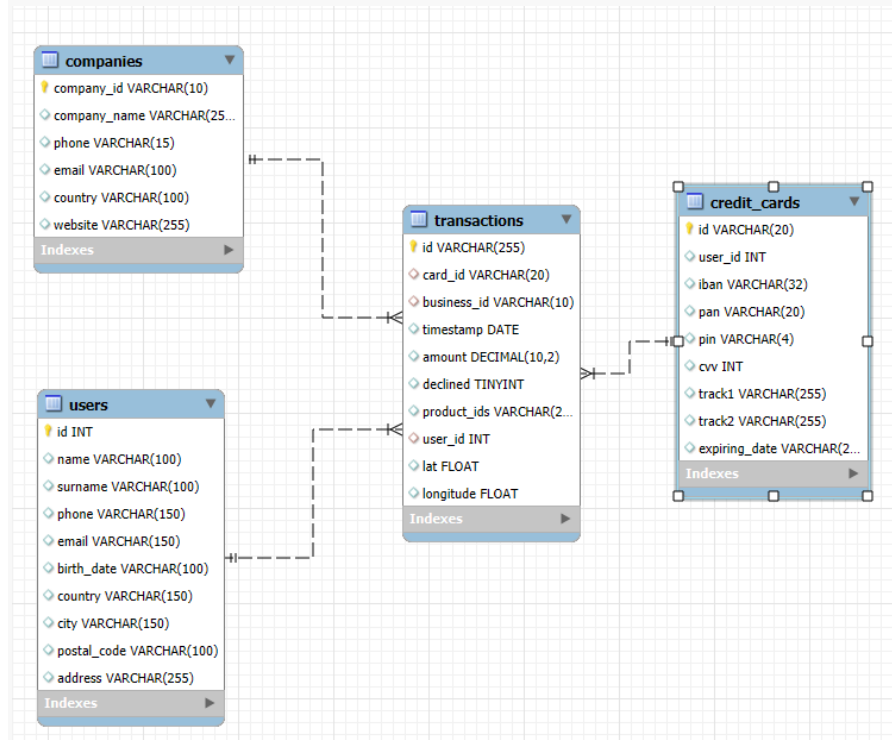# TASCA S4.01. Creació de Base de Dades

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:

Para llegar a tener la base de datos con modelo estrela he usado las siguientes consultas:

He tabla y luego he utilizado la funcion Load, que permite cargar archivos csv

Companies

```
1  •  CREATE TABLE companies(
2        company_id VARCHAR(10) PRIMARY KEY,
3        company_name VARCHAR(255),
4        phone VARCHAR(15),
5        email VARCHAR(100),
6        country VARCHAR(100),
7        website VARCHAR(255)
8     );
9
10 •  SELECT @@secure_file_priv;
11
12 •  LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/companies.csv'
13     INTO TABLE companies
14     FIELDS TERMINATED BY ','
15     ENCLOSED BY ''
16     LINES TERMINATED BY '\n'
17     IGNORE 1 ROWS;
18
19 •  SELECT * FROM companies;
20
```

company_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

companies 2 ×

Output

Action Output

#	Time	Action	Message
1	12:05:57	SELECT * FROM companies	100 row(s) returned

```sql
22 •  CREATE TABLE users(
23      id int PRIMARY KEY ,
24      name VARCHAR(100) ,
25      surname VARCHAR(100) ,
26      phone VARCHAR(150) ,
27      email VARCHAR(150) ,
28      birth_date VARCHAR(100) ,
29      country VARCHAR(150) ,
30      city VARCHAR(150) ,
31      postal_code VARCHAR(100),
32      address VARCHAR(255)
33    );
34
35 •  LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/users_usa.csv'
36      INTO TABLE users
37      FIELDS TERMINATED BY ','
38      OPTIONALLY ENCLOSED BY '"'
39      LINES TERMINATED BY '\r\n'
40      IGNORE 1 ROWS;
41
42 •  LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/users_uk.csv'
43      INTO TABLE users
44      FIELDS TERMINATED BY ','
45      OPTIONALLY ENCLOSED BY '"'
46      LINES TERMINATED BY '\r\n'
47      IGNORE 1 ROWS;
48
49 •  LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/users_ca.csv'
50      INTO TABLE users
51      FIELDS TERMINATED BY ','
52      OPTIONALLY ENCLOSED BY '"'
53      LINES TERMINATED BY '\r\n'
54      IGNORE 1 ROWS;
55
56 •  SELECT * FROM users;
57
58
59
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

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.

users 4 ✕

Output

Action Output

#	Time	Action	Message
✓	1 12:10:25	SELECT * FROM users	275 row(s) returned

credit_cards

```
62 •  CREATE TABLE credit_cards(
63      id VARCHAR(20) PRIMARY KEY,
64      user_id int,
65      iban VARCHAR(32),
66      pan VARCHAR(20),
67      pin VARCHAR(4),
68      cvv INT,
69      track1 VARCHAR(255),
70      track2 VARCHAR(255),
71      expiring_date VARCHAR(20)
72      );
73
74 •  LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/credit_cards.csv'
75      INTO TABLE credit_cards
76      FIELDS TERMINATED BY ','
77      ENCLOSED BY ''
78      LINES TERMINATED BY '\n'
79      IGNORE 1 ROWS;
80
81 •  SELECT * FROM credit_cards;
```

id	user_id	iban	pan	pin	cvv	track1	track2	expiring_date
CcU-2938	275	TR3019503122135768176386661	5424465566813633	3257	984	%B8383712448554646^WovsxejDpwiev...	%B7653863056044187=800716333673	10/30/22
CcU-2945	274	DO2685476374853747521656868689	5142423821948828	9080	887	%B4621311609958661^UftuyfsSeimxn^...	%B4149568437843501=510714033071	08/24/23
CcU-2952	273	BG45IVQL52710525608255	4556 453 55 5287	4598	438	%B2183285104307501^CddyytcUxwfdq...	%B6778580257827162=690685974007	06/29/21
CcU-2959	272	CR7242477244335841535	372461377349375	3583	667	%B7281111956795320^XocddijBckecd^...	%B4246154489281853=280522391678	02/24/24

credit_cards 10 ×

Output

Action Output

#	Time	Action	Message
✔ 1	11:21:52	SELECT * FROM credit_cards	275 row(s) returned

Transactions

La tabla Transactions es la tambla de hechos es la tabla central donde todas las otras entidandes hacen una relacion N-1, la tabla tiene como primary key id y como forgein key card_id hace referencia a la tabla credit card, business_id hace referencia a la tabla companies y user_id hace referencia a la tabla users

```
84 •  CREATE TABLE transactions(
85      id VARCHAR(255) PRIMARY KEY,
86      card_id VARCHAR(20),
87      business_id  VARCHAR(10) ,
88      timestamp date,
89      amount DECIMAL(10,2),
90      declined TINYINT,
91      product_ids VARCHAR(20),
92      user_id int,
93      lat FLOAT,
94      longitude FLOAT,
95      CONSTRAINT FK_TransactionsCredit FOREIGN KEY (card_id)
96      REFERENCES credit_cards(id),
97      CONSTRAINT FK_Transactionscompanies FOREIGN KEY (business_id)
98      REFERENCES companies(company_id),
99      CONSTRAINT FK_TransactionsUser FOREIGN KEY (user_id)
100     REFERENCES users(id)
101     );
102
```

```
102
103 ● LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/transactions.csv'
104     INTO TABLE transactions
105     FIELDS TERMINATED BY ';'
106     OPTIONALLY ENCLOSED BY '"'
107     LINES TERMINATED BY '\n'
108     IGNORE 1 ROWS;
109
110 ● SELECT * FROM transactions;
```

id	card_id	business_id	timestamp	amount	declined	product_ids	user_id	lat	longitude
02C6201E-D90A-1859-B4EE-88D2986D3...	CcU-2938	b-2362	2021-08-28	466.92	0	71, 1, 19	92	81.9185	-12.5276
0466A42E-47CF-8D24-FD01-C0B689713...	CcU-4219	b-2302	2021-07-26	49.53	0	47, 97, 43	170	-43.9695	-117.525
063FBA79-99EC-66FB-29F7-25726D176...	CcU-2987	b-2250	2022-01-06	92.61	0	47, 67, 31, 5	275	-81.2227	-129.05
0668296C-CDB9-A883-76BC-2E4C44F8C...	CcU-3743	b-2618	2022-01-26	394.18	0	89, 83, 79	265	-34.3593	-100.556

transactions 8 ×

Output

Action Output ▾

#	Time	Action	Message
● 1	12:13:21	SELECT * FROM transactions	587 row(s) returned

- Exercici 1

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

```
1    -- Nivell 1
2  ⊖ /*- Exercici 1
3    Realitza una subconsulta que mostri tots els usuaris amb més de 30 transaccions utilitzant almenys 2 taules.
4    */
5 ● ⊖ SELECT * FROM users WHERE id IN (
6    SELECT user_id FROM transactions GROUP BY user_id HAVING count(*) > 30);
```

id	name	surname	phone	email	birth_date	country	city	postal_code	address
92	Lynn	Riddle	1-387-885-4057	vitae.aliquet@outlook.edu	Sep 21, 1984	United States	Bozeman	61871	P.O. Box 712, 7907 Est St.
267	Ocean	Nelson	079-481-2745	aenean@yahoo.com	Dec 26, 1991	Canada	Charlottetown	85X 3P4	Ap #732-8357 Pede, Rd.
272	Hedwig	Gilbert	064-204-8788	sem.eget@icloud.edu	Apr 16, 1991	Canada	Tuktoyaktuk	Q4C 3G7	P.O. Box 496, 5145 Sapien Road
275	Kenyon	Hartman	082-871-7248	convallis.ante.lectus@yahoo.com	Aug 3, 1982	Canada	Richmond	R8H 2K2	8564 Facilisi. St.

users 70 ×

Output

Action Output ▾

#	Time	Action	Message
● 1	12:33:55	SELECT * FROM users WHERE id IN (SELECT user_id FROM transactions GROUP BY user_id HAVING count(*) >...	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.

```
8  ⊖ /*- Exercici 2
9    Mostra la mitjana d'amount per IBAN de les targetes de crèdit a la companyia Donec Ltd, utilitza almenys 2 taules.
10   */
11 ● SELECT coalesce(round(avg(amount),2))
12   FROM transactions tt
13   JOIN credit_cards cc ON tt.card_id = cc.id
14 ⊖ WHERE tt.business_id = (
15       SELECT company_id
16       FROM companies
17       WHERE company_name = "Donec Ltd"
18       AND declined = 0)
19   GROUP BY iban;
20
```

coalesce(round(avg(amount),2))
42.82

Result 99 ×

Output

Action Output ▾

#	Time	Action	Message
● 1	11:18:31	SELECT coalesce(round(avg(amount),2)) FROM transactions tt JOIN credit_cards cc ON tt.card_id = cc.id WHER...	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:

```
20   /*Nivell 2
21   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:*/
22  CREATE TABLE ActiveCard (
23      id VARCHAR(20) PRIMARY KEY,
24      status VARCHAR(50),
25      CONSTRAINT FK_CreditCardStatus FOREIGN KEY (id) REFERENCES credit_cards(id)
26  );
27
28  INSERT INTO activecard (id, status)
29  SELECT card_id, CASE WHEN sum(tr.declined) = 3 THEN "Desactiva"
30              ELSE "Activa" END ESTADO FROM (
31  SELECT * ,
32          ROW_NUMBER() OVER (PARTITION BY transactions.card_id ORDER BY transactions.timestamp DESC) ordenTr
33          FROM transactions) as tr WHERE tr.ordenTr BETWEEN 1 AND 3 GROUP BY card_id;
34
35  SELECT * FROM activecard;
```

id	status
CcU-2938	Activa
CcU-2945	Activa
CcU-2952	Activa
CcU-2959	Activa
CcU-2966	Activa

activecard 73

Output

Action Output

#	Time	Action	Message
1	12:38:48	SELECT * FROM activecard	275 row(s) returned

Exercici 1

Quantes targetes estan actives?

```
38   /*Exercici 1
39   Quantes targetes estan actives?*/
40  SELECT count(*) tarjetasActives FROM activecard WHERE status= "Activa" GROUP BY status;
```

tarjetasActives
275

Result 74

Output

Action Output

#	Time	Action	Message
1	12:39:11	SELECT count(*) tarjetasActives FROM activecard WHERE status= "Activa" GROUP BY status	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:
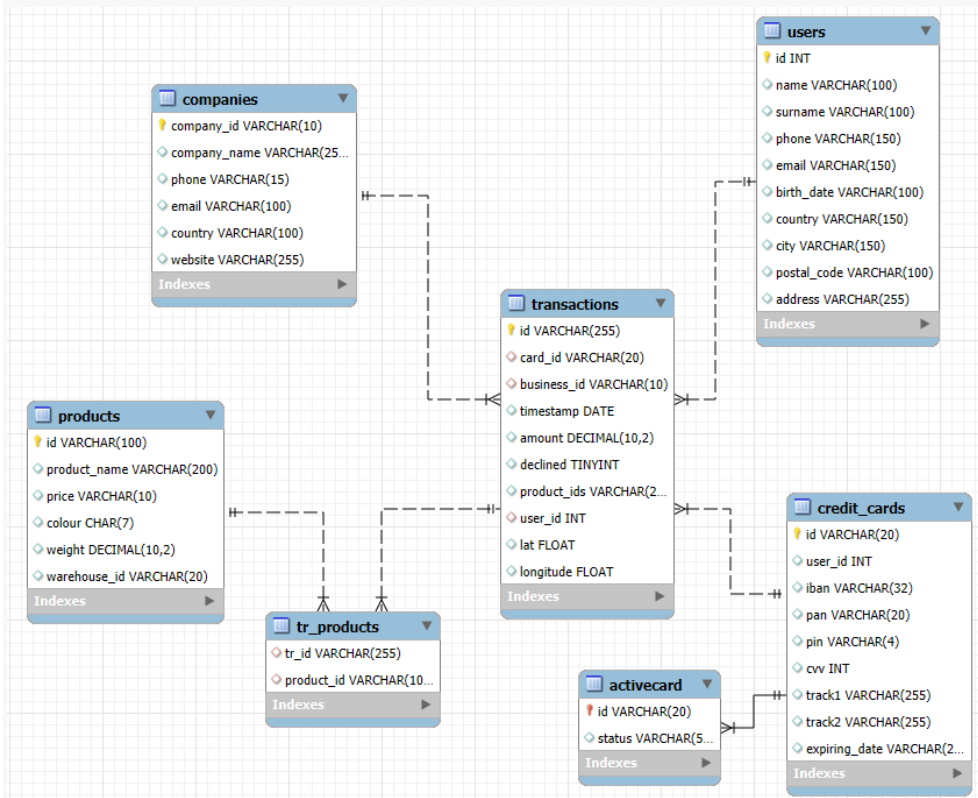
Creo la tabla tr_products para relacionar transacciones y productos. Cada fila de tr_products conecta una transacción (tr_id) con un producto (product_id). El INSERT INTO busca productos dentro de transactions.product_ids (una lista separada por comas) y los inserta en tr_products.

```
45    /*Nivell 3
46    Crea una taula amb la qual puguem unir les dades del nou arxiu products.csv amb la base de dades creada,
47    tenint en compte que des de transaction tens product_ids. Genera la següent consulta:*/
48  • CREATE TABLE products(
49        id VARCHAR(100) PRIMARY KEY,
50        product_name VARCHAR(200),
51        price VARCHAR(10)    ,
52        colour CHAR(7),
53        weight DECIMAL(10,2),
54        warehouse_id VARCHAR(20)
55    );
56
57  • LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/products.csv'
58    INTO TABLE products
59    FIELDS TERMINATED BY ','
60    ENCLOSED BY ''
61    LINES TERMINATED BY '\n'
62    IGNORE 1 ROWS;
63
64  • CREATE TABLE tr_products (
65        tr_id VARCHAR(255),
66        product_id VARCHAR(100),
67        FOREIGN KEY (tr_id) REFERENCES transactions(id) ON DELETE CASCADE,
68        FOREIGN KEY (product_id) REFERENCES products(id) ON DELETE CASCADE
69    );
70
71
```

#	Time	Action	Message
1	11:27:31	CREATE TABLE products(id VARCHAR(100) PRIMARY KEY, product_name VARCHAR(200), price VARCHAR...	0 row(s) affected
2	11:27:33	LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/products.csv' INTO TABLE products FIE...	100 row(s) affected Records: 100 Deleted: 0 Skipped: 0 Warnings: 0
3	11:27:37	CREATE TABLE tr_products (tr_id VARCHAR(255), product_id VARCHAR(100), FOREIGN KEY (tr_id) REFER...	0 row(s) affected

```
INSERT INTO tr_products (tr_id, product_id)
SELECT transactions.id AS tr_id, products.id AS product_id
FROM transactions
JOIN products
    ON FIND_IN_SET(CAST(products.id AS CHAR), REPLACE(transactions.product_ids,' ', '')) > 0;
```

Exercici 1

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

```
85    /*Exercici 1
86     Necessitem conèixer el nombre de vegades que s'ha venut cada producte.*/
87  •  SELECT product_name, COUNT(trp.tr_id) ventasproducto
88     FROM tr_products trp
89         JOIN products pr
90         ON trp.product_id = pr.id
91     GROUP BY pr.id;
92
93
```

product_name	ventasproducto
Direwolf Stannis	61
Karstark Dorne	48
palpatine chewbacca	60
skywalker ewok sith	61
dooku solo	49
Tarly Stark	65

Result 115 ×

Output

Action Output

#	Time	Action	Message
✓ 1	11:41:20	SELECT product_name, COUNT(trp.tr_id) ventasproducto FROM tr_products trp JOIN products pr ON trp.produ...	26 row(s) returned