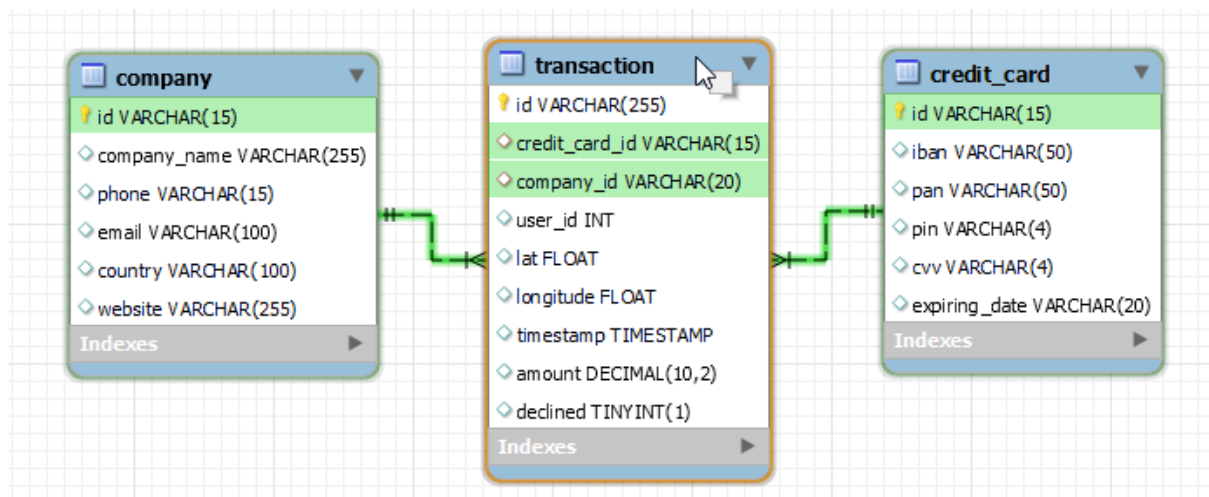


## Tasca S3.01. Nocions bàsiques SQL

### 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.



La tabla credit\_card tiene 275 registros y los siguientes 6 campos:

- id: es la **PK**. Podría llamarse id\_credit\_card para evitar confusiones con el campo id de las tablas transaction y company
- iban: es el International Bank Account Number
- pan: es el 'Personal Account Number' o número de la tarjeta de crédito
- pin: es el número secreto de 4 dígitos
- cvv: es el Valor de Verificación de Tarjeta
- expiring\_date: es la fecha de caducidad de la tarjeta

Relación: entre la tabla credit\_card y la tabla transaction existe una relación de 1 a muchos, lo que quiere decir que por cada credit\_card pueden existir muchas transacciones.

```

4 • USE transactions;
5
6 -- Creo la tabla credit_card
7 • CREATE TABLE IF NOT EXISTS credit_card (
8     id VARCHAR(15) PRIMARY KEY,
9     iban VARCHAR(50),
10    pan VARCHAR(50),
11    pin VARCHAR(4),
12    cvv VARCHAR(4),
13    expiring_date VARCHAR(20)
14 );
15
16
17 • ALTER TABLE transaction
18     ADD FOREIGN KEY (credit_card_id) REFERENCES credit_card(id);
19

```

Output

Action Output

#	Time	Action	Message
1	10:27:41	ALTER TABLE transaction ADD FOREIGN KEY (credit_card_id) REFERENCES credit_card(id)	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0

## Muestro la tabla credit\_card

```

6 -- Creo la tabla credit_card
7 • CREATE TABLE IF NOT EXISTS credit_card (
8     id VARCHAR(15) PRIMARY KEY,
9     iban VARCHAR(50),
10    pan VARCHAR(50),
11    pin VARCHAR(4),
12    cvv VARCHAR(4),
13    expiring_date VARCHAR(20)
14 );
15
16 • ALTER TABLE transaction
17     ADD FOREIGN KEY (credit_card_id) REFERENCES credit_card(id);
18
19 • SELECT * FROM credit_card;

```

Result Grid

	id	iban	pan	pin	cvv	expiring_date
▶	CcU-2938	TR301950312213576817638661	5424465566813633	3257	984	10/30/22
	CcU-2945	DO26854763748537475216568689	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	PT87806228135092429456346	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	CY49087426654774581266832110	511722 924833 2244	9562	867	09/16/22
	CcU-3008	LU507216693616119230	4485744464433884	1856	740	04/05/25

credit\_card 1 x

Output

Action Output

#	Time	Action	Message
1	10:27:41	ALTER TABLE transaction ADD FOREIGN KEY (credit_card_id) REFERENCES credit_card(id)	587 row(s) affected
2	10:33:27	SELECT * FROM credit_card	275 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.

Para actualizar el registro erróneo primero muestro el registro actual

*Antes*

```
20 • SELECT id, iban
21 FROM credit_card
22 where id = 'CcU-2938';
```

Result Grid | Filter Rows:

id	iban
CcU-2938	TR301950312213576817638661
NULL	NULL

para la actualización he utilizado la función update de la tabla credit\_card, con set le indico el dato que quiero actualizar que en este caso es el iban y con el where filtro por el id CcU-2938

*Después*

```
24 • update credit_card
25 set iban = 'R323456312213576817699999'
26 where id = 'CcU-2938';
27
28 • SELECT id, iban
29 FROM credit_card
30 where id = 'CcU-2938';
```

Result Grid | Filter Rows: | Edit:

id	iban
CcU-2938	R323456312213576817699999
NULL	NULL

En la imagen se observa el nuevo valor para el registro.

## EXERCICI 3

En la taula "transaction" ingresa un nou usuari amb la següent informació:

En esta consulta me ha dado un error relacionado con "las foreign key constraint"

Para solucionar esto he introducido en la tabla company la compañía b-9999 que no existía antes

```
48 • INSERT INTO company (id, company_name, phone, email, country, website)
49 VALUES ('b-9999', 'NA', '00 00 00 00', 'na@mail.com', 'NA', 'https://guardian.co.uk/settings');
```

Output			
Action Output			
#	Time	Action	Message
1	10:27:41	ALTER TABLE transaction ADD FOREIGN KEY (credit_card_id) REFERENCES credit_card(id)	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0
2	10:33:27	SELECT * FROM credit_card	275 row(s) returned
3	10:39:11	SELECT id, iban FROM credit_card where id = 'CcU-2938'	1 row(s) returned
4	10:51:02	update credit_card set iban = 'R323456312213576817699999' where id = 'CcU-2938'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
5	10:51:12	SELECT id, iban FROM credit_card where id = 'CcU-2938'	1 row(s) returned
6	10:51:46	INSERT INTO company (id, company_name, phone, email, country, website) VALUES (b-9999, 'NA', '00 00 00 00', 'na@mail.com', 'NA', 'https://guardia...	1 row(s) affected

como el error persistía entonces he buscado en la tabla credit\_card el registro id = 'CcU-9999' y efectivamente no está en dicha tabla:

```
48 • INSERT INTO company (id, company_name, phone, email, country, website)
49 VALUES ('b-9999', 'NA', '00 00 00 00', 'na@mail.com', 'NA', 'https://guardian.co.uk/settings');
50
51 • select * from credit_card
52 where id = 'CcU-9999';
53
```

Result Grid						
Filter Rows:						
	id	iban	pan	pin	cvv	expiring_date
*	NULL	NULL	NULL	NULL	NULL	NULL

credit\_card 4 x

Output			
Action Output			
#	Time	Action	Message
1	11:00:16	select * from credit_card where id = 'CcU-9999'	0 row(s) returned

Después he ingresado el nuevo registro en la tabla credit\_card

```
48 • INSERT INTO company (id, company_name, phone, email, country, website)
49 VALUES ('b-9999', 'NA', '00 00 00 00', 'na@mail.com', 'NA', 'https://guardian.co.uk/settings');
50
51 • select * from credit_card
52 where id = 'CcU-9999';
53
54 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-9999', 'iban-9999', 'pan', 'pin', 'cvv', '00/30/22');
```

Output			
Action Output			
#	Time	Action	Message
1	11:00:16	select * from credit_card where id = 'CcU-9999'	0 row(s) returned
2	11:03:57	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-9999', 'iban-9999', 'pan', 'pin', 'cvv', '00/30/22')	1 row(s) affected

Entonces es en este momento es cuando me permitió ingresar el nuevo usuario en la tabla transactions:

```

48 • INSERT INTO company (id, company_name, phone, email, country, website)
49 VALUES ('b-9999', 'NA', '00 00 00 00', 'na@mail.com', 'NA', 'https://guardian.co.uk/settings');
50
51 • select * from credit_card
52 where id = 'CcU-9999';
53
54 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-9999', 'iban-9999', 'pan','pin', 'cvv', '00/30/22'
55
56 • INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined)
57 VALUES ('108B1D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', '9999', '829.999', '-117.999', '111.11', '0');

```

Output			
Action Output			
#	Time	Action	Message
✓ 1	11:00:16	select * from credit_card where id = 'CcU-9999'	0 row(s) returned
✓ 2	11:03:57	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-9999', 'iban-9999', 'pan','pin', 'cvv', '00/30/22')	1 row(s) affected
✓ 3	11:04:51	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES ('108B1D1D-5B23-A76C-55EF-C568E49A...	1 row(s) affected

y para comprobar que el ingreso se ha realizado correctamente he hecho un select filtrando por el id del nuevo usuario.

```

48 • INSERT INTO company (id, company_name, phone, email, country, website)
49 VALUES ('b-9999', 'NA', '00 00 00 00', 'na@mail.com', 'NA', 'https://guardian.co.uk/settings');
50
51 • select * from credit_card
52 where id = 'CcU-9999';
53
54 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-9999', 'iban-9999', 'pan','pin', 'cvv', '00/30/
55
56 • INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined)
57 VALUES ('108B1D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', '9999', '829.999', '-117.999', '111.11', '0');
58
59 • SELECT *
60 FROM transaction
61 WHERE id = '108B1D1D-5B23-A76C-55EF-C568E49A99DD';
62

```

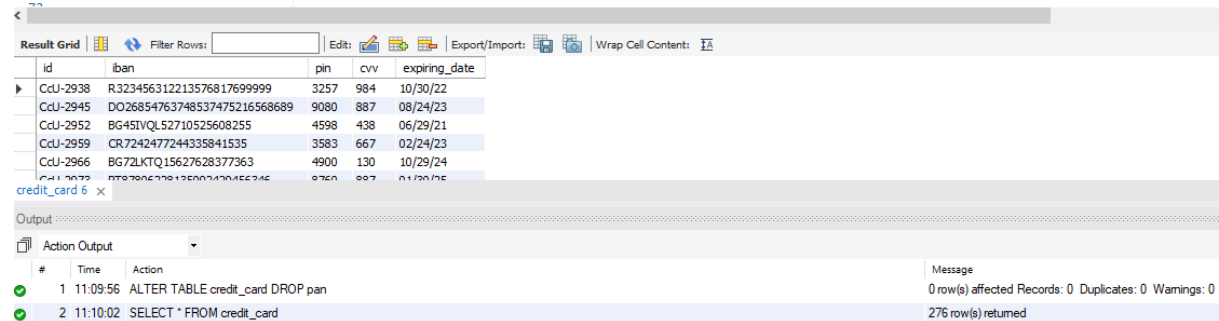
Result Grid									
Filter Rows:									
Edit: Export/Import: Wrap Cell Content: I A									
	id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
▶	108B1D1D-5B23-A76C-55EF-C568E49A99DD	CcU-9999	b-9999	9999	829.999	-117.999	NULL	111.11	0
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

transaction 5 ×			
Output			
Action Output			
#	Time	Action	Message
✓ 1	11:00:16	select * from credit_card where id = 'CcU-9999'	0 row(s) returned
✓ 2	11:03:57	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-9999', 'iban-9999', 'pan','pin', 'cvv', '00/30/22')	1 row(s) affected
✓ 3	11:04:51	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES ('108B1D1D-5B23-A76C-55EF-C568E49A...	1 row(s) affected
✓ 4	11:06:14	SELECT * FROM transaction WHERE id = '108B1D1D-5B23-A76C-55EF-C568E49A99DD'	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.

```
68 • ALTER TABLE credit_card DROP pan;
69
70 • SELECT *
71 FROM credit_card;
```



Para eliminar la columna pan de la tabla credit\_card he utilizado la función DROP que me permite eliminar un campo de una tabla.

## NIVELL 2

### EXERCICI 1

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

```
80 • DELETE FROM transaction WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02';
81
```



Para eliminar el registro he utilizado la función DELETE que me permite eliminar un registro determinado y también he utilizado el WHERE para filtrar el registro concreto que quiero eliminar. El where es muy importante ya que si no lo uso podría borrar toda la tabla.

### 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.

```

89 • CREATE VIEW `vistamarketing` AS
90 SELECT company_name as nombre_compañia, phone as teléfono, country as país, ROUND(AVG(amount),2) as media_compra
91 FROM transaction
92 JOIN company ON transaction.company_id = company.id
93 WHERE declined = '0'
94 GROUP BY company_id
95 ORDER BY 4 DESC;
96
97 • SELECT * FROM transactions.vistamarketing;

```

Result Grid				
Filter Rows: <input type="text"/>				
Export: <input type="button" value="Export"/>				
Wrap Cell Content: <input type="button" value="Wrap"/>				
	nombre_compañia	teléfono	país	media_compra
▶	Eget Ipsum Ltd	03 67 44 56 72	United States	481.86
	Sed Id Limited	07 28 18 18 13	United States	477.51
	Neque Tellus Incorporated	04 43 18 34 19	Ireland	477.10
	Nunc Sit Incorporated	07 28 42 63 63	Norway	461.83
	Non Magna LLC	06 71 73 13 17	United Kingdom	458.74
	Morosa Mauris Egesta Inc	00 38 53 75 61	Netherlands	451.70

vistamarketing 7 x

Output			
Action Output			
#	Time	Action	Message
✓ 1	11:21:08	CREATE VIEW `vistamarketing` AS SELECT company_name as nombre_compañia, phone as teléfono, country as país, ROUND(AVG(amount),2) as me...	0 row(s) affected
✓ 2	11:21:13	SELECT * FROM transactions.vistamarketing	101 row(s) returned

Para crear la vista he utilizado la función CREATE VIEW y dentro de la consulta he usado declined =0 para filtrar las compañías que tienen compras que han sido realizadas y no rechazadas.

Para mostrar los datos de la vista he hecho un select de vistamarketing.

### EXERCICI 3

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

```

101 • SELECT * FROM transactions.vistamarketing
102 WHERE país = 'Germany';
103

```

Result Grid				
Filter Rows: <input type="text"/>				
Export: <input type="button" value="Export"/>				
Wrap Cell Content: <input type="button" value="Wrap"/>				
	nombre_compañia	teléfono	país	media_compra
▶	Ac Industries	09 34 65 40 60	Germany	396.15
	Auctor Mauris Corp.	05 62 87 14 41	Germany	308.99
	Ac Fermentum Incorporated	06 85 56 52 33	Germany	293.57
	Aliquam PC	01 45 73 52 16	Germany	280.34
	Rutrum Non Inc.	02 66 31 61 09	Germany	266.90
	Nunc Interdum Incorporated	05 18 15 48 13	Germany	242.95
	Convallis In Incorporated	06 66 57 29 50	Germany	60.99
	Augue Foundation	06 88 43 15 63	Germany	15.05

vistamarketing 8 x

Output			
Action Output			
#	Time	Action	Message
✓ 1	11:25:48	SELECT * FROM transactions.vistamarketing WHERE país = 'Germany'	8 row(s) returned

Para esta consulta he utilizado el where para filtrar en la vistamarketing las compañías que residen en Alemania.

## 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:

Lo paso que he seguido en este ejercicio son los siguientes:

1. Descargar los archivos: estructura\_datos\_user y datos\_introducir\_user
2. Abrir los archivos en MySQL
3. Ejecutar estructura\_datos\_user

```
4 • CREATE INDEX idx_user_id ON transaction(user_id);
5
6 • CREATE TABLE IF NOT EXISTS user (
7     id INT PRIMARY KEY,
8     name VARCHAR(100),
9     surname VARCHAR(100),
10    phone VARCHAR(150),
11    email VARCHAR(150),
12    birth_date VARCHAR(100),
13    country VARCHAR(150),
14    city VARCHAR(150),
15    postal_code VARCHAR(100),
16    address VARCHAR(255),
17    FOREIGN KEY(id) REFERENCES transaction(user_id)
18 );
```

Output

Action Output

#	Time	Action	Message
✓ 1	12:26:45	DROP TABLE user	0 row(s) affected
✓ 2	12:27:03	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), FOREIGN KEY(id) REFERENCES transaction(user_id) )	0 row(s) affected

4. Ejecutar datos\_introducir\_user



```
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 (
5 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
6 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
7 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
8 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
9 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
10 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
11 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
12 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
13 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
14 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
15 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
16 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
17 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
18 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (

"1", "Zeus", "Gamble", "1-282-581-0551", "interdum.enim@protonmail.edu";
"2", "Garrett", "Mcconnell", "(718) 257-2412", "integer.vitae.nibh@protonmail.org";
"3", "Ciaran", "Harrison", "(522) 598-1365", "interdum.feugiat@aol.org";
"4", "Howard", "Stafford", "1-411-740-3269", "ornare.egestas@icloud.edu";
"5", "Hayfa", "Pierce", "1-554-541-2077", "et.malesuada.fames@hotmail.org";
"6", "Joel", "Tyson", "(718) 288-8020", "gravidam.nunc.sed@yahoo.ca";
"7", "Rafael", "Jimenez", "(817) 689-0478", "eget@outlook.ca";
"8", "Nissim", "Franks", "(692) 157-3469", "egestas.aliqum.fringilla@google.com";
"9", "Mannix", "Mcclain", "(590) 883-2184", "aliquam.nisl@outlook.com";
"10", "Robert", "Mccarthy", "(324) 746-6771", "fermentum@protonmail.com";
"11", "Joan", "Baird", "(981) 429-8106", "et@outlook.net";
"12", "Benedict", "Wheeler", "1-515-824-2855", "tincidunt.donec.vitae@hugoboss.com";
"13", "Allegra", "Stanton", "1-927-753-6488", "proin.eget@protonmail.ca";
"14", "Sara", "Flynn", "1-311-646-9333", "integer@outlook.net";
"15", "Noelani", "Patrick", "1-723-488-5894", "sem.magna@google.com";

Output
Action Output
# Time Action Message
258 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "258", "Blaze", "Daniel", "087-870-... 1 row(s) affected
259 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "259", "Lane", "Paul", "044-254-6... 1 row(s) affected
260 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "260", "Heather", "Bucks", "053-58... 1 row(s) affected
261 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "261", "Stone", "Robbins", "076-2... 1 row(s) affected
262 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "262", "Slade", "Downs", "034-22... 1 row(s) affected
263 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "263", "Grace", "Rowe", "071-756... 1 row(s) affected
264 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "264", "Violet", "Weber", "019-661... 1 row(s) affected
265 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "265", "Brett", "Kirby", "076-166-2... 1 row(s) affected
266 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "266", "Ima", "Hendricks", "065-95... 1 row(s) affected
267 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "267", "Keiko", "Guen", "034-741... 1 row(s) affected
268 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "268", "Chloe", "Keth", "022-178... 1 row(s) affected
269 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "269", "Alko", "Chaney", "026-660... 1 row(s) affected
270 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "270", "Ocean", "Nelson", "079-48... 1 row(s) affected
271 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "271", "Clark", "Olson", "029-086... 1 row(s) affected
272 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "272", "Haley", "Fitzpatrick", "055-... 1 row(s) affected
273 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "273", "Elton", "Roberson", "096-3... 1 row(s) affected
274 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "274", "Leandra", "Cherry", "089-2... 1 row(s) affected
275 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "275", "Hedwig", "Gilbert", "064-2... 1 row(s) affected
276 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "276", "Hilary", "Ferguson", "060-7... 1 row(s) affected
277 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "277", "Jameson", "Hunt", "024-73... 1 row(s) affected
278 12:28:00 INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "278", "Kenyon", "Hartman", "082-... 1 row(s) affected
279 12:28:00 SET foreign_key_checks = 1 0 row(s) affected
```

## 5. muestro la tabla user para ver sus campos y registros

```
95 • SELECT *
96 FROM user;
97
```

id	name	surname	phone	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	gravidam.nunc.sed@yahoo.ca	Oct 15, 1989	United States	Nashville	96838	888-2799 Amet Street
7	Rafael	Jimenez	(817) 689-0478	eget@outlook.ca	Dec 4, 1981	United States	Hillsboro	29874	8627 Malesuada Rd.
8	Nissim	Franks	(692) 157-3469	egestas.aliqum.fringilla@google.ca	Aug 1, 1993	United States	Jackson	61750	Ap #251-7144 Integer St.
9	Mannix	Mcclain	(590) 883-2184	aliquam.nisl@outlook.com	Jan 24, 1987	United States	Richmond	35987	647-3080 Lacus. St.
10	Robert	Mccarthy	(324) 746-6771	fermentum@protonmail.com	Apr 30, 1984	United States	Eugene	85526	P.O. Box 773, 3594 Ornare St.

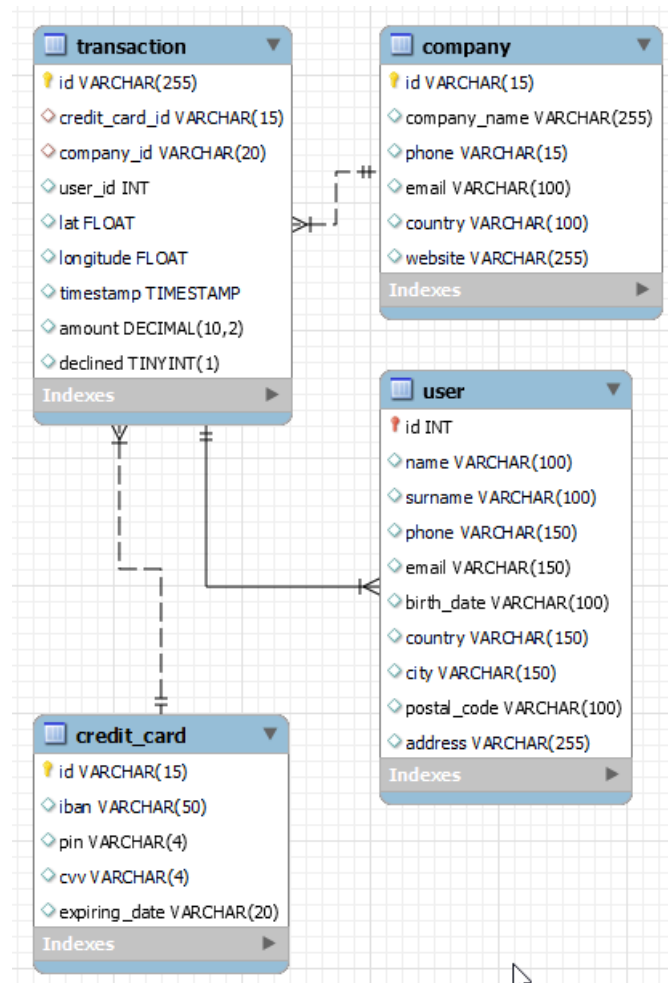
user 21 x

Output

Action Output

#	Time	Action	Message
1	12:29:32	SELECT * FROM user	275 row(s) returned

6. Observo el modelo y detecto que la dirección de la relación entre las tablas transaction y la tabla user está invertida.



7. Al intentar hacer el cambio de la dirección de la relación entre las tablas transaction y user me ha salido el siguiente error:

Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`transactions`.`#sql-1b7c\_a`, CONSTRAINT `transaction\_ibfk\_3` FOREIGN KEY (`user\_id`) REFERENCES `user` (`id`))

Para resolver esto he buscado el id en la tabla transaction que no se encuentra en la tabla user:

```

112 • SELECT user_id
113 FROM transaction
114 WHERE user_id NOT IN (SELECT id FROM user);
115

```

Result Grid	Filter Rows:	Export:	Wrap
user_id			
9999			

he corroborado que el id 9999 de la tabla user no existe (lo había agregado en el ejercicio 3 del nivel 1)

```

115 • SELECT *
116 FROM user
117 WHERE id = '9999'; # corroboro que id = 9999 no se encuentrs en user

```

## 8. Para solucionarlo he creado un nuevo registro en la tabla user id = 9999

```

119 # agrego el registro con id = 9999 en la tabla user
120 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "9999", "xx", "xx", "xx", "xx", "xx", "xx", "xx", "xx", "xx");
121

```

#	Time	Action	Message
1	11:57:18	SELECT * FROM user WHERE id = '9999'	0 row(s) returned
2	11:58:08	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "9999", "xx", "xx", "xx", "xx", "xx", "xx", "xx", "xx", "xx");	1 row(s) affected

#	Time	Action	Message
1	11:57:18	SELECT * FROM user WHERE id = '9999'	0 row(s) returned
2	11:58:08	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "9999", "xx", "xx", "xx", "xx", "xx", "xx", "xx", "xx", "xx");	1 row(s) affected
3	12:00:29	ALTER TABLE transaction ADD FOREIGN KEY (user_id) REFERENCES user(id)	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0

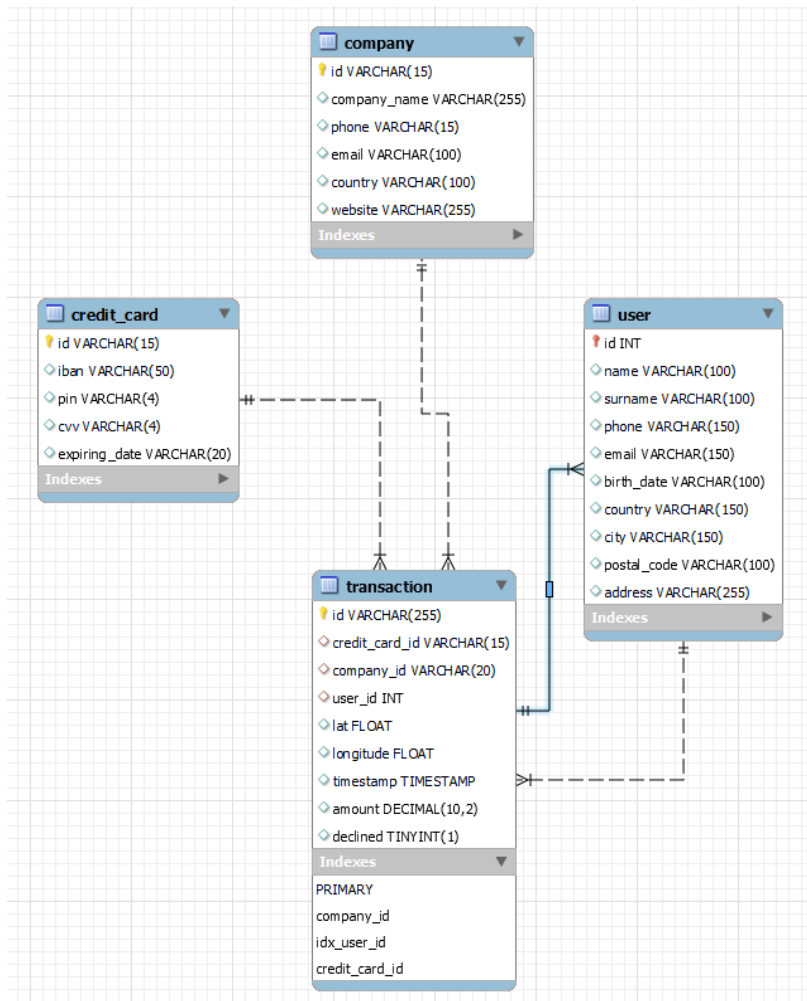
## 9. Creo la relación en la dirección correcta entre las tablas transaction y user

```

125 • ALTER TABLE transaction
126 ADD FOREIGN KEY (user_id) REFERENCES user(id);

```

#	Time	Action	Message
1	11:57:18	SELECT * FROM user WHERE id = '9999'	0 row(s) returned
2	11:58:08	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "9999", "xx", "xx", "xx", "xx", "xx", "xx", "xx", "xx", "xx");	1 row(s) affected
3	12:00:29	ALTER TABLE transaction ADD FOREIGN KEY (user_id) REFERENCES user(id)	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0



10. Borro la relación inicial con la dirección incorrecta entre las tablas transaction y user

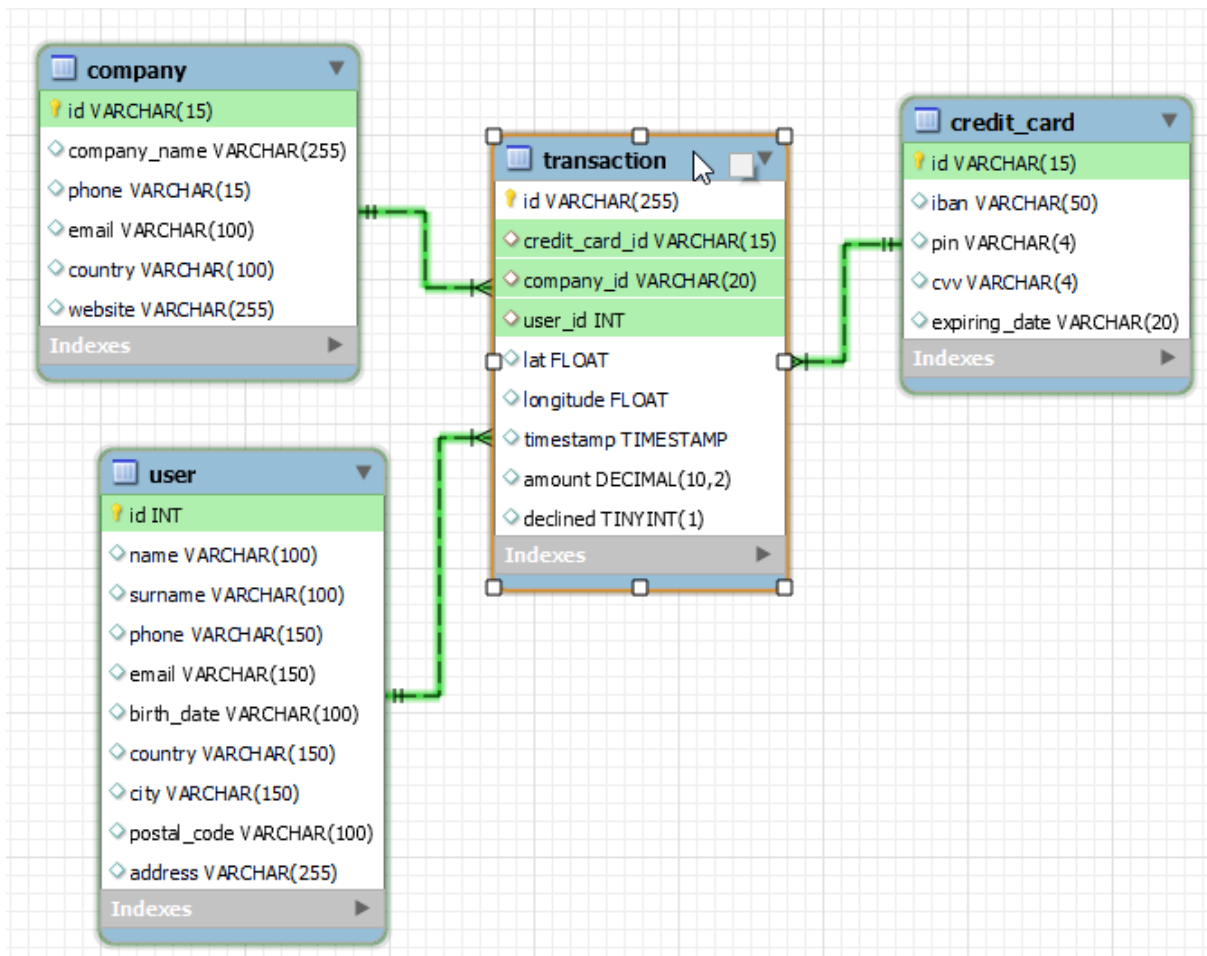
```

130 ALTER TABLE user
131 DROP FOREIGN KEY user_ibfk_1;
132

```

#	Time	Action	Message
1	11:57:18	SELECT * FROM user WHERE id = '9999'	0 row(s) returned
2	11:58:08	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ('9999', 'xx', 'xx', 'xx', 'xx', 'xx', 'xx', 'xx', 'xx', 'xx')	1 row(s) affected
3	12:00:29	ALTER TABLE transaction ADD FOREIGN KEY (user_id) REFERENCES user(id)	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0
4	12:01:57	ALTER TABLE user DROP FOREIGN KEY user_ibfk_1	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

11. Observo el diagrama y defino que a partir de este diagrama, empiezo a comparar las diferencias con respecto al diagrama del ejercicio 1 nivel 3:



12. Realizo los siguientes cambios en las tablas credit\_card:

- He agregado el campo “fecha actual” de tipo DATE
- He cambiado de tipo de datos de los campos

```
134 • ALTER TABLE credit_card
135     ADD fecha_actual DATE; # Añado el campo fecha_actual
136
137 • ALTER TABLE credit_card
138     MODIFY id VARCHAR(20) not null; # cambio del tipo de datos de id
139
140
```

Output

#	Time	Action
✓ 1	12:17:07	ALTER TABLE credit_card ADD fecha_actual DATE
✓ 2	12:21:55	ALTER TABLE credit_card MODIFY id VARCHAR(20) not null

13. En la tabla company he eliminado el campo website

```
140 • ALTER TABLE company
141 DROP COLUMN website; # Elimino el campo website
142
```

Output

Action Output

#	Time	Action
✓ 1	12:24:37	ALTER TABLE company DROP COLUMN website

14. En la tabla user he realizado los siguientes cambios:

- se ha cambiado el nombre de la tabla de user a “data\_user”
- se ha cambiado el nombre del campo email por “personal\_email”

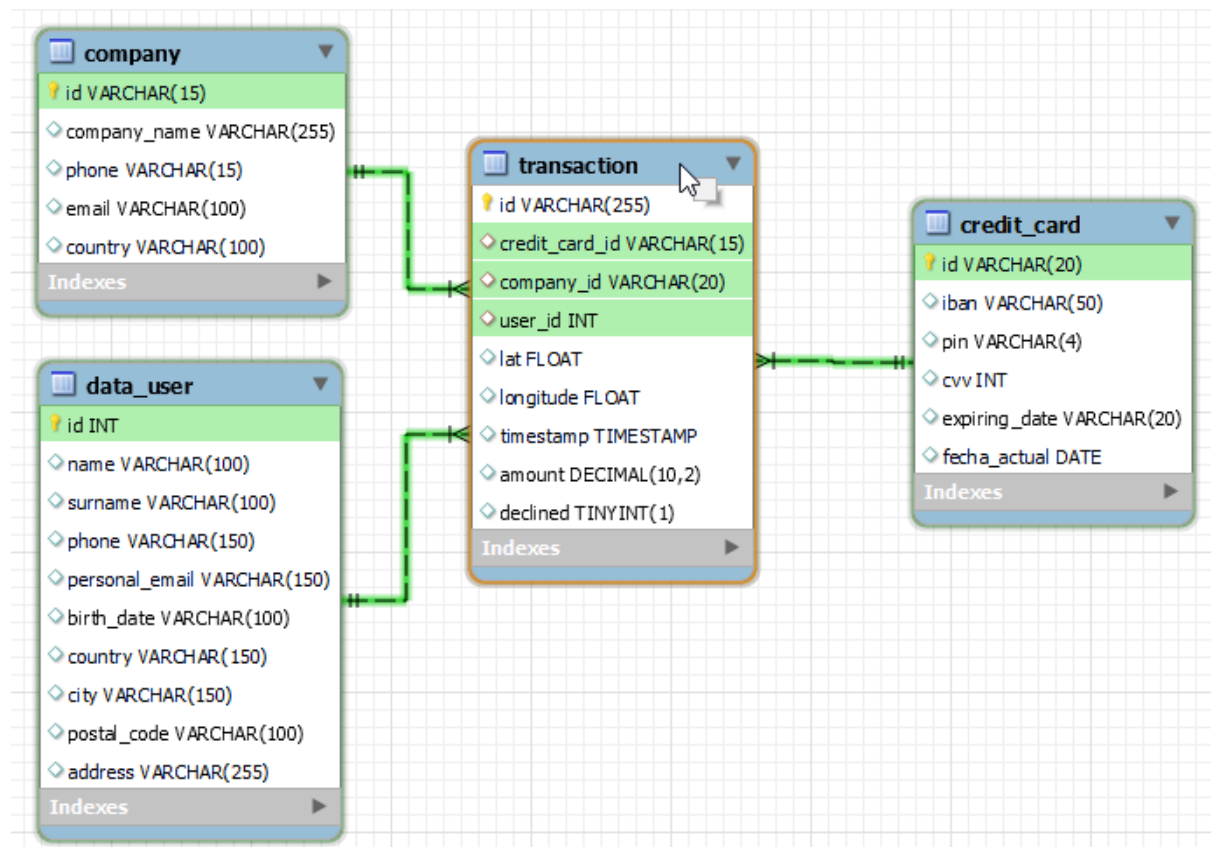
```
147 • RENAME TABLE user to data_user; # cambio de nombre
148
149 • ALTER TABLE data_user
150 CHANGE email personal_email VARCHAR(150);
151
```

Output

Action Output

#	Time	Action
✓ 1	12:37:20	ALTER TABLE data_user CHANGE email personal_email VARCHAR(150)

## 15. 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.

```

156 • CREATE VIEW `informetecnico` AS
157 SELECT transaction.id as id_transacción, name as nombre_usuario, surname as apellido_usuario, iban, company_name as nombre_com
158 FROM transaction
159 JOIN data_user ON transaction.user_id = data_user.id
160 JOIN company ON transaction.company_id = company.id
161 JOIN credit_card ON transaction.credit_card_id = credit_card.id
162 ORDER BY transaction.id DESC;
163
164 • SELECT * FROM transactions.informetecnico;

```

Result Grid					
Filter Rows: <input type="text"/>					
Export: <input type="button" value=""/>					
Wrap Cell Content: <input type="button" value=""/>					
	id_transacción	nombre_usuario	apellido_usuario	iban	nombre_compañía
▶	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
	FD89D518-AE8D-77DC-E450-B8083FBD3187	Hilda	Levy	LT053237077744561475	Malesuada PC
	FD2E8957-414B-BEEC-E9AD-59AA7A8A6290	Hedwig	Gilbert	GE84848451582810541526	Neque Tellus Imperdiet Corp.

informetecnico 12 x

Output				
Action Output				
#	Time	Action	Message	
✓ 1	13:11:21	CREATE VIEW `informetecnico` AS SELECT transaction.id as id_transacción, name as nombre_usuario, surname as apellido_usuario, iban, company_na...	0 row(s) affected	
✓ 2	13:11:28	SELECT * FROM transactions.informetecnico	587 row(s) returned	