

Tarea S4. Jianji Chen

Nivell 1

Exercici 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 requistes:

(1) crear un esquema “tarea_s4” y 4 tablas, e introducir datos

```
4 • CREATE DATABASE tarea_s4;
5 • USE tarea_s4;

7 • CREATE TABLE companies(
8     company_id VARCHAR(20) PRIMARY KEY,
9     company_name VARCHAR(255),
10    phone VARCHAR(15),
11    email VARCHAR(100),
12    country VARCHAR(100),
13    website VARCHAR(255));

15 • CREATE TABLE credit_cards(
16     id VARCHAR(20) PRIMARY KEY,
17     user_id INT REFERENCES users(id),
18     iban VARCHAR(50),
19     pan VARCHAR(50),
20     pin VARCHAR(4),
21     cvv INT,
22     track1 VARCHAR(255),
23     track2 VARCHAR(255),
24     expiring_date VARCHAR(20));

26 • CREATE TABLE users(
27     id INT PRIMARY KEY,
28     name VARCHAR(100),
29     surname VARCHAR(100),
30     phone VARCHAR(150),
31     email VARCHAR(150),
32     birth_date VARCHAR(100),
33     country VARCHAR(150),
34     city VARCHAR(150),
35     postal_code VARCHAR(100),
36     address VARCHAR(255));
```

```

38 • CREATE TABLE transactions(
39     id VARCHAR(255) PRIMARY KEY,
40     card_id VARCHAR(15) REFERENCES credit_cards(id),
41     business_id VARCHAR(20) REFERENCES companies(company_id),
42     timestamp TIMESTAMP,
43     amount DECIMAL(10, 2),
44     declined BOOLEAN,
45     product_ids VARCHAR(255),
46     user_id INT REFERENCES users(id),
47     lat FLOAT,
48     longitude FLOAT,
49     FOREIGN KEY (business_id) REFERENCES companies(company_id),
50     FOREIGN KEY (card_id) REFERENCES credit_cards(id),
51     FOREIGN KEY (user_id) REFERENCES users(id));

53 • LOAD DATA LOCAL INFILE
54     "C:\\EEE\\IT Academy_Analisis de Dades\\Especialitzacio_DA\\Tasca S4.01. Creacio de Base de Dades\\companies.csv"
55     INTO TABLE companies
56     CHARACTER SET 'UTF8MB4'
57     FIELDS TERMINATED BY ','
58     ENCLOSED BY '"'
59     LINES TERMINATED BY '\r\n'
60     IGNORE 1 LINES;

62 • LOAD DATA LOCAL INFILE
63     "C:\\EEE\\IT Academy_Analisis de Dades\\Especialitzacio_DA\\Tasca S4.01. Creacio de Base de Dades\\credit_cards.csv"
64     INTO TABLE credit_cards
65     CHARACTER SET 'UTF8MB4'
66     FIELDS TERMINATED BY ','
67     ENCLOSED BY '"'
68     LINES TERMINATED BY '\n'
69     IGNORE 1 LINES;

71 • LOAD DATA LOCAL INFILE
72     "C:\\EEE\\IT Academy_Analisis de Dades\\Especialitzacio_DA\\Tasca S4.01. Creacio de Base de Dades\\users_ca.csv"
73     INTO TABLE users
74     CHARACTER SET 'UTF8MB4'
75     FIELDS TERMINATED BY ','
76     ENCLOSED BY '"'
77     LINES TERMINATED BY '\r\n'
78     IGNORE 1 LINES;

80 • LOAD DATA LOCAL INFILE
81     "C:\\EEE\\IT Academy_Analisis de Dades\\Especialitzacio_DA\\Tasca S4.01. Creacio de Base de Dades\\users_uk.csv"
82     INTO TABLE users
83     CHARACTER SET 'UTF8MB4'
84     FIELDS TERMINATED BY ','
85     ENCLOSED BY '"'
86     LINES TERMINATED BY '\r\n'
87     IGNORE 1 LINES;

```

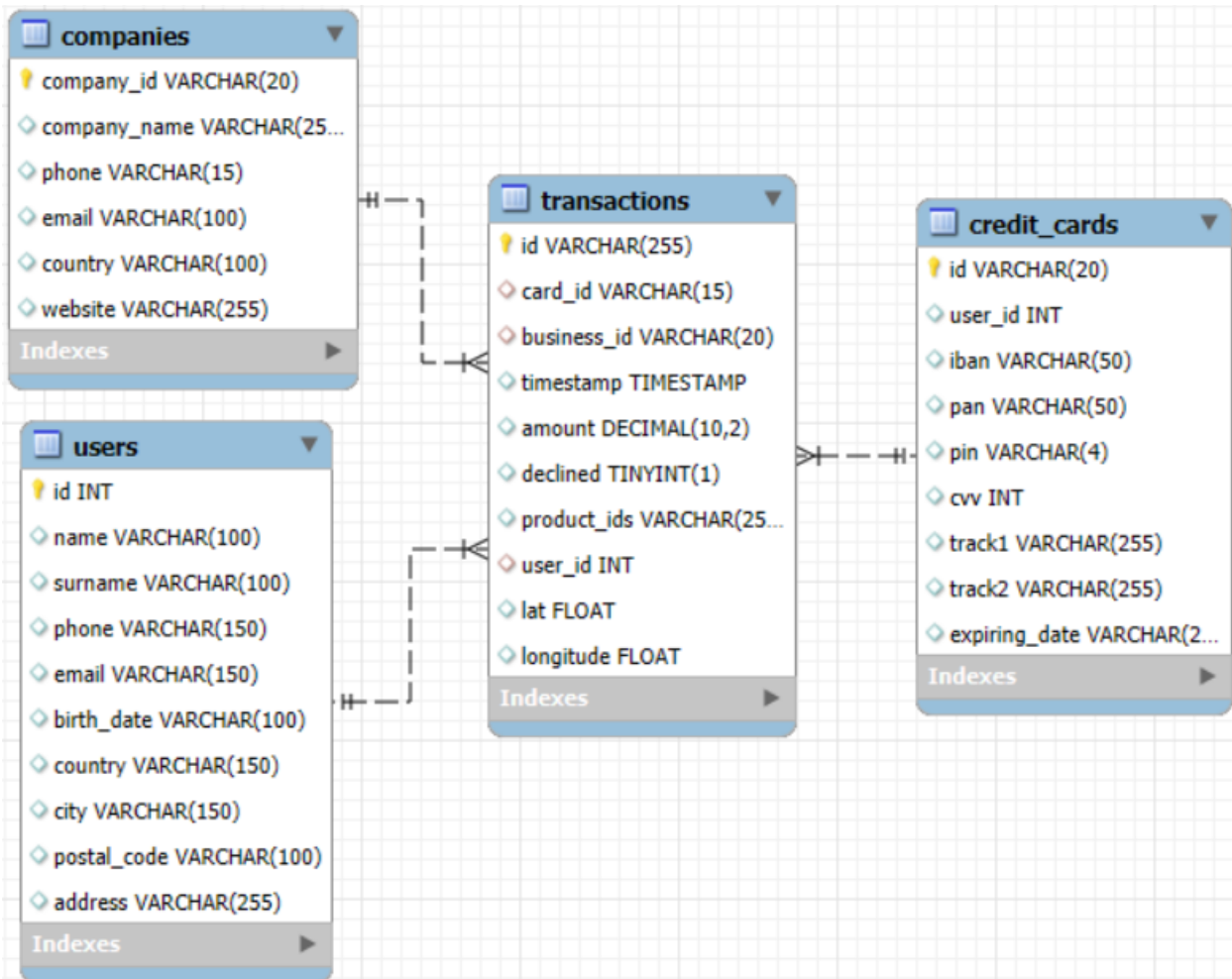
```

89 • LOAD DATA LOCAL INFILE
90 "C:\\EEE\\IT Academy_Analisis de Dades\\Especialitazacio_DA\\Tasca S4.01. Creacio de Base de Dades\\users_usa.csv"
91 INTO TABLE users
92 CHARACTER SET 'UTF8MB4'
93 FIELDS TERMINATED BY ','
94 ENCLOSED BY '"'
95 LINES TERMINATED BY '\\r\\n'
96 IGNORE 1 LINES;

98 • LOAD DATA LOCAL INFILE
99 "C:\\EEE\\IT Academy_Analisis de Dades\\Especialitazacio_DA\\Tasca S4.01. Creacio de Base de Dades\\transactions.csv"
100 INTO TABLE transactions
101 CHARACTER SET 'UTF8MB4'
102 FIELDS TERMINATED BY ';'
103 ENCLOSED BY '"'
104 LINES TERMINATED BY '\\r\\n'
105 IGNORE 1 LINES;

```

(2) crear un diagrama de modelo de estrella de las 4 tablas



Exercici 1

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

```
109 • SELECT u.*
110 FROM users u
111 INNER JOIN transactions t
112 ON u.id = t.user_id
113 GROUP BY u.id
114 HAVING COUNT(t.id) > 30;
```

	id	name	surname	phone	email	birth_date	country	city
▶	92	Lynn	Riddle	1-387-885-4057	vitae.aliquet@outlook.edu	Sep 21, 1984	United States	Bozeman
	267	Ocean	Nelson	079-481-2745	aenean@yahoo.com	Dec 26, 1991	Canada	Charlottetown
	272	Hedwig	Gilbert	064-204-8788	sem.eget@icloud.edu	Apr 16, 1991	Canada	Tuktoyaktuk
	275	Kenyon	Hartman	082-871-7248	convallis.ante.lectus@yahoo.com	Aug 3, 1982	Canada	Richmond

Exercici 2

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

```
119 • SELECT c.company_name, cc.iban, AVG(t.amount) AS mean_sale
120 FROM transactions t
121 INNER JOIN credit_cards cc
122 ON t.card_id = cc.id
123 INNER JOIN companies c
124 ON t.business_id = c.company_id
125 WHERE c.company_name = "Donec Ltd"
126 GROUP BY c.company_name, cc.iban;
```

	company_name	iban	mean_sale
▶	Donec Ltd	PT87806228135092429456346	203.715000

Nivell 2

Exercici 1

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: Quantes targetes estan actives?

(1) Crear la taula

```
134 • CREATE TABLE card_status
135     SELECT tt.card_id,
136            STR_TO_DATE(cc.expiring_date, '%m/%d/%Y') AS expiring_date,
137            tt.id AS transaction_id,
138            tt.timestamp, tt.declined
139     FROM credit_cards cc
140     JOIN (
141         SELECT *,
142                ROW_NUMBER() OVER (PARTITION BY card_id ORDER BY timestamp DESC) AS rn
143         FROM transactions) tt
144     ON cc.id = tt.card_id
145     WHERE tt.rn <= 3;
146 • SELECT * FROM card_status;
```

card_id	expiring_date	transaction_id	timestamp	declined
CdU-2938	2022-10-30	AD85A78A-8829-5746-93A0-8B7A792EBC18	2022-03-12 09:23:10	0
CdU-2938	2022-10-30	F1A598A2-86C5-50A9-F1CE-FB1D69866C39	2022-03-09 20:53:59	0
CdU-2938	2022-10-30	55166D02-D74C-6A63-6C54-8678467649B4	2022-02-24 11:01:42	0
CdU-2945	2023-08-24	7DC26247-20EC-53FE-E555-B6C2E55CA5D5	2022-02-04 15:52:56	0
CdU-2945	2023-08-24	FE96CE47-BD59-381C-4E18-E3CA3D44E8FF	2021-06-15 00:26:29	1
CdU-2952	2021-06-29	72997E96-DC2C-A4D7-7C24-66C302F8AE5A	2022-01-30 15:16:36	0

(2) Consulta: cantidad de tarjetas activas actualmente

```
148 • SELECT COUNT(DISTINCT card_id) AS num_card_active
149     FROM card_status
150     WHERE expiring_date > NOW();
```

num_card_active
32

Nivell 3

Exercici 1

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

(1) crear la tabla "products" e introducir los datos

```
159 • CREATE TABLE products(  
160     id INT PRIMARY KEY,  
161     product_name VARCHAR(100),  
162     price VARCHAR(20),  
163     colour VARCHAR(20),  
164     weight DECIMAL(10, 1),  
165     warehouse_id VARCHAR(20));  
166  
167 • LOAD DATA LOCAL INFILE  
168     "C:\\EEE\\IT Academy_Analisis de Dades\\Especialitzacio_DA\\Tasca S4.01. Creacio de Base de Dades\\products.csv"  
169     INTO TABLE products  
170     CHARACTER SET 'UTF8MB4'  
171     FIELDS TERMINATED BY ','  
172     ENCLOSED BY ''''  
173     LINES TERMINATED BY '\\n'  
174     IGNORE 1 LINES;  
  
176 • UPDATE products  
177     SET price = REPLACE(price, "$", "")  
178     WHERE id <> "";  
179  
180 • ALTER TABLE products  
181     MODIFY price DECIMAL(10, 2);  
182  
183 • SELECT * FROM products;
```

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

	id	product_name	price	colour	weight	warehouse_id
▶	1	Direwolf Stannis	161.11	#7c7c7c	1.0	WH-4
	2	Tarly Stark	9.24	#919191	2.0	WH-3
	3	duel tourney Lannister	171.13	#d8d8d8	1.5	WH-2

(2) encontrar la cantidad máxima de productos por transacción

```

186 • SELECT SUM(LENGTH(product_ids) - LENGTH(REPLACE(product_ids, ",", "")) + 1) AS num_product
187 FROM transactions
188 GROUP BY id
189 ORDER BY num_product DESC
190 LIMIT 1;

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:	Fetch rows:
num_product				
4				

(3) crear una tabla “numbers” de numeros de 1 hasta 4, la cantidad maxima de productos por transaction, para separar product_ids en la tabla transactions

```

194 • CREATE TABLE numbers(
195     n int);
196 • INSERT INTO numbers VALUES (1),(2),(3),(4);

```

(4) separar product_ids y crear una nueva tabla.

```

199 • CREATE TABLE transactions_products
200 SELECT
201     id AS transaction_id,
202     SUBSTRING_INDEX(SUBSTRING_INDEX(t.product_ids, ",", n.n), ",", -1) product_id
203 FROM transactions t
204 INNER JOIN numbers n
205 ON n.n <= LENGTH(t.product_ids) - LENGTH(REPLACE(t.product_ids, ",", "")) + 1;
206
207 • ALTER TABLE transactions_products
208     MODIFY product_id INT,
209     ADD FOREIGN KEY (transaction_id) REFERENCES transactions(id),
210     ADD FOREIGN KEY (product_id) REFERENCES products(id);
211
212 • 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			

(5) consulta: el número de veces que se ha vendido cada producto.


```

215 • SELECT tp.product_id, p.product_name, COUNT(DISTINCT tp.transaction_id) AS num_transactions
216 FROM transactions t
217 JOIN transactions_products tp
218 ON t.id = tp.product_id
219 JOIN products p
220 ON tp.product_id = p.id
221 WHERE t.declined = 0
222 GROUP BY tp.product_id
223 ORDER BY num_transactions DESC;

```

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

	product_id	product_name	num_transactions
▶	23	riverlands north	68
	79	Direwolf riverlands the	66
	2	Tarly Stark	65

Mostrar el modelo final de tablas

