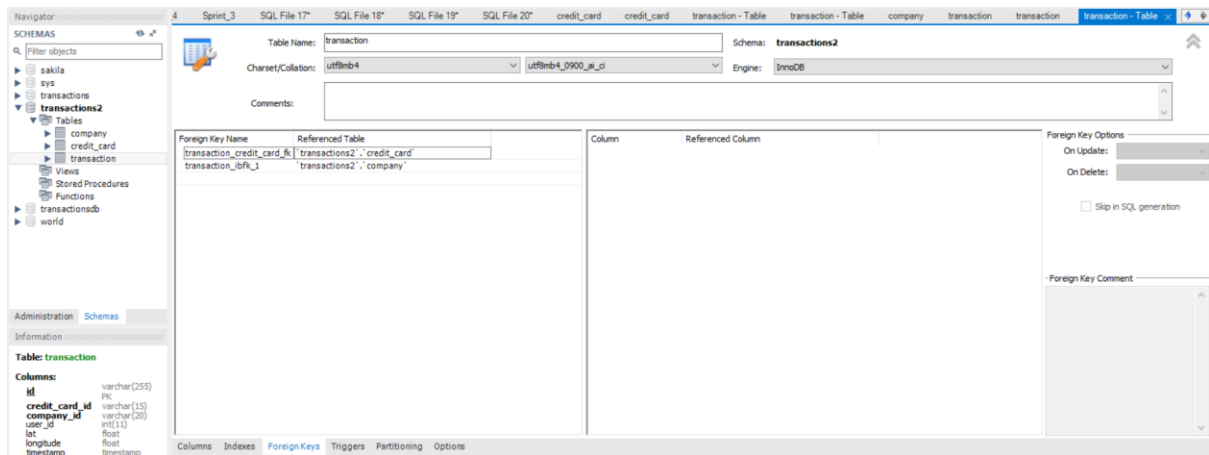
```
CREATE TABLE IF NOT EXISTS credit_card
(
    id VARCHAR (20) PRIMARY KEY,
    iban VARCHAR (50),
    pan VARCHAR (30),
    pin VARCHAR (10),
    cvv VARCHAR (10),
    expiring_date VARCHAR (20))
;
```

A l'hora d'importar les dades, he trobat un problema, ja que les "expiring_date" estan en format: mes/dia/any, per això he decidit definir-lo com a VARCHAR i un cop importat, modificar el tipus de data, perquè la detecti correctament.

```
# Cambiamos el tipo de dato de expiring_date:
UPDATE credit_card
SET expiring_date = STR_TO_DATE(expiring_date, '%m/%d/%y');
```

```
ALTER TABLE credit_card MODIFY COLUMN expiring_date DATE;
```

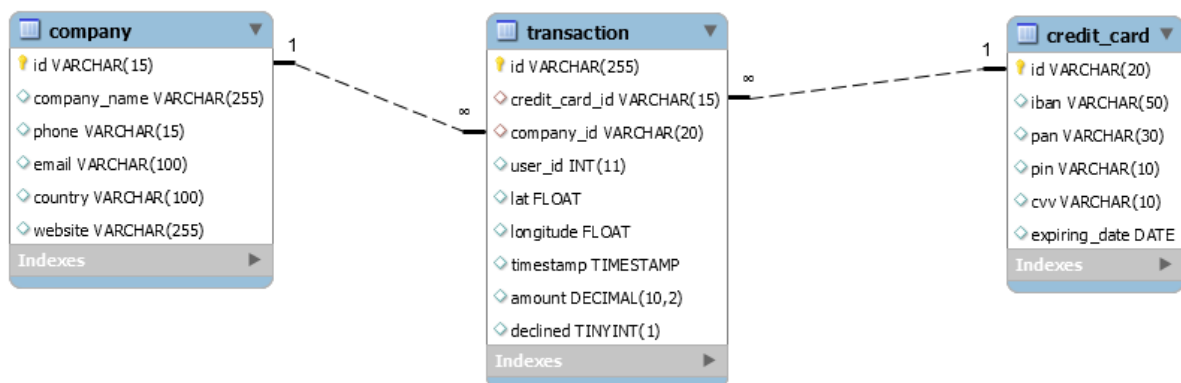
A continuació, he definit les relacions entre taules a través de les Foreign Keys.



El mateix programa ens indica que podriem fer aquesta modificació mitjançant la següent “Query”:

```
ALTER TABLE `transactions2`.`transaction`
ADD INDEX `transaction_credit_card_fk_idx` (`credit_card_id` ASC) VISIBLE;
;
ALTER TABLE `transactions2`.`transaction`
ADD CONSTRAINT `transaction_credit_card_fk`
FOREIGN KEY (`credit_card_id`)
REFERENCES `transactions2`.`credit_card` (`id`)
ON DELETE NO ACTION
ON UPDATE NO ACTION;
```

D'aquesta manera queden vinculades aquestes dues taules. De manera que el diagrama final és el següent:



Podem observar que la base de dades està formada per 3 taules:

A la taula “Company” podem trobar la informació de diferents empreses:

- L'identificador (id) [**Primary Key**] - varchar
- El nom de l'empresa (company_name) - varchar

- El telèfon (phone) - varchar
- el correu electrònic (email) - varchar
- El país (Country) - varchar
- La pàgina web (website) - varchar

A la taula "Transaction" trobem la informació de les transaccions que s'han dut a terme amb targetes de crèdit de cada companyia:

- L'identificador únic de cada transacció (id) [**Primary Key**] - varchar
- L'identificador de cada targeta de crèdit (credit_card_id [**Foreing Key**]) - varchar
- l'identificador de la companyia a la qual pertany la targeta de crèdit (company_id) [**Foreing Key**] - varchar
- L'identificador de l'usuari de la targeta de crèdit (user_id) - interger
- La geolocalització (lat, longitude) - float
- La data de la transacció (timestamp) - TIMESTAMP
- La quantitat (amount) - decimal
- Ens indica si la targeta ha estat rebutjada o no (reclined) és una variable de tipus booleà (0 o 1) - boolean

A la taula credit_card trobem la informació de les targetes de credit que han fet la transacció:

- L'identificador únic de cada targeta de crèdit (id) [**Primary Key**] - varchar
- L'IBAN del compte - varchar
- El PAN - varchar
- El PIN - varchar
- El CVV- varchar
- La data de caducitat (expiring_date) - DATE

Les taules 'transaction' i 'company' estan relacionades per la foreing key 'company_id' que es troba relacionada amb 'id' de la taula company.

La taula credit_card es troba relacionada amb transaction per la foreing key 'credit_card_id', que es relaciona amb el 'id' de la taula credit_card.

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.

The screenshot shows a database management interface with two SQL queries and their results.

Query 1:

```
10
11 • SELECT* from credit_card;
12
13 • UPDATE credit_card
14   SET iban = 'R323456312213576817699999'
15   WHERE id= 'CcU-2938';
```

Output:

#	Time	Action	Message
1	14:17:11	UPDATE credit_card SET iban = 'R323456312213576817699999' WHERE id= 'CcU-2938'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0

Query 2:

```
25 • SELECT iban
26   FROM credit_card
27   WHERE id='CcU-2938';
28
```

Result Grid:

iban
R323456312213576817699999

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

```
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');
```

En executar aquest query, dona un error, ja que la companyia 'CcU-9999' no existeix a la taula "company". Aquesta és clau Forana, per tant, ha de tenir registre a la taula "Company", el que he fet és afegir un registre amb aquest "company_id" a la taula "company". El mateix passa amb "credit_card_id":

```
SELECT id FROM company WHERE id = 'b-9999';
```

id
NULL

- Introdueixo un nou 'id' a la taula company, la resta de dades, les deixaré en blanc perquè no les conec:

```
INSERT INTO company (id)
VALUES ('b-9999');
```

995 12:59:19 INSERT INTO company (id) VALUES (b-9999)

1 row(s) affected

- Introdueixo un nou 'id' a la taula 'credit_card':

```
INSERT INTO credit_card (id)
VALUES('CcU-9999');
```

996 13:02:03 INSERT INTO credit_card (id) VALUES(CcU-9999)

1 row(s) affected

- Ara si, introduïm les dades de la transacció que ens demanen:

```
INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined)
VALUES ('10881D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', '9999', '829.999', '-117.999', '111.11', '0');
```

997 13:03:59 INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES (... 1 row(s) affected

```
337 • SELECT *
338 FROM transaction
339 WHERE id = '10881D1D-5B23-A76C-55EF-C568E49A99DD';
340
```

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
10881D1D-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

- Exercici 4

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

```
344
345 • ALTER TABLE credit_card
346 DROP COLUMN pan;
347
348 • SELECT *
349 FROM credit_card;
350
```

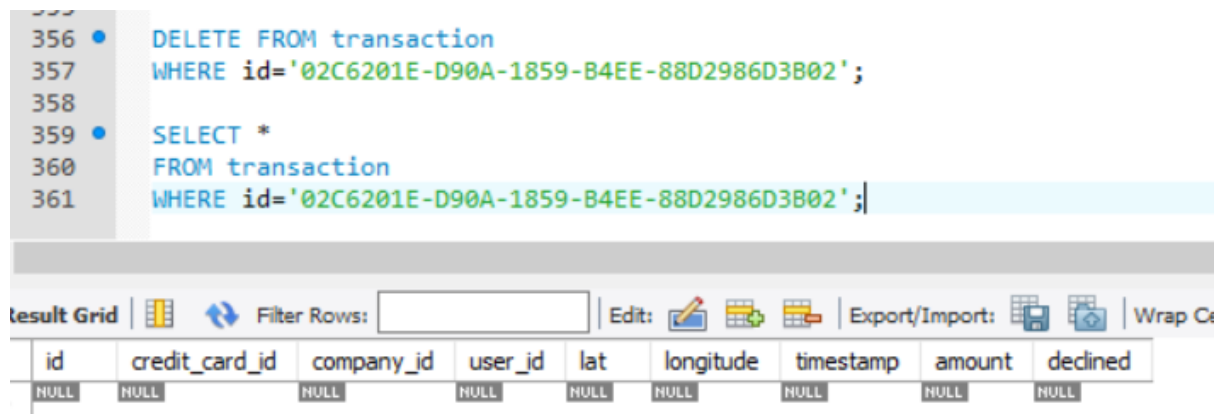
id	iban	pin	cvv	expiring_date
CcU-2938	R323456312213576817699999	3257	984	2022-10-30
CcU-2945	DO26854763748537475216568689	9080	887	2023-08-24
CcU-2952	BG45IVQL52710525608255	4598	438	2021-06-29
CcU-2959	CR7242477244335841535	3583	667	2023-02-24
CcU-2966	BG72LKTQ15627628377363	4900	130	2024-10-29
CcU-2973	PT87806228135092429456346	8760	887	2025-01-30

Nivell 2

Exercici 1

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

```
356 • DELETE FROM transaction
357 WHERE id='02C6201E-D90A-1859-B4EE-88D2986D3B02';
358
359 • SELECT *
360 FROM transaction
361 WHERE id='02C6201E-D90A-1859-B4EE-88D2986D3B02';|
```



The screenshot shows a database interface with a SQL editor and a result grid. The SQL editor contains two queries: a DELETE statement and a SELECT statement, both targeting the 'transaction' table with a specific ID. The result grid below shows a single row with all columns set to NULL.

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL




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.

Si tenim en compte les transaccions declinades, he creat una vista a partir d'aquesta consulta:

```
CREATE VIEW VistaMarketing_1 AS
SELECT company_name as 'Nom de la companyia',
       phone as 'Telèfon de contacte',
       country as 'País de residència',
       AVG(amount) as 'Mitjana de la compra'
FROM transaction LEFT JOIN company
ON transaction.company_id=company.id
GROUP BY company_name, phone, country
ORDER BY AVG(amount) desc
;
```





Resultat:

Result Grid  Filter Rows: <input type="text"/> Export:  Wrap Cell Content: 				
	Nom de la companyia	Telèfon de contacte	País de residència	Mitjana de la compra
▶	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
	Egestas Nunc Sed Limited	06 01 02 70 47	Italy	406.110000
	Nunc Sit Incorporated	07 28 42 63 63	Norway	405.355000

* Els 'alias' que he fet servir, únicament tenen finalitat estètica per fer més comprensible la taula de la vista creada.

Si no tenim en compte les transaccions declinades:

```
CREATE VIEW VistaMarketing AS
SELECT company_name as 'Nom de la companyia',
       phone as 'Telèfon de contacte',
       country as 'País de residència',
       AVG(amount) as 'Mitjana de la compra'
FROM transaction LEFT JOIN company
ON transaction.company_id=company.id
WHERE declined =0
GROUP BY company_name, phone, country
ORDER BY AVG(amount) desc
;
```


Result Grid   Filter Rows: <input type="text"/> Export:  Wrap Cell Content: 				
	Nom de la companyia	Telèfon de contacte	País de residència	Mitjana de la compra
▶	Eget Ipsum Ltd	03 67 44 56 72	United States	481.860000
	Sed Id Limited	07 28 18 18 13	United States	477.510000
	Neque Tellus Incorporated	04 43 18 34 19	Ireland	477.100000
	Nunc Sit Incorporated	07 28 42 63 63	Norway	461.830000
	Non Magna LLC	06 71 73 13 17	United Kingdom	458.740000
	Maecenas Malesuada Fringilla Inc.	09 38 53 76 61	Netherlands	451.290000
	Erat LLP	03 18 88 77 79	Netherlands	448.440000
	Tortor Nunc Commodor Company	05 35 92 77 16	United States	447.110000
	Justo Eu Arcu Ltd	08 42 56 71 52	Italy	444.160000
	Pede Cum Ltd	07 62 26 48 38	Norway	442.320000
	Vestibulum Lorem PC	02 02 87 33 40	Belgium	428.400000
	Mauris Institute	05 29 60 36 87	Sweden	427.710000

* Els 'alias' que he fet servir, únicament tenen finalitat estètica per fer més comprensible la taula de la vista creada.

Exercici 3

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

```
SELECT *  
FROM vistamarketing_1  
WHERE 'País de residència' = 'Germany';
```

Si filtro d'aquesta manera la vista, no em dona cap resultat. Però consultant la taula resulta que sí que hi ha companyies amb seu a 'Germany'. Sembla que en fer servir alies amb caràcters especials a la vista que he creat en comparar les dues cadenes de text, no troba coincidències.

Es pot solucionar demanant a la SELECT que faci servir la mateixa configuració a l'hora de comparar les "string":

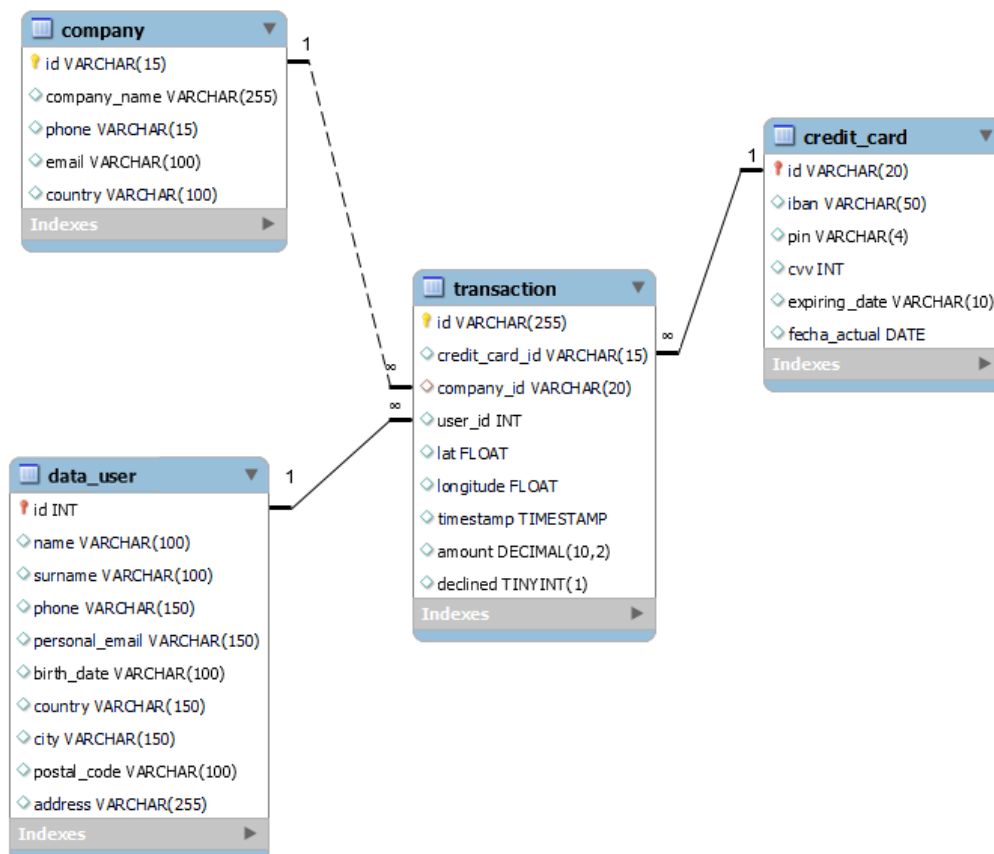
```
SELECT *  
FROM vistamarketing_1  
WHERE CONVERT(`País de residència` USING utf8) = CONVERT('Germany' USING utf8);
```

	Nom de la companyia	Telefòn de contacte	País de residència	Mitjana de la compra
▶	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

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:



Observant l'esquema el primer que veig és que ha importat la taula 'users'. Però les relacions que ha establert de 'transactions' a 'credit_card' i 'user', estan malament, ja que s'han establert les FK al revés.

A més, veig que ha fet diferents canvis en les taules que ja teníem:

- **Taula 'company':**
 - Ha eliminat el camp 'website' de la taula 'company'.
- **Taula 'credit_card':**
 - Ha canviat el tipus de dada del 'cvv' (INT) i 'expiring_date' (VARCHAR)
 - Ha afegit el camp 'fecha_actual' DATE
- **Taula 'user':**
 - li ha posat el nom de 'personal_email'.

Per important la taula 'user', primer l'hem de crear:

```
CREATE TABLE IF NOT EXISTS user (  
  id INT PRIMARY KEY,  
  name VARCHAR(100),  
  surname VARCHAR(100),  
  phone VARCHAR(150),  
  email VARCHAR(150),  
  birth_date VARCHAR(100),  
  country VARCHAR(150),  
  city VARCHAR(150),  
  postal_code VARCHAR(100),  
  address VARCHAR(255)  
);
```

I relacionar-la amb la taula 'transaction'. Però en fer-ho, ens dona error perquè hi ha un valor a la taula 'transaction' que no hi és a la taula 'user'. És el valor id=9999, llavors, hem d'afegir-lo a la taula 'user' per poder establir aquesta relació.

```
INSERT INTO user (id) VALUES (9999);
```

Ara sí, podem fer la relació:

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

1308 10:57:17 INSERT INTO user (id) VALUES (9999)

1 row(s) affected

1309 10:58:51 ALTER TABLE transaction ADD CONSTRAINT fk_user_id FOREIGN KEY (user_id) REFERENCES user(id)

587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0

He decidit no eliminar el camp 'website' de la taula 'company' per què no estic segura si és un error i sempre seré a temps d'eliminar-ho si és necessari.

Podem canviar el tipus de dada de 'cvv' i 'expiring_date' de la taula 'credit_card':

```
434
435 #canviem el tipus de dada de 'cvv':
436
437 • ALTER TABLE credit_card
438   MODIFY COLUMN cvv INT;
439
440 # canviem el tipus de dada a expiring_date:
441
442 • ALTER TABLE credit_card
443   MODIFY COLUMN expiring_date VARCHAR(10);
444
445
```

<

Output

Action Output

#	Time	Action
✓ 1313	12:49:29	ALTER TABLE credit_card MODIFY COLUMN cvv INT
✓ 1314	12:51:17	ALTER TABLE credit_card MODIFY COLUMN expiring_date VARCHAR(10)

Afegim el camp fecha_actual:

```
ALTER TABLE credit_card
ADD fecha_actual DATE;
```

En aquest cas, només crea un camp on s'aniran registrant la data en el moment de fer els nous ingressos de dades a la taula, però en els registres que ja estan ingressats, el valro serà NULL.

Si volem posar la data actual a tots els registres:

```
ALTER TABLE credit_card
ADD fecha_actual_2 TIMESTAMP DEFAULT CURRENT_TIMESTAMP;

UPDATE credit_card
SET fecha_actual_2 = CURRENT_TIMESTAMP
WHERE fecha_actual IS NULL;
```

En la següent imatge podem veure la diferència.

id	iban	pin	cvv	expiring_date	fecha_actual	fecha_actual_2
CcU-2938	R323456312213576817699999	3257	984	2022-10-30	NULL	2024-04-03 13:00:47
CcU-2945	DO26854763748537475216568689	9080	887	2023-08-24	NULL	2024-04-03 13:00:47
CcU-2952	BG451VQL52710525608255	4598	438	2021-06-29	NULL	2024-04-03 13:00:47
CcU-2959	CR7242477244335841535	3583	667	2023-02-24	NULL	2024-04-03 13:00:47
CcU-2966	BG72LKTQ15627628377363	4900	130	2024-10-29	NULL	2024-04-03 13:00:47
CcU-2973	PT87806228135092429456346	8760	887	2025-01-30	NULL	2024-04-03 13:00:47
CcU-2980	DE39241881883086277136	5075	596	2022-07-24	NULL	2024-04-03 13:00:47
CcU-2987	GE89681434837748781813	2298	797	2023-10-31	NULL	2024-04-03 13:00:47
CcU-2994	BH62714428368066765294	7545	595	2022-02-28	NULL	2024-04-03 13:00:47
CcU-3001	CY49087426654774581266832110	9562	867	2022-09-16	NULL	2024-04-03 13:00:47
CcU-3008	LU507216693616119230	1856	740	2025-04-05	NULL	2024-04-03 13:00:47
CcU-3015	PS119398216295715968342456821	3246	822	2022-01-31	NULL	2024-04-03 13:00:47

Per últim, podem canviar el nom del camp 'email' per 'personal_email':

```
ALTER TABLE user
CHANGE COLUMN email personal_email VARCHAR (150);
```

id	name	surname	phone	personal_email	birth_date	country	city	postal_code	address
1	Zeus	Gamble	1-282-581-0551	interdum.enim@protonmail.edu	Nov 17, 1985	United States	Lowell	73544	348-7818 Sagittis St.
2	Garrett	Mcconnell	(718) 257-2412	integer.vitae.nibh@protonmail.org	Aug 23, 1992	United States	Des Moines	59464	903 Sit Ave
3	Ciaran	Harrison	(522) 598-1365	interdum.feugiat@aol.org	Apr 29, 1998	United States	Columbus	56518	736-2063 Tellus St.
4	Howard	Stafford	1-411-740-3269	ornare.egestas@icloud.edu	Feb 18, 1989	United States	Kailua	77417	Ap #545-2244 Erat. Rd.
5	Hayfa	Pierce	1-554-541-2077	et.malesuada.fames@hotmail.org	Sep 26, 1998	United States	Sandy	31564	341-2821 Ultrices Av.
6	Joel	Tyson	(718) 288-8020	gravida.nunc.sed@yahoo.ca	Oct 15, 1989	United States	Nashville	96838	888-2799 Amet Street
7	Rafael	Jimenez	(817) 689-0478	euget@outlook.ca	Dec 4, 1981	United States	Hillsboro	29874	8627 Malesuada Rd.

En les relacions, hi ha un gran error, ja que el company ha establert les Foreign Keys a les teules 'credit_card' i 'user' i hauria d'haver-les establert a 'transaction'.

Per solucionar aquest error, primer hauríem d'eliminar aquestes relacions:

```
ALTER TABLE credit_card
```

```
DROP FOREIGN KEY nom_fk;
```

```
ALTER TABLE user
```

```
DROP FOREIGN KEY nom_fk;
```

```
ALTER TABLE transaction
```

```
ADD CONSTRAINT card_fk
```

```
FOREIGN KEY (credit_card_id)
```

```
REFERENCES credit_card (id);
```

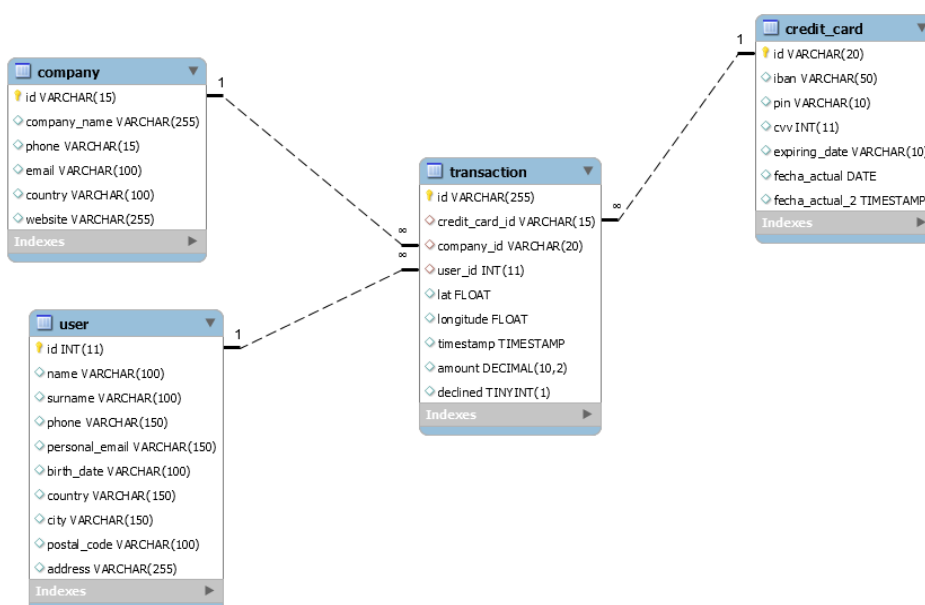
```
ALTER TABLE transaction
```

```
ADD CONSTRAINT user_fk
```

```
FOREIGN KEY (user_id)
```

```
REFERENCES user (id);
```

En aquest cas no adjunto imatges perquè aquestes relacions ja les he establert en els exercicis anteriors. Amb tots els canvis, l'esquema quedaria de la següent manera:



Exercici 2

L'empresa també et sol·licita crear una vista anomenada "InformeTecnico" que contingui la següent informació:

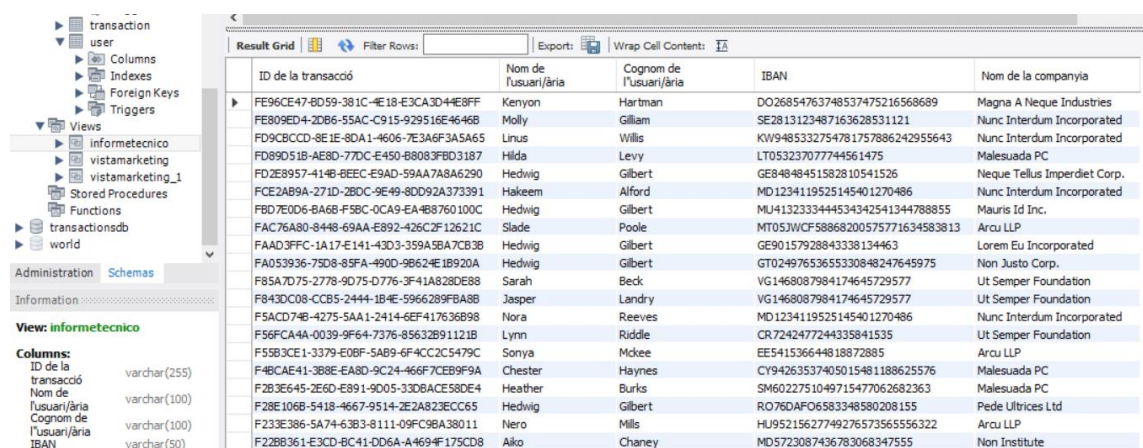
- ID de la transacció
- Nom de l'usuari/ària
- Cognom de l'usuari/ària
- IBAN de la targeta de crèdit usada.
- Nom de la companyia de la transacció realitzada.

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'.

```
CREATE VIEW InformeTecnico AS
SELECT transaction.id AS 'ID de la transacció',
       user.name AS "Nom de l'usuari/ària",
       user.surname AS 'Cognom de l'usuari/ària',
       credit_card.iban AS 'IBAN',
       company.company_name AS 'Nom de la companyia'
FROM Transaction LEFT JOIN company
ON transaction.company_id = company.id
LEFT JOIN credit_card
ON transaction.credit_card_id = credit_card.id
LEFT JOIN user
ON transaction.user_id = user.id
ORDER BY transaction.id desc;
```

El resultat de la vista creada és:



ID de la transacció	Nom de l'usuari/ària	Cognom de l'usuari/ària	IBAN	Nom de la companyia
FE96CE47-8D59-381C-4E18-E3CA3D-44E8FF	Kenyon	Hartman	DO26854763748537475216568689	Magna A Neque Industries
FE809ED4-2D86-55AC-C915-929516E-4646B	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-8EEC-E9AD-59AA7A8A6290	Hedwig	Gilbert	GE84848451582810541526	Neque Tellus Imperdiet Corp.
FCE2AB9A-271D-2BDC-9E49-8DD92A373391	Hakeem	Alford	MD1234119525145401270486	Nunc Interdum Incorporated
FBD7E0D6-8A68-F5BC-0CA9-EA4B8760100C	Hedwig	Gilbert	MU413233344453432541344788855	Mauris Id Inc.
FAC76A80-8448-69AA-E892-426C2F12621C	Slade	Poole	MT05JWCF58868200575771634583813	Arcu LLP
FAAD3FFC-1A17-E141-43D3-359A5BA7CB38	Hedwig	Gilbert	GE90157928843338134463	Lorem Eu Incorporated
FA053936-75D8-85FA-490D-96624E1B920A	Hedwig	Gilbert	GT02497653655330848247645975	Non Justo Corp.
F85A7D75-2778-9D75-D776-3F41A828DE88	Sarah	Beck	VG1468087984174645729577	Ut Semper Foundation
F843DC08-CCB5-2444-1B4E-5966289FBA88	Jasper	Landry	VG1468087984174645729577	Ut Semper Foundation
F5ACD74B-4275-5AA1-2414-6EF417636898	Nora	Reeves	MD1234119525145401270486	Nunc Interdum Incorporated
F56FCA4A-0039-9F64-7376-85632B91121B	Lynn	Riddle	CR7242477244335841535	Ut Semper Foundation
F55B3CE1-3379-E0BF-5AB9-6F4CC2C5479C	Sonya	Mckee	EE541536644818872885	Arcu LLP
F4BCAE41-388E-EA8D-9C24-466F7CEB9F9A	Chester	Haynes	CY94263537405015481188625576	Malesuada PC
F2B3E645-2E6D-E891-9005-330BACE58DE4	Heather	Burks	SM6022751049715477062682363	Malesuada PC
F28E1068-5418-4667-9514-2E2A823ECC65	Hedwig	Gilbert	RO76DAFO6583348580208155	Pede Ultrices Ltd
F233E386-5A74-63B3-8111-09FC9BA38011	Nero	Mills	HU95215627749276573565556322	Arcu LLP
F22B8361-E3CD-BC41-0D6A-A4694F175CD8	Aiko	Chaney	MD5723087436783068347555	Non Institute