

# Nivell 1

## N1-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.

Creem la taula amb contingut varchar per evitar possibles errors de conversió de format. Definim ID com la clau principal.

```
1 • DROP TABLE credit_card;
2 • CREATE TABLE credit_card
3   (id VARCHAR(8) ,
4    iban VARCHAR(40),
5    pan VARCHAR(25),
6    pin VARCHAR(4),
7    cvv VARCHAR(3),
8    expiring_date VARCHAR(20),
9    PRIMARY KEY (id));
10
```

✓ 4 17:08:23 CREATE TABLE credit\_card (id VARCHAR(8) , iban VARCHAR(40), pan VARCHAR(25), pin VARCHAR(4), cvv VARCHAR(3), expiring\_date VAR... 0 row(s) affected

Un cop creada la taula, importem les dades del fitxer sql “datos\_introducir\_credit”

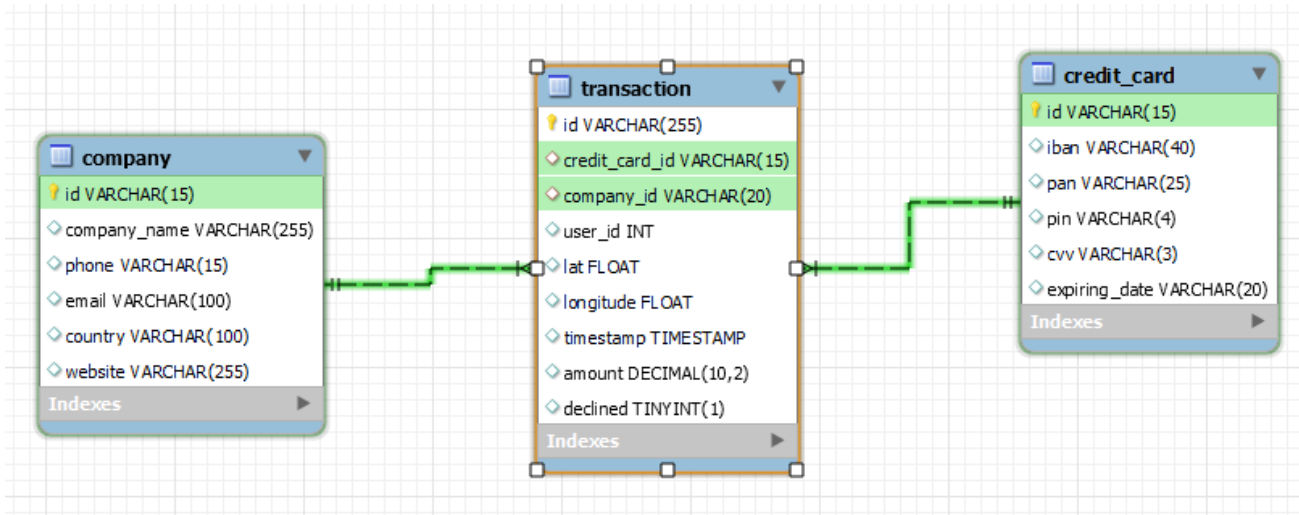
Un cop la taula ja té registres, podem relacionar-ne la clau principal amb la clau forània de la taula transaction, que correspon a l’ID de la targeta.

```
ALTER TABLE transaction
ADD CONSTRAINT FK_ccid
FOREIGN KEY(credit_card_id) REFERENCES credit_card(id);
```

289 17:32:11 ALTER TABLE transaction ADD CONSTRAINT FK\_ccid FOREIGN KEY(credit\_card\_id) REFERENCES credit\_card(id)

587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0

L’esquema resultant inclou una taula principal de fets, “transaction” que es relaciona amb les dues taules de dimensions, “company” i “credit\_card”.



## N1-Exercici 2

El departament de Recursos Humans ha identificat un error en el número de compte de l'usuari amb ID CcU-2938. La informació que ha de mostrar-se per a aquest registre és: R323456312213576817699999. Recorda mostrar que el canvi es va realitzar.

Faig un update del registre que té la ID ccu-2938 i li canvio a l'iban correcte

```

1 • UPDATE transactions.credit_card
2   SET iban = 'R323456312213576817699999'
3   WHERE id = "CcU-2938";
  
```

292 17:55:53 UPDATE transactions.credit\_card SET iban = 'R323456312213576817699999' WHERE id = "CcU-2938" 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0

Per mostrar que s'ha fet el canvi, selecciono el registre perquè es mostri en pantalla la id i l'iban:

```

SELECT id, iban
FROM transactions.credit_card
WHERE id = "CcU-2938";
  
```

|   | id       | iban                      |
|---|----------|---------------------------|
| ▶ | CcU-2938 | R323456312213576817699999 |
| • | NULL     | NULL                      |

293 17:59:20 SELECT id, iban FROM transactions.credit\_card WHERE id = "CcU-2938"

1 row(s) returned

## N1-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        |

Si intentem crear la transacció directament, dona error perquè les relacions entre taules no ens deixen crear registres on no existeixin la companyia o la targeta de crèdit.

Per tant, creo la companyia i la targeta de crèdit primer a les taules corresponents, i després introdueixo la transacció.

```
INSERT INTO company(id) VALUES ("b-9999");
```

```
INSERT INTO credit_card(ID) VALUES ("CcU-9999");
```

|   |      |          |   |                   |           |
|---|------|----------|---|-------------------|-----------|
| ✓ | 1285 | 10:58:58 | INSERT INTO company(id) VALUES ("b-9999")       | 1 row(s) affected | 0.032 sec |
| ✓ | 1286 | 10:59:00 | INSERT INTO credit_card(ID) VALUES ("CcU-9999") | 1 row(s) affected | 0.000 sec |

Faig un insert dins de la taula transaction i indico els noms de les columnes i els continguts:

```

> 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");

```

|   |      |          |   |                   |           |
|---|------|----------|---|-------------------|-----------|
| ✓ | 1287 | 10:59:03 | INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitu... | 1 row(s) affected | 0.000 sec |
|---|------|----------|---|-------------------|-----------|

## N1-Exercici 4

Des de recursos humans et solliciten eliminar la columna "pan" de la taula credit\_card. Recorda mostrar el canvi realitzat.

```
ALTER TABLE credit_card
DROP COLUMN pan;
```

|   |      |          |   |  |  |
|---|------|----------|---|--|--|
| ✓ | 1289 | 11:08:38 | ALTER TABLE credit_card DROP COLUMN pan | 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0 |  |
|---|------|----------|---|--|--|

Per confirmar que s'ha esborrat la columna, fem un show columns:

```

37 • SHOW COLUMNS
38 FROM credit_card;

```

|   | Field         | Type        | Null | Key | Default     | Extra |
|---|---------------|-------------|------|-----|-------------|-------|
| ▶ | id            | varchar(15) | NO   | PRI | <b>NULL</b> |       |
|   | iban          | varchar(40) | YES  |     | <b>NULL</b> |       |
|   | pin           | varchar(4)  | YES  |     | <b>NULL</b> |       |
|   | cvv           | varchar(3)  | YES  |     | <b>NULL</b> |       |
|   | expiring_date | varchar(20) | YES  |     | <b>NULL</b> |       |

|   |      |          |                               |                   |
|---|------|----------|-------------------------------|-------------------|
| ✓ | 1290 | 11:10:06 | SHOW COLUMNS FROM credit_card | 5 row(s) returned |
|---|------|----------|-------------------------------|-------------------|

## Nivell 2

### N2-Exercici 1

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

Eliminem la línia especificant que sigui la que conté la ID indicada:

```
DELETE FROM transaction
```

```
WHERE ID = "02C6201E-D90A-1859-B4EE-88D2986D3B02";
```

```
3 10:17:55 DELETE FROM transaction WHERE ID = "02C6201E-D90A-1859-B4EE-88D298... 1 row(s) affected
```

### N2-Exercici2

*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.*

Creem un View amb el títol VistaMarketing i a dins d'aquest fem un join de les taules "transaction" i "company" per poder treure la mitjana de la primera taula (agrupant per companyia) i la resta de dades de la taula company, i les ordenem per ordre descendent de la columna mitjana.

```
CREATE VIEW VistaMarketing AS
SELECT company_name, phone, country, AVG(amount) as mitjana
FROM transaction
JOIN company
ON transaction.company_id = company.id
WHERE declined <>0
GROUP BY company_id
ORDER BY mitjana DESC;
```

```
5 10:20:08 CREATE VIEW VistaMarketing AS SELECT company_name, phone, country, A... 0 row(s) affected
```

### N2-Exercici 3

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

Selecciono les línies on la condició Country sigui = “Germany”

```
select *  
FROM VistaMarketing  
WHERE country ="Germany";
```

|   | company_name               | phone          | country | mitjana    |
|---|----------------------------|----------------|---------|------------|
| ▶ | 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 | Germanv | 206.465000 |

6 10:21:06 select \* FROM VistaMarketing WHERE country ="Germany" 7 row(s) returned

## Nivell 3

### N3-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:*

Importem i executem els 2 scripts amb l'estructura de la taula Users i els continguts de la taula users: A l'estructura veiem que ens posa una foreign key a l'ID d'usuari, que és la primary key. Esborrem aquesta línia abans d'executar l'script:

```
CREATE TABLE IF NOT EXISTS user (  
    id INT PRIMARY KEY,  
    name VARCHAR(100),  
    surname VARCHAR(100),  
    phone VARCHAR(150),  
    email VARCHAR(150),  
    birth_date VARCHAR(100),  
    country VARCHAR(150),  
    city VARCHAR(150),  
    postal_code VARCHAR(100),  
    address VARCHAR(255),  
    FOREIGN KEY(id) REFERENCES transaction(user_id)
```

Importem el contingut de la taula (script datos\_introducir\_user)

Quan intentem crear la foreign key a la taula principal, que és la de transaccions ens dona un error perquè hi ha usuaris que no existeixen: Demanem que ens els mostri

```
SELECT user_id  
FROM transaction  
WHERE user_id NOT IN (SELECT id FROM user);
```

Com que a un exercici anterior hem creat una transacció per a l'usuari 9999 que no existeix a la taula d'usuaris, creem aquest usuari abans de crear la clau forania.

```
INSERT INTO user(id) VALUES ("9999");
```

M'he oblidat de fer l'screenshot al crear la línia, per tant ara em diu que ja existeix.

```
8 10:24:52 INSERT INTO user(id) VALUES ("9999") Error Code: 1062. Duplicate entry '9999' for key 'user.PRIMARY'
```

Creem la Foreign key que enllaci les taules:

```
ALTER TABLE transaction
ADD CONSTRAINT FK_UserID
FOREIGN KEY(user_id) REFERENCES user(id);
```

Comprovant els dos esquemes trobem les següents diferències:

A l'esquema "original", la taula user es diu data\_user, i no inclou la columna birth-date

Renombrem la taula...

```
RENAME TABLE user to data_user;
```

```
9 10:38:00 RENAME TABLE user to data_user 0 row(s) affected
```

I n'eliminem la columna birth-date

```
11 10:40:29 ALTER TABLE data_user DROP COLUMN birth_date 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
```

La taula company no inclou la columna Website. L'eliminem

```
ALTER TABLE company
DROP COLUMN website;
```

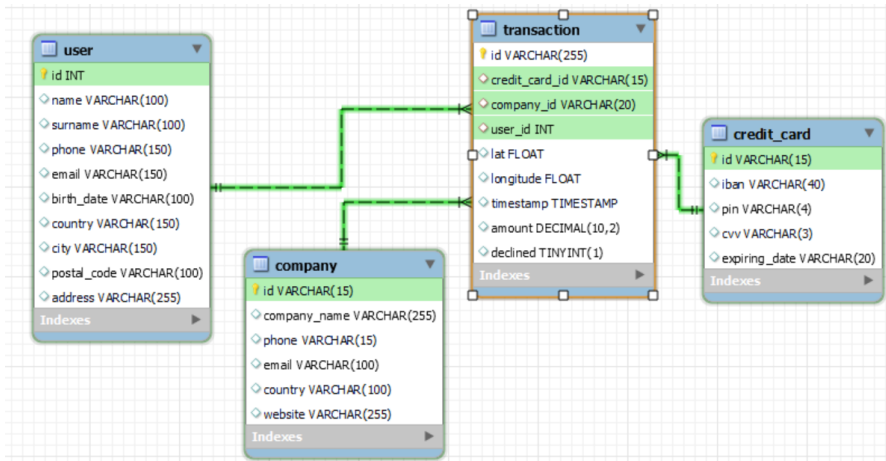
```
12 10:41:30 ALTER TABLE company DROP COLUMN website 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
```

A la taula credit\_card, la longitud de l'id és de 20 enlloc de 15, l'iban és de 50, el cvv és INT i a més hi ha una columna amb la data actual: fecha\_actual DATE. Fem tots els canvis amb Alter Table, modify i Add:

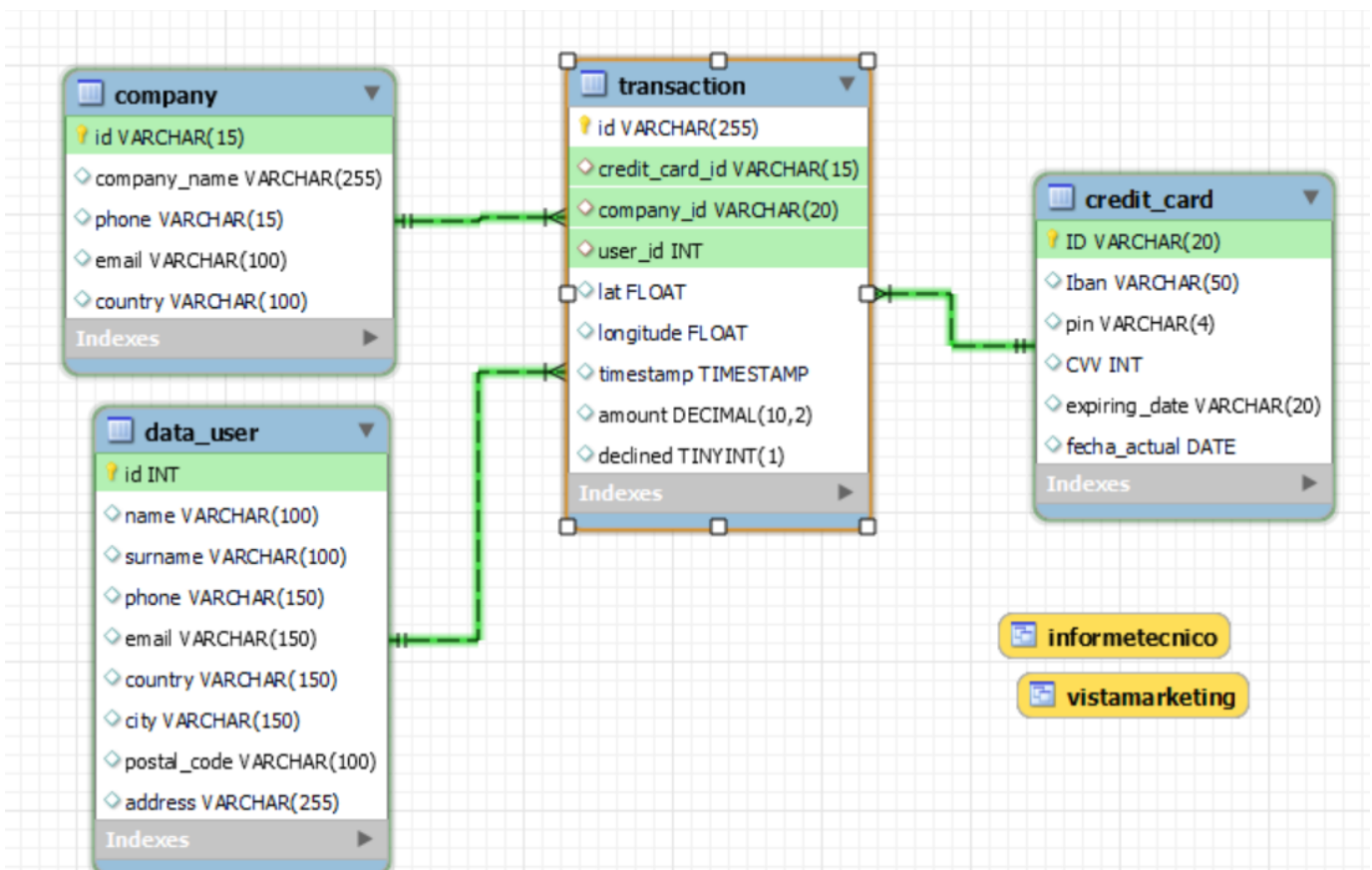
```
ALTER TABLE credit_card
MODIFY ID VARCHAR(20),
MODIFY Iban VARCHAR(50),
MODIFY CVV INT,
ADD fecha_actual DATE;
```

```
16 10:52:59 ALTER TABLE credit_card MODIFY ID VARCHAR(20), MODIFY Iban VARCH... 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
```

Abans dels canvis:



Després dels canvis:



## N3-Exercici 2

L'empresa també et sollicita 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.*

Fem un View amb un select que ajunti les columnes de les taules que contenen la diferent informació requerida.

```
CREATE VIEW InformeTecnico AS
SELECT transaction.id AS ID_transaccio, user.name AS nom_usuari, user.surname AS cognom_usuari, iban, company_name
FROM transaction
JOIN user
ON transaction.user_id = user.id
JOIN company
ON transaction.company_id = company.id
JOIN credit_card
ON transaction.credit_card_id = credit_card.id;
```

✓ 1878 13:17:09 CREATE VIEW InformeTecnico AS SELECT transaction.id AS ID\_transaccio, ... 0 row(s) affected

Fem un select de la vista que hem creat, ordenada per la columna ID\_transaccio en ordre descendent:

```
SELECT*
from informeTecnico
ORDER BY ID_transaccio DESC;
```

|   | ID_transaccio                        | nom_usuari | cognom_usuari | iban                           | company_name                 |
|---|--------------------------------------|------------|---------------|--------------------------------|------------------------------|
| ▶ | FE96CE47-8D59-381C-4E18-E3CA3D44E8FF | Kenyon     | Hartman       | DO26854763748537475216568689   | Magna A Neque Industries     |
|   | FE809ED4-2DB6-55AC-C915-929516E4646B | Molly      | Gilliam       | SE2813123487163628531121       | Nunc Interdum Incorporated   |
|   | FD9CBCCD-8E1E-8DA1-4606-7E3A6F3A5A65 | Linus      | Willis        | KW9485332754781757886242955643 | Nunc Interdum Incorporated   |
|   | FD89D51B-AE8D-77DC-E450-B8083FBD3187 | Hilda      | Levy          | LT053237077744561475           | Malesuada PC                 |
|   | FD2E8957-414B-BEEC-E9AD-59AA7A8A6290 | Hedwig     | Gilbert       | GE84848451582810541526         | Neque Tellus Imperdiet Corp. |
|   | FCE2AB9A-271D-2BDC-9E49-8DD92A373391 | Hakeem     | Alford        | MD1234119525145401270486       | Nunc Interdum Incorporated   |

✓ 1879 13:20:22 SELECT\* from informeTecnico ORDER BY ID\_transaccio DESC 587 row(s) returned