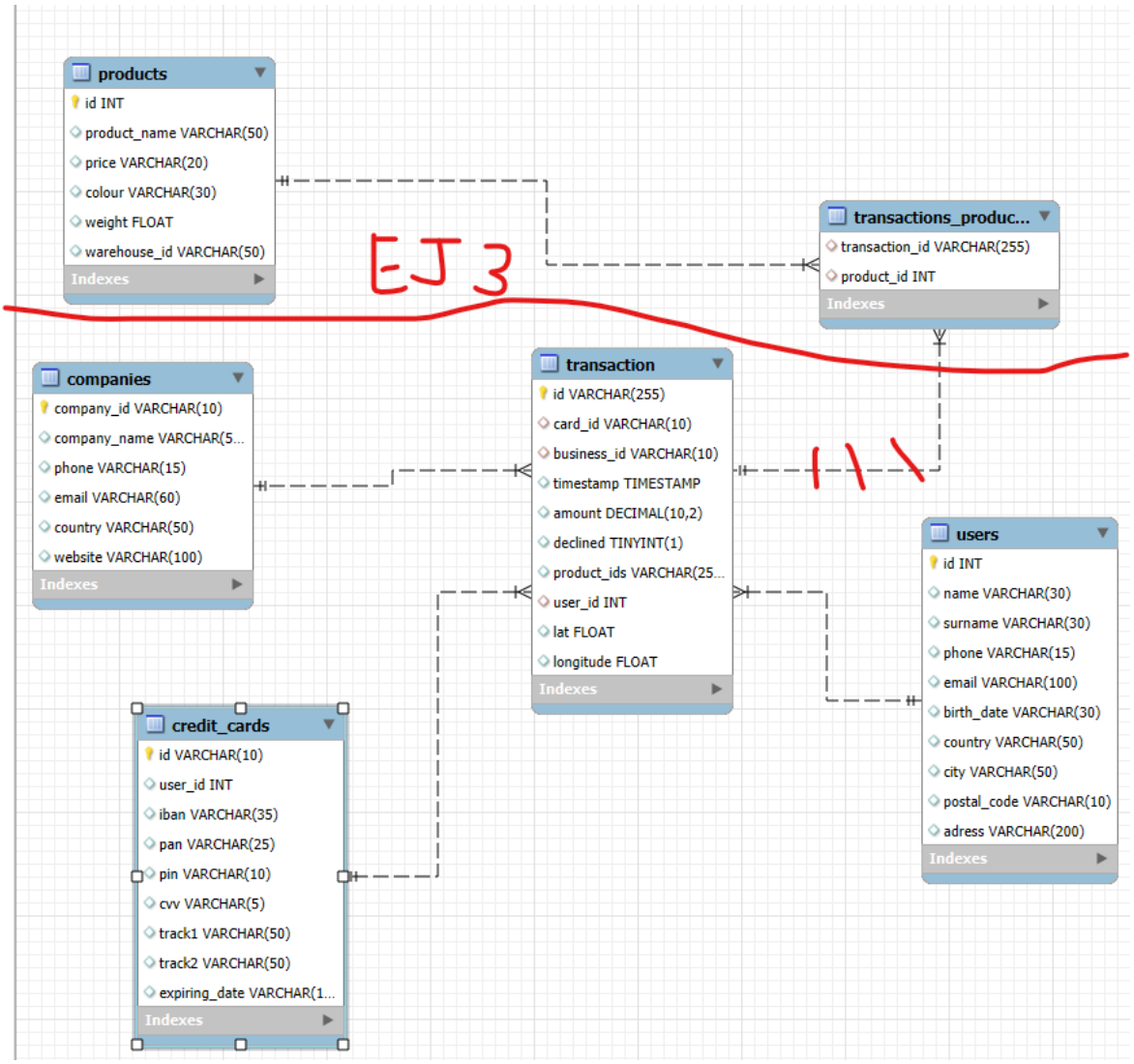
Joel Gallarte

Sprint 4

Nivell 1

Descàrrega els arxius CSV, estudia'ls i dissenya una base de dades amb un esquema d'estrella que contingui, almenys 4 taules de les quals puguis realitzar les següents consultes:

- Exercici 1



La estructura en estrella quedaría así. (No había visto que la tabla productos era para el nivel 3 y la he incluido ya en el esquema). Es un esquema en estrella con una relación de 1 a muchos desde cualquier tabla a transaction.

```sql
-- Nivell 1
-- Exercici 1
create database sprint4;
use sprint4;

create table credit_cards(
id varchar(10) primary key,
user_id int,
iban varchar(35),
pan varchar(25),
pin varchar(10),
cvv varchar(5),
track1 varchar(50),
track2 varchar(50),
expiring_date varchar(10)
);

create table companies(
company_id varchar(10) primary key,
company_name varchar(50),
phone varchar(15),
email varchar(60),
country varchar(50),
website varchar(100)
);
```

```sql
create table products(
id int primary key,
product_name varchar(50),
price varchar(20),
colour varchar(30),
weight float,
warehouse_id varchar(50)
);

create table users(
id int primary key,
name varchar(30),
surname varchar(30),
phone varchar(15),
email varchar(100),
birth_date varchar(30),
country varchar(50),
city varchar (50),
postal_code varchar(10),
adress varchar(200)
);
```

```sql
create table transaction(
id varchar(255) primary key,
card_id varchar(10),
business_id varchar(10),
timestamp timestamp,
amount decimal(10,2),
declined boolean,
product_ids varchar(255),
user_id int,
lat float,
longitude float,

foreign key (card_id) references credit_cards(id),
foreign key (user_id) references users(id),
foreign key (business_id) references companies(company_id)
);
```

```
LOAD DATA INFILE 'C:\\Users\\xXSrBiscuitXx\\Desktop\\ITACADEMY\\especializacion\\sprint4\\users_usa.csv'
INTO TABLE users
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\r\n'
ignore 1 rows;
LOAD DATA INFILE 'C:\\Users\\xXSrBiscuitXx\\Desktop\\ITACADEMY\\especializacion\\sprint4\\users_uk.csv'
INTO TABLE users
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\r\n'
ignore 1 rows;
LOAD DATA INFILE 'C:\\Users\\xXSrBiscuitXx\\Desktop\\ITACADEMY\\especializacion\\sprint4\\users_ca.csv'
INTO TABLE users
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\r\n'
ignore 1 rows;

LOAD DATA INFILE 'C:\\Users\\xXSrBiscuitXx\\Desktop\\ITACADEMY\\especializacion\\sprint4\\credit_cards.csv'
INTO TABLE credit_cards
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n'
ignore 1 rows;

LOAD DATA INFILE 'C:\\Users\\xXSrBiscuitXx\\Desktop\\ITACADEMY\\especializacion\\sprint4\\companies.csv'
INTO TABLE companies
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n'
ignore 1 rows;

LOAD DATA INFILE 'C:\\Users\\xXSrBiscuitXx\\Desktop\\ITACADEMY\\especializacion\\sprint4\\products.csv'
INTO TABLE products
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n'
ignore 1 rows;

LOAD DATA INFILE 'C:\\Users\\xXSrBiscuitXx\\Desktop\\ITACADEMY\\especializacion\\sprint4\\transactions.csv'
INTO TABLE transaction
FIELDS TERMINATED BY ';'
ENCLOSED BY '"'
LINES TERMINATED BY '\r\n'
ignore 1 rows;
```

- **Realitza una subconsulta que mostri tots els usuaris amb més de 30 transaccions utilitzant almenys 2 taules.**

```
115 •  select name, surname, conteo
116    from (select user_id as id, count(*) as conteo
117    from transaction
118    group by user_id
119    having conteo>30
120    order by conteo desc) as subquery
121    join (
122     select name,surname, id
123     from users
124    ) as users on users.id = subquery.id;
125
```

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

name	surname	conteo
Hedwig	Gilbert	76
Ocean	Nelson	52
Kenyon	Hartman	48
Lynn	Riddle	39

Result 6

Output

Action Output

#	Time	Action	Message
1	11:54:59	select name, surname, conteo from (select user_id as id, count(*) as conteo from transaction group by user_id having conteo>30 order by conteo de...	4 row(s) returned

- **Mostra la mitjana d'amount per IBAN de les targetes de crèdit a la companyia Donec Ltd, utilitza almenys 2 taules.**

```
125     -- Exercici 2
126 •   select iban, round(avg(transaction.amount),2) as mediaGasto
127     from transaction
128     join credit_cards on card_id = credit_cards.id
129     join companies on business_id = company_id
130     where company_name="Donec ltd" and declined=0
131     group by iban;
```

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

iban	mediaGasto
PT87806228135092429456346	42.82

Nivell 2

Crea una nova taula que reflecteixi l'estat de les targetes de crèdit basat en si les últimes tres transaccions van ser declinades i genera la següent consulta:

- Exercici 1

- **Quantes targetes estan actives?**

```sql
create table activeCards(
    id varchar(10),
    active bool,

    foreign key (id) references credit_cards(id)
);

insert into activeCards (id)
select id from credit_cards;

-- vista -- cardsdeclined
with transacciones_ordenadas as (
    select card_id, timestamp, declined, row_number() over (partition by card_id order by timestamp desc) as rn
    from transaction
)
select card_id
from transacciones_ordenadas
where rn <= 3  -- Solo las 3 últimas transacciones
group by card_id
having sum(case when declined = 1 then 1 else 0 end) = 3;
```

```sql
update activeCards
left join cardsdeclined cd on activeCards.id = cardsdeclined.card_id
set activeCards.active = case
    when cardsdeclined.card_id is not null then false
    else true
end;

select * from activecards;
```

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

id	active
CcU-2980	1
CcU-3001	1
CcU-3008	1
CcU-3015	1
CcU-3022	1
CcU-3029	1

totes.(275)

Nivell 3

Crea una taula amb la qual puguem unir les dades del nou arxiu products.csv amb la base de dades creada, tenint en compte que des de transaction tens product_ids. Genera la següent consulta:

- Exercici 1

- **Necessitem conèixer el nombre de vegades que s'ha venut cada producte.**

```sql
create table transactions_products(
transaction_id varchar (255),
product_id int,

foreign key (transaction_id) references transaction(id),
foreign key (product_id) references products(id)
);

insert into transactions_products (transaction_id, product_id)
select transaction.id,
       cast(trim(SUBSTRING_INDEX(SUBSTRING_INDEX(product_ids, ',', n), ',', -1)) as unsigned) as producto_id
from transaction
join (
  select 1 as n
  union all select 2
  union all select 3
  union all select 4
  union all select 5
) as numeros
on numeros.n <= 1 + length(product_ids) - length(replace(product_ids, ',', ''));

select * from transactions_products;
```

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

transaction_id	product_id
02C6201E-D90A-1859-B4EE-88D2986D3B02	19
02C6201E-D90A-1859-B4EE-88D2986D3B02	1
02C6201E-D90A-1859-B4EE-88D2986D3B02	71
0466A42E-47CF-8D24-FD01-C0B689713128	43
0466A42E-47CF-8D24-FD01-C0B689713128	97
0466A42E-47CF-8D24-FD01-C0B689713128	47
063FBA79-99EC-66FB-29F7-25726D1764A5	5

transactions_products 13

Output

Action Output

#	Time	Action	Message
1	12:53:01	insert into transactions_products (transaction_id, product_id) select transaction.id, cast(trim(SUBSTRING_INDEX(SUBSTRING_INDEX(product...	1457 row(s) affected Records: 1457 Duplicates: 0 Warnings: 0

```sql
-- Nivell 3
-- Exercici 1

select product_id, count(*)
from transactions_products
group by product_id;
```

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

product_id	count(*)
1	61
2	65
3	51
5	49
7	54
11	48

Result 9

Output

Action Output

#	Time	Action	Message
1	13:16:20	select product_id, count(*) from transactions_products group by product_id	26 row(s) returned