

Tasca S4.01. Creació de Base de Dades

★ NIVELL 1:

Descarrega 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:

- **Creació i disseny de la base de dades**

Primer de tot, he començant fent una nova base de dades amb el comand 'CREATE DATABASE', li he posat el nom de 'bd':

```
CREATE DATABASE IF NOT EXISTS bd;  
USE bd;
```

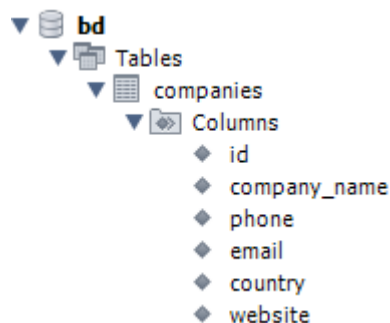
Per crear les taules he fet el següent:

➤ Taula 'companies'

Mirant els valors que contenia la taula 'companies' desde el bloc de notes, he creat la taula segons les columnes que contenia (id, company_name, phone, email, country i website) i la quantitat de números o caràcters que conté (tipus de valor i la longitud d'aquest). He indicat que la PK es 'id', he posat tots els valors en tipus 'varchar', més endavant relacionaré la taula amb la resta:

```
CREATE TABLE IF NOT EXISTS companies (  
    id VARCHAR(15) PRIMARY KEY,  
    company_name VARCHAR(255),  
    phone VARCHAR(15),  
    email VARCHAR(100),  
    country VARCHAR(100),  
    website VARCHAR(255)  
);
```

Com es pot veure a l'esquema de l'esquerra, se'ns ha creat la base de dades i la taula 'companies':










Ara que ja tenim la taula creada, hem d'importar els arxius.
Per carregar els arxius passem els arxius a la carpeta 'Uploads' que es pot trobar en aquesta ruta:

Este equipo > Windows (C:) > ProgramData > MySQL > MySQL Server 8.0 > Uploads

Passem tots els arxius a aquesta carpeta:

Nombre

 companies
 credit_cards
 products
 transactions
 users_ca
 users_uk
 users_usa

Desde MySQL per importar-los fem servir el comand 'LOAD DATA INFILE' i l'indiquem la ruta de on hem posar els arxius de la carpeta 'uploads' i després amb 'INTO TABLE' indiquem a quina taula volem importar-los, després amb 'FIELDS TERMINATED BY' li he indicat les comes, ja que mitjançant les comes és com estan separats els camps. Al final de comand he indicat 'IGNORE 1 LINES' per a que no afegeixi els encapçalats dels camps, i els ignori.

```
LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/companies.csv"
INTO TABLE companies
FIELDS TERMINATED BY ','
IGNORE 1 LINES;
```

Comprovació:

✓ 4 10:29:57 LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/companies.csv" INTO TABLE companies FIELDS TERMINATED BY ','; 100 row(s) affected Records: 100 Deleted: 0 Skipped: 0 Warnings: 0

Si fem un 'SELECT' de tota la taula 'companies', comprovem que els arxius s'han carregat correctament:

```
SELECT *
FROM companies;
```

Resultat:

id	company_name	phone	email	country	website
b-2222	Ac Fermentum Incorporated	06 85 56 52 33	donec.porttitor.tellus@yahoo.net	Germany	https://instagram.com/site
b-2226	Magna A Neque Industries	04 14 44 64 62	risus.donec.nibh@icloud.org	Australia	https://whatsapp.com/group/9
b-2230	Fusce Corp.	08 14 97 58 85	risus@protonmail.edu	United States	https://pinterest.com/sub/cars
b-2234	Convallis In Incorporated	06 66 57 29 50	mauris.ut@aol.couk	Germany	https://cnn.com/user/110
b-2238	Ante Iaculis Nec Foundation	08 23 04 99 53	sed.dictum.proin@outlook.ca	New Zealand	https://netflix.com/settings
b-2242	Donec Ltd	01 25 51 37 37	at.iaculis@hotmail.couk	Norway	https://nytimes.com/user/110
b-2246	Sed Nunc Ltd	02 62 64 73 48	nibh@yahoo.org	United Kingdom	https://cnn.com/one
b-2250	Amet Nulla Donec Corporation	07 15 25 14 74	mattis.integer.eu@protonmail.net	Italy	https://netflix.com/sub/cars
b-2254	Nascetur Ridiculus Mus Inc.	06 26 87 61 84	suspendisse.dui@icloud.net	United States	https://ebay.com/sub
b-2258	Vestibulum Lorem PC	02 02 87 33 40	aenean.massa.integer@aol.net	Belgium	https://pinterest.com/sub/cars
b-2262	Gravida Sagittis LLP	03 81 28 33 97	turpis.vitae@google.ca	Sweden	https://naver.com/site
b-2266	Mus Aenean Eget Foundation	06 25 15 52 43	mi.duis@hotmail.net	Sweden	https://instagram.com/group/9

Comprovació:

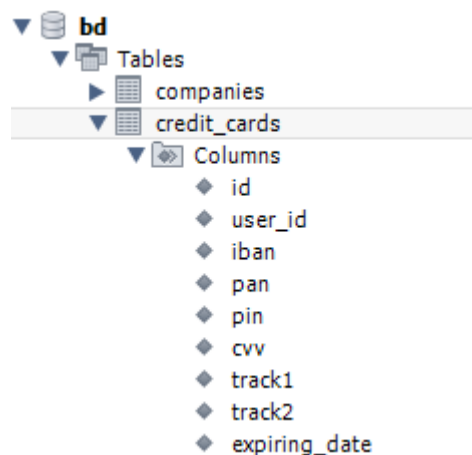
✓ 754 11:54:42 select * from companies LIMIT 0, 1000 100 row(s) returned

➤ Taula 'credit_cards'

He fet el mateix que he fet amb la taula 'companies', he mirat les columnes que tenia l'arxiu (id, user_id, iban, pan, pin, cvv, track1, track2 i expiring_date) el tipus de dada i la seva longitud i he creat la taula segons els valors que conté l'arxiu:

```
CREATE TABLE IF NOT EXISTS credit_cards (  
    id varchar(20) PRIMARY KEY,  
    user_id VARCHAR(5),  
    iban VARCHAR(50),  
    pan VARCHAR(25),  
    pin VARCHAR(4),  
    cvv VARCHAR(3),  
    track1 VARCHAR(200),  
    track2 VARCHAR(200),  
    expiring_date VARCHAR(10)  
);
```

Podem veure que s'ha creat a l'esquema de 'bd':



Per importar l'arxiu fem igual que amb la taula 'companies', amb 'LOAD DATA INFILE' seleccionem la ruta de l'arxiu que volem importar, amb 'INTO TABLE' posem el nom de la taula on volem carregar-lo, com en aquest arxiu els camps també estan separats per comes ' , ' ho he indicat a 'FIELDS TERMINATED BY', i per no afegir els encapçalats he fet 'IGNORE 1 LINES'.

```
LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/credit_cards.csv"  
INTO TABLE credit_cards  
FIELDS TERMINATED BY ' , '  
IGNORE 1 LINES;
```

Comprovació:

6 10:34:20 LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/credit_cards.csv" INTO TABLE credit_cards FIELDS TERMINATED BY ' , ' ... 275 row(s) affected Records: 275 Deleted: 0 Skipped: 0 Warnings: 0

Un cop importat l'arxiu comprovem que sigui correcte fent un 'SELECT' de tota la taula:

```
SELECT *  
FROM credit_cards;
```

Resultat:

id	user_id	iban	pan	pin	cvv	track1	track2	expiring_date
CdU-2938	275	TR301950312213576817638661	5424465566813633	3257	984	%B8383712448554646^WovsxejDpwiev^8604114227	%B7653863056044187=800716333673	10/30/22
CdU-2945	274	DO26854763748537475216568689	5142423821948828	9080	887	%B4621311609958661^UftuyfsSeimxn^061062824177	%B4149568437843501=510714033071	08/24/23
CdU-2952	273	BG451VQL52710525608255	4556 453 55 5287	4598	438	%B2183285104307501^CddytytUxwfdq^590795543079	%B6778580257827162=6906859740077	06/29/21
CdU-2959	272	CR7242477244335841535	372461377349375	3583	667	%B7281111956795320^XocddjBckecd^0901625373	%B4246154489281853=280522391678	02/24/23
CdU-2966	271	BG72LKTQ15627628377363	448566 886747 7265	4900	130	%B4728932322756223^JhlgvsuFbmwgj^7202289494377	%B2318571115599881=890821578475	10/29/24
CdU-2973	270	PT87806228135092429456346	544 58654 54343 384	8760	887	%B4761405253275637^HjnnipoBlejrl^710851563271	%B7816169831446746=1310277729	01/30/25
CdU-2980	269	DE39241881883086277136	402400 7145845969	5075	596	%B7320483593870549^OokqzxrHpsed^4901739473	%B2474313962214151=041221913175	07/24/22
CdU-2987	268	GE89681434837748781813	3763 747687 76666	2298	797	%B4750646345146674^PjmlryfGwwtrf^8305171276	%B5441935173418615=410370453677	10/31/23
CdU-2994	267	BH62714428368066765294	344283273252593	7545	595	%B1583759784015674^GmqoyhtUtoqrn^2507851177	%B4141467473024349=650680955074	02/28/22
CdU-3001	266	CY49087426654774581266832110	511722 924833 2244	9562	867	%B6227288756728648^AwxdifcFmgvdy^280880461178	%B3429355750963453=530526830573	09/16/22
CdU-3008	265	LU507216693616119230	4485744464433884	1856	740	%B7182449430529226^MkoutyHTfdvpo^170817547375	%B6235123731781366=940563816678	04/05/25
CdU-3015	264	PS119398216295715968342456821	3784 662233 17389	3246	822	%B5776250106724742^OvzvkrCwrovm^5305842489774	%B3561372148267521=190547316677	01/31/22

Comprovació:

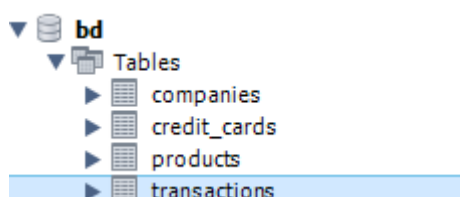
764 12:18:54 SELECT * FROM credit_cards LIMIT 0, 1000 275 row(s) returned

➤ Taula 'transactions'

Tornem a fer el mateix, creem la taula 'transactions' mitjançant les dades de l'arxiu, en aquest cas, conté les columnes id (es la PK), card_id, business_id, timestamp, amount, declined, products_ids, user_id, lat i longitude. La majoria de les dades son en format 'varchar', menys la dada que está en format 'timestamp' per a que s'indiqui la data i l'hora de quan s'han efectuat les transaccions, l'amount está en decimals per indicar el cèntims, declined está en format 'boolean' si es 0 es que no s'ha declinat la transacció, i si es 1 es que sí que s'ha declinat, user_id está en format 'interger' ja que son números, i la latitud i longitud en format 'float':

```
CREATE TABLE IF NOT EXISTS transactions (  
    id VARCHAR(255) PRIMARY KEY,  
    card_id VARCHAR(10),  
    business_id VARCHAR(10),  
    timestamp TIMESTAMP,  
    amount DECIMAL(10, 2),  
    declined BOOLEAN,  
    products_ids VARCHAR(20),  
    user_id INT,  
    lat FLOAT,  
    longitude FLOAT  
);
```

Veiem que a l'esquema s'ha creat la taula:



Per importar els arxius, fem 'LOAD DATA INFILE' amb la ruta de l'arxiu 'transactions' de la carpeta 'Uploads', amb 'INTO TABLE' seleccionem la taula on ho volem importar, en aquest cas a la taula 'transactions', en aquest arxiu els camps estan separats per punt i coma ';', ho indiquem a 'FIELDS TERMINATED BY', seguit de 'IGNORE 1 LINES' per ignorar els encapçalats.

Aquí es pot veure que estan separats per ';'

```
id;card_id;business_id;timestamp;amount;declined;product_ids;user_id;lat;longitude
108B1D1D-5B23-A76C-55EF-C568E49A05DD;CcU-2938;b-2222;2021-07-07 17:43:16;293.57;0;59;275;83.7839152128;-178.860353536
```

Codi:

```
LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/transactions.csv"
INTO TABLE transactions
FIELDS TERMINATED BY ';'
IGNORE 1 LINES;
```

Comprovació:

```
13 10:44:44 LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/transactions.csv" INTO TABLE transactions FIELDS TERMINATED BY ';' IGNORE 1 LINES ... 587 row(s) affected Records: 587 Deleted: 0 Skipped: 0 Warnings: 0
```

Un cop importat, fem la comprovació fent un select de tota la taula 'transactions':

```
SELECT *
FROM transactions;
```

Resultat:

id	card_id	business_id	timestamp	amount	declined	products_ids	user_id	lat	longitude
02C6201E-D90A-1859-B4EE-88D2986D3B02	CcU-2938	b-2362	2021-08-28 23:42:24	466.92	0	71, 1, 19	92	81.9185	-12.5276
0466A42E-47CF-8D24-FD01-C0B689713128	CcU-4219	b-2302	2021-07-26 07:29:18	49.53	0	47, 97, 43	170	-43.9695	-117.525
063FBA79-99EC-66FB-29F7-25726D1764A5	CcU-2987	b-2250	2022-01-06 21:25:27	92.61	0	47, 67, 31, 5	275	-81.2227	-129.05
0668296C-CDB9-A883-76BC-2E4C44F8C8AE	CcU-3743	b-2618	2022-01-26 02:07:14	394.18	0	89, 83, 79	265	-34.3593	-100.556
06CD9AA5-9B42-D684-DDDD-A5E394FEB9A9	CcU-2959	b-2346	2021-10-26 23:00:01	279.93	0	43, 31	92	33.7381	158.298
07A46D48-31A3-7E87-65B9-0DA902AD109F	CcU-3225	b-2386	2021-06-28 21:11:42	340.87	1	47, 23	272	38.8342	92.1905
09DE92CE-6F27-2BB7-13B5-9385B2B388E2	CcU-3071	b-2298	2021-05-11 20:40:06	303.05	1	67, 7	275	71.1706	10.5757
0A476ED9-0C13-1962-F87B-D3563924B539	CcU-4359	b-2302	2022-02-26 20:33:54	430.49	0	29, 41, 11	221	-56.4901	114.801
08EB80B7-9D66-1707-CE4B-9DC7E71914B5	CcU-3141	b-2338	2022-03-04 14:54:35	288.81	1	19, 41, 29, 3	272	23.3264	-13.6037
0C7C3A33-9947-3BC1-846D-7BE3D0D17598	CcU-3309	b-2434	2021-04-10 20:58:41	103.44	1	89, 31	272	63.3615	-68.6667
0CE957A6-CCAA-2B7A-6839-8A4B1B324853	CcU-3435	b-2506	2022-02-02 07:29:36	428.69	1	83, 43, 73, 61	269	-69.3537	-10.26
0DD2E608-5C9E-D1B3-4999-B99F43AD735A	CcU-2959	b-2234	2021-04-17 05:30:17	252.47	1	7, 47, 17	275	9.68811	130.282

Comprovació:

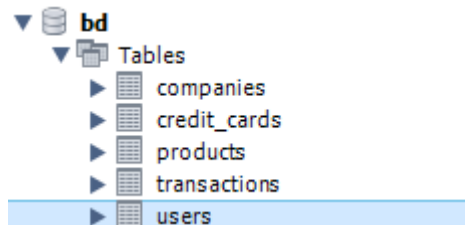
```
821 12:50:57 SELECT * FROM transactions LIMIT 0, 1000 587 row(s) returned
```

➤ Taula 'users'

He creat una taula anomenada 'users' per ajuntar les 3 taules dels usuaris de Canada, United Kingdom i de USA. Aquesta taula conté les columnes 'id' (que es la PK), 'name', 'surname', 'phone', 'email', 'birth_date', 'country', 'city', 'postal_code' i 'address'. En aquesta taula la data de naixement està indicada d'una altre forma, posant primer el mes de manera escrita abreujat, el dia i després l'any de naixement, com el format 'date' no és així l'he posat el format 'varchar' per a que no doni problemes quan s'importi, he posat la longitud de les dades segons he vist a l'arxiu en el bloc de notes:

```
CREATE TABLE IF NOT EXISTS users (  
    id INT PRIMARY KEY,  
    name VARCHAR(20),  
    surname VARCHAR(20),  
    phone VARCHAR(15),  
    email VARCHAR(100),  
    birth_date VARCHAR(15),  
    country VARCHAR(100),  
    city VARCHAR(100),  
    postal_code VARCHAR(10),  
    address VARCHAR(200)  
);
```

Podem veure que s'ha creat la taula a la base de dades:



Per importar els tres arxius, he anat arxiu per arxiu, he començat amb la taula de 'users_usa', indicant la ruta d'aquest arxiu de la carpeta 'Uploads' amb 'LOAD DATA INFILE'.

Amb 'INTO TABLE' he posat 'users' que es la taula que just he creat. Veien l'arxiu en el bloc de notes, es pot veure que tots els camps estan separats per comes ' , ' i a més el camp de la data de naixement està indicada entre cometes dobles ' " ', això ho indico a 'FIELDS TERMINATED BY' (l'indico les comes), i després amb 'ENCLOSED BY' l'indico les cometes dobles, per indicar-li que el valor està comprès entre aquestes comilles.

Indicant 'LINES TERMINATED BY '\r\n' li estem dient que les línies acaben quan dona el salt de línia. Tornem a posar 'IGNORE 1 LINES' per ignorar el encapçalat. Fem el mateix per amb els altres 2 arxius més de 'users_uk' i de 'users_ca'.

Aquí es pot veure com en el bloc de notes podem veure que totes les dades estan separades mitjançant les comes, i a més, quan s'indica la data de naixement, aquesta està entre cometes dobles:

```
203,Jarrold,Fields,010-741-8105,sit.amet@google.couk,"Jan 6, 1982",Canada,Baddeck,K3X 6Z5,441-8969 Rhoncus Road
```

Codi:

```
-- Taula 'USA'
LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/users_usa.csv"
INTO TABLE users
FIELDS TERMINATED BY ',' ENCLOSED BY '"'
LINES TERMINATED BY '\r\n'
IGNORE 1 LINES;
-- Taula 'UK'
LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/users_uk.csv"
INTO TABLE users
FIELDS TERMINATED BY ',' ENCLOSED BY '"'
LINES TERMINATED BY '\r\n'
IGNORE 1 LINES;
-- Taula 'CA'
LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/users_ca.csv"
INTO TABLE users
FIELDS TERMINATED BY ',' ENCLOSED BY '"'
LINES TERMINATED BY '\r\n'
IGNORE 1 LINES;
```

Comprovació:

```
15 11:20:23 LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/users_usa.csv" INTO TABLE users FIELDS TERMINATED BY ',' EN... 150 row(s) affected Records: 150 Deleted: 0 Skipped: 0 Warnings: 0
16 11:20:25 LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/users_uk.csv" INTO TABLE users FIELDS TERMINATED BY ',' ENC... 50 row(s) affected Records: 50 Deleted: 0 Skipped: 0 Warnings: 0
17 11:20:27 LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/users_ca.csv" INTO TABLE users FIELDS TERMINATED BY ',' ENC... 75 row(s) affected Records: 75 Deleted: 0 Skipped: 0 Warnings: 0
```

Fem un 'SELECT' de tota la taula de 'users' un cop hem importat el 3 arxius, per comprovar que s'han carregat correctament:

```
SELECT *
FROM users;
```

Resultat:

id	name	surname	phone	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.
5	Hayfa	Pierce	1-554-541-2077	et.malesuada.fames@hotmail.org	Sep 26, 1998	United States	Sandy	31564	341-2821 Ultrices Av.
6	Joel	Tyson	(718) 288-8020	gravida.nunc.sed@yahoo.ca	Oct 15, 1989	United States	Nashville	96838	888-2799 Amet Street
7	Rafael	Jimenez	(817) 689-0478	eget@outlook.ca	Dec 4, 1981	United States	Hillsboro	29874	8627 Malesuada Rd.
8	Nissim	Franks	(692) 157-3469	egestas.aliquam.fringilla@google.ca	Aug 1, 1993	United States	Jackson	61750	Ap #251-7144 Integer St.
9	Mannix	Mcclain	(590) 883-2184	aliquam.nisl@outlook.com	Jan 24, 1987	United States	Richmond	35987	647-3080 Lacus. St.
10	Robert	Mccarthy	(324) 746-6771	fermentum@protonmail.com	Apr 30, 1984	United States	Eugene	85526	P.O. Box 773, 3594 Ornare St.
11	Joan	Baird	(981) 429-8106	et@outlook.net	Feb 25, 1990	United States	Lincoln	35211	P.O. Box 687, 8917 Ligula St.
12	Benedict	Wheeler	1-515-824-2855	tincidunt.donec.vitae@hotmail.couk	Aug 6, 1999	United States	Lewiston	92393	748-8694 Porttitor Avenue
13	Allegra	Stanton	1-927-753-6488	proin.eget@protonmail.ca	May 19, 1990	United States	Kearney	14947	4457 Ante. Av.
14	Sara	Flynn	1-311-646-9333	integer@outlook.net	Dec 27, 1988	United States	Warren	20288	P.O. Box 865, 4397 Ante St.
15	Noelani	Patrick	1-723-488-5894	sem.magna@google.com	Sep 17, 1993	United States	Orlando	47987	596-5044 Sapien. Street
16	Eric	Roth	1-218-549-8253	lorem.sit@yahoo.net	Sep 7, 1988	United States	Reading	96697	P.O. Box 541, 5137 Non Road

Comprovació:

```
92 10:19:54 SELECT * FROM users LIMIT 0, 1000 275 row(s) returned
```


Un cop tinc les 4 taules creades i ja he importat les dades, veiem que el diagrama ha quedat així:

users	companies	transactions	credit_cards
id INT	id VARCHAR(15)	id VARCHAR(255)	id VARCHAR(20)
name VARCHAR(20)	company_name VARCHAR(255)	card_id VARCHAR(10)	user_id VARCHAR(5)
surname VARCHAR(20)	phone VARCHAR(15)	business_id VARCHAR(10)	iban VARCHAR(50)
phone VARCHAR(15)	email VARCHAR(100)	timestamp TIMESTAMP	pan VARCHAR(25)
email VARCHAR(100)	country VARCHAR(100)	amount DECIMAL(10,2)	pin VARCHAR(4)
birth_date VARCHAR(15)	website VARCHAR(255)	declined TINYINT(1)	cvv VARCHAR(3)
country VARCHAR(100)		products_ids VARCHAR(20)	track1 VARCHAR(200)
city VARCHAR(100)		user_id INT	track2 VARCHAR(200)
postal_code VARCHAR(10)		lat FLOAT	expiring_date VARCHAR(10)
address VARCHAR(200)		longitude FLOAT	
Indexes	Indexes	Indexes	Indexes

Identifico que la 'tabla de hechos' (Fact Table) es 'transactions' i les taules 'companies', 'credit_cards' i 'users' son les taules de dimensions. A la taula 'transactions' es a on es relacionen les altres taules de mètriques, és a dir, és com el punt d'unió. Ara identificaré les FK i relacionaré les taules. Per crear les relacions primer he creat l'índex a la taula 'transactions' per poder establir després la foreign key.

Per relacionar la taula 'companies' amb 'transactions', he creat l'índex a 'transactions' amb l'id de business, per després afegir la FK 'business_id' relacionant-la amb 'id' de la taula 'companies':

```
ALTER TABLE transactions
ADD INDEX idx_business_id (business_id ASC);

ALTER TABLE transactions
ADD FOREIGN KEY (business_id) REFERENCES companies(id);
```


Per relacionar la taula 'credit_cards' amb 'transactions', he creat l'índex a 'transactions' amb 'card_id', per després afegir la FK 'card_id' relacionant-la amb 'id' de la taula 'credit_cards':

```
ALTER TABLE transactions
```

```
ADD INDEX idx_card_id (card_id ASC);
```

```
ALTER TABLE transactions
```

```
ADD FOREIGN KEY (card_id) REFERENCES credit_cards(id);
```

92 13:42:22 ALTER TABLE transactions ADD FOREIGN KEY (card_id) REFERENCES credit_cards(id) 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

Per relacionar la taula 'users' amb 'transactions', he creat l'índex a 'transactions' amb l'id de users, per després afegir la FK 'user_id' relacionant-la amb 'id' de la taula 'users':

```
ALTER TABLE transactions
```

```
ADD INDEX idx_user_id (user_id ASC);
```

```
ALTER TABLE transactions
```

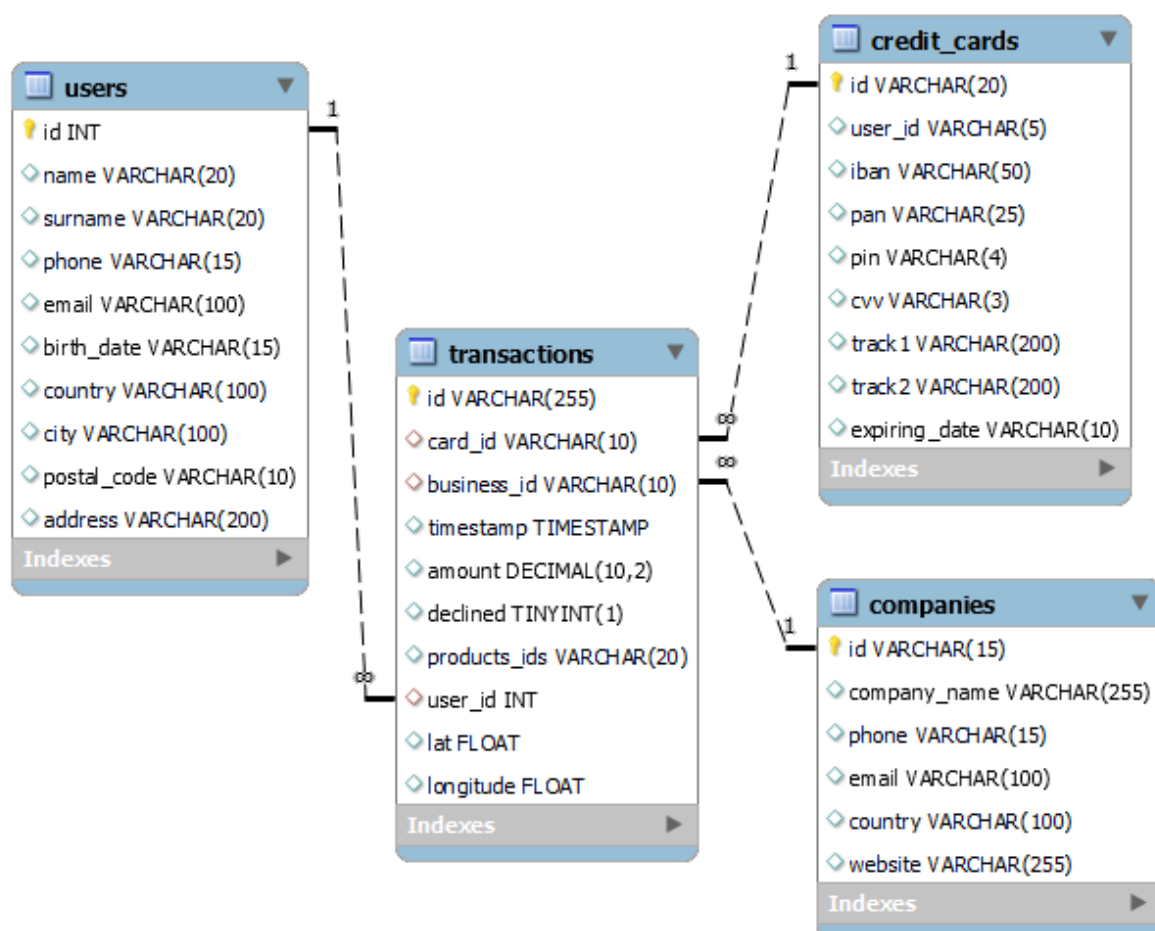
```
ADD FOREIGN KEY (user_id) REFERENCES users(id);
```

94 13:43:27 ALTER TABLE transactions ADD FOREIGN KEY (user_id) REFERENCES users(id) 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

Si ara visualitzem el diagrama veiem que s'han establert les relacions, de manera que es un diagrama en model estrella, no és relacional.

Les taules de dimensions tenen una relació 1:n amb la taula 'transactions'. La taula 'transactions' és la taula principal d'aquest model relacional, compté mètriques (dades quantitatives) i també camps clau que són els que es relacionen amb les taules de dimensions.

Les taules de dimensions ('companies', 'credit_cards' i 'users'), son les que s'uneixen a la Fact Table mitjançant els camps clau, tenen atributs (descripcions que donen més informació sobre les dades i permeten analitzar amb més detall) i claus primàries (PK) que son identificadors únics.



Ara sí que es pot realitzar la consulta.

❖ EXERCICI 1:

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

He fet la subquery desde el WHERE, al SELECT he indicat totes les dades que em semblen que son rellevants sobre els usuaris (id, nom, cognom, pais, ciutat, codi postal i telèfon), a la clàusula WHERE he indicat que busqui els id dels usuaris que coincideixen que a la taula transactions tenen més de 30 transaccions (filtrant amb HAVING, i posant la funció d'agregació COUNT per a que compti el total de transaccions, per després poder indicar que solament vull filtrar aquelles que son superiors a 30):

```
SELECT
    id AS 'ID',
    name AS 'Nom',
    surname AS 'Cognom',
    country AS 'Pais',
    city AS 'Ciutat',
    postal_code AS 'Codi postal',
    phone AS 'Telèfon'
FROM users
WHERE users.id IN (SELECT user_id
                   FROM transactions
                   GROUP BY user_id
                   HAVING COUNT(transactions.id) >30);
```

Resultat:

ID	Nom	Cognom	Pais	Ciutat	Codi postal	Telèfon
92	Lynn	Riddle	United States	Bozeman	61871	1-387-885-4057
267	Ocean	Nelson	Canada	Charlottetown	85X 3P4	079-481-2745
272	Hedwig	Gilbert	Canada	Tuktoyaktuk	Q4C 3G7	064-204-8788
275	Kenyon	Hartman	Canada	Richmond	R8H 2K2	082-871-7248

Comprovació:

187 12:23:56 SELECT id AS 'ID', name AS 'Nom', surname AS 'Cognom', country AS 'Pais', city AS 'Ciutat', postal_code AS 'Codi postal', phone AS 'Telèfon' FROM users WHERE users.id IN (SELECT user_id FROM transactions GROUP BY user_id HAVING COUNT(transactions.id) >30); 4 row(s) returned

❖ EXERCICI 2:

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

Al 'SELECT' he indicat el iban de les targetes de crèdit de la taula 'credit_cards' i la mitja d'amount de la taula 'transactions', amb la funció AVG per calcular la mitjana, a aquesta l'he aplicat el ROUND, per a que surtin solament 2 decimals.

Després he fet un JOIN entre les taules 'transactions' i 'credit_cards' i després entre 'transactions' i 'companies'.

Al filtre WHERE he filtrat directament per el nom de la companyia, en aquest cas 'Donec Ltd', mitjançant el 'company_name' de la taula 'companies'. Ho he agrupat per l'iban.

```
SELECT
    credit_cards.iban AS 'IBAN',
    ROUND(AVG(transactions.amount),2) AS 'Mitja (€)'
FROM transactions
JOIN credit_cards
ON transactions.card_id = credit_cards.id
JOIN companies
ON transactions.business_id = companies.id
WHERE companies.company_name = 'Donec Ltd'
GROUP BY credit_cards.iban;
```

Resultat:

IBAN	Mitja (€)
PT87806228135092429456346	203.72

Comprovació:

```
30 17:10:31 SELECT credit_cards.iban AS 'IBAN', ROUND(AVG(transactions.amount),2) AS 'Mitja (€)' FROM transactions JOIN credit_cards ON transac... 1 row(s) returned
```

★★ 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.

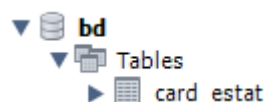
He creat una taula a la que li he anomenat 'card_estat', li he afegit les columnes 'id' i 'IBAN' de la taula de 'credit_cards' i també he creat una nova columna a la que li he donat el nom de 'estat', utilitzant un 'CASE', indicant amb un 'COUNT' que si el recompte de transaccions declinades (posant que declined ha de ser 1, ja que és un boolean i vol dir que li estic indicant les declinades) és igual o major que 3, que indiqui que en aquesta situació el seu estat es 'bloquejada', i amb el 'ELSE' que indiqui que segueix activa (en aquest cas, amb el 'ELSE' indico que en tot cas que no compleixi la funció que li he indicat abans de que sigui major o igual a 3, doncs si no compleix aquesta situació que indiqui que és activa, ja que solament apareixerà que és activa en els casos on sigui menor que 3).

En el 'FROM' he indicat que es de la taula 'transactions' i he fet un 'JOIN' amb la taula 'credit_cards' a través del id de les targetes de crèdit. Ho he agrupat per l'id i l'IBAN. No li he posar un 'ORDER BY', ja que automàticament ho ordena per l'id de menor a major.

```
CREATE TABLE card_estat AS
SELECT
    credit_cards.id,
    credit_cards.iban,
    CASE
        WHEN (COUNT(transactions.declined) = 1) >= 3 THEN 'bloquejada'
        ELSE 'activa'
    END AS 'Estat'
FROM transactions
JOIN credit_cards
ON transactions.card_id = credit_cards.id
GROUP BY credit_cards.id, credit_cards.iban;
```

213 11:50:21 CREATE TABLE card_estat AS SELECT credit_cards.id, credit_cards.iban, CASE WHEN COUNT(transactions.id) >= 3 THEN bloqu... 275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0

Veiem que s'ha creat la taula a la base de dades 'bd':



Si fem un SELECT de tota la taula

```
SELECT *  
FROM card_estat;
```

Resultat (totes les targetes estan actives, no hi ha ninguna bloquejada):

id	iban	Estat
CcU-2938	TR301950312213576817638661	activa
CcU-2945	DO26854763748537475216568689	activa
CcU-2952	BG45IVQL52710525608255	activa
CcU-2959	CR7242477244335841535	activa
CcU-2966	BG72LKTQ15627628377363	activa
CcU-2973	PT87806228135092429456346	activa
CcU-2980	DE39241881883086277136	activa
CcU-2987	GE89681434837748781813	activa
CcU-2994	BH62714428368066765294	activa
CcU-3001	CY49087426654774581266832110	activa
CcU-3008	LU507216693616119230	activa
CcU-3015	PS119398216295715968342456821	activa
CcU-3022	GT91695162850556977423121857	activa
CcU-3029	AZ62317413982441418123739746	activa
CcU-3036	AZ39336002925842865843941994	activa
CcU-3043	TN6488143310514852179535	activa
CcU-3050	FR5167744369175836831854477	activa
CcU-3057	LU931822574697545215	activa
CcU-3064	PS146965545449253377627273133	activa
CcU-3071	NO8923814763512	activa
CcU-3078	IS025127145884623279548733	activa
CcU-3085	BE63114723972437	activa
CcU-3092	RO65LSOD1166122125447487	activa
CcU-3099	PT26105275356823705537218	activa
CcU-3106	AT684251637751136592	activa
CcU-3113	IE26LCGT47732173572752	activa
CcU-3120	RS72655766666166237144	activa
CcU-3127	PT83533461438644342816864	activa
CcU-3134	BG23MYJQ52668951824779	activa
CcU-3141	CH4437804777669672438	activa
CcU-3148	FI6261403224677114	activa
CcU-3155	AD2777204763277722050982	activa
CcU-3162	HU56074054826233730628233311	activa
CcU-3169	AT658218806585843788	activa
CcU-3176	LV84DASY1039958872222	activa
CcU-3183	GE00157078842228124462	activa

Comprovació:

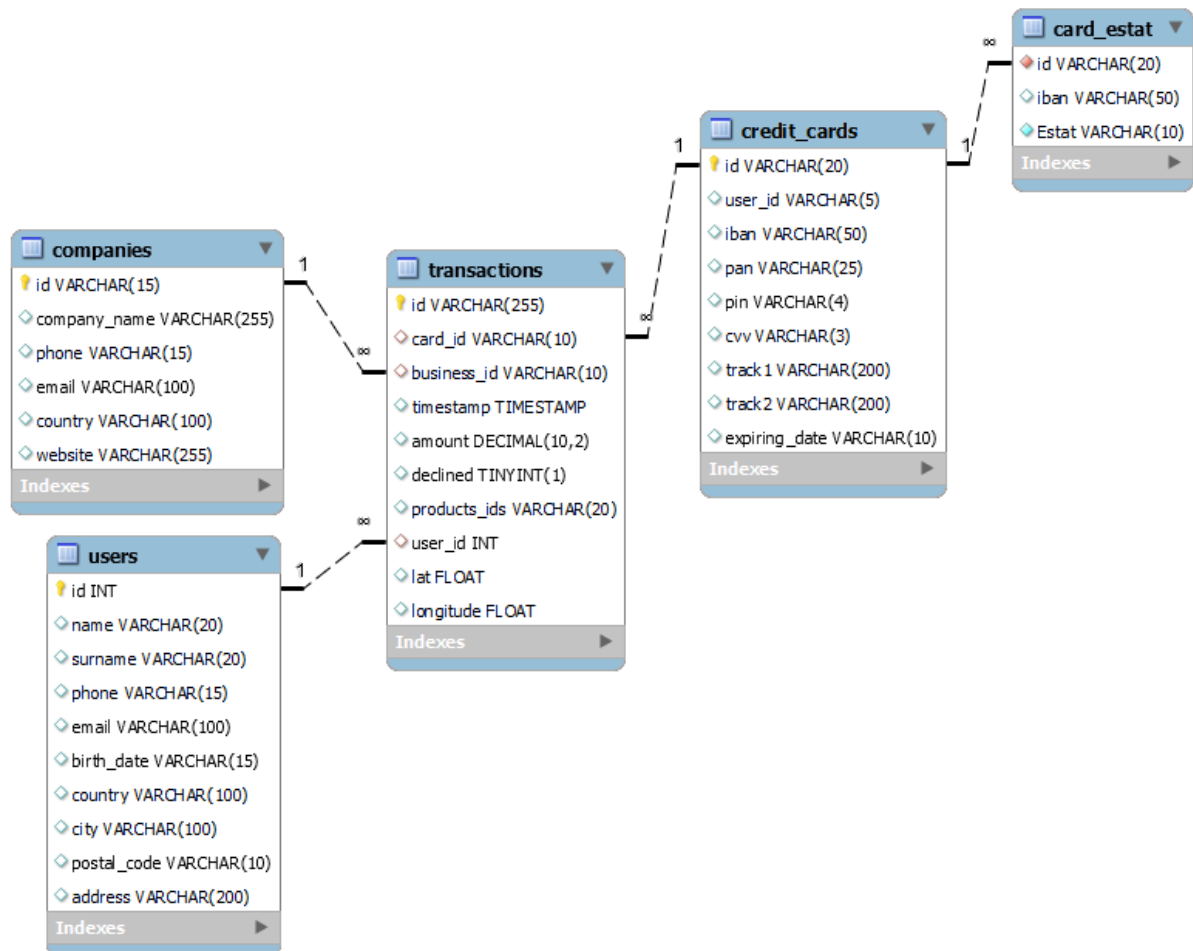
215 11:55:50 SELECT * FROM card_estat LIMIT 0, 1000

275 row(s) returned

Ara que ja tenim la taula de 'card_estat' creada l'he de relacionar amb la taula 'credit_cards', per fer-ho he modificat canvis a la taula 'card_estat', indicant que afegeixi la FK id i la relacioni amb 'id' de la taula 'credit_cards'

```
ALTER TABLE card_estat  
ADD FOREIGN KEY (id) REFERENCES credit_cards(id);
```

Veient el diagrama ens ha quedat així, ara ja no es un model en estrella, ara es de 'copo de nieve', ja que ara en el diagrama es veu que té la 'tabla de hechos' (Fact table), les de dimensions ('companies', 'users' i 'credit_cards') i que de una taula de dimensions (credit_cards) surt una altre taula de dimensions (card_estat). És a dir, a més de les taules de dimensions, aquestes contenen altres taules de dimensions associades a aquestes, adoptant el model en 'copo de nieve', fent que el model sigui més dimensional.



❖ EXERCICI 1:

Quantes targetes estan actives?

Per sapiguer la quantitat de targetes actives en el 'SELECT' he fet un 'COUNT' dels id de les targetes, per comptar-les, agafant les dades de la taula 'card_estat', i amb el filtre 'WHERE' indico que solament vull sapiguer aquelles que a l'estat posa que esta 'activa'.

He comprovat també que si en el 'SELECT' faig un 'COUNT(DISTINCT id)' segueix sortint el mateix resultat, ho havia fet per comprovar que no hi hagués ningun valor duplicat, pero en aquest cas no hi ha.

El resultat és que estan totes les targetes actives, no hi ha ninguna bloquejada.

```
SELECT
  COUNT(id) AS 'Targetes actives'
FROM
  card_estat
WHERE
  Estat = 'activa';
```

Resultat:

Targetes actives
275

Comprovació:

222 12:17:44 SELECT COUNT(id) AS 'Targetes actives' FROM card_estat WHERE Estat = 'activa' LIMIT 0, 1000 1 row(s) returned

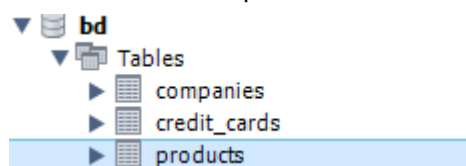
★★★ NIVELL 3:

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:

Per crear la taula 'products' he mirat en el bloc de notes com era l'arxiu, les columnes que contenia (id, product_name, price, colour, weight i warehouse_id) i segons les dades que conté he posat la longitud convenient i he posat tots els valors en format 'varchar':

```
CREATE TABLE IF NOT EXISTS products (  
    id VARCHAR(20) PRIMARY KEY,  
    product_name VARCHAR(50),  
    price VARCHAR(10),  
    colour VARCHAR(7),  
    weight VARCHAR(4),  
    warehouse_id VARCHAR(10)  
);
```

Podem veure que s'ha creat:



Per importar les dades, amb 'LOAD DATA INFILE' indiquem la ruta de l'arxiu 'products' de la carpeta 'Uploads', amb 'INTO TABLE' seleccionem la taula 'products' que acabem de crear, i com aquestes dades estan separades per comes, ho indiquem a 'FIELDS TERMINATED BY', i per no incloure els encapçalats fem 'IGNORE 1 LINES'.

```
LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/products.csv"  
INTO TABLE products  
FIELDS TERMINATED BY ','  
IGNORE 1 LINES;
```

Comprovació:

9 10:38:37 LOAD DATA INFILE "C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/products.csv" INTO TABLE products FIELDS TERMINATED BY ',' I... 100 row(s) affected Records: 100 Deleted: 0 Skipped: 0 Warnings: 0

Un cop importat, comprovem amb un 'SELECT' de tota la taula 'products':

```
SELECT *  
FROM products;
```

Resultat:

id	product_name	price	colour	weight	warehouse_id
1	Direwolf Stannis	\$161.11	#7c7c7c	1	WH-4
2	Tarly Stark	\$9.24	#919191	2	WH-3
3	duel tourney Lannister	\$171.13	#d8d8d8	1.5	WH-2
4	warden south duel	\$71.89	#111111	3	WH-1
5	skywalker ewok	\$171.22	#bdbdbd	3.2	WH-0
6	dooku solo	\$136.60	#c4c4c4	0.8	WH--1
7	north of Casterly	\$63.33	#b7b7b7	0.6	WH--2
8	Winterfell	\$32.37	#383838	1.4	WH--3
9	Winterfell	\$76.40	#b5b5b5	1.2	WH--4
10	Karstark Dorne	\$119.52	#f4f4f4	2.4	WH--5
11	Karstark Dorne	\$49.70	#141414	2.7	WH--6
12	duel Direwolf	\$181.60	#a8a8a8	2.1	WH--7

Comprovació:

```
802 12:39:40 SELECT * FROM products LIMIT 0, 1000 100 row(s) returned
```

He intentat crear una taula intermitja de 'transaction_product', per fer l'unió entre 'products' i 'transactions' per no em sortia com separar els 'id', perquè a la taula 'transactions' els 'products_ids' estan separats per comes.

Per relacionar la taula 'products' amb 'transactions', he creat l'índex a 'transactions' amb l'id de 'products'.

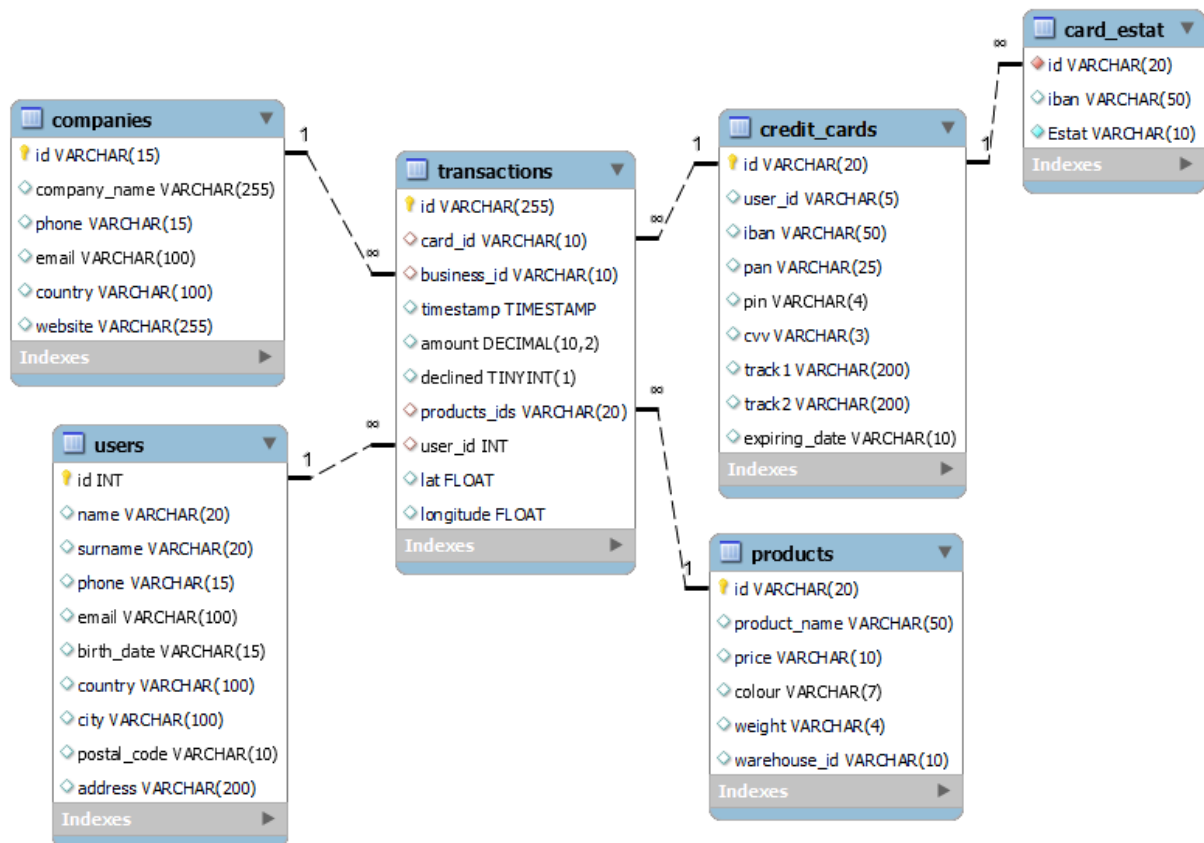
Al establir la relació entre 'products' i 'transactions' em sortia l'error 1452, que ho he solucionat desactivant les FK, he creat la connexió ('products_ids' relacionant-la amb 'id' de la taula 'products'), i després he tornat a activar-les:

```
ALTER TABLE transactions  
ADD INDEX idx_products_ids (products_ids ASC);  
-- solucionar error:  
SET FOREIGN_KEY_CHECKS=0;  
-- establir connexió:  
ALTER TABLE transactions  
ADD FOREIGN KEY (products_ids) REFERENCES products(id);  
-- les activem un altre cop:  
SET FOREIGN_KEY_CHECKS=1;
```

```
93 13:42:39 ALTER TABLE transactions ADD FOREIGN KEY (products_ids) REFERENCES products(id)
```

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

Amb la taula 'products' creada el diagrama ha quedat així:



❖ EXERCICI 1:

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

En aquesta consulta, en el 'SELECT' he fet un 'COUNT' dels id de les transaccions de la taula 'transactions' per sapiguer la quantitat de vegades que s'han venut els productes, seguit del nom del producte, del seu 'ID' i del preu. He fet un 'JOIN' entre la taula 'transactions' i 'products' mitjançant els id dels productes.

He aplicat el filtre 'WHERE' que la columna 'declined' ha de ser 0, per filtrar solament aquells productes que s'han venut i la seva transacció no ha sigut denegada. Ho he agrupat per 'ID', nom del producte i del seu preu unitari i ho he ordenat de majors vendes a menys.

Fent l'exercici vaig veure que hi han productes amb el mateix nom, però el seu 'ID' és diferent, entenent que son productes diferents, ja que també tenen preus diferents, és per aquesta raó que a l'exercici he indicat el seu 'ID' i el preu, per entendre que encara que el nom d'alguns productes coincideix, no son el mateix.

```
SELECT
  COUNT(transactions.id) AS 'Quantitat de vegades venut',
  products.product_name AS 'Nom del producte',
  products.id AS 'ID del producte',
  products.price AS 'Preu per unitat'
FROM transactions
JOIN products
ON transactions.products_ids = products.id
WHERE transactions.declined = 0
GROUP BY products.id, products.product_name, products.price
ORDER BY COUNT(transactions.id) DESC;
```

Resultat:

Quantitat de vegades venut	Nom del producte	ID del producte	Preu per unitat
7	kingsblood Littlefinger the	53	\$137.81
6	duel	43	\$59.80
6	skywalker ewok	5	\$171.22
5	riverlands north	23	\$169.96
5	Tully maester Tarly	29	\$167.20
5	Direwolf Littlefinger	37	\$26.66
5	Direwolf Stannis	59	\$114.77
5	duel tourney	83	\$26.51
5	skywalker ewok	89	\$172.78
4	Direwolf Stannis	1	\$161.11
4	dooku solo	19	\$60.33
4	Winterfell Lannister	61	\$28.01
4	Winterfell	67	\$195.94
4	north of Casterly	7	\$63.33
4	Direwolf riverlands the	79	\$132.86
3	Karstark Dorne	11	\$49.70
3	palpatine chewbacca	13	\$139.59
3	Tarly Stark	2	\$9.24
3	duel tourney Lannister	3	\$171.13
3	Lannister	31	\$85.02
3	Lannister Barratheon Dir...	41	\$141.01
3	Dorne bastard	73	\$114.09
3	jinn Winterfell	97	\$65.25
2	skywalker ewok sith	17	\$91.89
2	Tully	47	\$82.15

Comprovació:

211 13:48:44 SELECT products.product_name AS 'Nom del producte', COUNT(transactions.id) AS 'Quantitat de vegades' FROM transactions JOIN products ... 23 row(s) returned