

Alumna: Milene Quiles Gonçalves

Sprint 3

Nivell 1

Ex 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.

A continuación describo los pasos seguidos:

1) Yo crio la tabla "credit_card"

```
1  -- creamos table credit_card
2
3  •  USE transactions;
4
5  •  CREATE TABLE IF NOT EXISTS `credit_card` (
6      `id` VARCHAR(45) NOT NULL,
7      `iban` VARCHAR(45) NOT NULL,
8      `pan` VARCHAR(45) NULL,
9      `pin` VARCHAR(45) NULL,
10     `cvv` VARCHAR(45) NULL,
11     `expiring_date` VARCHAR(45) NULL,
12     PRIMARY KEY (`id`));
```

2) Luego, inserto los datos del archivo "datos_introducir_credit".

Ex 2. El departament de Recursos Humans ha identificat un error en el número de compte de l'usuari amb el: IBAN CcU-2938. Es requereix actualitzar la informació que identifica un compte bancari a nivell internacional (identificat com "IBAN"): TR323456312213576817699999. Recorda mostrar que el canvi es va realitzar.

Result Grid

Filter Rows:

Edit:

Export/Import:

	id	iban	pan	pin	cvv	expiring_date
▶	CcU-2938	TR301950312213576817638661	5424465566813633	3257	984	10/30/22
★	NULL	NULL	NULL	NULL	NULL	NULL

```
UPDATE credit_card
SET iban = "TR323456312213576817699999"
WHERE id = "CcU-2938";
```

Result Grid

Filter Rows:

Edit:

Export/Import:

	id	iban	pan	pin	cvv	expiring_date
▶	CcU-2938	TR323456312213576817699999	5424465566813633	3257	984	10/30/22
✱	NULL	NULL	NULL	NULL	NULL	NULL

Ex 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

Para poder adicionar los datos a la tabla transaction, tuve primero que adicionar el registro de los datos foreign key en las tablas relacionadas. A continuación describo los pasos seguidos:

1) El código

```

12  -- EX 3.
13  • INSERT INTO company (id) value ("b-9999");
14
15  • INSERT INTO credit_card (id) value ("CcU-9999");
16
17  • INSERT INTO transaction (Id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) values (
18      "10881D1D-5B23-A76C-55EF-C568E49A99DD", "CcU-9999", "b-9999", "9999", "829.999", "-117.999", "111.11", "0");
19

```

2) El resultado

```

1  • SELECT * FROM transaction
2  WHERE ID = "10881D1D-5B23-A76C-55EF-C568E49A99DD";

```

Result Grid									
Filter Rows:									
Edit: Export/Import: Wrap Cell Content:									
	id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
▶	10881D1D-5B23-A76C-55EF-C568E49A99DD	CcU-9999	b-9999	9999	829.999	-117.999	NULL	111.11	0
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Ex 4. Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Recordar mostrar el canvi realitzat.

```

22  -- EX 4.
23  • ALTER TABLE credit_card
24  DROP COLUMN pan;
25
26  • DESCRIBE credit_card;

```

Result Grid					
Filter Rows:					
Export:					
	Field	Type	Null	Key	Default
▶	id	varchar(45)	NO	PRI	NULL
	iban	varchar(45)	YES		NULL
	pin	varchar(45)	YES		NULL
	cvv	varchar(45)	YES		NULL
	expiring_date	varchar(45)	YES		NULL

Nivell 2

Ex 1. Elimina de la taula transaction el registre amb ID 02C6201E-D90A-1859-B4EE-88D2986D3B02 de la base de dades

1) Pesquisando los datos de la transacción antes de eliminarlos

```
33 • SELECT *
34 FROM
35 transaction
36 WHERE id = "02C6201E-D90A-1859-B4EE-88D2986D3B02";
37
```

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

2) El código y el resultado despues de ejecutarlo

```
31 -- EX 1.
32
33 • DELETE FROM transaction
34 WHERE id = "02C6201E-D90A-1859-B4EE-88D2986D3B02";
35
36 • SELECT *
37 FROM
38 transaction
39 WHERE id = "02C6201E-D90A-1859-B4EE-88D2986D3B02";
40
```



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

Ex 2. La sección de marketing desea tener acceso a información específica para realizar análisis y estrategias efectivas. Se ha solicitado crear una vista que proporcione detalles clave sobre las compañías y sus transacciones. Será necesaria que crees una vista llamada VistaMarketing que contenga la siguiente información: Nombre de la compañía. Teléfono de contacto. País de residencia. Promedio de compra realizado por cada compañía. Presenta la vista creada, ordenando los datos de mayor a menor promedio de compra.

El código utilizado para crear la vista y el resultado después de ejecutarlo

```
324      -- EX 2.
325      -- Creamos la vista del promedio de compras realizadas por cada compañía
326      • CREATE VIEW VistaMarketing AS
327      SELECT company_name, phone, country, avg_amount
328      FROM company
329      JOIN (
330          SELECT company_id, AVG(amount) AS avg_amount
331          FROM transaction
332          WHERE declined = FALSE
333          GROUP BY company_id
334      ) as avg_transaction
335      ON company.id = avg_transaction.company_id
336      ORDER BY avg_amount DESC;
```

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

Result Grid				
Filter Rows: <input type="text"/>				
Export:  Wrap Cell Content: 				
	company_name	phone	country	avg_amount
▶	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
	Tortor Nunc Commodo Company	05 35 92 77 16	United States	447.110000
	Justo Eu Arcu Ltd	08 42 56 71 52	Italy	444.160000
	Pede Cum Ltd	07 62 26 48 38	Norway	442.320000
	Vestibulum Lorem PC	02 02 87 33 40	Belgium	428.400000
	Mauris Institute	05 29 60 36 87	Sweden	427.710000
	Aliquet Diam Limited	02 76 61 47 46	United States	425.640000
	Mus Aenean Eget Foundation	06 25 15 52 43	Sweden	419.970000
	Sed LLC	01 63 16 26 52	Belgium	416.660000
	Vivamus Dapibus Foundation	02 22 42 22 22	United Kingdom	414.520000

stamarketing 1 x

Ex 3. Filtra la vista VistaMarketing per a mostrar només les companyies que tenen el seu país de residència en "Germany"

```
339      -- EX 3.
340      -- Filtro las compañías de Alemania
341      • SELECT * FROM transactions.vistamarketing
342      WHERE country = "Germany";
343
344
345
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
company_name	phone	country	avg_amount
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
Convallis In Incorporated	06 66 57 29 50	Germany	60.990000
Augue Foundation	06 88 43 15 63	Germany	15.050000

Nivell 3

Ex 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 les següents modificacions (s'espera que realitzin 6 canvis):

- 1) *Creemos la tabla users, con su primary key, foreign key y un index en transaction(user_id)*
- 2) *Insertamos los datos de user (del archivo datos_introducir_user)*

```

59      -- EX 1.
60      -- 1) Creamos la tabla user
61
62      ● CREATE INDEX idx_user_id ON transaction(user_id);
63
64      ● CREATE TABLE IF NOT EXISTS user (
65          id INT PRIMARY KEY,
66          name VARCHAR(100),
67          surname VARCHAR(100),
68          phone VARCHAR(150),
69          email VARCHAR(150),
70          birth_date VARCHAR(100),
71          country VARCHAR(150),
72          city VARCHAR(150),
73          postal_code VARCHAR(100),
74          address VARCHAR(255),
75          FOREIGN KEY(id) REFERENCES transaction(user_id)
76      );
77
78      ● SET foreign_key_checks = 0;
79
80      -- 2) Insertamos datos de user
81      ● INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES
82      ● INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES
83      ● INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES
84      ● INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES

```

3) Eliminamos la campo website de la tabla company:

4) Cambiamos el nombre de la columna email para personal_email en la tabla user

5) cambiamos el tipo de datos de los campos: credit_card (id) a VARCHAR(20), iban VARCHAR(50), pin VARCHAR(4), cvv INT, spire_date VARCHAR(10) y creamos el foreign key para cambiar la relacion con la tabla transaction


```

649      -- 3) Eliminamos el campo website de la tabla company
650 • ALTER TABLE company
651     DROP COLUMN website;
652
653      -- 4) cambiamos el nombre de la columna email
654 • ALTER TABLE user
655     RENAME COLUMN email TO personal_email;
656
657      -- 5) cambiamos el tipo de los datos en la tabla credit_card
658      -- y creamos el foreign key para cambiar la relacion con la tabla transaction
659 • ALTER TABLE credit_card
660     CHANGE COLUMN `id` `id` VARCHAR(20) NOT NULL ,
661     CHANGE COLUMN `iban` `iban` VARCHAR(50) NULL DEFAULT NULL ,
662     CHANGE COLUMN `pin` `pin` VARCHAR(4) NULL DEFAULT NULL ,
663     CHANGE COLUMN `cvv` `cvv` VARBINARY(4) NULL DEFAULT NULL ,
664     CHANGE COLUMN `expiring_date` `expiring_date` VARCHAR(10) NULL DEFAULT NULL;
665
666 • ALTER TABLE `transactions`.`credit_card`
667     ADD CONSTRAINT `card_transaction`
668         FOREIGN KEY (`id`)
669         REFERENCES `transactions`.`transaction` (`credit_card_id`)
670         ON DELETE NO ACTION
671         ON UPDATE NO ACTION;
672

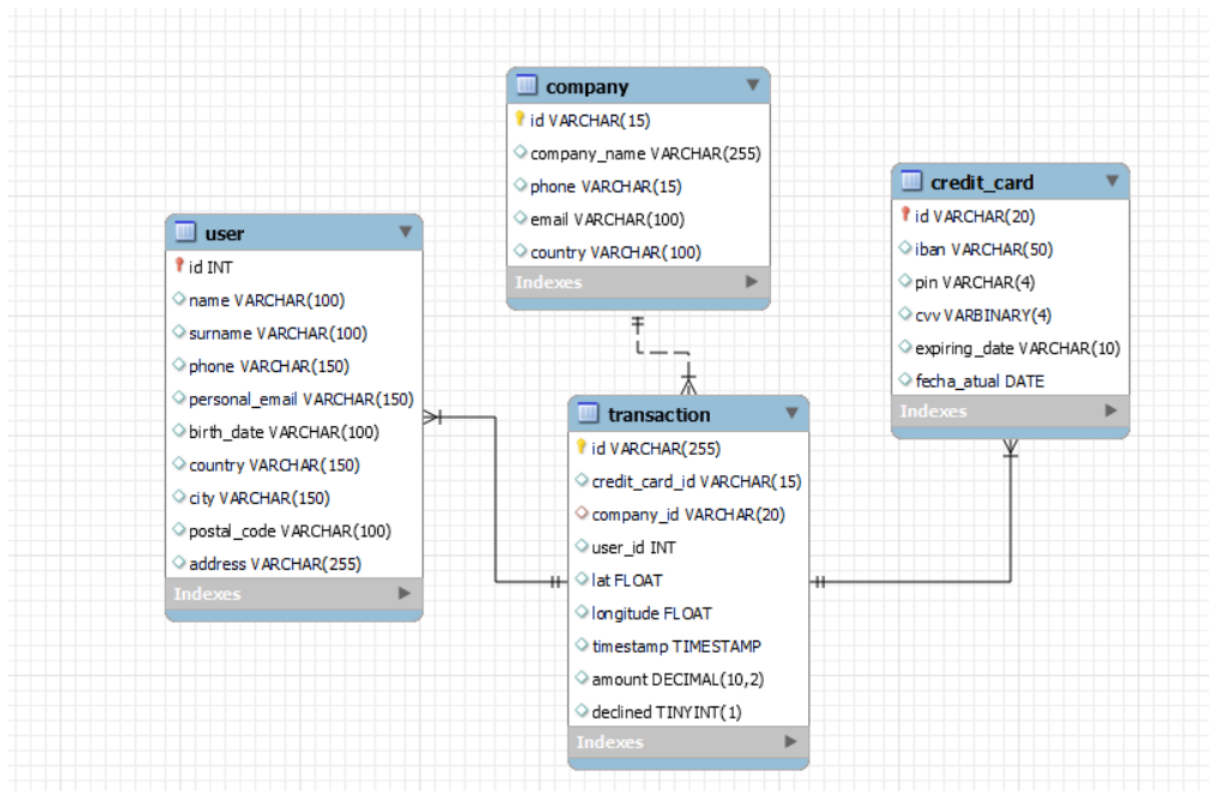
```

6) *adicionamos el campo fecha_atual DATE a credit_card*

```

669      -- 6) adicionamos columna fecha_atual a la tabla credit_card
670 • ALTER TABLE credit_card
671     ADD fecha_atual DATE;
---
```

El diagrama final del modelo:



Ex 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.

El codigo:

```
-- Ex 3.
CREATE VIEW InformeTecnico AS
SELECT
    transaction.id AS transaction_id,
    user.name AS user_name,
    user.surname AS user_surname,
    credit_card.iban AS iban_card,
    company.company_name AS company_name
FROM transaction
LEFT JOIN user ON transaction.user_id = user.id
LEFT JOIN credit_card ON transaction.credit_card_id = credit_card.ID
LEFT JOIN company ON transaction.company_id = company.ID
ORDER BY transaction_id;
```

El resultado de la vista:

1 • `SELECT * FROM transactions.informetecnico;`

transaction_id	user_name	user_surname	iban_card	company_name
0466A42E-47CF-8D24-FD01-C0B689713128	William	Benjamin	MD1234119525145401270486	Nunc Interdum Incorporated
063FBA79-99EC-66FB-29F7-25726D1764A5	Kenyon	Hartman	GE89681434837748781813	Amet Nulla Donec Corporation
0668296C-CD89-A883-76BC-2E4C44F8C8AE	Chloe	Keith	BA542358041365401657	Non Institute
06CD9AA5-9B42-D684-DDDD-A5E394FEB999	Lynn	Riddle	CR7242477244335841535	Ut Semper Foundation
07A46D48-31A3-7E87-65B9-0DA902AD109F	Hedwig	Gilbert	FI9398462343991818	Lacus Quisque Associates
09DE92CE-6F27-2BB7-13B5-9385B2B388E2	Kenyon	Hartman	NO8923814763512	Elit Etiam Laoreet Associates
0A476ED9-0C13-1962-F87B-D35639248539	Sasha	Emerson	PK1842602717357586645371	Nunc Interdum Incorporated
08EB80B7-9D66-1707-CE4B-9DC7E71914B5	Hedwig	Gilbert	CH4437804777669672438	Magna Incorporated
0C7C3A33-9947-3BC1-846D-7BE3D0D17598	Hedwig	Gilbert	CR2081908447841410311	Tristique Neque Venenatis Institute
0CE957A6-CCAA-2B7A-6839-8A4B1B324853	Haley	Fitzpatrick	LV78EROL4956681623444	Sapient Nunc Pulvinar LLP
0DD2E608-5C9E-D1B3-4999-B99F43AD735A	Kenyon	Hartman	CR7242477244335841535	Convallis In Incorporated
1017AA59-3D5F-7A4C-1992-D151A8D1FA0A	Ocean	Nelson	CY042153714905187371339...	Non Institute

InformeTecnico 1 x