

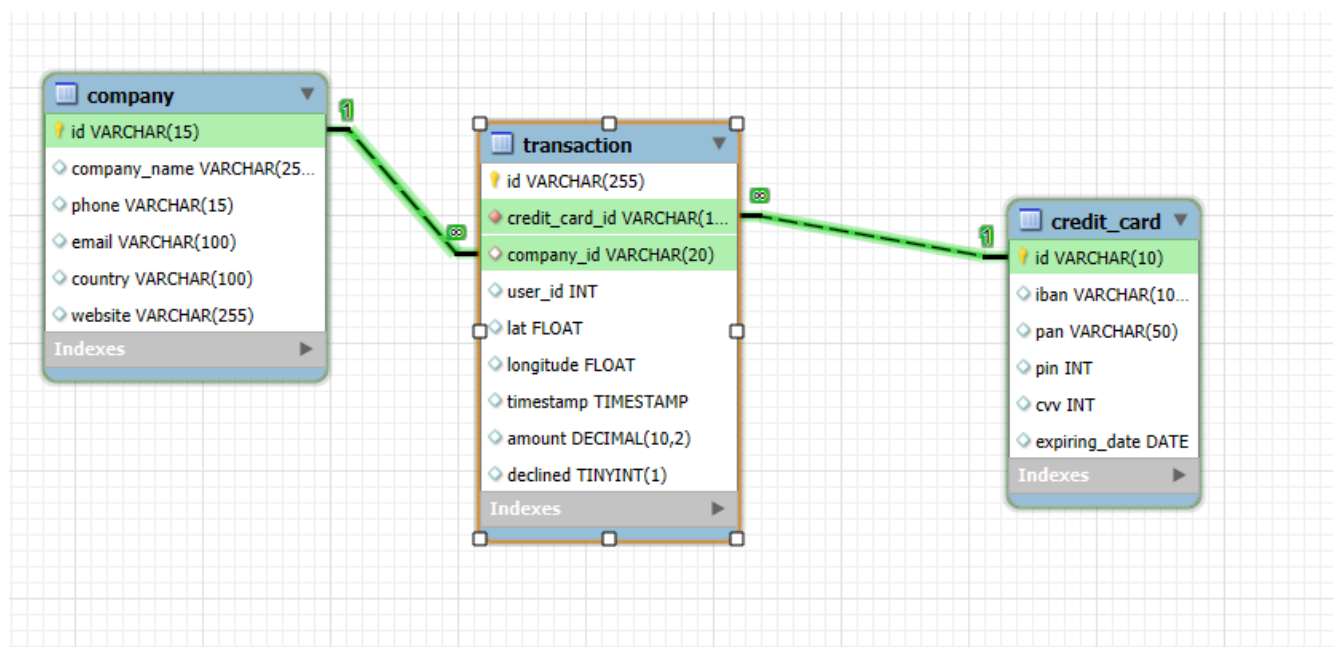


Nivell 1

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.

```
CREATE TABLE IF NOT EXISTS credit_card (  
  id VARCHAR(10) NOT NULL PRIMARY KEY,  
  iban VARCHAR(100),  
  pan VARCHAR(50),  
  pin INT,  
  cvv INT,  
  expiring_date VARCHAR(10),  
  CONSTRAINT FK_transaction_creditcard  
  FOREIGN KEY (ID) REFERENCES transaction(credit_card_id)  
  ON UPDATE CASCADE  
  ON DELETE SET NULL  
);
```



El digrama formado es un modelo estrella donde las tablas company y credit_card se relacionan directamente con la tabla de transaction. Las tablas credit_card y company son dimensiones, company con la informacion de cada empresa y credit_card con la informacion de cada tarjeta de credito. Ambos tienen relacion de 1 a muchos con transaction. También he tenido que actualizar el campo expiring_date para que muestre la fecha correctamente

9	•	UPDATE credit_card
10		SET expiring_date = STR_TO_DATE(expiring_date, '%m/%d/%y');
11	•	ALTER TABLE credit_card
12		MODIFY COLUMN expiring_date DATE;
13		
14		

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

Fetch rows:

id	iban	pan	pin	cvv	expiring_date
CcU-3925	NL40SLES4851032267	448 56746 54875 257	6038	485	2020-03-18
CcU-4618	CY403262761606781187503...	5267481137212858	1606	377	2020-03-24
CcU-3666	DE35385647561242046858	4556 7844 3557 4824	4798	930	2020-03-26
CcU-4226	CR73483061667272971113	492924 3966325775	3580	659	2020-04-13
CcU-3645	PS774972273648295735416...	345444856288771	3688	415	2020-04-17

credit_card 5

ApplyRevert

Output

Action Output

#	Time	Action	Message	Duration / Fetch
5042	12:35:04	select * from credit_card	5000 row(s) returned	0.000 sec / 0.000 sec
5043	12:35:14	UPDATE credit_card SET expiring_date = STR_TO_DATE(expiring_date, '%m/%d/%y')	5000 row(s) affected Rows matched: 5000 Changed: 5000 Warnings: 0	0.250 sec
5044	12:35:19	select * from credit_card	5000 row(s) returned	0.016 sec / 0.000 sec
5045	12:35:56	ALTER TABLE credit_card MODIFY COLUMN expiring_date DATE	5000 row(s) affected Records: 5000 Duplicates: 0 Warnings: 0	0.172 sec
5046	12:36:10	select * from credit_card	5000 row(s) returned	0.015 sec / 0.000 sec

Exercici 2

El departament de Recursos Humans ha identificat un error en el número de compte associat a la targeta de crèdit amb ID CcU-2938. La informació que ha de mostrar-se per a aquest registre és: TR323456312213576817699999. Recorda mostrar que el canvi es va realitzar.

21	-- EJERCICIO 2 --
22	-- El departament de Recursos Humans ha identificat un error en el número de compte associat a la targeta de crèdit amb ID CcU-2938.
23	-- La informació que ha de mostrar-se per a aquest registre és: TR323456312213576817699999. Recorda mostrar que el canvi es va realitzar.
24	
25	UPDATE credit_card
26	SET iban = "TR323456312213576817699999"
27	WHERE id = "CcU-2938" ;
28	
29	• select * from credit_card
30	where id = "CcU-2938";
31	
32	
33	

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

id	iban	pan	pin	cvv	expiring_date
CcU-2938	TR323456312213576817699999	5424465566813633	3257	984	2022-10-30
	NULL	NULL	NULL	NULL	NULL

credit_card 14

ApplyRevert

Output

Action Output

#	Time	Action	Message	Duration / Fetch
27	13:31:21	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_creditcard FOREIGN KEY (credit_card_id) REFERENCE...	Error Code: 1826. Duplicate foreign key constraint name 'fk_transaction_creditcard'	0.000 sec
28	13:32:03	ALTER TABLE transaction DROP FOREIGN KEY fk_transaction_creditcard	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
29	13:32:09	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_creditcard FOREIGN KEY (credit_card_id) REFERENCE...	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings: 0	3.750 sec
30	13:40:07	UPDATE credit_card SET iban = "TR323456312213576817699999" WHERE id = "CcU-2938"	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.015 sec
31	13:41:07	select * from credit_card where id = "CcU-2938"	1 row(s) returned	0.000 sec / 0.000 sec

Exercici 3

En la taula "transaction" ingressa una nova transacció 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

En este ejercicio no sabia bien si deberia agregar la informacion tambien a las tablas credit_card y company_id ya que son FK y estos valores no se encuentran ingresados en sus respectivas tablas. Es decir el credit_card_id Ccu-9999 no existe en la tabla credit_card y el company_id b-9999 no existe en la tabla company. Por lo tanto no sabia si habia que crear esos registros o saltarme la comprobacion, ya que el ejercicio no dice nada de crear esos registros en las otras tablas, por lo que en este caso use SET FOREIGN_KEY_CHECKS = 0, aunque luego vi que no estaba permitido. Mas adelante en el ejercicio 1 del nivel 3 he cambiado esos valores a null, aunque no tengo claro todavia si deberia haber creado los registros en las demas tablas o no.

```
• SET FOREIGN_KEY_CHECKS = 0;

• INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined, timestamp)
VALUES ("108B1D1D-5B23-A76C-55EF-C568E49A99DD", "CcU-9999", "b-9999", "9999", "829.999", "-117.999", "111.11", "0", NOW())
);

• SET FOREIGN_KEY_CHECKS = 1;
```

Exercici 4

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

39 -- EJERCICIO 4 --
40 -- Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Recordra mostrar el canvi realitzat.
41
42 • ALTER TABLE credit_card
43 DROP COLUMN pan;
44
45 • select * from credit_card

id	iban	pin	cvv	expiring_date
CcS-4857	XX4857591835292505850771	1819	467	09/27/25
CcS-4858	XX8581768137002436094025	3964	817	12/28/28
CcS-4859	XX7826930491423553609370	4983	277	11/26/26
CcS-4860	XX5559590368835304645299	6876	661	07/27/27
CcS-4861	XX2035182877195191627307	5710	398	04/25/26

credit_card 13 x

Output

#	Time	Action	Message	Duration / Fetch
45	11:29:24	SELECT * FROM transactions.transaction	100001 row(s) returned	0.016 sec / 0.328 sec
46	11:29:25	select * from credit_card	5000 row(s) returned	0.000 sec / 0.000 sec
47	11:29:25	select * from company	100 row(s) returned	0.000 sec / 0.000 sec
48	11:55:18	ALTER TABLE credit_card DROP COLUMN pan	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.031 sec
49	11:55:31	select * from credit_card	5000 row(s) returned	0.000 sec / 0.015 sec

Exercici 1

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

```
46 -- NIVEL 2 --
47 -- EJERCICIO 1--
48 -- Elimina de la taula transaction el registre amb ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD de la base de dades.
49
50 • DELETE FROM transaction
51 WHERE id = "000447FE-B650-4DCF-85DE-C7ED0EE1CAAD";
52
53 • select * from transaction where id = "000447FE-B650-4DCF-85DE-C7ED0EE1CAAD"
```

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

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

transaction 1 x

Apply

Revert

Output

Action Output

#	Time	Action	Message	Duration / Fetch
35	12:39:53	ALTER TABLE transaction ADD CONSTRAINT fk_transaction_creditcard FOREIGN KEY (credit_card_id) RE...	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('transactions', '#sql-11a0_6...	0.359 sec
36	12:40:22	SELECT t.company_id, COUNT(*) AS cnt FROM transaction t LEFT JOIN company c ON t.company_id = c.id ...	0 row(s) returned	0.079 sec / 0.000 sec
37	12:40:22	SELECT t.credit_card_id, COUNT(*) AS cnt FROM transaction t LEFT JOIN credit_card cc ON t.credit_card_id = cc.id ...	1 row(s) returned	0.109 sec / 0.000 sec
38	12:42:43	DELETE FROM transaction WHERE id = "000447FE-B650-4DCF-85DE-C7ED0EE1CAAD"	1 row(s) affected	0.031 sec
39	12:43:06	select * from transaction where id = "000447FE-B650-4DCF-85DE-C7ED0EE1CAAD"	0 row(s) returned	0.000 sec / 0.000 sec

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.

```
53 -- EJERCICIO 2 --
54 -- La secció de màrqueting desitja tenir accés a informació específica per a realitzar anàlisi i estratègies efectives.
55 -- S'ha sol·licitat crear una vista que proporcioni detalls clau sobre les companyies i les seves transaccions.
56 -- Serà necessària que creïs una vista anomenada VistaMarketing que contingui la següent informació: Nom de la companyia. Telèfon de contacte.
57 -- País de residència. Mitjana de compra realitzat per cada companyia. Presenta la vista creada, ordenant les dades de major a menor mitjana de compra.
58
59 • CREATE VIEW VistaMarketing as
60 SELECT company_name, phone, country, AVG(amount) as media_compras
61 FROM transaction JOIN company ON company_id = company.id
62 GROUP BY company_id
63 ORDER BY media_compras DESC;
64
65 SELECT * FROM VistaMarketing
66
```

Result Grid

Filter Rows:

Exports:

Wrap Cell Content:

company_name	phone	country	media_compras
Ac Fermentum Incorporated	06 85 56 52 33	Germany	284.867160
Pretium Neque Corp.	07 77 48 55 28	Australia	276.158330
Urna Convallis Associates	06 01 24 77 04	United States	274.235011
At Associates	09 56 61 10 65	New Zealand	272.214870
Metus Vitae Associates	08 25 44 40 66	Australia	270.080965

VistaMarketing 2 x

Read Only

Output

Action Output

#	Time	Action	Message	Duration / Fetch
51	12:54:43	drop view VistaMarketing	0 row(s) affected	0.016 sec
52	12:56:01	select * from company	100 row(s) returned	0.000 sec / 0.000 sec
53	12:56:09	SELECT * FROM transactions.transaction	100000 row(s) returned	0.015 sec / 0.360 sec
54	12:56:31	CREATE VIEW VistaMarketing as SELECT company_name, phone, country, AVG(amount) as media_compras...	0 row(s) affected	0.031 sec
55	12:56:34	SELECT * FROM VistaMarketing	100 row(s) returned	0.687 sec / 0.000 sec

Exercici 3

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

```
65 -- EJERCICIO 3 --
66 -- Filtra la vista VistaMarketing per a mostrar només les companyies que tenen el seu país de residència en "Germany" --
67
68 • CREATE OR REPLACE VIEW VistaMarketing AS
69 SELECT company_name, phone, country, AVG(amount) as media_compras
70 FROM transaction JOIN company ON company_id = company.id
71 WHERE country = "Germany"
72 GROUP BY company_id
73 ORDER BY media_compras DESC;
74
75 • select * from vistamarketing;
76
77
```

Result Grid

Filter Rows:

Export:

Wrap Cell Contents:

company_name	phone	country	media_compras
Ac Fermentum Incorporated	06 85 56 52 33	Germany	284.867160
Nunc Interdum Incorporated	05 18 15 48 13	Germany	259.319156
Convallis In Incorporated	06 66 57 29 50	Germany	257.745376
Ac Industries	09 34 65 40 60	Germany	255.147288
Rutrum Non Inc.	02 66 31 61 09	Germany	255.136927

vistamarketing 3 x

Read Only

Output

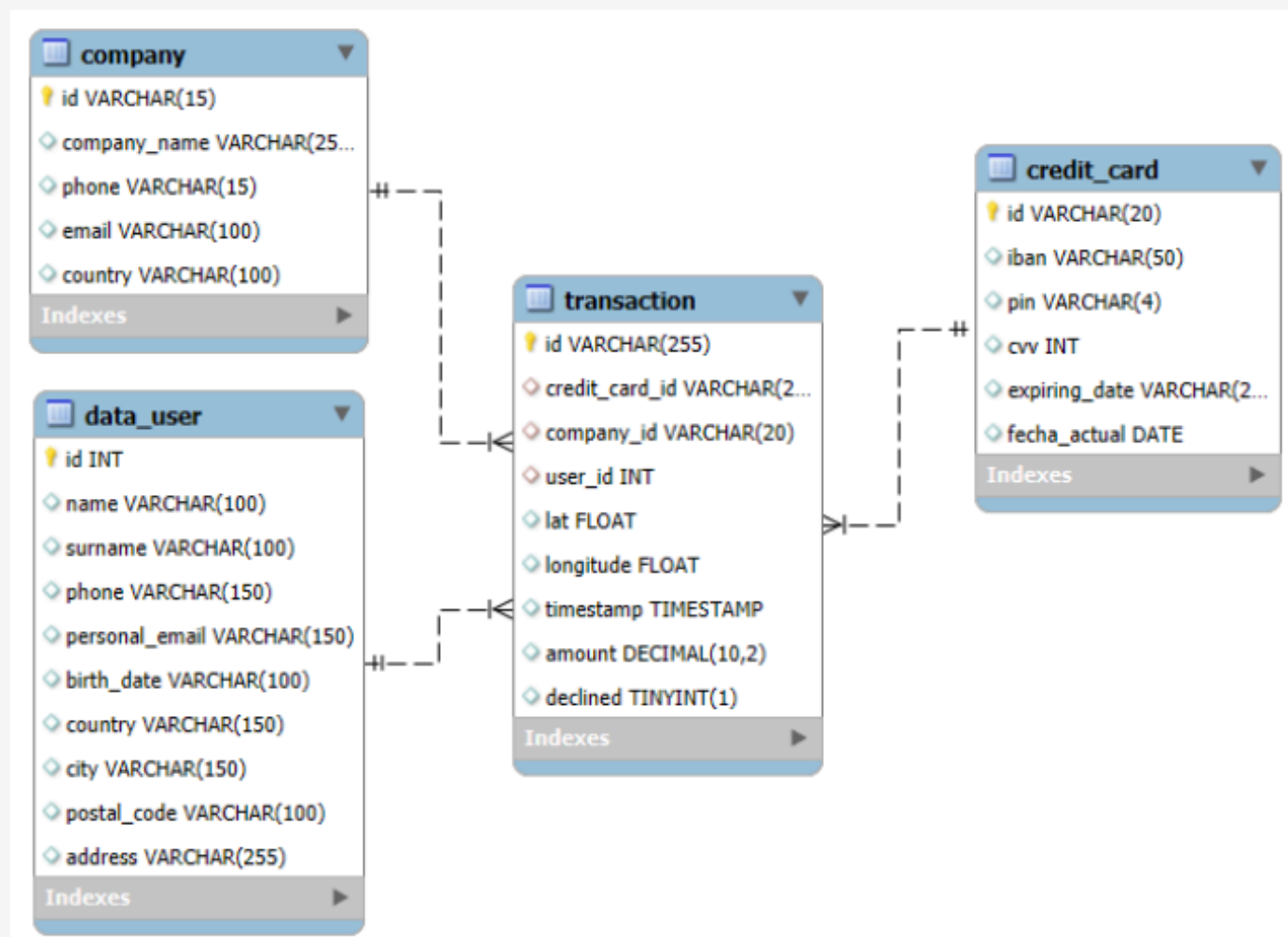
Action Output

#	Time	Action	Message	Duration / Fetch
53	12:56:09	SELECT * FROM transactions.transaction	100000 row(s) returned	0.015 sec / 0.360 sec
54	12:56:31	CREATE VIEW VistaMarketing as SELECT company_name, phone, country, AVG(amount) as media_compra...	0 row(s) affected	0.031 sec
55	12:56:34	SELECT * FROM VistaMarketing	100 row(s) returned	0.687 sec / 0.000 sec
56	13:00:32	CREATE OR REPLACE VIEW VistaMarketing AS SELECT company_name, phone, country, AVG(amount) as...	0 row(s) affected	0.031 sec
57	13:00:46	select * from vistamarketing	8 row(s) returned	0.141 sec / 0.000 sec

☆☆☆ 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 primero fue crear la tabla user como ha sido proporcionada en el ejercicio e insertar los datos.

```
1  CREATE TABLE IF NOT EXISTS user (  
2      id CHAR(10) PRIMARY KEY,  
3      name VARCHAR(100),  
4      surname VARCHAR(100),  
5      phone VARCHAR(150),  
6      email VARCHAR(150),  
7      birth_date VARCHAR(100),  
8      country VARCHAR(150),  
9      city VARCHAR(150),  
10     postal_code VARCHAR(100),  
11     address VARCHAR(255)  
12 );  
13
```

Luego compare mi diagrama con el diagrama dado en el ejercicio y las primeras observaciones y modificaciones que hice fueron las siguientes:

- De la tabla COMPANY: eliminar la columna WEBSITE
- De la tabla USER: cambiar nombre de la tabla a DATA_USER, cambiar formato de ID a (INT), cambiar nombre de la columna EMAIL a PERSONAL_EMAIL
- De la tabla CREDIT_CARD: cambiar formato ID a VARCHAR(20), cambiar formato IBAN a VARCHAR(50), cambiar formato PIN a VARCHAR(4), cambiar formato expiring_date a VARCHAR(20) y agregar FECHA_ACTUAL DATE
- De la tabla TRANSACTION: cambiar formato CREDIT_CARD_ID a VARCHAR(20), cambiar formato COMPANY_ID a VARCHAR(20)

Por ultimo establecer los y ver que los PK y los FK esten correctos y sino establecerlos.

```

89      -- HACER LAS MODIFICACIONES DE LAS OBSERVACIONES --
90  ●   ALTER TABLE company
91      DROP COLUMN  website;
92
93  ●   RENAME TABLE user TO data_user;
94
95  ●   ALTER TABLE data_user
96      MODIFY COLUMN id INT NOT NULL AUTO_INCREMENT,
97      CHANGE COLUMN email personal_email VARCHAR(150);
98
99  ●   ALTER TABLE credit_card
100     MODIFY COLUMN id VARCHAR(20) NOT NULL,
101     MODIFY COLUMN iban VARCHAR(50),
102     MODIFY COLUMN pin VARCHAR(4),
103     MODIFY COLUMN expiring_date VARCHAR(20),
104     ADD COLUMN fecha_actual DATE;
105
106  ●   ALTER TABLE transaction
107     MODIFY COLUMN credit_card_id VARCHAR(20) NULL,
108     MODIFY COLUMN company_id VARCHAR(20) NULL,
109     MODIFY COLUMN user_id INT NULL;

```

Luego lo que hice fue eliminar las FK que habia creado anteriormente, buscar datos en las tablas huérfanos, que en este caso son los del ejercicio 3 del nivel 1 que habia agregado anteriormente con el SET FOREIGN_KEY_CHECKS = 0, y lo que hice fue establecer esos valores a null ya que esos valores no existen en las tablas hijas como company, data_user y credit_card, como esos valores no existen y no hay informacion para agregarlos a esa tabla, solamente los estableci en valores null.

Primero busque los valores huérfanos

```

146  -- BUSCAR HUERFANOS Y CORREGIRLO PARA PODER CREAR LAS FK --
147  ●   SELECT DISTINCT company_id
148      FROM `transaction`
149      WHERE company_id IS NOT NULL AND company_id NOT IN (SELECT id FROM company);
150
151  ●   SELECT DISTINCT credit_card_id
152      FROM `transaction`
153      WHERE credit_card_id IS NOT NULL AND credit_card_id NOT IN (SELECT id FROM credit_card);
154
155  ●   SELECT DISTINCT user_id
156      FROM `transaction`
157      WHERE user_id IS NOT NULL AND user_id NOT IN (SELECT id FROM data_user);
158

```

Luego los setie a null


```

159      -- CORREGIR HUERFANOS, SETEANDOLOS A NULL YA QUE N
160 •    UPDATE `transaction`
161      LEFT JOIN company
162      ON `transaction`.company_id = company.id
163      SET `transaction`.company_id = NULL
164      WHERE `transaction`.company_id IS NOT NULL
165      AND company.id IS NULL;
166
167 •    UPDATE `transaction`
168      LEFT JOIN credit_card
169      ON `transaction`.credit_card_id = credit_card.id
170      SET `transaction`.credit_card_id = NULL
171      WHERE `transaction`.credit_card_id IS NOT NULL
172      AND credit_card.id IS NULL;
173
174 •    UPDATE `transaction`
175      LEFT JOIN data_user
176      ON `transaction`.user_id = data_user.id
177      SET `transaction`.user_id = NULL
178      WHERE `transaction`.user_id IS NOT NULL
179      AND data_user.id IS NULL;

```

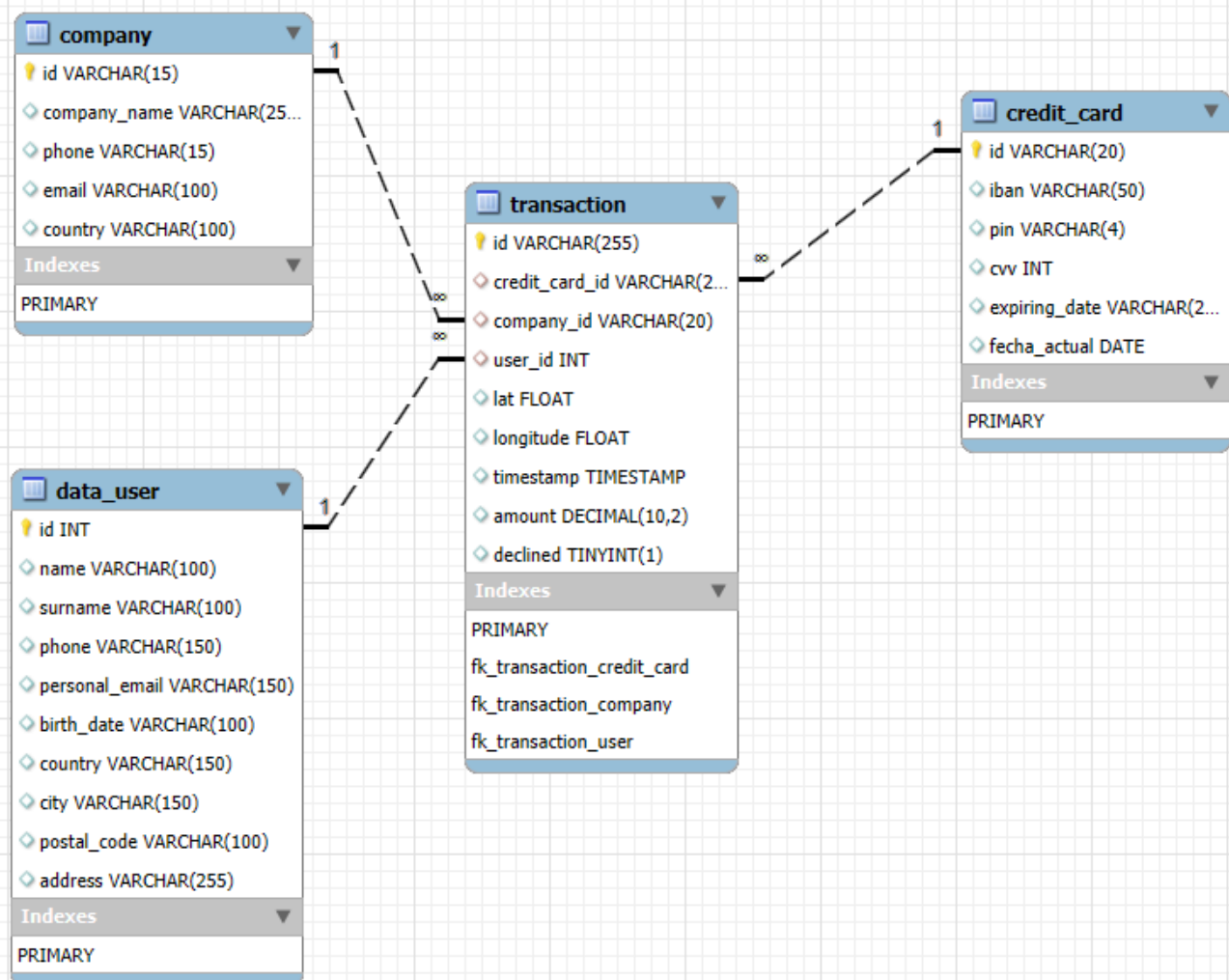
Y luego finalmente estableci las FK de manera correcta

```

181      -- AHORA SI , AGREGAR LAS FK --
182
183 •    ALTER TABLE `transaction`
184      ADD CONSTRAINT fk_transaction_credit_card
185      FOREIGN KEY (credit_card_id)
186      REFERENCES credit_card(id)
187      ON DELETE SET NULL
188      ON UPDATE CASCADE;
189
190 •    ALTER TABLE `transaction`
191      ADD CONSTRAINT fk_transaction_company
192      FOREIGN KEY (company_id)
193      REFERENCES company(id)
194      ON DELETE SET NULL
195      ON UPDATE CASCADE;
196
197 •    ALTER TABLE `transaction`
198      ADD CONSTRAINT fk_transaction_user
199      FOREIGN KEY (user_id)
200      REFERENCES data_user(id)
201      ON DELETE SET NULL
202      ON UPDATE CASCADE;

```

Finalmente el diagrama a quedado asi:



Exercici 2

L'empresa també us demana 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.
- Assegureu-vos d'incloure informació rellevant de les taules que coneixereu i utilitzeu àlies per canviar de nom columnes segons calgui.

Mostra els resultats de la vista, ordena els resultats de forma descendent en funció de la variable ID de transacció.

```
214 • CREATE OR REPLACE VIEW InformeTecnico AS
215 SELECT `transaction`.id AS transaction_id,
216 data_user.name AS user_name,
217 data_user.surname AS user_surname,
218 data_user.phone AS user_phone,
219 credit_card.iban,
220 company.company_name AS company_name,
221 company.phone AS company_phone,
222 `transaction`.timestamp AS transaction_date
223 FROM `transaction`
224 LEFT JOIN data_user
225     ON `transaction`.user_id = data_user.id
226 LEFT JOIN credit_card
227     ON `transaction`.credit_card_id = credit_card.id
228 LEFT JOIN company
229     ON `transaction`.company_id = company.id;
230
231 -- MOSTRAR VISTA --
232
233 • SELECT * FROM InformeTecnico
234 ORDER BY transaction_id DESC;
```

Result Grid | Filter Rows: | Exports: | Wrap Cell Content: | Fetch rows:

	transaction_id	user_name	user_surname	user_phone	iban	company_name	company_phone	transaction_date
▶	FFFD31D6-9495-47CE-B54A-7DB8E1CC274B	Bmrgli	Tprvvmrc	+58-917-5294	XX794814451211289182490922	Turpis Company	09 69 48 65 52	2021-08-05 12:26:24
	FFFCF76D-ECF0-4985-A2D0-82A7B75998FC	Dfrled	Vilqcjdl	+38-243-7637	XX636251701647892036676034	Amet Nulla Donec Corporation	07 15 25 14 74	2023-06-17 19:10:30
	FFFC9E8D-27C7-4ADE-98F2-7533EF4DF126	Securp	Faofvqfy	+76-822-2041	XX162677143304223631437567	Nunc Interdum Incorporated	05 18 15 48 13	2019-10-30 02:31:37
	FFFB270D-F53A-4D5D-9666-E5307C53CC84	Ggzjpa	Uirzjulh	+45-591-9299	XX395114267082019952567052	Viverra Donec Foundation	03 33 12 32 73	2024-05-13 03:42:03
	FFF9E3CE-234E-408C-A8EF-F9CAD577224A	Yshimq	Zpsjsleed	+41-970-7949	XX8845462156537570367941	Convallis In Incorporated	06 66 57 29 50	2022-12-17 20:57:55
	FFF9E178-6CD2-4DF9-99B0-49AE068809B1	Jevepx	Xwcwzwnm	+51-367-9854	XX321405515711654384711481	Mus Aenean Eget Foundation	06 25 15 52 43	2023-02-08 10:01:48

InformeTecnico 21 ×