

1 Descripció	2
2 Nivell 1.....	2
2.1 Exercici 1.....	2
2.1.1 Creació de la taula credit_card.....	2
2.1.2 Insertar les dades a credit_card	4
2.1.3 Modificar expiring_date a tipus DATE.....	5
2.1.4 Crear relació entre taules company i transaction	6
2.2 Exercici 2.....	7
2.2.1 Resolució	7
2.3 Exercici 3.....	8
2.3.1 Procés per trobar la solució	8
2.3.2 Codi sql de la solució final.....	12
2.4 Exercici 4.....	13
2.4.1 Codi sql de la Solució	14
3 Nivell 2.....	14
3.1 Exercici 1.....	14
3.1.1 Codi sql de la solució	14
3.2 Exercici 2.....	17
3.2.1 Codi sql de la solució sense crear la vista	17
3.2.2 Generació del codi sql de la vista i de la pròpia vista VistaMarketing.....	18
3.3 Exercici 3.....	21
3.3.1 Codi sql de la solució	21
4 Nivell 3.....	22
4.1 Exercici 1.....	22
4.1.1 Creació taula user	23
4.1.2 Introduir dades taula user.....	24
4.1.3 Creació relació taula user amb taula transaction.....	27
4.1.4 Comprovacions taula company.....	30
4.1.5 Comprovacions taula transaction i relacions amb altres taules	31
4.1.6 Comprovacions taula credit_card	32
4.1.6.1 Convertir columna expiring_date de DATE a VARCHAR(10)	33
4.1.6.2 Afegir columna fecha_actual.....	35
5 ANNEXOS.....	36
5.1 Creació de la taula de l'Exercici 1 Nivell 1	36

1 Descripció

En aquest sprint, es simula una situació empresarial en la qual has de realitzar diverses manipulacions en les taules de la base de dades. Al seu torn, hauràs de treballar amb índexs i vistes. En aquesta activitat, continuaràs treballant amb la base de dades que conté informació d'una empresa dedicada a la venda de productes en línia. En aquesta tasca, començaràs a treballar amb informació relacionada amb targetes de crèdit

2 Nivell 1

2.1 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.

2.1.1 Creació de la taula credit_card

Per decidir la longitud d'alguns camps de la taula *credit_card*, veure l'Annex del següent apartat [Annex 5.1. Creació de la taula de l'Exercici 1 Nivell 1](#)

Directament amb codi creo la taula *credit_card* tal com la vull.

```

1 -- NIVELL 1
2 -- EXERCICI 1 (Solució 1)
3 • USE transactions;
4 • CREATE TABLE IF NOT EXISTS credit_card (
5     id VARCHAR(8) PRIMARY KEY,
6     iban VARCHAR(50) NOT NULL,
7     pan VARCHAR(19) NOT NULL,
8     pin SMALLINT NOT NULL,
9     cvv SMALLINT NOT NULL,
10    /*Por que los datos que insertamos no son formato fecha*/
11    expiring_date VARCHAR(8) NOT NULL
12 );
13
14

```

Comprovo que ho fa bé amb SHOW CREATE TABLE:

Table	Create Table
credit_card	CREATE TABLE `credit_card` (`id` varchar(8) NOT NULL, `iban` varchar(50) NOT NULL, `pan` varchar(19) NOT NULL, `pin` smallint NOT NULL, `cvv` smallint NOT NULL, `expiring_date` varchar(8) NOT NULL)

Action Output	
# Time Action	Message
1 10:43:02 SHOW CREATE TABLE credit_card	1 row(s) returned

Si selecciono i faig Open Value in Viewer veig el codi:

The dialog box shows the SQL code for creating the credit_card table:

```

1 CREATE TABLE `credit_card` (
2     `id` varchar(8) NOT NULL,
3     `iban` varchar(50) NOT NULL,
4     `pan` varchar(19) NOT NULL,
5     `pin` smallint NOT NULL,
6     `cvv` smallint NOT NULL,
7     `expiring_date` varchar(8) NOT NULL,
8     PRIMARY KEY (`id`)
9 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

```

At the bottom of the dialog, it says "Data Length: 298 bytes".

2.1.2 Insertar les dades a credit_card

Executo el codi `datos_introducir_credit.sql`, afegint abans USE `transactions`.

```

1   -- Fem activa la BBDD transaction
2 • USE transactions;
3
4   -- ALTER TABLE credit_card MODIFY expiring_date VARCHAR(8);
5   -- Insertam els dades de credit_card
6 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2938', 'TR301950312213576817638661', '542446
7 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2945', 'D026854763748537475216568689', '5142
8 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2952', '8645IVQL52710525608255', '4556 453 5
9 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2959', 'CR7242477244335841535', '37246137734
10 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2966', 'BG72LKTQ15627628377363', '448566 886
11 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2973', 'PT87806228135092429456346', '544 586
12 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2980', 'DE39241881883086277136', '402400 714
13 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2987', 'GE89681434837748781813', '3763 74768
14 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-2994', 'BH62714428368066765294', '3442832732
15 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3001', 'CY49087426654774581266832110', '5117
16 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3008', 'LU507216693616119230', '448574446443
17 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3015', 'PS119398216295715968342456821', '378
18 • INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-3022', 'GT91695162850556977423121857', '5164

Output:
Action Output
# Time Action Message
193 12:35:08 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4275', 'AE491827... 1 row(s) affected
194 12:35:08 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4282', 'MK253474... 1 row(s) affected
195 12:35:08 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4289', 'PS346287... 1 row(s) affected
196 12:35:08 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4296', 'BE548496... 1 row(s) affected
197 12:35:08 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4303', 'FI1146379... 1 row(s) affected
198 12:35:08 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4310', 'ME270222... 1 row(s) affected
199 12:35:08 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4317', 'HU291746... 1 row(s) affected
200 12:35:08 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4324', 'CR293442... 1 row(s) affected
201 12:35:08 INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4331', 'KW94853 1 row(s) affected

```

Visualitzo la taula, per tal que es vegi que té 275 files.

Result Grid					
#	Time	Action	Message		
269	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4835', 'PT34592171131763200132583', 3723 677744 22550, 1149, 680, '01/08/24')	1 row(s) affected		
270	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4842', 'SA2156708581957118818229', 3774 636724 83250, 4655, 750, '11/11/24')	1 row(s) affected		
271	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4849', 'SE2813123487163628531121', 5223363813491514, 9992, 779, '03/21/25')	1 row(s) affected		
272	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4856', 'TR373872558313545667124286', 349528235713651, 9086, 974, '05/19/23')	1 row(s) affected		
*					

credit card 1 >

Output:

#	Time	Action	Message		
269	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4807', 'LB19298...', 1 row(s) affected)			
270	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4814', 'MR4845...', 1 row(s) affected)			
271	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4821', 'LT25314...', 1 row(s) affected)			
272	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4828', 'BG11LM...', 1 row(s) affected)			
273	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4835', 'PT34592...', 1 row(s) affected)			
274	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4842', 'SA21567...', 1 row(s) affected)			
275	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4849', 'SE28131...', 1 row(s) affected)			
276	12:35:08	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('Ccu-4856', 'TR37387...', 1 row(s) affected)			
277	12:54:42	SELECT * FROM transactions.credit_card LIMIT 0, 50000	275 row(s) returned		

2.1.3 Modificar `expiring_date` a tipus DATE

Com he resolt anteriorment, per insertar els valors a la taula `credit_card`, la variable `expiring_date` no l'hem creat com a DATE, ja que els valors eren cadenes de caràcters i ens fallava quan executàvem el codi `datos_introducir_credit.sql`. Però ara, encara que no ens ho demani l'enunciat la volem convertir a tipus DATE, doncs `expiring_date` són dates.

Hem intentat provar la funció directament STR_TO_DATE amb UPDATE però ens ha fallat.

```
12:56:10    UPDATE credit_card SET expiring_date=STR_TO_DATE(expiring_date_old,
'%m/%d/%Y') Error Code: 1175. You are using safe update mode and you tried to update a table
without a WHERE that uses a KEY column. To disable safe mode, toggle the option in Preferences
-> SQL Editor and reconnect.      0.016 sec
```

Ho corregim afegint una nova columna que li diem `expiring_date` que serà de tipus DATE i la anterior `expiring_date_old`.

I utilitzem la funció STR_TO_DATE per convertir la cadena de text a tipus DATE. Veiem el codi:

```
48 -- EXERCICI 1 (Modificació expiring_date de VARCHAR a DATE)
49
50 • ALTER TABLE credit_card RENAME COLUMN expiring_date to expiring_date_old;
51 • ALTER TABLE credit_card ADD COLUMN expiring_date DATE;
52 -- Sinó UPDATE falla, ya que no utiliza una condicion WHERE
53 • SET SQL_SAFE_UPDATES=0;
54 • UPDATE credit_card SET expiring_date=STR_TO_DATE(expiring_date_old, '%m/%d/%Y');
55 • ALTER TABLE credit_card DROP COLUMN expiring_date_old;
56 -- Restablecemos el pode UPDATE SAFE como teníamos.
57 • SET SQL_SAFE_UPDATES=1;
```

Action Output		Message
#	Time	Action
1	13:07:30	ALTER TABLE credit_card RENAME COLUMN expiring_date to expiring_date_old
2	13:07:30	ALTER TABLE credit_card ADD COLUMN expiring_date DATE
3	13:07:30	SET SQL_SAFE_UPDATES=0
4	13:07:30	UPDATE credit_card SET expiring_date=STR_TO_DATE(expiring_date_old, "%m/%d/%Y")
5	13:07:30	ALTER TABLE credit_card DROP COLUMN expiring_date_old
6	13:07:30	SET SQL_SAFE_UPDATES=1

The screenshot shows a MySQL Workbench interface. At the top, there's a toolbar with various icons for database management. Below the toolbar, a query editor window displays the following SQL command:

```
1 • SELECT * FROM transactions.credit_card;
```

Below the query editor is a results grid titled "Result Grid". The grid has columns labeled "id", "iban", "pan", "pin", "cvv", and "expiring_date". The data in the grid is as follows:

	id	iban	pan	pin	cvv	expiring_date
▶	CcU-2938	TR301950312213576817638661	5424465566813633	3257	984	2022-10-30
	CcU-2945	DO26854763748537475216568689	5142423821948828	9080	887	2023-08-24
	CcU-2952	BG45IVQL52710525608255	4556 453 55 5287	4598	438	2021-06-29
	CcU-2959	CR7242477244335841535	372461377349375	3583	667	2023-02-24
	CcU-2966	BG72LKTQ15627628377363	448566 886747 7265	4900	130	2024-10-29

I ja tinc `expiring_date` de tipus DATE.

2.1.4 Crear relació entre taules company i transaction

Afegeixo la clau externa a la taula transaction amb un ALTER. Faig ADD CONSTRAINT `fk_credit_card_id`, per tal que es vegi aquest nom com a clau externa relacionada amb el camp `credit_card_id`.

The screenshot shows the MySQL Workbench interface with the SQL editor open. The code entered is:

```
59 -- EXERCICI 1 (Crear relacions amb altres taules)
60 • USE transactions;
61 • ALTER TABLE transaction ADD CONSTRAINT fk_credit_card_id FOREIGN KEY transaction(credit_card_id) REFERENCES credit_card(id);
```

Below the SQL editor is an "Output" pane showing the execution results:

Action Output	#	Time	Action	Message
	1	13:39:30	USE transactions	0 row(s) affected
	2	13:39:30	ALTER TABLE transaction ADD CONSTRAINT fk_credit_card_id FOREIGN KEY transaction(credit_card... 587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0	

Podem comprovar la la relació creada amb SHOW CREATE.

2.3 Exercici 3

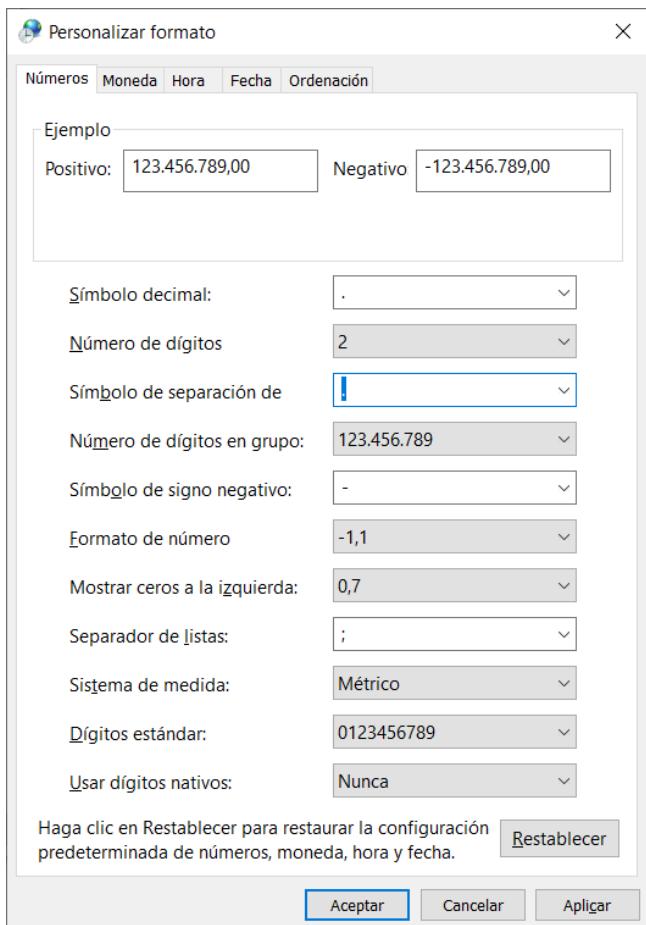
En la taula "transaction" ingressa un nou usuari amb la següent informació:

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

2.3.1 Procés per trobar la solució

La professora Alana pot ometre aquest apartat i anar directament a l'apartat següent [2.3.2. Codi sql de la solució final](#) si es vol veure directament la solució, ja que en aquest apartat expliquem com falla la primera versió de codi.

El primer objectiu és automatitzar en Excel el codi de SQL per tal que sigui més ràpid si hi haguessin moltes files a inserta al codi. A continuació canvio el format regional, per tal que els decimals es converteixin a punts a l'excel.



Pego els valors a l'Excel i els transposo:

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

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

Creo la fórmula de SQL en Excel. Així em serviria per si hagués d'afegir molts registres de cop i no volgués copiar-la a mà per tots els valors. El resultat de la fórmula la pego al Workbench i poso entre comes alguns dels valors que no són numèrics.

```
=CONCATENAR("INSERT INTO transaction (";A13;" , ";B13;" , ";C13;" , ";D13;" , ";E13;" , ";F13;" , ";G13;" , ";H13;") VALUES (";A14;" , ";B14;" , ";C14;" , ";D14;" , ";E14;" , ";F14;" , ";G14;" , ";H14;")")
```

```

77 -- EXERCICI 3 (Opció 1. L'execució dóna error.
78 • USE transactions;
79 • INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES
80 ('108B1D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', 9999, 829.999, -117.999, 111.11, 0);
81
Output
Action Output
# Time Action Message
1 11:21:24 USE transactions 0 row(s) affected
2 11:21:24 INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES ... Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('transactions'.`transact...

```

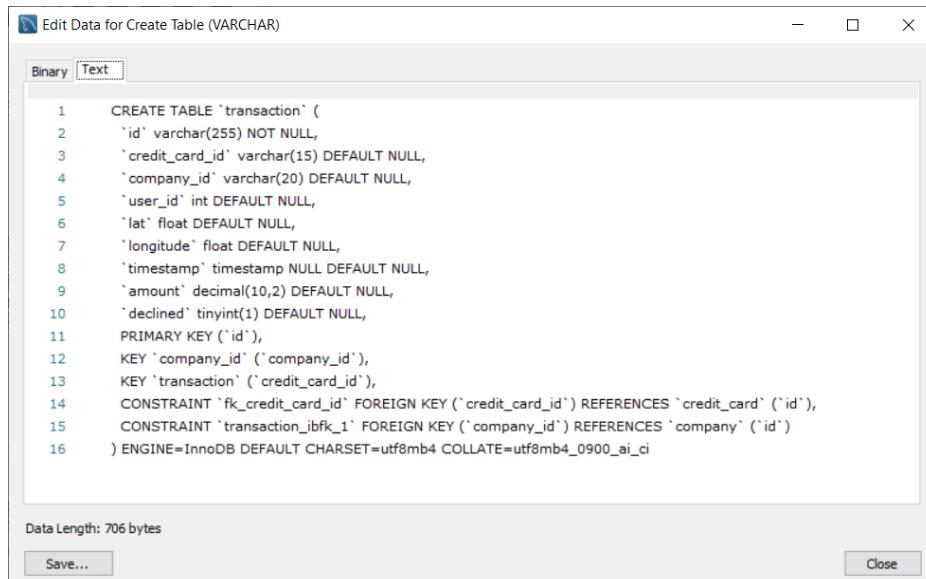
Ara ja puc tornar a deixar la configuració regional com tenia (els punts com a decimals)

La consulta falla, es clar veig que falla per tres motius. No hi ha una company amb el *id*='b-9999', ni tampoc un *credit_card_id*='CcU-9999' i també no hem entrat a INSERT el camp *timestamp*. Faig un SHOW CREATE TABLE per veure si el camp *timestamp* admet NULL i veig que sí.

```

32 • USE transactions;
33 • SHOW CREATE TABLE transaction;

```



Miro si la taula company admet valors NULL, als camp que són diferents als de *id* i veig que sí.

```

39 • USE transactions;
40 • SHOW CREATE TABLE company;

```

```

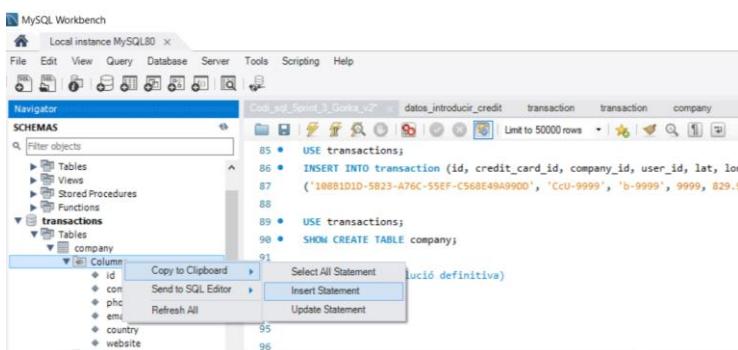
1 CREATE TABLE `company` (
2     `id` varchar(15) NOT NULL,
3     `company_name` varchar(255) DEFAULT NULL,
4     `phone` varchar(15) DEFAULT NULL,
5     `email` varchar(100) DEFAULT NULL,
6     `country` varchar(100) DEFAULT NULL,
7     `website` varchar(255) DEFAULT NULL,
8     PRIMARY KEY (`id`)
9 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

```

Data Length: 336 bytes

Save... **Close**

En canvi a la taula *credit_card_id*, sé que no admet NULL ja que així ho he imposat jo quan l'he creat.



Per no tenir que copiar totes les columnes de company, faig el següent al Workbench i després PASTE (veure “pantallazo” anterior) i genero el codi INSERT de company de forma automàtica. El mateix per credit_card. Després modifico el codi per entrar els valor corresponents.

A credit_card com que no admet NULL en molts dels camps excepte *expiring_date*, he d'entrar alguns valors a les columnes.

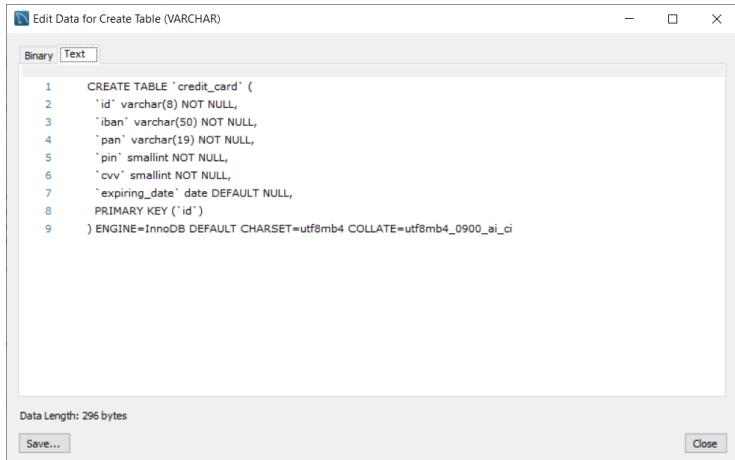
Modifico la taula *credit_card* amb ALTER per tal que admeti NULL, al camp *expiring_date*.

```

100 • USE transactions;
101 • ALTER TABLE credit_card MODIFY expiring_date DATE NULL;

```

Faig SHOW TABLE i ja veig que expiring_date ja admet NULL.



```

1 CREATE TABLE `credit_card` (
2   `id` varchar(8) NOT NULL,
3   `iban` varchar(50) NOT NULL,
4   `pan` varchar(19) NOT NULL,
5   `pin` smallint NOT NULL,
6   `cvv` smallint NOT NULL,
7   `expiring_date` date DEFAULT NULL,
8   PRIMARY KEY (`id`)
9 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

```

Data Length: 296 bytes

Save... Close

2.3.2 Codi sql de la solució final

Per si no s'ha llegit l'apartat [2.3.1. Procés per trobar la solució](#), explico una mica el codi sql.

No puc introduir a *transaction* els valors d'un nou usuari a l'enunciat, doncs no hi ha una company amb el *id*=’b-9999’, ni tampoc un *credit_card_id*=’CcU-9999’ i per tant això trencaria la integritat referencial que està imposada a la taula *transaction* amb *company* i amb *credit_card*.

Aleshores en primer lloc, introduceixo a *company*, una companyia amb amb id=’b-9999’ i els altres camps amb NULL. Per *credit_card* ho introduceixo amb el *credit_card_id*=’CcU-9999’ i els altres valors poso alguna cosa, ja que tenia imposat que no podien ser NULL excepte *expiring_date*¹

¹ Podria haver modificat la taula *credit_card* amb amb varis ALTER per tal que tots els camps acceptessin NULL, però ho he fet d'aquesta manera.

```

92      -- EXERCICI 3 (Solució definitiva)
93 •  USE transactions;
94 •  INSERT INTO company
95     (id,company_name,phone,email,country,website)
96     VALUES ('b-9999',NULL, NULL, NULL, NULL,NULL);
97 •  INSERT INTO credit_card
98     (id,iban,pan,pin,ccv,expiring_date)
99     VALUES ('CcU-9999','',' ','-1,-1,NULL');

```

Output:

Action Output	#	Time	Action	Message
	1	12:36:44	USE transactions	0 row(s) affected
	2	12:36:44	INSERT INTO company (id,company_name,phone,email,country,website) VALUES ('b-9999',NULL, NULL, NULL, NULL,NULL);	1 row(s) affected
	3	12:36:44	INSERT INTO credit_card (id,iban,pan,pin,ccv,expiring_date) VALUES ('CcU-9999','',' ','-1,-1,NULL');	1 row(s) affected

Ara ja podré fer un INSERT del que em demanaven a l'exercici, ja que sí hi ha integritat referencial entre *company* i *transaction* i *credit_card* i *transaction*.

```

102 •  USE transactions;
103 •  INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, declined) VALUES
104     ('108B1D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', 9999, 829.999, -117.999, NULL,111.11, 0);
105

```

Output:

Action Output	#	Time	Action	Message
	1	12:39:33	USE transactions	0 row(s) affected
	2	12:39:33	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, decl...)	1 row(s) affected

Per verificar que hem afegit a la taula *transaction* el registre amb els camps que ens demanaven a l'enunciat.

```

106 •  USE transactions;
107 •  SELECT * FROM transaction t
108 WHERE t.id='108B1D1D-5B23-A76C-55EF-C568E49A99DD'
109
110

```

Result Grid:

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

2.4 Exercici 4

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

2.4.1 Codi sql de la Solució

```

111      -- EXERCICI 4
112 • USE transactions;
113 • ALTER TABLE credit_card DROP COLUMN pan;
114
-
8 12:47:03 ALTER TABLE credit_card DROP COLUMN pan
0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

```

Comprovo que la columna pan s'ha eliminat.

```
115 • SELECT * FROM transactions.credit_card;
```

```
116
```

	id	iban	pin	cvv	expiring_date
▶	CcU-2938	a	3257	984	2022-10-30
	CcU-2945	DO26854763748537475216568689	9080	887	2023-08-24
	CrLI-2952	RG45TVOI 52710525608255	4598	438	2021-06-29

3 Nivell 2

3.1 Exercici 1

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

3.1.1 Codi sql de la solució

Faig codi per buscar quin registre es i després utilitzaré el visualitzador de Workbench

Ja tinc el registre identificat.

```

119      -- EXERCICI 1
120 •  USE transactions;
121 •  SELECT
122    *
123   FROM transaction t
124  WHERE t.id='02C6201E-D90A-1859-B4EE-88D2986D3B02'
125
126
127

```

The screenshot shows the MySQL Workbench interface. At the top, there is a code editor window with the following SQL code:

```

-- EXERCICI 1
USE transactions;
SELECT
  *
FROM transaction t
WHERE t.id='02C6201E-D90A-1859-B4EE-88D2986D3B02'

```

Below the code editor is a "Result Grid" table with one row of data:

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
02C6201E-D90A-1859-B4EE-88D2986D3B02	CdU-2938	b-2362	92	81.9185	-12.5276	2021-08-28 23:42:24	466.92	0

At the bottom of the interface, there is a "transaction 10" tab and an "Output" panel showing the history of actions:

- 1 12:53:54 USE transactions 0 row(s) affected
- 2 12:53:54 SELECT * FROM transaction t WHERE t.id='02C6201E-D90A-1859-B4EE-88D2986D3B02' LIMIT 0, 500... 1 row(s) returned

Ara des de Workbench l'elimino:

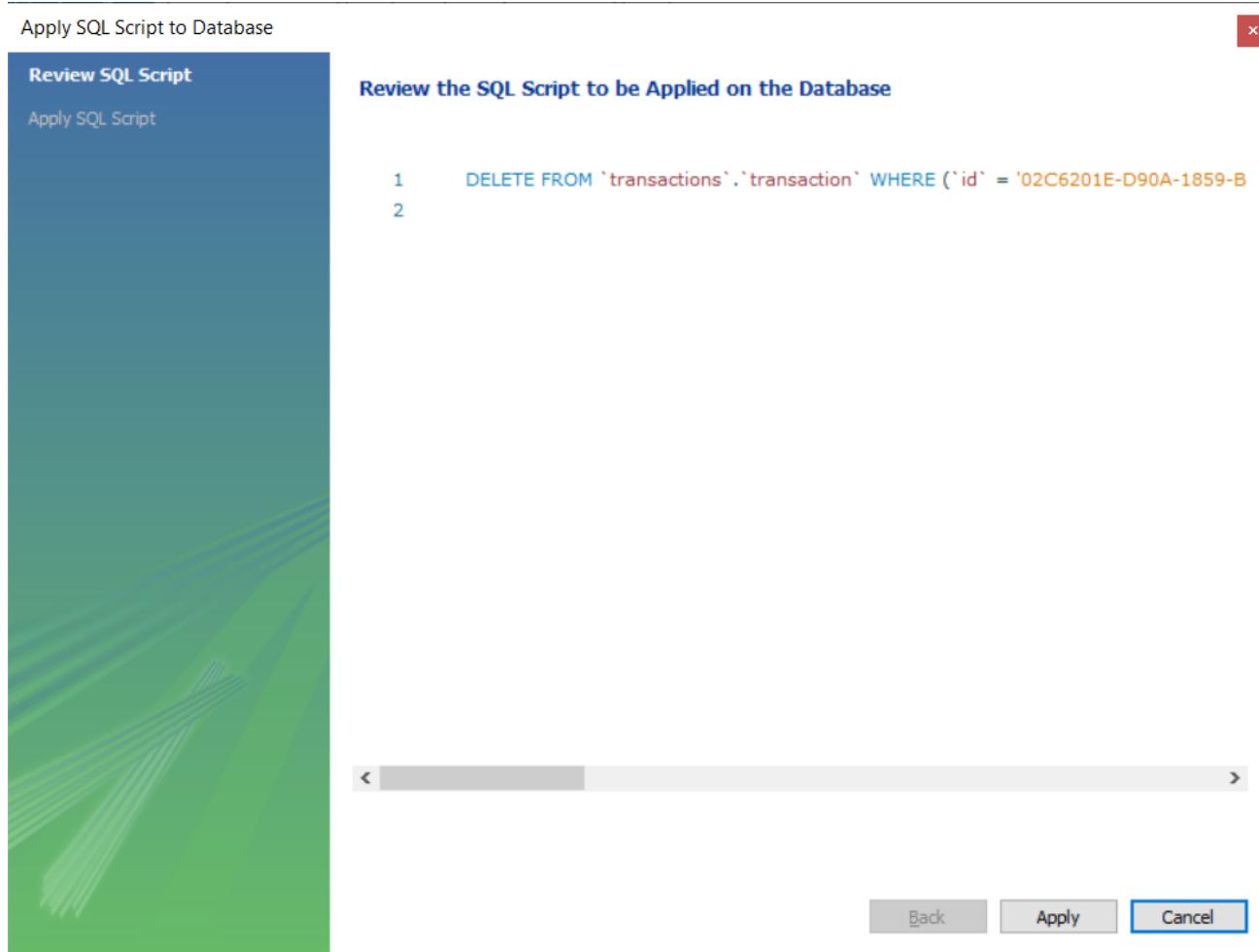
The screenshot shows the MySQL Workbench interface with the same SQL code as before. A context menu is open over the row in the Result Grid table, specifically over the first column of the first row. The menu options include:

- Open Value in Editor
- Set Field to NULL
- Mark Field Value as a Function/Literal
- Delete Row(s)** (highlighted in blue)
- Load Value From File...
- Save Value To File...
- Copy Row
- Copy Row (with names)
- Copy Row (quoted)
- Copy Row (with names, unquoted)
- Copy Row (with names, tab separated)
- Copy Row (tab separated)
- Copy Field
- Copy Field (quoted)
- Paste Row
- Capitalize Text
- Lowercase Text
- UPPERCASE Text

The "Delete Row(s)" option is selected. The message area at the bottom right of the interface shows:

HEREt.id='02C6201E-D90A-1859-B4EE-88D2986D3B02' LIMIT 0, 500... 1 row(s) returned

Després faig Apply i ja em surt el codi per eliminar el registre:



Podria fer Apply, però com que vull conservar el codi, copio el codi i ho executo de nou:

```
119      -- EXERCICI 1  
120 • USE transactions;  
121 • SELECT  
122 *  
123 FROM transaction t  
124 WHERE t.id='02C6201E-D90A-1859-B4EE-88D2986D3B02';  
125 • DELETE FROM transaction WHERE (id = '02C6201E-D90A-1859-B4EE-88D2986D3B02');  
--  
✓ 5 12:57:07 DELETE FROM transaction WHERE (id = '02C6201E-D90A-1859-B4EE-88D2986D3B02') 1 row(s) affected
```

3.2 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.

3.2.1 Codi sql de la solució sense crear la vista

Primer fem el codi fem un INNER JOIN per relacionar *company* i *transaction*. Aleshores com que només fem GROUP BY *t.company_id*, la mitjana només serà sobre aquesta agrupació.

```

127 -- EXERCICI 2
128 • USE transactions;
129 • SELECT
130   c.company_name Nom_companyia,
131   c.phone Telefon,
132   c.country País,
133   AVG(t.amount) Mitja_compra_realitzada
134   FROM transaction t
135   INNER JOIN company c ON t.company_id=c.id
136   GROUP BY t.company_id
137   ORDER BY Mitja_compra_realitzada DESC;
138

```

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

Nom_companyia	Telefon	País	Mitja_compra_realitzada
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 Duis Institute	05 35 93 32 44	Netherlands	442.520000

transaction 10 Result 16 x

Action Output

#	Time	Action	Message
1	13:21:18	USE transactions	0 row(s) affected
2	13:21:18	SELECT c.company_name Nom_companyia, c.phone Telefon, c.country País, AVG(t.amount) Mitja_com... <td>101 row(s) returned</td>	101 row(s) returned

Primer de tot faig activa la BBDD transactions, abans de crear la vista.

```
139 • USE transactions;
```

```
140
```

Output

Action Output

#	Time	Action
1	10:29:34	USE transactions

3.2.2 Generació del codi sql de la vista i de la pròpia vista VistaMarketing.

Ara creo una nova vista i pego el codi anterior sota CREATE VIEW i li poso el nom *VistaMarketing*.

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Create a new view in the active schema in the connected server

Name: new_view

Filter objects

Local instance MySQL80

Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

EMAS

any_in_some_subqueries

skalia

sys

Tables

Views

Stored Procedures

Functions

transactions

Tables

Views

visamarketing

Stored Procedures

Functions

world

Name: new_view

The name of the view is parsed automatically from the DDL statement. The DDL is parsed automatically while you type.

DDL:

```

1 • CREATE VIEW VistaMarketing AS
2   SELECT
3     c.company_name Nom_companyia,
4     c.phone Telefon,
5     c.country País,
6     AVG(t.amount) Mitja_compra_realitzada
7   FROM transaction t
8   INNER JOIN company c ON t.company_id=c.id
9   GROUP BY t.company_id
10  ORDER BY Mitja_compra_realitzada DESC;

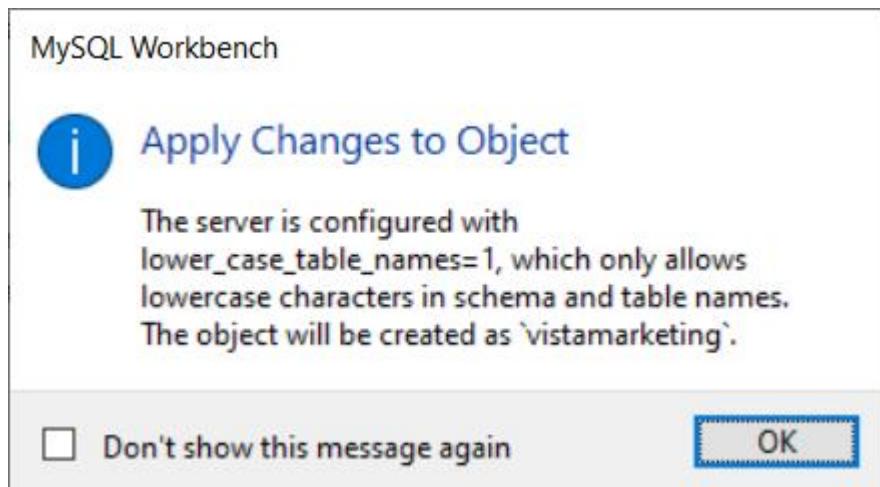
```

View

Output

Apply Revert

Faig Apply



Ara em surt el codi SQL:

Review SQL Script

Review the SQL Script to be Applied on the Database

Online DDL

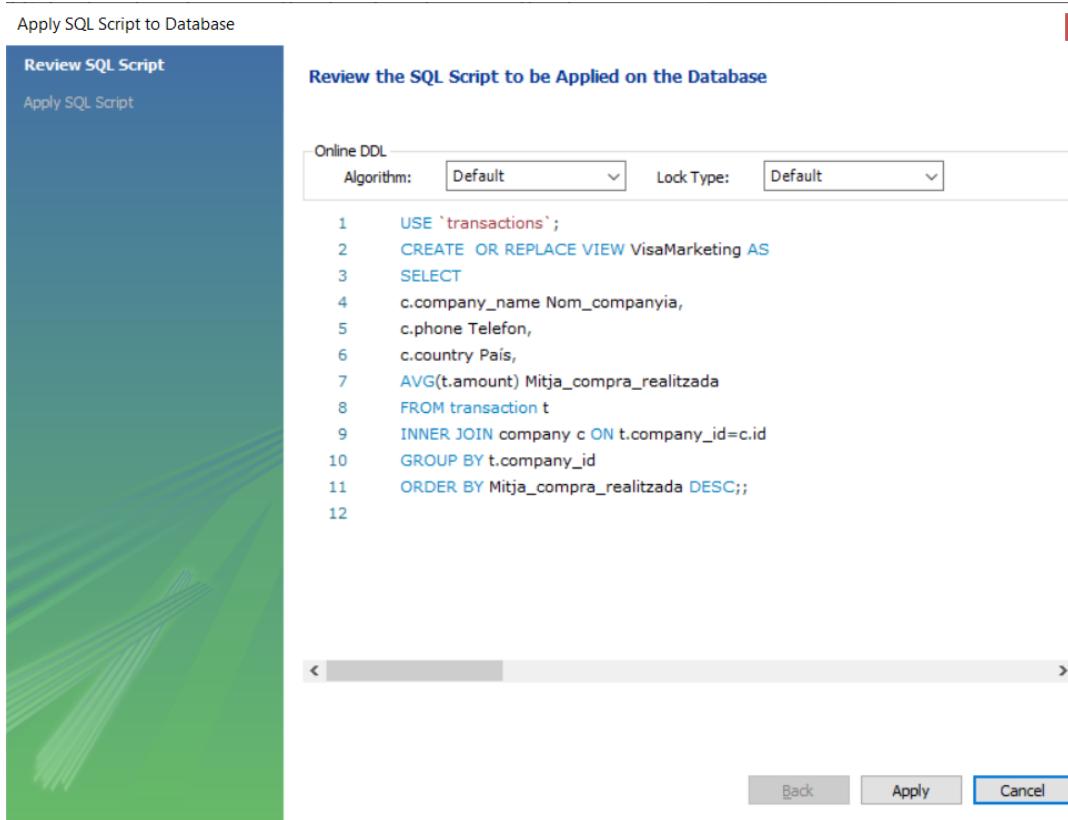
Algorithm:	Default	Lock Type:	Default
------------	---------	------------	---------

```
1 USE `transactions`;
2 CREATE OR REPLACE VIEW VistaMarketing AS
3 SELECT
4     c.company_name Nom_companyia,
5     c.phone Telefon,
6     c.country País,
7     AVG(t.amount) Mitja_compra_realitzada
8     FROM transaction t
9     INNER JOIN company c ON t.company_id=c.id
10    GROUP BY t.company_id
11    ORDER BY Mitja_compra_realitzada DESC;;
```

< >

Back Apply Cancel

Ara ja puc fer Apply



On si faig Apply em genera el codi de la vista²

```

CREATE
DEFINER = "root"@"localhost"
SQL SECURITY DEFINER
VIEW `transactions`.`vistamarketing` AS
SELECT
    `c`.`company_name` AS `Nom_companyia`,
    `c`.`phone` AS `Telefon`,
    `c`.`country` AS `Pais`,
    AVG(t.amount) AS `Mitja_compra_realitzada`
FROM
    (transactions.transaction t
     JOIN transactions.company c ON ((t.company_id = c.id)))
GROUP BY t.company_id
ORDER BY Mitja_compra_realitzada DESC
  
```

Si faig sobre la VistaMarketing un SELECT puc veure de nou els resultats, sense tenir que aplicar el codi original

² M'he adonat després de fer la vista al WorkBench que es pot crear la vista directament per codi sql fent CREATE VIEW nom_vista AS i després el codi SQL. Realment ho faré sempre així que és més ràpid quan tingui que crear-la.

The screenshot shows the MySQL Workbench interface. On the left, the database browser displays the schema for 'vistamarketing'. A query editor window titled 'Codi_sql_Sprint_3_Gorka_v3*' contains the SQL command: 'SELECT * FROM transactions.vistamarketing;'. The results grid shows 101 rows of company data. The output pane at the bottom shows the execution log: '1 10:37:06 SELECT * FROM transactions.vistamarketing LIMIT 0, 50000' and '101 row(s) returned'.

Nom_companyia	Telofon	País	Mitja_compra_realitzada
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 55 71 52	Italy	443.635000
Eget Tincidunt Duis 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 00	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
Magna A Neque Industries	04 14 44 64 62	Australia	396.315000
Amet Luctus Vulpitate Foundation	03 18 54 24 19	Canada	390.329000
Aliquam PC	01 45 73 52 16	Germany	385.265000
Neque Tellus Incorporated	04 43 18 34 19	Ireland	364.005000

3.3 Exercici 3

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

3.3.1 Codi sql de la solució

Ara puc treballar sobre la vista Germany com si fos una taula i filtrar amb un WHERE pel país que volem.

The screenshot shows the MySQL Workbench interface with three main panes:

- Top pane:** Shows the SQL editor with the following code:


```
1 • USE transactions;
2 • SELECT * FROM vistamarketing v
3 WHERE v.País='Germany'
```
- Middle pane:** Shows the "Result Grid" containing the following data:

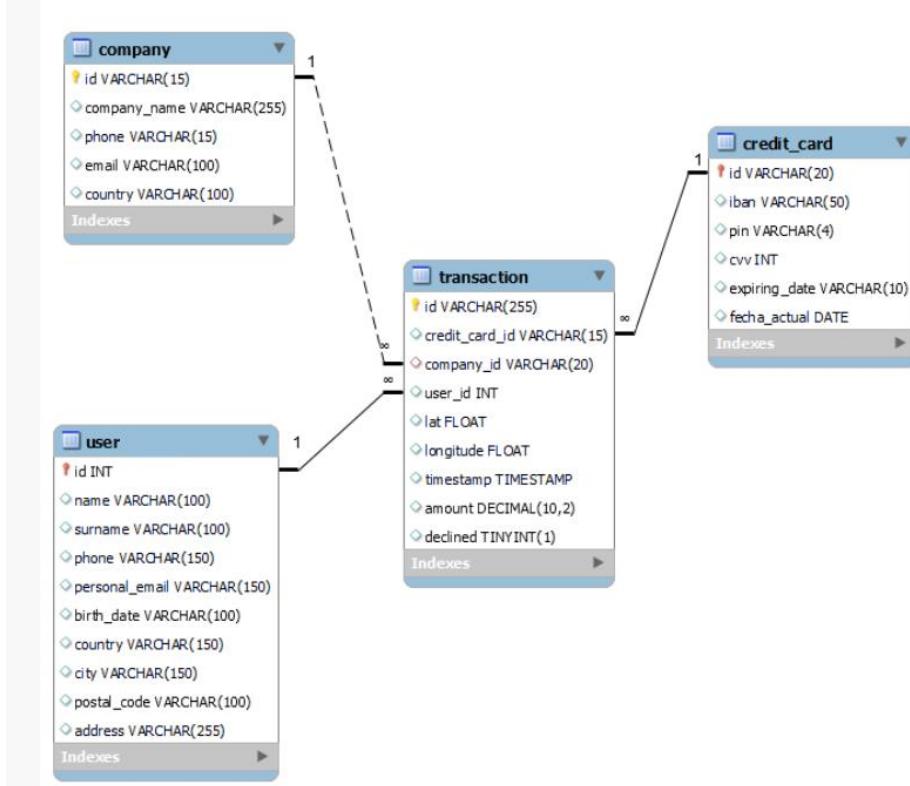
	Nom_companyia	Telefon	País	Mitja_compra_realitzada
▶	Aliquam PC	01 45 73 52 16	Germany	385.265000
	Ac Industries	09 34 65 40 60	Germany	289.645000
	Rutrum Non Inc.	02 66 31 61 09	Germany	266.900000
	Nunc Interdum Incorporated	05 18 15 48 13	Germany	244.025238
	Augue Foundation	06 88 43 15 63	Germany	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
- Bottom pane:** Shows the "vistamarketing 3" session output with the following logs:

#	Time	Action	Message
1	10:40:47	USE transactions	0 row(s) affected
2	10:40:47	SELECT * FROM vistamarketing v WHERE v.País='Germany' LIMIT 0, 50000	8 row(s) returned

4 Nivell 3

4.1 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:



4.1.1 Creació taula user

Obre la consulta de l'enunciat *estructura_datos_user.sql* i afageix USE transactions i l'executo.

Screenshot of MySQL Workbench showing the creation of the 'user' table in the 'transactions' schema.

Navigator:

- SCHEMAS: transactions
- Tables: company, credit_card, transaction, user (highlighted)
- Views: vistamarketing
- Stored Procedures
- Functions

Codi_sql_Sprint_3_Gorka_v4:

```

143 -- NIVELL 3
144 -- EXERCICI 1
145 • USE transactions;
146 • CREATE INDEX idx_user_id ON transaction(user_id);
147     -- La taula transactions ja en codi de l'enunciat.
148 • CREATE TABLE IF NOT EXISTS user (
149     id INT PRIMARY KEY,
150     name VARCHAR(100),
151     surname VARCHAR(100),
152     phone VARCHAR(150),
153     email VARCHAR(150),
154     birth_date VARCHAR(100),
155     country VARCHAR(150),
156     city VARCHAR(150),
157     postal_code VARCHAR(100),
158     address VARCHAR(255),
159     FOREIGN KEY(id) REFERENCES transaction(user_id)
160 );
161
  
```

Table: company

Columns:

id	varchar(15) PK
company_name	varchar(255)
phone	varchar(15)
email	varchar(100)
country	varchar(100)
website	varchar(255)

La taula ja està creada.

Ara introduïm les dades, obrint la consulta *datos_introducir_user.sql*

4.1.2 Introduir dades taula user

```

1 • USE transactions;
2 -- Insertamos datos de user
3 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
4 •   "1", "Zeus", "Gamble", "1-282-581-0551",
5 •   "2", "Garrett", "Lowell", "73544", "348-7818 Sagittis St."
6 •   "3", "Ciaran", "Lowell", "73544", "348-7818 Sagittis St."
7 •   "4", "Howard", "Lowell", "73544", "348-7818 Sagittis St."
8 •   "5", "Hayfa", "Pittsburgh", "15222", "123 Elmwood Ave."
9 •   "6", "Joel", "Tyre", "702-369-0220", "123 Elmwood Ave."
10 •  "7", "Rafael", "Pittsburgh", "15222", "123 Elmwood Ave."
11 •  "8", "Nissim", "Pittsburgh", "15222", "123 Elmwood Ave."
12 •  "9", "Mannix", "Pittsburgh", "15222", "123 Elmwood Ave."
13 •  "10", "Robert", "Pittsburgh", "15222", "123 Elmwood Ave."
14 •  "11", "Joan", "Baltimore", "301-555-0123", "123 Elmwood Ave."
15 •  "12", "Benedict", "Baltimore", "301-555-0123", "123 Elmwood Ave."
16 •  "13", "Allegra", "Baltimore", "301-555-0123", "123 Elmwood Ave."
17 •  "14", "Sara", "Baltimore", "301-555-0123", "123 Elmwood Ave."
18 •  "15", "Noelani", "Baltimore", "301-555-0123", "123 Elmwood Ave."
19 •  "16", "Eric", "Baltimore", "301-555-0123", "123 Elmwood Ave."
20 •  "17", "Bruce", "Baltimore", "301-555-0123", "123 Elmwood Ave."

```

Veiem que falla i pego l'error:

Segurament falla, ja que hi ha algun user a *transactions* que no estan a la taula *users*.

11:33:31 *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", "Nov 17, 1985", "United States", "Lowell", "73544", "348-7818 Sagittis St.")* Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('transactions`.`user', CONSTRAINT `user_ibfk_1` FOREIGN KEY (`id`) REFERENCES `transaction` (`user_id`)) 0.015 sec

Repasso el codi de la creació de la taula user i veig un error. En el codi, en la següent sentència:

FOREIGN KEY(id) REFERENCES transaction(user_id)

El *id* és una clau primària a la taula *user*, però no és una clau externa de la taula *transaction*. En tot cas és *user_id* de *transactions* una clau externa de la taula *user*.

Com ha anomenat aquesta clau. Ho puc veure directament al Workbench (també ho podria fer amb SHOW CREATE TABLE).



Així doncs eliminaré aquesta clau externa:

```
143      -- NIVELL 3
144      -- EXERCICI 1
145
146 •  ALTER TABLE user DROP FOREIGN KEY user_ibfk_1;
147
148
149
```

The screenshot shows the MySQL Workbench interface with the 'Output' tab selected. At the top, there's a message bar indicating the execution time: '1 12:07:58'. Below it, a table titled 'Action Output' shows one row: 'ALTER TABLE user DROP FOREIGN KEY user_ibfk_1'. To the right of the table, a 'Message' panel displays the status: '0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0'. This confirms that the foreign key constraint has been successfully dropped from the 'user' table.

Un cop eliminada, ara sí puc introduir totes les dades a la taula *user*;

```

Codi_sql_Sprint_3_Gorka_v4*  vistamarketing  datos_introducir_user  estructura_datos_user
1 • USE transactions;
2 -- Insertamos datos de user
3 • INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (
4 •     "1", "Zeus", "Galaxy", "010-12345678", "zeus@gmail.com", "1985-01-01", "USA", "New York", "10000", "123 Main St"
5 •     "2", "Garrett", "Hawkeye", "010-98765432", "garrett.hawkeye@avengers.com", "1988-05-20", "USA", "Los Angeles", "90210", "555 Park Ave"
6 •     "3", "Ciaran", "Fitzsimons", "010-12345678", "ciaran.fitzsimons@avengers.com", "1990-02-15", "Ireland", "Dublin", "A12345", "123 Fitzsimons St"
7 •     "4", "Howard", "Potts", "010-12345678", "howard.potts@avengers.com", "1985-07-10", "USA", "Chicago", "60601", "123 Stark Tower"
8 •     "5", "Hayfa", "Al-Malik", "010-12345678", "hayfa.al-malik@avengers.com", "1989-03-05", "Saudi Arabia", "Riyadh", "123 Al-Saud St"
9 •     "6", "Joel", "Tyree", "010-12345678", "joel.tyree@avengers.com", "1987-08-25", "USA", "Detroit", "48226", "123 Tyree St"
10 •    "7", "Rafael", "Dominguez", "010-12345678", "rafael.dominguez@avengers.com", "1986-09-12", "Spain", "Barcelona", "08008", "123 Dominguez St"
11 •    "8", "Nissim", "Ezra", "010-12345678", "nissim.ezra@avengers.com", "1984-11-15", "Israel", "Tel Aviv", "123 Ezra St"
12 •    "9", "Mannix", "Luther", "010-12345678", "mannix.luther@avengers.com", "1983-07-01", "USA", "Washington D.C.", "20500", "123 Mannix St"
13 •    "10", "Robert", "Downey Jr.", "010-12345678", "robert.downey.jr@avengers.com", "1965-04-04", "USA", "Los Angeles", "90210", "123 Iron Man St"
14 •    "11", "Joan", "Bennet", "010-12345678", "joan.bennet@avengers.com", "1982-01-15", "USA", "New York", "10001", "123 Quinjet St"
15 •    "12", "Benedict", "Cumberbatch", "010-12345678", "benedict.cumberbatch@avengers.com", "1976-07-04", "UK", "London", "SW1A 1AA", "123 Captain America St"
16 •    "13", "Allegra", "Kapoor", "010-12345678", "allegra.kapoor@avengers.com", "1985-05-20", "USA", "Chicago", "60601", "123 Captain America St"
17 •    "14", "Sara", "Fowler", "010-12345678", "sara.fowler@avengers.com", "1987-02-10", "USA", "Los Angeles", "90210", "123 Captain America St"
18 •    "15", "Noelani", "Wong", "010-12345678", "noelani.wong@avengers.com", "1989-07-15", "USA", "San Francisco", "94109", "123 Captain America St"
19 •    "16", "Eric", "Reed", "010-12345678", "eric.reed@avengers.com", "1984-08-25", "USA", "Los Angeles", "90210", "123 Captain America St"
19 •    "17", "Bruce", "Banner", "010-12345678", "bruce.banner@avengers.com", "1972-02-15", "USA", "Chicago", "60601", "123 Captain America St"

```

Output

#	Time	Action	Message
270	12:10:00	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) V...	1 row(s) affected
271	12:10:00	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) V...	1 row(s) affected
272	12:10:00	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) V...	1 row(s) affected
273	12:10:00	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) V...	1 row(s) affected
274	12:10:00	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) V...	1 row(s) affected
275	12:10:00	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) V...	1 row(s) affected
276	12:10:00	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) V...	1 row(s) affected

Finalment he de canviar el nom de *email* a *personal_email* tal com ens diu al diagrama de l'enunciat:

Per això faig:

```

110      -- Canvio el nom email a personal_email
111 •    ALTER TABLE user RENAME COLUMN email TO personal_email;
112

```

#	Time	Action	Message
1	13:14:24	ALTER TABLE user CHANGE COLUMN email personal_email VARCHAR(150)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

Faig un SELECT per veure quants files té la taula *user*

```

151 • USE transactions;
152 • SELECT
153 *
154 FROM user;
155

```

Result Grid

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.

user 23

Output

#	Time	Action	Message
1	13:16:12	USE transactions	0 row(s) affected
2	13:16:12	SELECT * FROM user LIMIT 0, 50000	275 row(s) returned

Així veiem que la taula *user* té 275 files.

4.1.3 Creació relació taula *user* amb taula *transaction*

```

155 • ALTER TABLE transaction ADD CONSTRAINT fk_user_id FOREIGN KEY transaction(user_id) REFERENCES user(id);
156
157

```

Output

#	Time	Action	Message	Duration / Fetch
1	12:32:08	ALTER TABLE transaction ADD CONSTRAINT fk_user_id FOREIGN KEY transaction(user_id) REFERENCES user(id)	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('transactions`.`#sql-17f8_25', CONSTRAINT `fk_user_id` FOREIGN KEY(`user_id`) REFERENCES `user`(`id`))	0.031 sec

Veig que surt un error, probablement perquè la taula *transaction* té *user_id* que no es troben a la taula *user*.

12:32:08 ALTER TABLE transaction ADD CONSTRAINT fk_user_id FOREIGN KEY transaction(user_id) REFERENCES user(id) Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('transactions`.`#sql-17f8_25', CONSTRAINT `fk_user_id` FOREIGN KEY(`user_id`) REFERENCES `user`(`id`)) 0.031 sec

En primer lloc miro quants usuaris de la taula *transaction* no es troben a la taula *user*. Es clar és l'usuari 9999 que havíem entrat en exercicis anteriors. Però miro si poden haver més *users_id* que estiguin a *transaction* i no estiguin a la taula *user*.

```

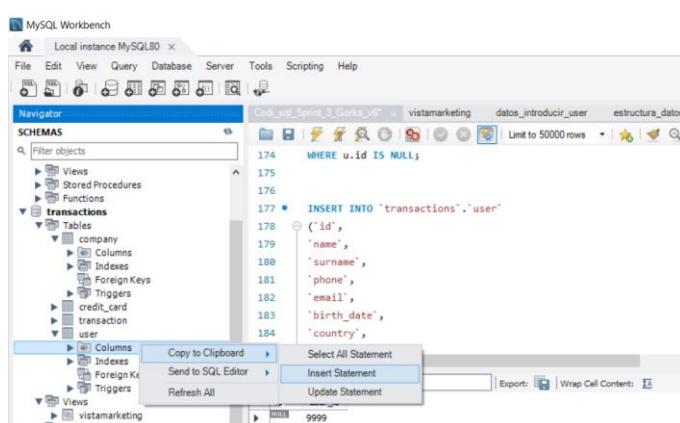
167      -- Miro quants user_id de la taula transaction no estan a la taula user
168 • USE transactions;
169 • SELECT
170   u.id,
171   t.user_id
172   FROM user u
173   RIGHT JOIN transaction t ON u.id=t.user_id
174   WHERE u.id IS NULL;
175
176
177

```

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

	id	user_id
▶	HULL	9999

El que faig és insertar un nou usuari a la taula *user*. Primer faig *Insert Statement* del Workbench per tal que em generi un codi orientatiu i així hagi de picar menys codi.



Ara modifco el codi per afegir el registre que havia dit.

```

177      -- Inserto l'element que està en la taula transaction i no està a la taula user
178 • USE transactions;
179 • INSERT INTO user
180   (id,name,surname,phone,personal_email,birth_date,country,city,postal_code,address)
181   VALUES
182   (9999,NULL,NULL,NULL,NULL,NULL,NULL,NULL,NULL);
183
184
185
186
187
188
189
190
191
192 • SHOW CREATE TABLE company;

```

Action Output

#	Time	Action	Message
1	13:17:50	USE transactions	0 row(s) affected
2	13:17:50	INSERT INTO user (id,name,surname,phone,personal_email,birth_date,country,city,postal_code,address) ...	1 row(s) affected

Ara en teoria podré crear la relació externa entre *user_id* i *id* que abans ens ha fallat.

```

184      -- Intento afegir clau externa a user_id respecte id de la taula user però falla
185 • ALTER TABLE transaction ADD CONSTRAINT fk_user_id FOREIGN KEY transaction(user_id) REFERENCES user(id);
186
< Output : Action Output
# Time Action
1 13:19:57 ALTER TABLEtransaction ADD CONSTRAINTfk_user_id FOREIGN KEYtransaction(user_id) REFERE... Message
587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0

187 • USE transactions;
188 • SHOW CREATE TABLE transaction;
189
< Result Grid | Filter Rows: Export: Wrap Cell Content: Table Create Table
transaction CREATE TABLE `transaction` ( `id` varchar(255) NOT NULL, `credit_card_id` varchar(15) DEFAULT NULL, `company_id` varc

```

```

< Result 24 x
Output : Action Output
# Time Action Message
1 13:21:09 USE transactions 0 row(s) affected
2 13:21:09 SHOW CREATE TABLE transaction 1 row(s) returned

```

Faig SHOW CREATE TABLE per veure la nova clau externa a la taula *transaction*:

Edit Data for Create Table (VARCHAR)

Binary	Text
--------	------

```

1 CREATE TABLE `transaction` (
2   `id` varchar(255) NOT NULL,
3   `credit_card_id` varchar(15) DEFAULT NULL,
4   `company_id` varchar(20) DEFAULT NULL,
5   `user_id` int DEFAULT NULL,
6   `lat` float DEFAULT NULL,
7   `longitude` float DEFAULT NULL,
8   `timestamp` timestamp NULL DEFAULT NULL,
9   `amount` decimal(10,2) DEFAULT NULL,
10  `declined` tinyint(1) DEFAULT NULL,
11  PRIMARY KEY (`id`),
12  KEY `company_id` (`company_id`),
13  KEY `transaction` (`credit_card_id`),
14  KEY `idx_user_id` (`user_id`),
15  CONSTRAINT `fk_credit_card_id` FOREIGN KEY (`credit_card_id`) REFERENCES `credit_card` (`id`),
16  CONSTRAINT `fk_user_id` FOREIGN KEY (`user_id`) REFERENCES `user` (`id`),
17  CONSTRAINT `transaction_ibfk_1` FOREIGN KEY (`company_id`) REFERENCES `company` (`id`)
18 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

```

Data Length: 815 bytes

4.1.4 Comprovacions taula *company*

Faig SHOW CREATE TABLE

```
143      -- NIVELL 3
144      -- EXERCICI 1
145 •  SHOW CREATE TABLE company;
```

		Result Grid	Filter Rows:	Export:	Wrap Cell Content:
	Table	Create Table			
▶	company	CREATE TABLE `company` (`id` varchar(15) NOT NULL, `company_name` varchar(255) DEFAULT NULL, `phone` varchar(15) DEFAULT NULL, `email` varchar(100) DEFAULT NULL, `country` varchar(100) DEFAULT NULL, `website` varchar(255) DEFAULT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci			

Result 2		
Output		
Action Output		Message
#	Time	Action
✓ 1	10:48:23	SHOW CREATE TABLE company

1 row(s) return

Edit Data for Create Table (VARCHAR)

Binary Text

```
1 CREATE TABLE `company` (
2   `id` varchar(15) NOT NULL,
3   `company_name` varchar(255) DEFAULT NULL,
4   `phone` varchar(15) DEFAULT NULL,
5   `email` varchar(100) DEFAULT NULL,
6   `country` varchar(100) DEFAULT NULL,
7   `website` varchar(255) DEFAULT NULL,
8   PRIMARY KEY (`id`)
9 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

Data Length: 336 bytes

Save... Close

I l'únic que veig comparant amb el diagrama de l'enunciat es que he d'eliminar la columna *website*.

```

194      -- Elimino columna website
195 • USE transactions;
196 • ALTER TABLE company DROP COLUMN website;
197

```

Output:

#	Time	Action
1	13:29:16	USE transactions
2	13:29:16	ALTER TABLE company DROP COLUMN website

4.1.5 Comprovacions taula *transaction* i relacions amb altres taules

Fent un SHOW CREATE TABLE veiem que té tots els camps del diagrama i les tres Claus externes (fk_credit_card_id, fk_user_id, transaction_ibfk_1) amb les tres altres taules del diagrama. El id també es clau primària com ha de ser.

Edit Data for Create Table (VARCHAR)

Binary Text

```

2      `id` varchar(255) NOT NULL,
3      `credit_card_id` varchar(15) DEFAULT NULL,
4      `company_id` varchar(20) DEFAULT NULL,
5      `user_id` int DEFAULT NULL,
6      `lat` float DEFAULT NULL,
7      `longitude` float DEFAULT NULL,
8      `timestamp` timestamp NULL DEFAULT NULL,
9      `amount` decimal(10,2) DEFAULT NULL,
10     `declined` tinyint(1) DEFAULT NULL,
11     PRIMARY KEY (`id`),
12     KEY `company_id` (`company_id`),
13     KEY `transaction` (`credit_card_id`),
14     KEY `idx_user_id` (`user_id`),
15     CONSTRAINT `fk_credit_card_id` FOREIGN KEY (`credit_card_id`) REFERENCES `credit_card` (`id`),
16     CONSTRAINT `fk_user_id` FOREIGN KEY (`user_id`) REFERENCES `user` (`id`),
17     CONSTRAINT `transaction_ibfk_1` FOREIGN KEY (`company_id`) REFERENCES `company` (`id`)
18   ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

```

Data Length: 815 bytes

4.1.6 Comprovacions taula *credit_card*

```
201 • USE transactions;
202 • SHOW CREATE TABLE credit_card;
```

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

Table	Create Table
credit_card	CREATE TABLE `credit_card` (`id` varchar(8) NOT NULL, `iban` varchar(50) NOT NULL, `pin` smallint NOT NULL, `cvv` smallint NOT NULL, `expiring_date` date DEFAULT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

Result 1 ×

Output

Action Output

#	Time	Action	Message
1	13:44:16	USE transactions	0 row(s) affected
2	13:44:23	SHOW CREATE TABLE credit_card	1 row(s) returned

Edit Data for Create Table (VARCHAR)

Binary Text

```
1 CREATE TABLE `credit_card` (
2   `id` varchar(8) NOT NULL,
3   `iban` varchar(50) NOT NULL,
4   `pin` smallint NOT NULL,
5   `cvv` smallint NOT NULL,
6   `expiring_date` date DEFAULT NULL,
7   PRIMARY KEY (`id`)
8 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci
```

Data Length: 266 bytes

El camp id s'ha de modificar la longitud de VARCHAR com el diagrama. També modifiquem el tipus del camp *pin* a VARCHAR(4) com al diagrama de l'enunciat i el cvv a tipus INT. Com que SMALLINT permet valors més petits que INT no hi haurà problema amb aquesta segona conversió.

```

212 • USE transactions;
213 • ALTER TABLE credit_card MODIFY id VARCHAR(20) NOT NULL;
214 • ALTER TABLE credit_card MODIFY pin VARCHAR(4) NOT NULL;
215 • ALTER TABLE credit_card MODIFY cvv INT NOT NULL;
216

```

✓	1	23:44:46	USE transactions
✓	2	23:44:46	ALTER TABLE credit_card MODIFY id VARCHAR(20) NOT NULL
✓	3	23:44:46	ALTER TABLE credit_card MODIFY pin VARCHAR(4) NOT NULL
✓	4	23:44:46	ALTER TABLE credit_card MODIFY cvv INT NOT NULL

El problema seria el camp *pin* que tingués més dígits que 4. Anem a comprovar si hem perdut registres amb la conversió:

```

213      -- Per veure si perdo valors al canviar el pin de SMALLINT a VARCHAR(4)
214 • USE transactions;
215 • SELECT
216     COUNT(*)
217     FROM credit_card c
218     WHERE LENGTH(c.pin)=4;
219

```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
COUNT(*)				
275				

Result 4			
Output			
Action Output			Message
#	Time	Action	
✓	1	13:30:01 USE transactions	0 row(s) affected
✓	2	13:30:01 SELECT COUNT(*) FROM credit_card c WHERE LENGTH(c.pin)=4 LIMIT 0, 50000	1 row(s) returned

No hem perdut valors, doncs la taula *credit_card* té 276 files, i una de elles té un registre amb tot de NULLS i amb un id= 'CcU-9999' per això la query anterior ens ha donat 275 files.

4.1.6.1 Convertir columna expiring_date de DATE a VARCHAR(10)

La variable `expiring_date` inicialment la vaig convertir a DATE [2.1.3 Modificar expiring_date a tipus DATE](#) (exercicis Nivell 1), ja que en ningun moment em deia que no podia ser DATE, i preferia que tingués un format DATE per aprendre més.

Ara bé, en aquest exercici, m'indiquen que `expiring_date` sigui de tipus VARCHAR(8) segons el diagrama de l'enunciat.

Aquests són els resultats quan `expiring_date` és del tipus DATE.

	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	4556 453 55 5287	4598	438	2021-06-29
	CcU-2959	CR7242477244335841535	372461377349375	3583	667	2023-02-24
	CcU-2966	BG72LKTO15627628377363	448566 886747 7265	4900	130	2024-10-29

Aleshores executaré el següent codi, per tal que `expiring_date` passi al format inicial VARCHAR. Haurem de convertir a VARCHAR(10) i no pas VARCHAR(8), ja que al haver convertit al Nivell 1 `expiring_date` a DATE, ara aquest camp té valors del tipus `aaaa-mm-dd`, i per tant a caràcters ocuparà 10 posicions com a mínim. De fet quan he executat el codi amb VARCHAR(8) em fallava.

Per fer servir UPDATE d'una columna m'he de crear una altra columna. Aleshores `expiring_date` serà la columna de VARCHAR(10) i l'actual columna `expiring_date` de tipus DATE l'anomeno com `expiring_date_old`.

Com he explicat abans, he desactivat SQL_SAFE_UPDATES, ja que el UPDATE no utilitza un filtre WHERE.

```

232 • USE transactions;
233 • ALTER TABLE credit_card RENAME COLUMN expiring_date TO expiring_date_old;
234 • ALTER TABLE credit_card ADD COLUMN expiring_date VARCHAR(10);
235 • SET SQL_SAFE_UPDATES=0;
236 • UPDATE credit_card SET expiring_date = DATE_FORMAT(expiring_date_old, '%m/%d/%Y');
237 • ALTER TABLE credit_card DROP expiring_date_old;
238 • SET SQL_SAFE_UPDATES=1;
239
240

```

✓	37	16:12:04	SET SQL_SAFE_UPDATES=0
✓	38	16:12:04	UPDATE credit_card SET expiring_date = DATE_FORMAT(expiring_date_old, "%m/%d/%Y")
✓	39	16:12:04	SET SQL_SAFE_UPDATES=1
✓	40	16:12:26	SELECT * FROM transactions.credit_card LIMIT 0, 50000
✓	41	16:14:21	ALTER TABLE credit_card DROP expiring_date_old

Ja tenim el camp `expiring_date` de tipus VARCHAR(10):

	id	iban	pan	pin	cvv	expiring_date
▶	CcU-2938	R323456312213576817699999	5424465566813633	3257	984	10/30/2022
	CcU-2945	DO26854763748537475216568689	5142423821948828	9080	887	08/24/2023
	CcU-2952	BG45IVQL52710525608255	4556 453 55 5287	4598	438	06/29/2021
	CcU-2959	CR7242477244335841535	372461377349375	3583	667	02/24/2023
	CcU-2966	BG72LKTQ15627628377363	448566 886747 7265	4900	130	10/29/2024
	CcU-2973	PT87806228135092429456346	544 58654 54343 384	8760	887	01/30/2025
	CcU-2980	DE39241881883086277136	402400 7145845969	5075	596	07/24/2022

Ara per variar de l'esquema del Workbench podem veure els nous tipus de les variables tal com al diagrama.

4.1.6.2 Afegir columna *fecha_actual*

Ara faltaria afegir una variable *fecha_actual* tal com el diagrama, que l'omplim amb la data d'acabament de l'exercici.

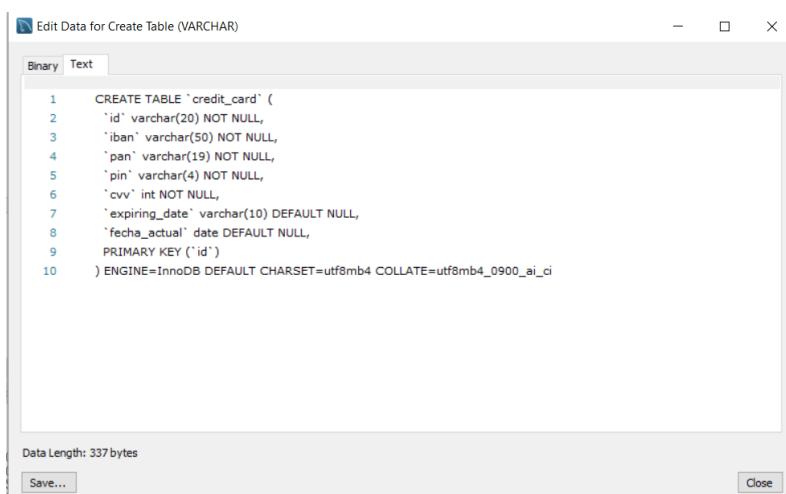
```

233 • USE transactions;
234 • ALTER TABLE credit_card ADD COLUMN fecha_actual DATE NULL;
235 • SET SQL_SAFE_UPDATES=0;
236 • UPDATE credit_card SET fecha_actual=CURDATE();
237 • SET SQL_SAFE_UPDATES=1;

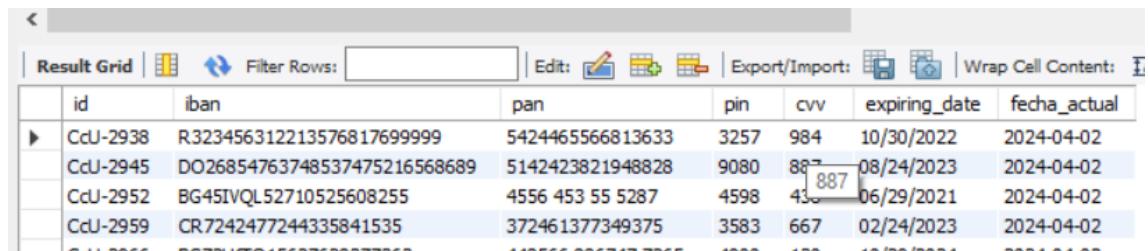
✓ 8 17:05:46 ALTER TABLE credit_card ADD COLUMN fecha_actual DATE NULL          0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✗ 9 17:06:35 UPDATE credit_card SET fecha_actual=CURDATE()                   Error Code: 1175. You are using safe update mode and you tried to update a table without a WHERE
✓ 10 17:06:55 SET SQL_SAFE_UPDATES=0                                         0 row(s) affected
✓ 11 17:06:55 UPDATE credit_card SET fecha_actual=CURDATE()                  276 row(s) affected Rows matched: 276 Changed: 276 Warnings: 0
✓ 12 17:06:56 SET SQL_SAFE_UPDATES=1                                         0 row(s) affected

```

Tenim les columnes i els tipus segons els diagrama de l'enunciat.



Aquí podem veure alguns registres, amb la *fecha_actual* la del dia de l'acabament de l'exercici.



	id	iban	pan	pin	cvv	expiring_date	fecha_actual
▶	CcU-2938	R323456312213576817699999	5424465566813633	3257	984	10/30/2022	2024-04-02
	CdU-2945	DO26854763748537475216568689	5142423821948828	9080	887	08/24/2023	2024-04-02
	CdU-2952	BG45IVQL52710525608255	4556 453 55 5287	4598	43	06/29/2021	2024-04-02
	CdU-2959	CR7242477244335841535	372461377349375	3583	667	02/24/2023	2024-04-02

5 ANNEXOS

5.1 Creació de la taula de l'Exercici 1 Nivell 1

Per la longitud dels diferents camps de la taula *credit_card*, m'he basat una mica en informació de la xarxa.

<https://wise.com/es/help/articles/2968880/paises-sepa-y-numeros-iban>

Pel número *iban* hem considerat 30 caràcters com a màxim.

<https://www.paycomet.com/news/glosario/que-es-el-pan/#:~:text=El%20PAN%20ac%C3%B3nimo%20de%20Personal,d%C3%A9bito%20virtuales%20y%20de%20prepago.>

El *pan* és el número de la targeta que com a molt és de 19 díigits..