

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.

Creemos una tabla con los siguientes campos, ID, Iban, Pan, Pin, Cvv, Expiring date donde ID será la primary key.

También he incluido una columna llamada Update_date la cual recoge la fecha en que se actualiza el sistema

```
CREATE TABLE credit_card (  
    Id VARCHAR(15) PRIMARY KEY,  
    Iban VARCHAR(40) NOT NULL UNIQUE,  
    Pan VARCHAR(30) NOT NULL,  
    Pin INT NOT NULL,  
    Cvv INT NOT NULL,  
    expiring_date VARCHAR(10) NOT NULL,  
    Update_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP);
```

También he incluido una columna llamada Update_date la cual recoge la fecha en que se actualiza el sistema. Este campo lo he añadido porque en los siguientes ejercicios se nos pide que conste la actualización de datos.

Ahora tenemos que indicar que el campo credit_card_id de la tabla transactions, será una foreign key de la tabla credit_Card. Eso lo haremos mediante el siguiente comando

```
ALTER TABLE transaction
```

```
ADD CONSTRAINT FK_Credit_card FOREIGN KEY (credit_card_id) REFERENCES credit_card (Id)
```

+++++++CORRECCIONES+++++++

Vamos a hacer un cambio en los campos CVV y PIN para que los mismos pase a ser VARCHAR. Estos Datos son esenciales para la seguridad del sistema de tarjetas y la información ha de grabarse bien.

Haremos el cambio con un ALTER TABLE

```
ALTER TABLE credit_card
```

```
MODIFY CVV VARCHAR(3) NOT NULL;
```

```
ALTER TABLE credit_card
```

```
MODIFY PIN VARCHAR(4) NOT NULL;
```

Para ver que se los datos se han grabado correctamente:

49 • DESCRIBE credit_Card;

50 /*****

<

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

	Field	Type	Null	Key	Default	Extra
▶	Id	varchar(20)	NO	PRI	NULL	
	Iban	varchar(50)	YES	UNI	NULL	
	PIN	varchar(4)	NO		NULL	
	CVV	varchar(3)	NO		NULL	
	expiring_date	varchar(10)	NO		NULL	
	fecha_actual	date	YES		NULL	

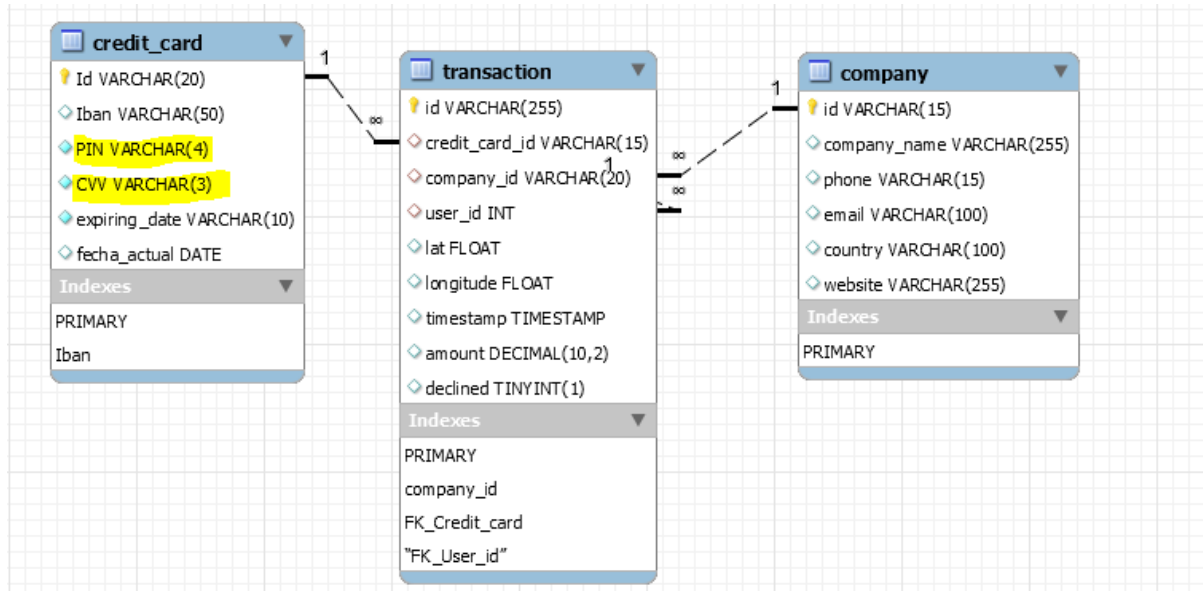
Result 3 x

Output

Action Output

#	Time	Action
✓ 1	17:14:22	DESCRIBE credit_Card

Quedarà el siguiente esquema Entidad-Relación:



+++++CORRECCIONES+++++

- Exercici 2

El departament de Recursos Humans ha identificat un error en el número de compte de 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.

Miramos la información que tiene el usuario

```

43 • select * from credit_Card
44 where id ="CcU-2938";

```

Result Grid							
Filter Rows: <input type="text"/>							
Edit: Export/Import: Wrap Cell Content:							
	Id	Iban	Pan	Pin	Cvv	expiring_date	Update_date
▶	CcU-2938	TR301950312213576817638661	5424465566813633	3257	984	10/30/22	2024-05-30 16:38:57
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

credit Card 6 x

Output

Action Output

#	Time	Action	Message
✓ 1	17:14:55	select * from credit_Card where id ="CcU-2938" LIMIT 0, 50000	1 row(s) returned

Cambiamos la información con el comando UPDATE

```

46 • UPDATE credit_Card
47 SET Iban = "R323456312213576817699999", Update_date = current_timestamp()
48 WHERE id = "CcU-2938";

```

Result Grid							
Filter Rows: <input type="text"/>							
Edit: Export/Import: Wrap Cell Content:							
	Id	Iban	Pan	Pin	Cvv	expiring_date	Update_date
▶	CcU-2938	R323456312213576817699999	5424465566813633	3257	984	10/30/22	2024-05-30 17:26:20
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

credit Card 12 x

Output

Action Output

#	Time	Action	Message
✓ 1	17:26:56	SELECT * FROM credit_Card WHERE id ="CcU-2938" LIMIT 0, 50000	1 row(s) returned

En el Update_Date camp, se puede ver el día y la hora en que se ha actualizado dicho registro.

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

Usamos el comando INSERT para incluir el nuevo registro con los datos solicitados

INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined)

VALUES ('108B1D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', '9999', '829.999', '-117.999', '111.11', '0');

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

Action Output		
Time	Action	Message
1 17:26:56	SELECT * FROM credit_Card WHERE id ="CcU-2938" LIMIT 0, 50000	1 row(s) returned
2 17:39:49	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timesta...	Error Code: 1292. Incorrect datetime value: "for column 'timestamp' at row 1
3 17:40:13	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timesta...	Error Code: 1064. You have an error in your SQL syntax; check the manual that correspon...
4 17:42:03	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount....	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (transac...

/*****correcciones*****/

Obtenemos un error. Ese error se debe a que el nuevo registro que queremos introducir tiene tres campos ("credit_card_id", "company_id" y "User_id") que no existen en las tablas maestros.

Procederemos deshabilitar temporalmente las foreign key, insertar el registro y volver a insertar las foreign key*/

ALTER TABLE transaction
DROP FOREIGN KEY transaction_ibfk_1;

```
ALTER TABLE transaction
DROP FOREIGN KEY FK_Credit_card;
```

```
ALTER TABLE TRANSACTION
DROP FOREIGN KEY "FK_User_id";
```

```
ALTER TABLE TRANSACTION
DROP PRIMARY KEY;
```

Volvemos a insertar de nuevo el registro

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

Consulta para ver que se ha grabado correctamente:

109 • `select * from transaction;`
110

	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
	108B1D1D-5B23-A76C-55EF-C568E49A05DD	CcU-2938	b-2222	275	83.7839	-178.86	2021-07-07 17:43:16	293.57	0
	17561134-37A4-E316-2C3C-C5541EDAD761	CcU-2938	b-2362	92	-10.8529	18.0969	2021-09-15 05:23:32	371.57	0
	185C9F2C-67A2-8EC0-0BB6-93789CC59F05	CcU-2938	b-2362	92	9.55928	165.726	2021-07-25 12:34:59	285.24	0
	18B2472B-DA8B-36F7-B0F2-7DDA688B73D1	CcU-2938	b-2362	87	-41.7722	98.2897	2021-09-24 08:33:44	391.38	0
	26D8A738-B651-A620-1EEA-D8E2CD353BEB	CcU-2938	b-2362	92	15.6872	25.2507	2021-05-28 15:21:36	238.99	0
	4DA5B1CE-E88C-23D1-AE6A-B7589E4FB2D5	CcU-2938	b-2362	82	39.2187	63.5563	2021-03-28 05:01:44	378.86	0

transaction 28 x

Output

Action Output

#	Time	Action	Message
1	18:24:42	select * from transaction LIMIT 0, 50000	587 row(s) returned

Ahora tenemos que volver a insertar todas las foreign keys en la tabla transaction

```
ALTER TABLE TRANSACTION
ADD CONSTRAINT FK_Company FOREIGN KEY (company_id) REFERENCES company(id);
```

```
ALTER TABLE TRANSACTION
ADD CONSTRAINT FK_Credit_card FOREIGN KEY (credit_card_id) REFERENCES credit_card(id);
```

```
ALTER TABLE TRANSACTION
ADD CONSTRAINT FK_User_id FOREIGN KEY (user_id) REFERENCES data_user(id);
```

```
ALTER TABLE TRANSACTION
```

ADD PRIMARY KEY (id);

/*****

- Exercici 4

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

```
74 • ALTER TABLE credit_card
75 DROP COLUMN PAN;
76
77 • select * from credit_Card
```

Id	Iban	Pin	Cvv	expiring_date	Update_date
CdU-2938	R323456312213576817699999	3257	984	10/30/22	2024-05-30 17:26:20
CdU-2945	DO26854763748537475216568689	9080	887	08/24/23	2024-05-30 16:38:58
CdU-2952	BG45IVQL52710525608255	4598	438	06/29/21	2024-05-30 16:38:58
CdU-2959	CR7242477244335841535	3583	667	02/24/23	2024-05-30 16:38:58
CdU-2966	BG72LKTQ15627628377363	4900	130	10/29/24	2024-05-30 16:38:58

#	Time	Action	Message
4	17:42:03	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amou...	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (trans...
5	17:52:51	select * from credit_Card LIMIT 0, 50000	275 row(s) returned
6	17:53:28	ALTER TABLE credit_card DROP COLUMN PAN	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
7	17:53:33	select * from credit_Card LIMIT 0, 50000	275 row(s) returned

Nivell 2

Exercici 1

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

Primero tenemos que localizar la transacción.

```
select * from transaction
where id = "02C6201E-D90A-1859-B4EE-88D2986D3B02";
```

Borraremos dicha transacción con el el statement delete. Hay que tener en cuenta no olvidar incluir una cláusula WHERE donde indiquemos el ID que queremos eliminar. De lo contrario, borraremos todos los registros de la tabla.

```
DELETE FROM TRANSACTIONS.transaction
WHERE id = "02C6201E-D90A-1859-B4EE-88D2986D3B02";
```

En estos momentos hay 587 registros. Al borrarlo deberemos obtener 586.

97 • `select * from transaction; # aparecen 586 registros.`

Result Grid								
Filter Rows:								
Edit: Export/Import: Wrap Cell Content:								
id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
0466A42E-47CF-8D24-FD01-C0B689713128	CcU-4219	b-2302	170	-43.9695	-117.525	2021-07-26 07:29:18	49.53	0
063FBA79-99EC-66FB-29F7-25726D1764A5	CcU-2987	b-2250	275	-81.2227	-129.05	2022-01-06 21:25:27	92.61	0
0668296C-CDB9-A883-76BC-2E4C44F8C8AE	CcU-3743	b-2618	265	-34.3593	-100.556	2022-01-26 02:07:14	394.18	0
06CD9AA5-9B42-D684-DDDD-ASE394FEBA99	CcU-2959	b-2346	92	33.7381	158.298	2021-10-26 23:00:01	279.93	0
07A46D48-31A3-7E87-65B9-0DA902AD109F	CcU-3225	b-2386	272	38.8342	92.1905	2021-06-28 21:11:42	340.87	1
09DE92CE-6F27-2BB7-13B5-9385B2B3B8E2	CcU-3071	b-2298	275	71.1706	10.5757	2021-05-11 20:40:06	303.05	1
...

#	Time	Action	Message
8	09:54:26	DELETE FROM TRANSACTIONS.transaction WHERE id = "02C6201E-D90A-1859-B4EE-88D2986D3B02"	0 row(s) affected
9	09:54:55	DELETE FROM TRANSACTIONS.transaction WHERE id = "02C6201E-D90A-1859-B4EE-88D2986D3B02"	0 row(s) affected
10	09:56:21	select * from transaction where id = "02C6201E-D90A-1859-B4EE-88D2986D3B02" #	0 row(s) returned
11	09:57:17	select * from transaction LIMIT 0, 50000	586 row(s) returned

Al buscar de nuevo la transacción ya no debe aparecer

101 • `SELECT * FROM transaction`

102 `WHERE id = "02C6201E-D90A-1859-B4EE-88D2986D3B02"; # Obtenemos NULL.`

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 14 x

Output			
#	Time	Action	Message
✓ 14	10:17:24	SELECT DISTINCT company.country FROM transactions.transaction LEFT JOIN transa...	15 row(s) returned
✓ 15	10:17:34	SELECT* FROM transactions.transaction LEFT JOIN transactions.company ON transact...	586 row(s) returned
✓ 16	10:19:34	SELECT * FROM transaction WHERE id = "02C6201E-D90A-1859-B4EE-88D2986D3B02"	0 row(s) returned
✓ 17	10:20:43	SELECT * FROM transaction WHERE id = "02C6201E-D90A-1859-B4EE-88D2986D3B02"	0 row(s) returned

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.

Tendremos que unir las tablas Transactions y Company con un JOIN, filtrar el campo decline para que sólo muestre las transacciones realizadas agrupamos los datos por Nombre de Compañía, Teléfono y País y calcular el importe promedio de las ventas. Finalmente, guardaremos los datos en una vista

/***CORRECCIÓN *****/**

Para eliminar el order by de la view Vistamarketing la he modificado con el siguiente comando

```
CREATE OR REPLACE VIEW VistaMarketing AS
```

```
SELECT    company.company_name    ,    company.phone    ,    company.country    ,  
AVG(transaction.amount) AS Sales_Average
```

```
FROM transactions.transaction
```

```
LEFT JOIN transactions.company
```

```
ON transaction.company_id = company.id
```

```
WHERE transaction.declined = 0
```

```
GROUP BY company.company_name , company.phone , company.country;
```

```

193 • CREATE OR REPLACE VIEW VistaMarketing AS
194 SELECT company.company_name , company.phone , company.country , AVG(transaction.amount)
195 FROM transactions.transaction
196 LEFT JOIN transactions.company
197 ON transaction.company_id = company.id
198 WHERE transaction.declined = 0
199 GROUP BY company.company_name , company.phone , company.country;

```

200

Result Grid				
	company_name	phone	country	Sales_Average
▶	Nunc Interdum Incorporated	05 18 15 48 13	Germany	242.947692
	Amet Nulla Donec Corporation	07 15 25 14 74	Italy	92.610000
	Non Institute	06 77 15 31 14	United Kingdom	297.041333
	Ut Semper Foundation	01 60 36 33 06	Sweden	277.972241
	Ac Fermentum Incorporated	06 85 56 52 33	Germany	293.570000
	Ac Fermentum	06 85 56 52 3	Germany	111.110000
	Lorem Eu Incorporated	01 83 66 62 07	Canada	258.644615

vistamarketing 48 x

Output				
	Action Output			
#	Time	Action	Message	
✓ 1	19:27:29	CREATE OR REPLACE VIEW VistaMarketing AS SELECT company.company_name , co...	0 row(s) affected	
✓ 2	19:27:35	SELECT * FROM vistamarketing LIMIT 0, 50000	101 row(s) returned	

Hacemos la consulta que nos piden, incluyendo el order by Sales average.

```

203 • SELECT * FROM vistamarketing

```

```

204 order by 4 desc;

```

205

```

206 /*****

```

Result Grid				
	company_name	phone	country	Sales_Average
▶	Eget Ipsum Ltd	03 67 44 56 72	United States	481.860000
	Sed Id Limited	07 28 18 18 13	United States	477.510000
	Neque Tellus Incorporated	04 43 18 34 19	Ireland	477.100000
	Nunc Sit Incorporated	07 28 42 63 63	Norway	461.830000
	Non Magna LLC	06 71 73 13 17	United Kingdom	458.740000
	Maecenas Malesuada Fringilla Inc.	09 38 53 76 61	Netherlands	451.290000
	Erat LLP	03 18 88 77 79	Netherlands	448.440000

vistamarketing 51 x

Output				
	Action Output			
#	Time	Action	Message	
✓ 1	19:29:22	SELECT * FROM vistamarketing order by 4 desc LIMIT 0, 50000	101 row(s) returned	

```

/*****
*****/

```

Exercici 3

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

Haremos una consulta usando la vista "Vistamarketing" como si fuera una tabla más y filtrando por país = "Germany"

```
131 • SELECT *
132 FROM vistamarketing
133 WHERE country = "Germany";
```

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

	company_name	phone	country	Sales_Average
▶	Ac Industries	09 34 65 40 60	Germany	396.150000
	Auctor Mauris Corp.	05 62 87 14 41	Germany	308.990000
	Ac Fermentum Incorporated	06 85 56 52 33	Germany	293.570000
	Aliquam PC	01 45 73 52 16	Germany	280.340000
	Rutrum Non Inc.	02 66 31 61 09	Germany	266.900000
	Nunc Interdum Incorporated	05 18 15 48 13	Germany	242.947692

vistamarketing 5 x

Output

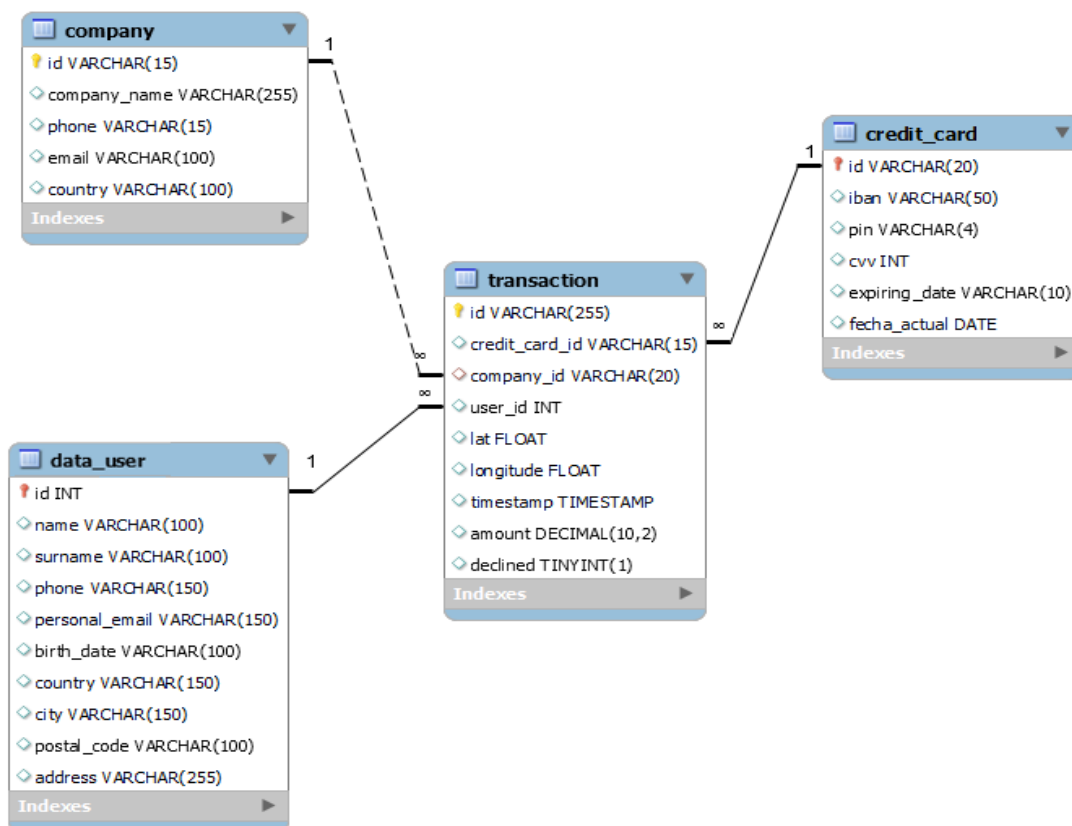
Action Output

#	Time	Action	Message
✓ 1	12:10:42	SELECT * FROM vistamarketing LIMIT 0, 50000	100 row(s) returned
✓ 2	12:14:45	SELECT * FROM vistamarketing WHERE country = "Germany" LIMIT 0, 50000	8 row(s) returned

Nivell 3

Exercici 1

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



Modificaciones:

1 - Mi compañero ha creado una base de datos llamado usuarios ("users" para registrar de manera única a los diferentes usuarios que tenemos y un detalle de información suya (Nombre, apellidos, teléfono, fecha de nacimientos, dirección postal y de mail)

Pero se ha equivocado definir el campo "id" como primary key y como foreign Key al mismo tiempo. Eso es contradictorio. Un campo es foreign key o es primary key pero no pueden las dos cosas a la misma vez.

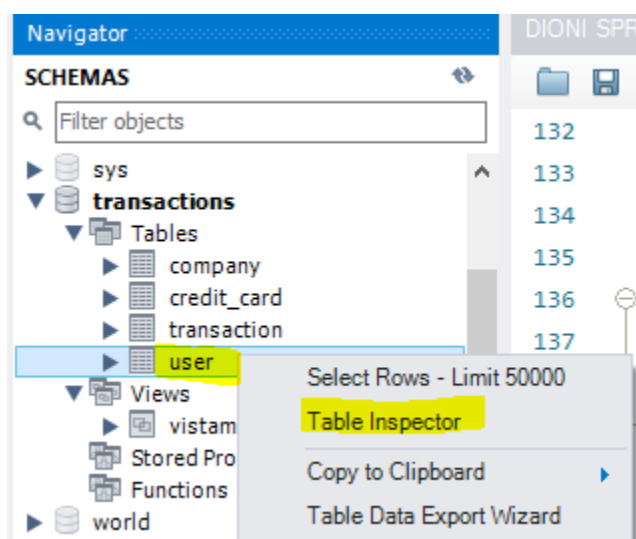
En este caso no el campo ID es un primary key. Con lo cual hay que eliminar el constraint del foreign key.

ALTER TABLE user

DROP FOREIGN KEY user_ibfk_1;

Cuando mi compañero creo la constraint foregin key no le asignó ningún nombre. Pero automáticamente el MySQL le ha puesto el nombre "user_ibfk_1".

El nombre de la foreign key constraint lo he sacado del table inspector en MYSQL



DIONI SPRINT 2 DATA ANALY...		SQL File 2*		DIONI SPRINT 1		SQL File 5*		transactions.user		transactions.user					
Info		Columns		Indexes		Triggers		Foreign keys		Partitions		Grants		DDL	
Name				Schema		Table		Column		Referenced Sch...		Referenced Table		Referenced Col...	
user_ibfk_1				transactions		user		id		transactions		transaction		user_id	

Una vez ejecutada el comando, procedemos a comprobar que la foreign key se ha eliminado. Para ello, entramos en el inspector de tabla, pestaña DDL y allí veremos que la restricción de foreign key ya no está.

DIONI SPRINT 2 DATA ANALY...

SQL File 2*

DIONI SPRINT 1

SQL File 5*

transactions.user

transactions.user x

Info

Columns

Indexes

Triggers

Foreign keys

Partitions

Grants

DDL

DDL for transactions.user

1

2

3

4

5

6

7

8

9

10

11

12

13

CREATE TABLE `user` (

`id` int NOT NULL,

`name` varchar(100) DEFAULT NULL,

`surname` varchar(100) DEFAULT NULL,

`phone` varchar(150) DEFAULT NULL,

`email` varchar(150) DEFAULT NULL,

`birth_date` varchar(100) DEFAULT NULL,

`country` varchar(150) DEFAULT NULL,

`city` varchar(150) DEFAULT NULL,

`postal_code` varchar(100) DEFAULT NULL,

`address` varchar(255) DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci

2 - Por otra parte, ha hecho un cambio en la tabla transactions. Ha creado un índice en la tabla transaction en el campo USER_ID.

```
CREATE INDEX idx_user_id ON transaction(user_id);
```

El mismo se puede observar en la tabla inspector de la tabla transaction

DIONI SPRINT 2 DATA ANALY...

SQL File 2*

DIONI SPRINT 1

SQL File 5*

transactions.user

transactions.user

transactions.transaction x

Info

Columns

Indexes

Triggers

Foreign keys

Partitions

Grants

DDL

Indexes in Table

Visible	Key	Type	Unique	Columns
<input checked="" type="checkbox"/>	PRIMARY	BTREE	YES	id
<input checked="" type="checkbox"/>	company_id	BTREE	NO	company_id
<input checked="" type="checkbox"/>	FK_Credit_card	BTREE	NO	credit_card_id
<input checked="" type="checkbox"/>	idx_user_id	BTREE	NO	user_id

Index Details

Key Name:

Index Type:

Allows NULL:

Cardinality:

Comment:

User Comment:

Packed:

Unique:

Columns in table

Column	Type	Nullable	Indexes
id	varchar(255)	NO	PRIMARY
credit_card_id	varchar(15)	YES	FK_Credit_card
company_id	varchar(20)	YES	company_id
user_id	int	YES	idx_user_id
lat	float	YES	
longitude	float	YES	
timestamp	timestamp	YES	
amount	decimal(10,2)	YES	
declined	tinyint(1)	YES	

Create

Ahora que hemos creado la tabla "user", el campo "user_id " de la tabla transaction pasa a ser un foreign key de la tabla maestro user.

Eliminamos el índice de dicho campo mediante un alter table

```
ALTER TABLE transaction
DROP INDEX idx_user_id;
```

Ahora tenemos que convertir el campo "user_id" de la tabla Transaction en una foreign key relacionada con la tabla user recientemente creada.

```
ALTER TABLE TRANSACTION
ADD CONSTRAINT "FK_User_id" FOREIGN KEY (user_id) REFERENCES user(id).
```

3 - El campo ID de la tabla credit_card lo hemos creado como VARCHAR(15) pero ha de ser VARCHAR(20). Modificaremos estos con los siguientes comandos.

```
ALTER TABLE credit_Card.
```

```
MODIFY COLUMN Id VARCHAR(20)
```

4 - El campo Iban de la tabla credit_card lo hemos creado como VARCHAR(40) pero ha de ser VARCHAR(50). Modificaremos estos con los siguientes comandos.

Primero cambiamos el nombre de la columna

```
ALTER TABLE credit_card
```

```
RENAME COLUMN UPDATE_DATE TO FECHA_ACTUAL
```

Segundo cambiamos el tipo de datos.

```
ALTER TABLE credit_card
```

```
MODIFY COLUMN Iban VARCHAR(50)
```

5- El campo pin de la tabla credit_card lo hemos como INT pero ha de ser VARCHAR(4). Modificaremos estos con los siguientes comandos.

```
ALTER TABLE credit_card
```

```
MODIFY COLUMN pin VARCHAR(4)
```

6 – El campo fecha_actual de la tabla credit_card lo hemos creado como TIMESTAMP pero ha de ser DATE. También tenemos que cambiar el nombre del campo.

Primero cambiamos el nombre de la columna

```
ALTER TABLE credit_card
```

```
RENAME COLUMN UPDATE_DATE TO FECHA_ACTUAL;
```

Segundo cambiamos el tipo de datos de TIMESTAMP a DATE

```
ALTER TABLE credit_card
```

```
MODIFY COLUMN fecha_actual DATE;
```

7 – Tenemos que cambiar el nombre de la tabla User a Data_User

```
RENAME table user TO Data_User;
```

Los campos Cvv y expiring_date lo hemos creado correctamente de la misma forma que indica la tabla

/***CORRECCIONES*****/**

8 - Se nos pide que cambiemos el nombre de la columna email de la tabla data_user por personal_email.

```
ALTER TABLE data_user
```

```
RENAME COLUMN email TO personal_email;
```

9 - Eliminamos la columna website

```
ALTER TABLE COMPANY
```

```
DROP COLUMN WEBSITE;
```

SELECT * FROM company;

```

301 • select * from company;
302 /*****
303

```

id	company_name	phone	email	country
b-2222	Ac Fermentum Incorporated	06 85 56 52 33	donec.porttitor.tellus@yahoo.net	Germany
b-2226	Magna A Neque Industries	04 14 44 64 62	risus.donec.nibh@icloud.org	Australia
b-2230	Fusce Corp.	08 14 97 58 85	risus@protonmail.edu	United States
b-2234	Convallis In Incorporated	06 66 57 29 50	mauris.ut@aol.couk	Germany
b-2238	Ante Iaculis Nec Foundation	08 23 04 99 53	sed.dictum.proin@outlook.ca	New Zealand
b-2242	Donec Ltd	01 25 51 37 37	at.iaculis@hotmail.couk	Norway
b-2246	Sed Nunc Ltd	02 62 64 73 48	nibh@vahoo.ora	United Kinodom

company 57 x

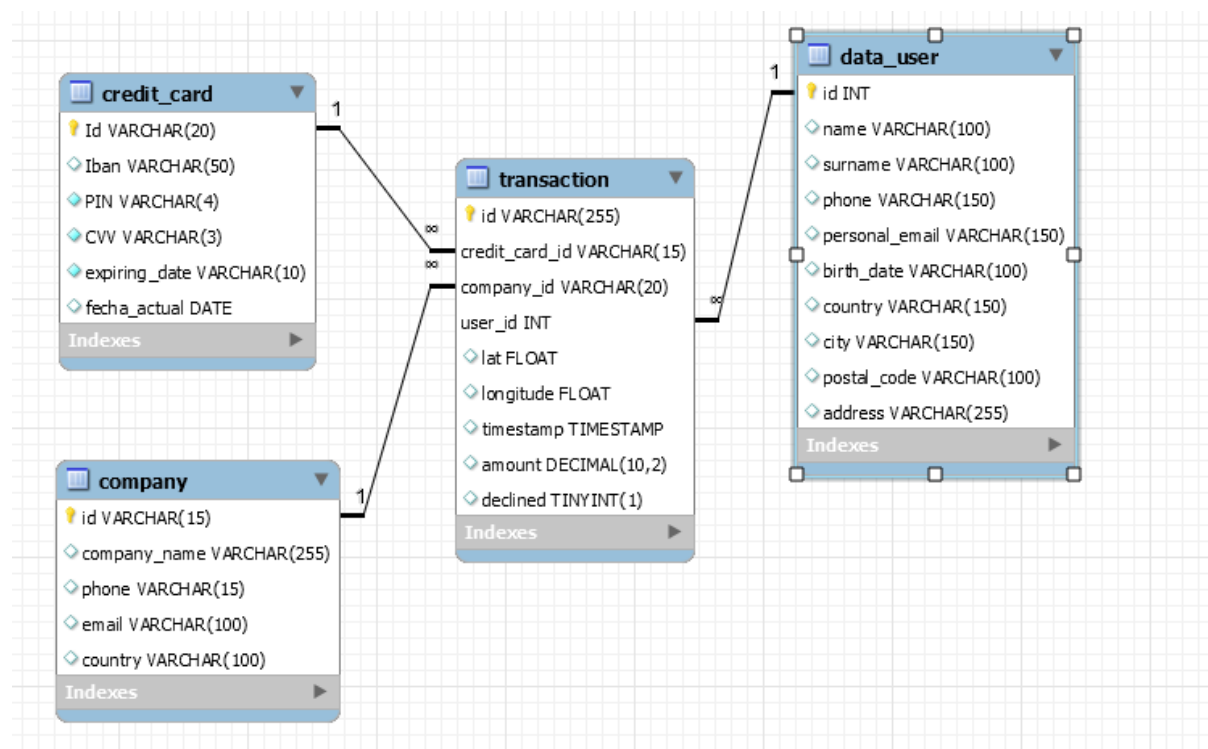
Output

Action Output

#	Time	Action	Message
1	19:43:56	select * from company LIMIT 0, 50000	101 row(s) returned

/***/

Al final el modelo queda de la siguiente forma:



d

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.

Tendremos que unir las tablas Transactions y Company con un JOIN, seleccionamos los campos requeridos y además añadimos el país y el importe de la compra porque me parecen datos relevantes. Finalmente, guardaremos los datos en una vista.

```
CREATE VIEW InformeTecnico AS
SELECT transaction.id AS ID_Transaccio, data_user.name AS Nom_usuari, data_user.surname AS Cognom,
credit_card.Iban as IBAN, company.company_name AS Company_name, transaction.amount AS Importe_compra ,company.country AS PAIS
FROM transactions.transaction
LEFT JOIN transactions.company
ON transaction.company_id = company.id
LEFT JOIN transactions.data_user
ON transaction.user_id = data_user.id
LEFT JOIN transactions.credit_card
ON transaction.credit_card_id = credit_card.Id
ORDER BY 1 DESC;
```

236 • `SELECT * FROM informetecnico;`

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

ID_Transaccio	Nom_usuari	Cognom	IBAN	Company_name	Importe_compra	PAIS
FE96CE47-8D59-381C-4E18-E3CA3D44E8FF	Kenyon	Hartman	DO26854763748537475216568689	Magna A Neque Industries	480.13	Australia
FE809ED4-2DB6-55AC-C915-929516E46468	Molly	Gilliam	SE2813123487163628531121	Nunc Interdum Incorporated	219.83	Germany
FD9CBCCD-8E1E-8DA1-4606-7E3A6F3A5A65	Linus	Willis	KW9485332754781757886242955643	Nunc Interdum Incorporated	42.32	Germany
FD89D51B-AE8D-77DC-E450-B8083FBD3187	Hilda	Levy	LT053237077744561475	Malesuada PC	200.72	Ireland
FD2E8957-414B-BEEC-E9AD-59AA7A8A6290	Hedwig	Gilbert	GE84848451582810541526	Neque Tellus Imperdiet Corp.	78.29	Ireland
FCE2AB9A-271D-2BDC-9E49-8DD92A373391	Hakeem	Alford	MD1234119525145401270486	Nunc Interdum Incorporated	335.56	Germany

informetecnico 13 x

Output

Action Output

#	Time	Action	Message
21	11:30:45	CREATE VIEW InformeTecnico AS SELECT transaction.id AS ID_Transaccio, data_u...	Error Code: 1050. Table 'InformeTecnico' already exists
22	11:30:56	SELECT * FROM informetecnico LIMIT 0, 50000	586 row(s) returned

1950