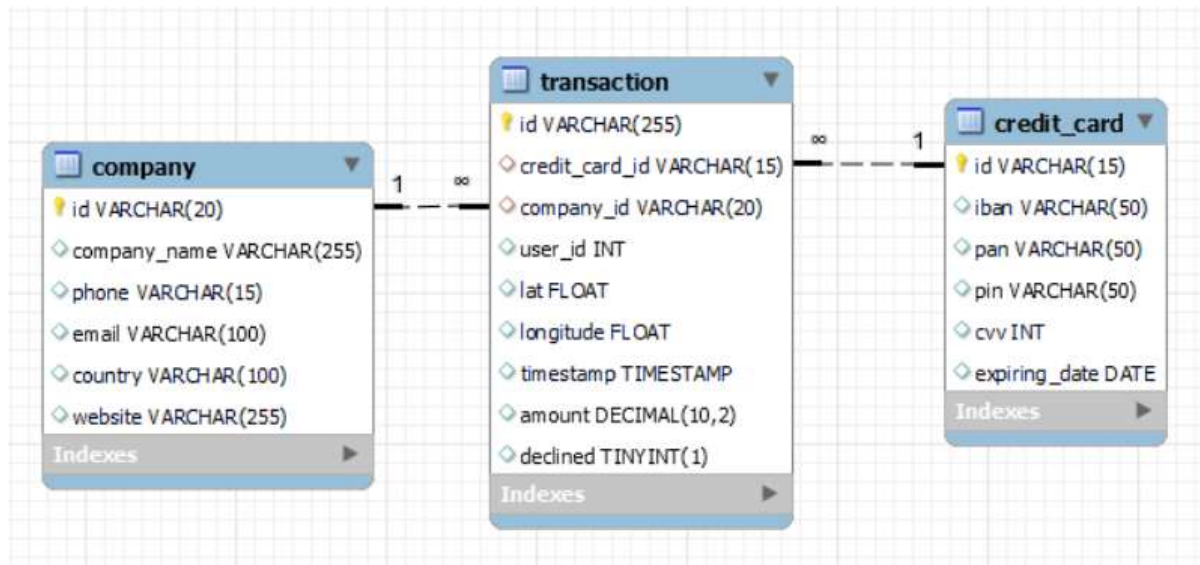


# Nivell 1

## - 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.



Aquí podem observar que tenim una **base de dades relacional**, on tenim per un costat la taula de fets (**transaction**) i dues taules de dimensions (**company** i **credit\_card**).

Per poder crear la taula **credit\_card**, introduir les dades i posteriorment relacionar-la amb les taules **transaction** i **company**, se han seguit els següents passos:

- Primer se analitzen les dades que tenim per introduir i es pot observar que tenen els següents camps:
  - **Id**: mostra el número d'identificació d'una targeta en concret i és la PK. Li assigno el valor Varchar amb longitud 15 pels valors dels dades.
  - **Iban**: mostra el número de compte de la targeta associada. Pongo una longitud més llarga, valor Varchar 50 perquè veig en les dades que tenen diferents mesures.
  - **Pan**: mostra el nº de targeta de la targeta de crèdit. Aquí em va a posar 15 perquè és el nombre de dígits que té una targeta, però en les dades facilitades en uns hi ha espais i en altres no, per tant, li deixo una longitud més llarga; Varchar 50.
  - **Pin**: mostra el nº pin de la targeta de crèdit. Li assigno Varchar 4 perquè el pin sol ser de 4 dígits.
  - **Cvv**: mostra el nº de CVV de la targeta de crèdit. Li assigno per defecte Varchar 50.
  - **Expiring date**: mostra la data de caducitat de la targeta de crèdit.

```

11 • CREATE TABLE credit_card_def ( #crear tabla donde tambien indicamos la PK
12     id varchar(15) NOT NULL,
13     iban varchar(50) DEFAULT NULL,
14     pan varchar(50) NOT NULL, #mas adelante me doy cuenta que necesito que pueda ser nulo
15     pin varchar(50) DEFAULT NULL,
16     cvv int DEFAULT NULL,
17     expiring_date DATE DEFAULT NULL,
18     PRIMARY KEY (`id`)
19 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

```

Output

#	Time	Action	Message
1	10:21:23	CREATE TABLE credit_card_def ( #crear tabla donde tambien indicamos la PK id varchar(15) NOT NULL, ...	0 row(s) affected

Para poder **crear la tabla**, primero tuve en cuenta que la fecha de caducidad no venía con formato ISO, con el predeterminado. Por ese motivo primero creo la tabla definitiva. Para posteriormente crear una tabla temporal donde introducir los datos y luego pasarla a la tabla definitiva con la fecha en formato predeterminado y no en string. También **crearemos la tabla definitiva** para poder trasladar luego los datos:

```

11 • CREATE TABLE credit_card_def ( #crear tabla donde tambien indicamos la PK
12     id varchar(15) NOT NULL,
13     iban varchar(50) DEFAULT NULL,
14     pan varchar(50) NOT NULL, #mas adelante me doy cuenta que necesito que pueda ser nulo
15     pin varchar(50) DEFAULT NULL,
16     cvv int DEFAULT NULL,
17     expiring_date DATE DEFAULT NULL,
18     PRIMARY KEY (`id`)
19 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;

```

Aquí podemos ver el código de la **tabla temporal** (no hice script de cuando lo ejecuté y no he podido poner el resultado que ha sido ok).

```

24 • CREATE TEMPORARY TABLE credit_card (
25     id VARCHAR(15),
26     iban VARCHAR(50),
27     pan VARCHAR(50),
28     pin VARCHAR(50),
29     cvv VARCHAR(50),
30     expiring_date VARCHAR(50)
31 );

```

Seguidamente, ejecutamos el script de **datos\_introducir\_credit**, para que inserte los datos en la tabla temporal.

A continuación, **inserto los datos** de la tabla temporal a la tabla definitiva:

```
34  #Inserto los datos el tabla definitiva desde la tabla temporal
35 • INSERT INTO credit_card_def (id, iban, pan, pin, cvv, expiring_date)
36  SELECT id, iban, pan, pin, cvv, STR_TO_DATE(expiring_date, '%m/%d/%Y')
37  FROM credit_card;
```

A continuación, **verificamos** que la tabla se ha creado correctamente, **eliminamos** la tabla temporal para que no de problemas y **renombramos** la tabla final para que tenga el nombre que queremos.

```
39  #Verifico que la tabla se ha creado correctamente
40 • SELECT *
41  FROM credit_card_def;
42
43  #Elimino la tabla temporal para que no haya problemas con el nombre
44 • DROP TEMPORARY table credit_card;
45
46  #Renombro la tabla definitiva
47 • RENAME TABLE credit_card_def TO credit_card;
```

Ahora tenemos la tabla creada con todos los datos introducidos y la fecha en formato predeterminado (y/m/d).

Por lo tanto, el siguiente paso, es relacionar las tablas mediante los campos en común. En este caso, **relacionamos la tabla credit\_card con la columna id a la tabla transactions con la columna credit\_card\_id**.

```
52 • ALTER TABLE transaction
53  ADD CONSTRAINT transaction_creditcard
54  FOREIGN KEY (credit_card_id)
55  REFERENCES credit_card(id);
```

Observando los datos que quedan, vemos que, en **pan**, de la tabla credit\_card, hay espacios en algunas filas y otras sin espacios. Ejecuto la siguiente consulta para poder **eliminar los espacios**.

```
63  #Borro los espacios vacios que existen en algunas lineas de la columna pan
64 • UPDATE credit_card
65  SET pan = REPLACE(pan, ' ', '');
66
```

Output			
Action Output			
#	Time	Action	Message
3	10:34:53	SELECT * FROM credit_card INNER JOIN transaction ON credit_card.id = transaction.credit_card_id LIM...	588 row(s) returned
4	10:35:40	UPDATE credit_card SET pan = REPLACE(pan, ' ', '')	0 row(s) affected Rows matched: 276 Changed: 0 Warnings: 0

Indicar, que se ve que no se hizo ningún cambio porque ya se hizo anteriormente.

**Compruebo** que la relación se ha creado correctamente haciendo un join de las tablas.

```
57 #Compruebo que funciona correctamente uniendo las tablas por la columna asignada
58 • SELECT *
59 FROM credit_card
60 INNER JOIN transaction
61 ON credit_card.id = transaction.credit_card_id;
62
```

Result Grid							
Filter Rows:							
Export:							
Wrap Cell Content:							
id	iban	pan	pin	cvv	expiring_date	id	
CcU-2938	R323456312213576817699999	5424465566813633	3257	984	2022-10-30	02C6201E-D90A-1859-B4EE-88D2986D3B02	
CcU-2938	R323456312213576817699999	5424465566813633	3257	984	2022-10-30	108B1D1D-5B23-A76C-55EF-C568E49A05DD	
CcU-2938	R323456312213576817699999	5424465566813633	3257	984	2022-10-30	17561134-37A4-E316-2C3C-C5541EDAD761	
CcU-2938	R323456312213576817699999	5424465566813633	3257	984	2022-10-30	185C9F2C-67A2-8EC0-0BB6-93789CC59F05	

Result 1			
Output			
Action Output			
#	Time	Action	Message
2	10:24:18	drop table credit_card_def	0 row(s) affected
3	10:34:53	SELECT * FROM credit_card INNER JOIN transaction ON credit_card.id = transaction.credit_card_id LIM...	588 row(s) returned

Y, por último, hacer una consulta de la tabla para ver todos **los datos correctamente**.

```
67 #Compruebo que queda todo correcto
68 • SELECT *
69 FROM credit_card;
70
```

Result Grid							
Filter Rows:							
Edit:							
Export/Import:							
Wrap Cell Content:							
id	iban	pan	pin	cvv	expiring_date		
CcU-2938	R323456312213576817699999	5424465566813633	3257	984	2022-10-30		
CcU-2945	DO26854763748537475216568689	5142423821948828	9080	887	2023-08-24		
CcU-2952	BG45IVQL52710525608255	4556453555287	4598	438	2021-06-29		
CcU-2959	CR7242477244335841535	372461377349375	3583	667	2023-02-24		
CcU-2966	BG72LKTQ15627628377363	4485668867477265	4900	130	2024-10-29		

credit_card 2			
Output			
Action Output			
#	Time	Action	Message
4	10:35:40	UPDATE credit_card SET pan = REPLACE(pan, '.', '')	0 row(s) affected Ro
5	10:37:26	SELECT * FROM credit_card LIMIT 0, 50000	276 row(s) returned



## - 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.

Primero **compruebo** la información existente:

```
78 • SELECT id, iban
79 FROM credit_card
80 WHERE id = 'CcU-2938';
```

Result Grid	
id	iban
CcU-2938	TR301950312213576817638661
NULL	NULL

Realizo el **cambio**:

```
82 • UPDATE credit_card
83 SET iban = 'R323456312213576817699999'
84 WHERE id = 'CcU-2938';
```

Output					
Action Output					
#	Time	Action	Message	Duration / Fetch	
4	10:52:01	SELECT * FROM credit_card LIMIT 0, 50000	275 row(s) returned	0.000 sec / 0.000	
5	11:01:19	SELECT id, iban FROM credit_card WHERE i...	Error Code: 1054. Unknown column 'CcU' in '...	0.000 sec	
6	11:01:38	SELECT id, iban FROM credit_card WHERE i...	1 row(s) returned	0.000 sec / 0.000	
7	11:05:46	UPDATE credit_card SET iban = 'R32345631...	1 row(s) affected Rows matched: 1 Changed:...	0.016 sec	

Por último, **compruebo** que todo se haya cambiado correctamente:

```
86 • SELECT id, iban
87 FROM credit_card
88 WHERE id = 'CcU-2938';
```

Result Grid	
id	iban
CcU-2938	R323456312213576817699999
NULL	NULL

Output					
Action Output					
#	Time	Action	Message	Duration / Fetch	
5	11:01:19	SELECT id, iban FROM credit_card WHERE i...	Error Code: 1054. Unknown column 'CcU' in '...	0.000 sec	
6	11:01:38	SELECT id, iban FROM credit_card WHERE i...	1 row(s) returned	0.000 sec / 0	
7	11:05:46	UPDATE credit_card SET iban = 'R32345631...	1 row(s) affected Rows matched: 1 Changed:...	0.016 sec	
8	11:06:58	SELECT id, iban FROM credit_card WHERE i...	1 row(s) returned	0.000 sec / 0	

## - Exercici 3

En la taula "transaction" ingresa 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

Al intentar insertar los datos primero en credit\_card me doy cuenta de que el campo pan lo puse como no nulo, entonces al intentar poner el usuario y no introducir el pan me daba error, por ese motivo, **modifico el tipo de dato a Default Null del campo pan.**

```
105 #Me doy cuenta que puse el PAN como valor no nulo y tengo que cambiarlo
106 • ALTER TABLE credit_card
107 MODIFY pan varchar(50) DEFAULT NULL;
```

Output					
Action Output					
#	Time	Action	Message	Duration / Fetc	
✓ 8	11:06:58	SELECT id, iban FROM credit_card WHERE i...	1 row(s) returned	0.000 sec / 0.1	
✓ 9	11:13:52	SELECT * FROM credit_card INNER JOIN tra...	587 row(s) returned	0.015 sec / 0.1	
✗ 10	11:21:21	INSERT INTO credit_card (id) VALUES ('CcU...	Error Code: 1364. Field 'pan' doesn't have a d...	0.000 sec	
✓ 11	11:23:56	ALTER TABLE credit_card MODIFY pan varc...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.094 sec	

A continuación, introduzco los datos en la tabla **credit\_card**:

```
105 #Primero inserto datos en la tabla credit_card
106 • INSERT INTO credit_card (id)
107 VALUES ('CcU-9999');
108
```

Output					
Action Output					
#	Time	Action	Message		
✗ 10	11:21:21	INSERT INTO credit_card (id) VALUES ('CcU...	Error Code: 1364. Field 'pan' doesn't have a d...		
✓ 11	11:23:56	ALTER TABLE credit_card MODIFY pan varc...	0 row(s) affected Records: 0 Duplicates: 0 ...		
✓ 12	11:25:35	SELECT * FROM credit_card INNER JOIN tra...	587 row(s) returned		
✓ 13	11:26:45	INSERT INTO credit_card (id) VALUES ('CcU...	1 row(s) affected		

Luego, en la tabla **transaction**:

```
109 #Inserto los datos en la tabla transaction
110 • INSERT INTO transaction (id, user_id, lat, longitude, amount, declined)
111 VALUES ('108B1D1D-5B23-A76C-55EF-C568E49A99DD', 9999, 829.999, -117.999, 111.11, 0);
112
```

Output

#	Time	Action	Message	Duration / Fetch
✓ 13	11:26:45	INSERT INTO credit_card (id) VALUES ('CcU-9999')	1 row(s) affected	0.000 sec
✓ 14	11:30:24	SELECT * FROM credit_card INNER JOIN tra...	587 row(s) returned	0.000 sec / 0.000
✗ 15	11:31:54	INSERT INTO transaction (user_id, lat, longit...	Error Code: 1364. Field 'id' doesn't have a def...	0.015 sec
✓ 16	11:32:53	INSERT INTO transaction (id, user_id, lat, lon...	1 row(s) affected	0.000 sec

Y por ultimo, en la tabla **company**:

```
113 #Inserto los datos en la tabla company
114 • INSERT INTO company (id)
115 VALUES ('b-9999');
116
```

Output

#	Time	Action	Message	Duration / Fet
✓ 14	11:30:24	SELECT * FROM credit_card INNER JOIN tra...	587 row(s) returned	0.000 sec / 0
✗ 15	11:31:54	INSERT INTO transaction (user_id, lat, longit...	Error Code: 1364. Field 'id' doesn't have a def...	0.015 sec
✓ 16	11:32:53	INSERT INTO transaction (id, user_id, lat, lon...	1 row(s) affected	0.000 sec
✓ 17	11:34:50	INSERT INTO company (id) VALUES ('b-9999')	1 row(s) affected	0.000 sec

Al intentar mostrar el resultado, me da error. Buscando la solución, veo que el problema está en que no he introducido en la tabla transaction la información que pertenece también a la tabla company y a la tabla credit\_card respectivamente.

Así que, añado la información que falta en la tabla **transaction**.

```
117 #Al intentar mostrar los resultados, me doy cuenta que en la tabla transaction no he puesto los valores que relacionan con las otras tablas
118 • UPDATE transaction
119 SET credit_card_id = 'CcU-9999',
120     company_id = 'b-9999'
121 WHERE id like '108B1D1D-5B23-A76C-55EF-C568E49A99DD';
122
```

#	Time	Action	Message	Duration
✓ 5	10:37:26	SELECT * FROM credit_card LIMIT 0, 50000	276 row(s) returned	0.000 sec
✓ 6	10:42:26	UPDATE transaction SET credit_card_id = 'CcU-9999', company_id = 'b-9999' WHERE id like '108B1D1...	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0	0.000 sec

Por último, muestro el **resultado**.

```

124 • SELECT t.id as Id, t.user_id, t.lat, t.longitude, t.amount, t.declined, credit_card.id as credit_card_id, company.id as company_id
125 FROM transaction as t
126 INNER JOIN credit_card on t.credit_card_id = credit_card.id
127 INNER JOIN company on company.id = t.company_id
128 WHERE t.id like '108B1D1D-5B23-A76C-55EF-C568E49A99DD'
129 AND credit_card.id like 'CcU-9999'
130 AND company.id like 'b-9999';
131

```

Result Grid									
Filter Rows: <input type="text"/>									
Exports:									
Wrap Cell Contents: <input type="checkbox"/>									
	Id	user_id	lat	longitude	amount	declined	credit_card_id	company_id	
▶	108B1D1D-5B23-A76C-55EF-C568E49A99DD	9999	829.999	-117.999	111.11	0	CcU-9999	b-9999	

Result 3 x

Output				
Action Output				
#	Time	Action	Message	
6	10:42:26	UPDATE transaction SET credit_card_id = 'CcU-9999', company_id = 'b-9999' WHERE id like '108B1D1D-5B23-A76C-55EF-C568E49A99DD';	0 row(s) affected Rows matched: 1 Changed: 0 Warnings: 0	
7	10:43:55	SELECT t.id as Id, t.user_id, t.lat, t.longitude, t.amount, t.declined, credit_card.id as credit_card_id, company.id as company_id	1 row(s) returned	

## - Exercici 4

Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit\_\*card. Recordar mostrar el canvi realitzat.

Primero **ver** los datos:

```

141 • SELECT *
142 FROM credit_card;

```

(No hice el script antes y por eso no están los resultados).

Luego ejecuto el comando para poder **borrar** la columna:

```

138 • ALTER TABLE credit_card
139 DROP COLUMN pan;

```

Output				
Action Output				
#	Time	Action	Message	
7	10:43:55	SELECT t.id as Id, t.user_id, t.lat, t.longitude, t.amount, t.declined, credit_card.id as credit_card_id, company.id as company_id	1 row(s) returned	
8	10:49:59	ALTER TABLE credit_card DROP COLUMN pan	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	



Muestro los **resultados**:

```
141 • SELECT *
142 FROM credit_card;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: [IA](#)

	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

credit\_card 4 x

Output

Action Output

	#	Time	Action	Message
✓	8	10:49:59	ALTER TABLE credit_card DROP COLUMN pan	0 row(s) affected Rec
✓	9	10:51:17	SELECT * FROM credit_card LIMIT 0, 50000	276 row(s) returned

## Nivell 2

### Exercici 1

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

Primero **revisar** la información:

```
154 #Primero reviso los datos:
155 • SELECT *
156 FROM transaction
157 WHERE id like '02C6201E-D90A-1859-B4EE-88D2986D3B02';
```

Luego, **eliminar** el registro:

```
148 • DELETE
149 FROM transaction
150 WHERE id like '02C6201E-D90A-1859-B4EE-88D2986D3B02';
151
```

Output

Action Output

#	Time	Action	Message
11	11:01:32	SELECT * FROM transaction WHERE id like '02C6201E-D90A-1859-B4EE-88D2986D3B02' LIMIT 0, 50000	1 row(s) returned
12	11:02:16	DELETE FROM transaction WHERE id like '02C6201E-D90A-1859-B4EE-88D2986D3B02'	1 row(s) affected

**Compruebo** que se ha borrado correctamente:

```
156 #Comando para comprobar que se ha borrado correctamente
157 • SELECT *
158 FROM transaction
159 WHERE id like '02C6201E-D90A-1859-B4EE-88D2986D3B02';
160
```

Result Grid

Filter Rows:

Edit:

Export/Import:

	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.

Primero, creo la **vista**:

```
161 • CREATE VIEW VistaMarketing AS
162 SELECT
163     company.company_name AS NomCompanyia,
164     company.phone AS Telefon,
165     company.country AS Pais,
166     AVG(transaction.amount) AS MitjanaCompra
167 FROM
168     company
169 JOIN
170     transaction ON company.id = transaction.company_id
171 GROUP BY
172     company.company_name, company.country, company.phone
173 ORDER BY
174     MitjanaCompra DESC;
175
```

Output

#	Time	Action	Message
20	11:17:27	drop view VistaMarketing	0 row(s) affected
21	11:17:33	CREATE VIEW VistaMarketing AS SELECT company.company_name AS NomCompanyia, company.phone AS Telefon, com...	0 row(s) affected

Muestro los **resultados**.

```
176 • SELECT *
177 FROM VistaMarketing;
```

NomCompanyia	Telefon	Pais	MitjanaCompra
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

VistaMarketing 10 x

Output

#	Time	Action	Message
21	11:17:33	CREATE VIEW VistaMarketing AS SELECT company.company_name AS NomCompanyia, company.phone AS Telefon, com...	0 row(s) affected
22	11:18:10	SELECT * FROM VistaMarketing LIMIT 0, 50000	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":

En este ejercicio, interpreto que lo que se busca es ejecutar un filtro en una vista. Por ese motivo, consulta la vista con el filtro en país:

```
182 • SELECT *
183 FROM VistaMarketing
184 WHERE Pais like 'Germany';
```

Result Grid				
Filter Rows:		Export:	Wrap Cell Content:	
NomCompanyia	Telefon	Pais	MitjanaCompra	
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	

VistaMarketing 11 x

Output

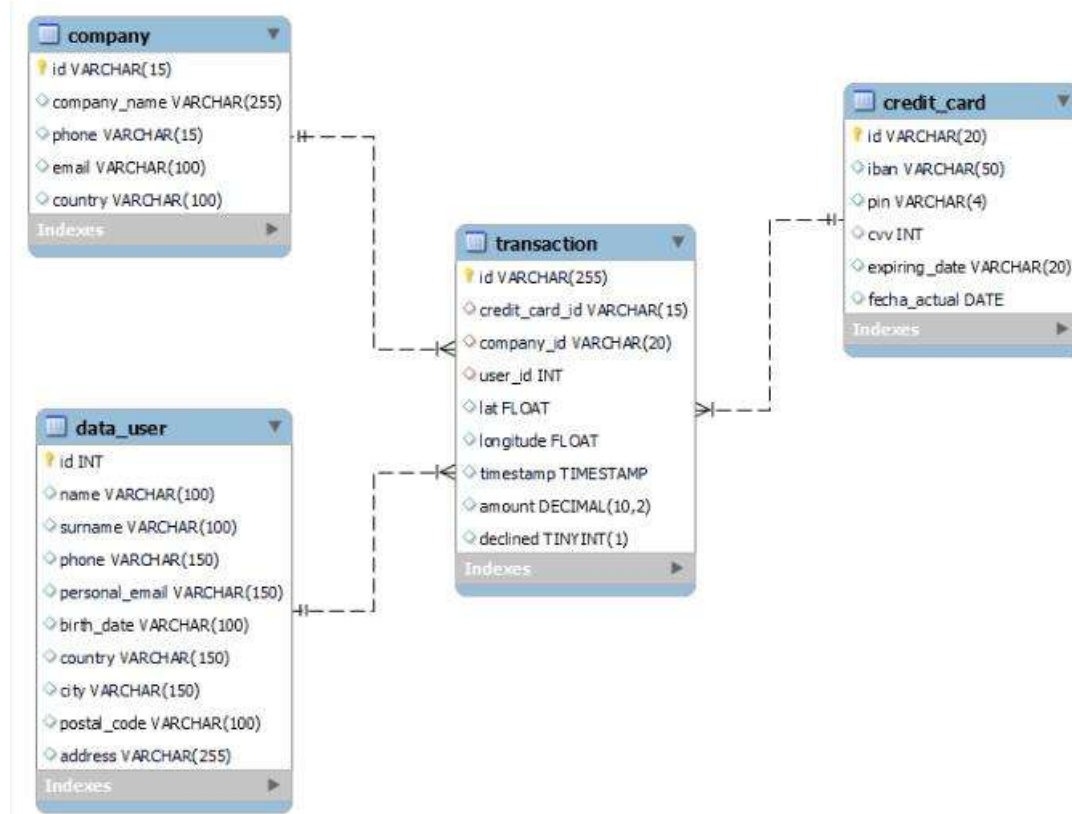
Action Output			
#	Time	Action	Message
✓ 22	11:18:10	SELECT * FROM VistaMarketing LIMIT 0, 50000	101 row(s) returned
✓ 23	11:19:27	SELECT * FROM VistaMarketing WHERE Pais like 'Germany' LIMIT 0, 50000	8 row(s) returned



# Nivell 3

## Exercici 1

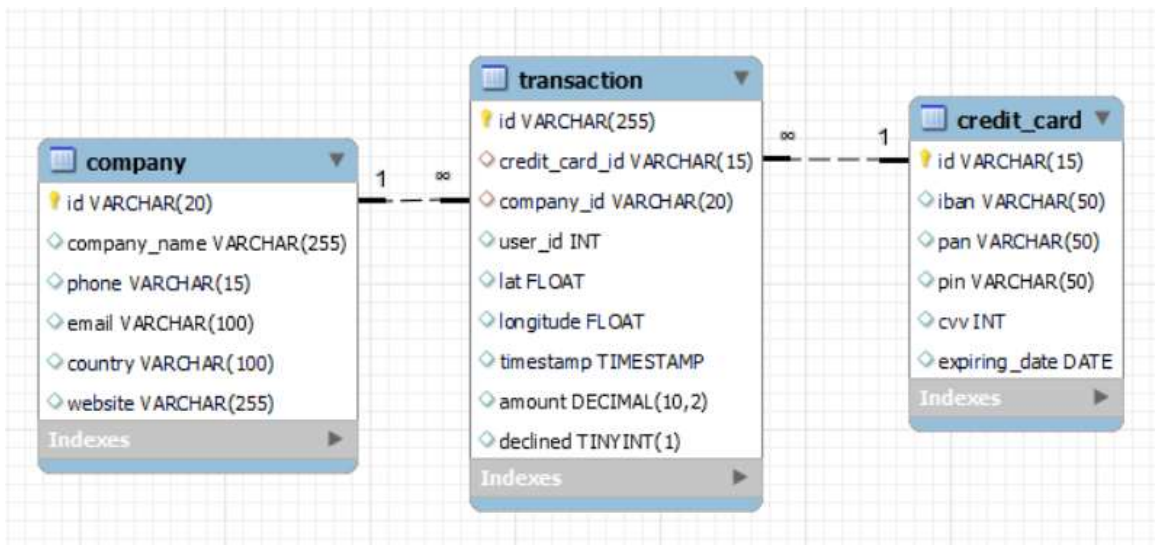
La setmana vinent tindràs una nova reunió amb els gerents de màrqueting. Un company del teu equip va realitzar modificacions en la base de dades, però no recorda com les va realitzar. Et demana que l'ajudis a deixar els comandos executats per a obtenir el següent diagrama:



### Recordatori

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

Primero, vemos lo que tenemos creado vs lo que hay creado y que cambios se realizaron para llegar ahí.



Analizaremos tabla por tabla y haremos las modificaciones correspondientes.

La tabla transaction no hay que modificar (más adelante, me doy cuenta de que si hay que modificar la para cuadrar la FK con la tabla data\_user).

- Por tanto, **elimino primero la FK:**

```
189 • ALTER TABLE transaction
190 DROP FOREIGN KEY transaction_creditcard;
```

#	Time	Action	Message
25	12:16:41	SHOW CREATE TABLE company	1 row(s) returned
26	12:19:06	ALTER TABLE transaction DROP FOREIGN KEY transac...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

- Como me seguía dando error, busqué todas las FK que afectaban a los cambios para eliminarlas y no tener futuros problemas con las relaciones entre ellas. Por tanto, también debo eliminar otra constraint que afectaba al campo id de la tabla transaction:

```
192 • ALTER TABLE transaction
193 DROP FOREIGN KEY transaction_ibfk_1;
```

#	Time	Action	Message	Duration / Fetch
33	12:28:01	ALTER TABLE company MODIFY id varchar(15)	Error Code: 1833. Cannot change column 'id': used in a foreign key constraint 'transaction_ibfk_1'...	0.000 sec
34	12:28:44	ALTER TABLE transaction DROP FOREIGN KEY transac...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.016 sec

- Introduzco de nuevo la FK:

```
204 • ALTER TABLE transaction
205 ADD CONSTRAINT transaction_company
206 FOREIGN KEY (company_id)
207 REFERENCES company(id);
208
```

Output				
Action Output				
#	Time	Action	Message	
43	12:48:02	ALTER TABLE transaction ADD CONSTRAINT transaccio...	Error Code: 1826. Duplicate foreign key constraint name 'transaction_ibfk_1'	
44	12:49:33	ALTER TABLE transaction ADD CONSTRAINT transaccio...	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0	

- De la tabla **Company**, haremos los siguientes cambios:

- Eliminar la columna website:

```
200 • ALTER TABLE company
201 DROP COLUMN website;
202
```

Output				
Action Output				
#	Time	Action	Message	
36	12:34:50	ALTER TABLE transaction ADD CONSTRAINT transaccio...	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0	
37	12:41:28	ALTER TABLE company DROP COLUMN website	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	

Continuamos con los cambios de la tabla **credit\_card**:

- Modifico el **tamaño** del valor **id**:

```
214 • ALTER TABLE credit_card
215 MODIFY id varchar(20);
216
```

Output				
Action Output				
#	Time	Action	Message	Duration / Fetch
341	13:16:31	ALTER TABLE transaction ADD CONSTR...	587 row(s) affected Records: 587 Duplicates: 0 ...	0.093 sec
342	13:17:52	ALTER TABLE credit_card MODIFY id var...	0 row(s) affected Records: 0 Duplicates: 0 War...	0.016 sec

- Modifico el **tamaño** del valor de **pin**:

```
211 • ALTER TABLE credit_card
212 MODIFY pin varchar(4);
213
```

Output				
Action Output				
#	Time	Action	Message	
44	12:49:33	ALTER TABLE transaction ADD CONSTRAINT transaccio...	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0	
45	12:51:16	ALTER TABLE credit_card MODIFY pin varchar(4)	276 row(s) affected Records: 276 Duplicates: 0 Warnings: 0	

- Modifico el valor **expiring\_date** de fecha a texto:

```
214 #Modifico la expiring_date de fecha a varchar:
215 • ALTER TABLE credit_card
216 MODIFY COLUMN expiring_date VARCHAR(20);
```

Output

#	Time	Action	Message
✓ 45	12:51:16	ALTER TABLE credit_card MODIFY pin varchar(4)	276 row(s) affected Records: 276 Duplicates: 0 Warnings: 0
✓ 46	12:54:02	ALTER TABLE credit_card MODIFY COLUMN expiring_d...	276 row(s) affected Records: 276 Duplicates: 0 Warnings: 0

- Por último, añado la columna **fecha\_actual** como fecha:

```
219 • ALTER TABLE credit_card
220 ADD COLUMN fecha_actual DATE;
```

Output

#	Time	Action	Message
✓ 46	12:54:02	ALTER TABLE credit_card MODIFY COLUMN expiring_d...	276 row(s) affected Records: 276 Duplicates: 0 Warnings: 0
✓ 47	12:56:39	ALTER TABLE credit_card ADD COLUMN fecha_actual ...	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

- Y añado los datos de la columna **fecha\_actual**:

```
223 • UPDATE credit_card
224 SET fecha_actual = CURDATE();
225
```

Output

#	Time	Action	Message
✗ 50	12:58:09	UPDATE credit_card SET fecha_actual = fecha	Error Code: 1054. Unknown column 'fecha' in 'field list'
✓ 51	12:59:18	UPDATE credit_card SET fecha_actual = CURDATE()	276 row(s) affected Rows matched: 276 Changed: 276 Warnings: 0

Para finalizar, tendremos que crear la tabla **user**:

- Primero creamos la tabla con el script facilitado **estructura\_datos\_user**.
- Introducimos los datos con el script **datos\_introducir\_user(1)**.
- Observo que la columna email tiene que ser **personal\_email**. Procedo a modificar el nombre de la columna:

```
228 • ALTER TABLE user
229 RENAME COLUMN email TO peronal_email;
```

Output

#	Time	Action	Message	Duration / Fet
✓ 335	13:04:41	SET foreign_key_checks = 1	0 row(s) affected	0.000 sec
✓ 336	13:07:48	ALTER TABLE user RENAME COLUMN e...	0 row(s) affected Records: 0 Duplicates: 0 War...	0.016 sec



- Al corregir, nos damos cuenta de que tengo mal el nombre, procedo a ponerlo correctamente:

```
237 #Al corregir me doy cuenta que tengo el mal el nombre:
```

```
238 • ALTER TABLE data_user
```

```
239 RENAME COLUMN peronal_email TO personal_mail;
```

```
240
```

Output

Action Output

#	Time	Action	Message	Duration / Fe
9	11:09:05	ALTER TABLE user_data RENAME COLUMN...	Error Code: 1146. Table 'transactions.user_da...	0.000 sec
10	11:09:53	ALTER TABLE data_user RENAME COLUMN...	0 row(s) affected Records: 0 Duplicates: 0 ...	0.047 sec

- También hay que cambiar el nombre de la tabla **user** a **data\_user**:

```
231 • RENAME TABLE user TO data_user;
```

Output

Action Output

#	Time	Action	Message
336	13:07:48	ALTER TABLE user RENAME COLUMN e...	0 row(s) affected Records: 0 Duplicates: 0 War...
337	13:11:19	RENAME TABLE user TO data_user	0 row(s) affected

- Añadir el **user 9999** que añadimos en el ejercicio anterior, para que los datos sean coincidentes:

```
241 INSERT INTO data_user (id)
```

```
242 VALUES (9999);
```

```
243
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
345	13:31:18	ALTER TABLE data_user INSERT INTO user (id) VALU...	Error Code: 1064. You have an error in your SQL syntax; check ...	0.000 sec
346	13:32:37	INSERT INTO data_user (id) VALUES (9999)	1 row(s) affected	0.000 sec

- Observo que se ha creado una FK de manera automática al introducir los datos, pero hay que ponerla correctamente. Por tanto, buscaremos primero el **nombre concreto de la constraint** para poder eliminarla:

```
248 #Añadir la PK a la tabla data_user. Primero buscamos el nombre de la FK para poder borrarla
```

```
249 • show create table data_user;
```

```
250
```

Result Grid



Filter Rows:



Export:



Wrap Cell Content:

Table	Create Table
data_user	CREATE TABLE 'data_user' ( 'id' int NOT NULL, 'name' varchar(100) DEFAULT NULL, 'surname' varchar(100) DEFAULT NU

- Posteriormente, **elimino la FK** de la tabla `user_data`;, porque se ha creado una FK de manera automática al introducir los datos, pero hay que ponerla correctamente:

```

248 • ALTER TABLE data_user
249 DROP FOREIGN KEY data_user_ibfk_1;

```

Output

#	Time	Action	Message
351	13:50:38	ALTER TABLE data_user DROP FOREIGN KEY transaction_company	Error Code: 1091. Can't DROP 'transaction_company'; check th
352	13:50:53	ALTER TABLE data_user DROP FOREIGN KEY data_user_ibfk_1	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

- Añado la FK a la tabla transaction:**

```

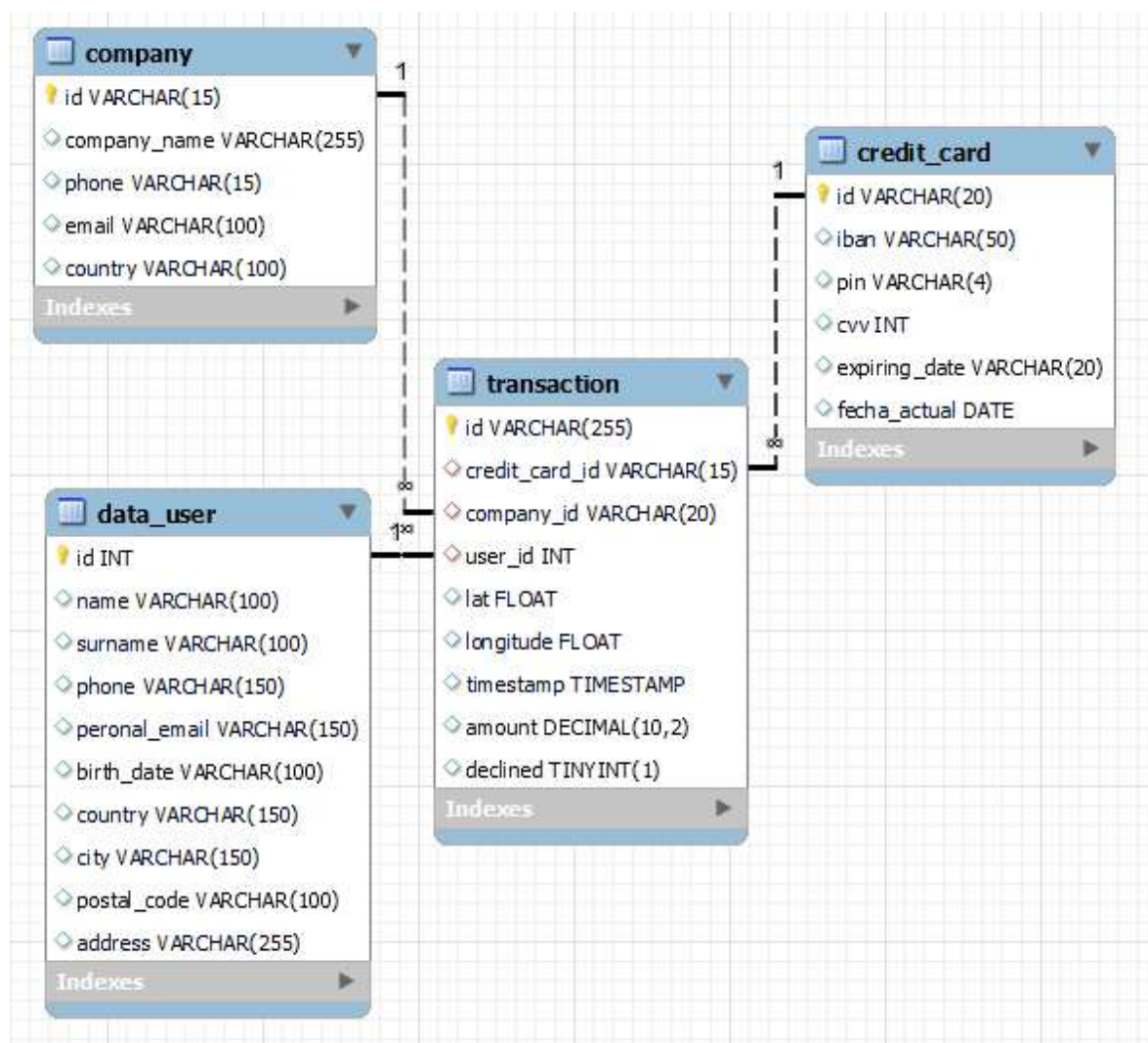
251 • ALTER TABLE transaction
252 ADD CONSTRAINT transaction_data_user
253 FOREIGN KEY (user_id)
254 REFERENCES data_user(id);

```

Output

#	Time	Action	Message
352	13:50:53	ALTER TABLE data_user DROP FOREIGN KEY data_user_ibfk_1	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
353	13:53:06	ALTER TABLE transaction ADD CONSTRAINT transaction_data_user FOREIGN KEY (user_id) REF...	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0

Muestro el resultado final:



## 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.

Primero **crearem** la vista:

```
261 • CREATE VIEW InformeTecnico AS
262 SELECT transaction.id AS Id_transaction, data_user.name AS Nom, data_user.surname AS Cognom, credit_card.iban AS IBAN, company.company_name AS Companyia
263 FROM transaction
264 INNER JOIN data_user ON transaction.user_id = data_user.id
265 INNER JOIN credit_card ON transaction.credit_card_id = credit_card.id
266 INNER JOIN company ON transaction.company_id = company.id;
267
```

#	Time	Action	Message	Duration
2	09:03:15	SELECT transaction.id AS Id_transaction, data_user.name AS Nom, data_user.surname AS Cognom, ...	587 row(s) returned	0.000 s
3	09:04:30	CREATE VIEW InformeTecnico AS SELECT transaction.id AS Id_transaction, data_user.name AS N...	0 row(s) affected	0.016 s

Muestro los **resultados** de la vista ordenados de manera descendiente en función de la variable ID de transaction:

```
269 • SELECT *
270 FROM InformeTecnico
271 ORDER BY Id_transaction DESC;
272
```

Id_transaction	Nom	Cognom	IBAN	Companyia
FE96CE47-BD59-381C-4E18-E3CA3D44E8FF	Kenyon	Hartman	DO26854763748537475216568689	Magna A Neque Industries
FE809ED4-2DB6-55AC-C915-929516E46468	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.

#	Time	Action	Message
4	09:08:37	SELECT * FROM InformeTecnico ORDER BY transaction.id DESC LIMIT 0, 50000	Error Code: 1054. Unknown column 'transaction.id' in 'order clause'
5	09:09:05	SELECT * FROM InformeTecnico ORDER BY Id_transaction DESC LIMIT 0, 50000	587 row(s) returned