

Tarea S3.01. Gestió de taules, índex i vistes

Nivel 1

Ejercicio 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.

Usando la bbdd “transactions”, primero creo la tabla nueva “credit_card” especificando sus campos correspondientes. La mayoría de los campos son puramente numéricos, pero se han registrado como VARCHAR porque actúan como códigos y su valor se debe registrar exactamente como aparece. El campo “expiring_date” también se registró como VARCHAR para evitar problemas a la hora de cargar los datos, y posteriormente se convirtió a formato DATE.

```
17      -- Crear tabla "credit_card"
18      CREATE TABLE IF NOT EXISTS credit_card (
19          id VARCHAR(20) NOT NULL,
20          iban VARCHAR(50),
21          pan VARCHAR(20),
22          pin VARCHAR(4),
23          cvv VARCHAR(3),
24          expiring_date VARCHAR(20), -- DATE('%d/%m/%Y')
25          PRIMARY KEY (id)
26      );
27      -- Añadimos FK en tabla transaction
28      -- Añadimos los datos: datos_introducir_credit.sql --> 275 filas
29      ALTER TABLE transaction
30      ADD FOREIGN KEY (credit_card_id) REFERENCES credit_card(id);
31
32      -- Una vez cargados los datos, modifico el tipo de dato de expiring_date
33      UPDATE credit_card
34      SET expiring_date = STR_TO_DATE(expiring_date, '%m/%d/%y');
35      ALTER TABLE credit_card
36      MODIFY COLUMN expiring_date DATE;
37
38      -- Compruebo:
39      SELECT * FROM transactions.credit_card;
```

id	iban	pan	pin	cvv	expiring_date
CcU-2938	TR301950312213576817638661	5424465566813633	3257	984	2022-10-30
CcU-2945	DO26854763748537475216568689	5142423821948828	9080	887	2023-08-24
CcU-2952	BG45IVQL52710525608255	4556 453 55 5287	4598	438	2021-06-29
CcU-2959	CR7242477244335841535	372461377349375	3583	667	2023-02-24
CcU-2966	BG72LKTQ15627628377363	448566 886747 7265	4900	130	2024-10-29

#	Time	Action	Message
1	21:11:02	UPDATE credit_card SET expiring_date = STR_TO_DATE(expir...	275 row(s) affected Rows matched: 275 Changed: 275 Warning...
2	21:11:07	ALTER TABLE credit_card MODIFY COLUMN expiring_date DA...	275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0
3	21:11:13	SELECT * FROM transactions.credit_card	275 row(s) returned

Ejercicio 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.

```

49      -- Compruebo previamente si existe el registro:
50 •   SELECT *
51     FROM credit_card
52    WHERE id = 'CcU-2938'; -- Sí existe
53      -- Actualizo el registro:
54 •   UPDATE credit_card
55      SET iban = 'R323456312213576817699999'
56      WHERE id = 'CcU-2938';
57      -- Compruebo:
58 •   SELECT *
59     FROM credit_card
60    WHERE id = 'CcU-2938';
61      -- El IBAN cambió de 'TR301950312213576817638661' a 'R323456312213576817699999'
62

```

Result Grid						
Filter Rows:						
Edit:						
Export/Import:						
Wrap Cell Content:						
	id	iban	pan	pin	cvv	expiring_date
▶	CcU-2938	R323456312213576817699999	5424465566813633	3257	984	2022-10-30
*	NULL	NULL	NULL	NULL	NULL	NULL

credit_card 3 x			
Output			
Action Output			
#	Time	Action	Message
✓ 4	21:16:05	SELECT * FROM credit_card WHERE id = 'CcU-2938'	1 row(s) returned
✓ 5	21:17:14	UPDATE credit_card SET iban = 'R323456312213576817699...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
✓ 6	21:17:18	SELECT * FROM credit_card WHERE id = 'CcU-2938'	1 row(s) returned

Ejercicio 3 –

En la taula "transaction" ingresa un nou usuari amb la següent informació:

Al intentar ingresar un nuevo registro en la tabla "transaction", aparece un error código 1042 por "constraint", ya que se estaría añadiendo un registro que no está en las tablas "company" y "credit_card", creando un conflicto. Lo correcto sería crear primero los registros correspondientes en las otras tablas y luego añadir el registro en "transaction", pero en este caso como no disponemos de la información necesaria para completar las otras tablas, vamos a quitar la obligatoriedad de mantener la coherencia con las foreign keys.

```
78      -- Registros antes de la adición:
79 •   SELECT COUNT(*)
80     FROM transaction; -- 587 registros
81      -- Adición:
82 •   SET foreign_key_checks = 0;
83 •   INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude,
84 •   SET foreign_key_checks = 1;
85      -- Compruebo:
86 •   SELECT COUNT(*)
87     FROM transaction; -- 588 registros
88
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
COUNT(*)			
588			

Result 5 ×

Output

Action Output

	#	Time	Action	Message
✓	9	21:26:10	SET foreign_key_checks = 0	0 row(s) affected
✓	10	21:26:15	INSERT INTO transaction (id, credit_card_id, company_id, user...	1 row(s) affected
✓	11	21:26:17	SET foreign_key_checks = 1	0 row(s) affected
✓	12	21:26:22	SELECT COUNT(*) FROM transaction	1 row(s) returned

Ejercicio 4 –

Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Recordar mostrar el canvi realitzat.

Primero muestro con un comando "DESCRIBE" la situación de la tabla antes de manipularla para demostrar que tiene 6 columnas iniciales, y luego al eliminar "pan" queda una tabla de 5 columnas:

```
93      -- Antes del cambio:
94 •   DESCRIBE credit_card; -- 6 filas de resultados corresponden a 6 columnas de la tabla
95      -- Elimino columna pan:
96 •   ALTER TABLE credit_card
97     DROP COLUMN pan;
98      -- Compruebo:
99 •   DESCRIBE credit_card; -- 5 filas, una columna menos
100 •  SELECT *
101     FROM credit_card; -- muestro la tabla entera
```

Result Grid					
Filter Rows: <input type="text"/>					
Edit:					
Export/Import:					
Wrap Cell Content:					
id	iban	pin	cvv	expiring_date	
CcU-2938	R323456312213576817699999	3257	984	2022-10-30	
CcU-2945	DO26854763748537475216568689	9080	887	2023-08-24	
CcU-2952	BG45IVQL52710525608255	4598	438	2021-06-29	
CcU-2959	CR7242477244335841535	3583	667	2023-02-24	
CcU-2966	BG72LKTQ15627628377363	4900	130	2024-10-29	
CcU-2973	PT87806228135092429456346	8760	887	2025-01-30	

credit_card 8 ×

Output

#	Time	Action	Message
✓ 13	21:29:34	DESCRIBE credit_card	6 row(s) returned
✓ 14	21:29:39	ALTER TABLE credit_card DROP COLUMN pan	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
✓ 15	21:29:43	DESCRIBE credit_card	5 row(s) returned
✓ 16	21:29:48	SELECT * FROM credit_card	275 row(s) returned

Nivel 2

Ejercicio 1 –

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

Inicialmente tenemos 587 registros en la tabla y después de borrar el registro indicado, tenemos 586:

```

110      -- Antes del cambio:
111 • SELECT *
112 FROM transaction; -- 587 registros
113 • SELECT *
114 FROM transaction
115 WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02'; -- compruebo que el registro existe
116      -- Elimino el registro indicado
117 • DELETE FROM transaction
118 WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02';
119      -- Compruebo:
120 • SELECT *
121 FROM transaction; -- 586 registros
122 • SELECT *
123 FROM transaction
124 WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02'; -- 0 rows porque el registro ya no existe

```

Result Grid									
id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined	
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	

transaction 9					Apply	Rever
Output						
#	Time	Action	Message	Duration / Fetch		
✓ 1017	12:04:31	DELETE FROM transaction WHERE id = '02C6201E...	1 row(s) affected	0.000 sec		
✓ 1018	12:04:34	SELECT * FROM transaction	586 row(s) returned	0.000 sec / 0.000 sec		
✓ 1019	12:04:41	SELECT * FROM transaction WHERE id = '02C6201...	0 row(s) returned	0.000 sec / 0.000 sec		

Ejercicio 2 –

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.

Se ha creado una nueva vista:

```
135 • CREATE VIEW VistaMarketing AS
136 SELECT c.company_name, c.phone, c.country, ROUND(AVG(t.amount), 2) AS transaction_average
137 FROM company AS c
138 JOIN transaction AS t
139 ON c.id = t.company_id
140 WHERE t.declined = 0
141 GROUP BY c.company_name, c.phone, c.country
142 ORDER BY transaction_average DESC;
143 -- Compruebo:
144 • SELECT * FROM transactions.vistamarketing;
```

company_name	phone	country	transaction_average
Eget Ipsum Ltd	03 67 44 56 72	United States	481.86
Sed Id Limited	07 28 18 18 13	United States	477.51
Neque Tellus Incorporated	04 43 18 34 19	Ireland	477.10
Nunc Sit Incorporated	07 28 42 63 63	Norway	461.83
Non Magna LLC	06 71 73 13 17	United Kingdom	458.74
Maecenas Malesuada Fringilla Inc.	09 38 53 76 61	Netherlands	451.29

vistamarketing 24 x

Output

Action Output

#	Time	Action	Message
✓ 1	21:38:00	SELECT * FROM transactions.vistamarketing	100 row(s) returned

Ejercicio 3 –

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

Son 8 las empresas que tienen sede en Alemania:

```
149 • SELECT *
150 FROM vistamarketing
151 WHERE country = 'Germany'; -- 8 empresas
```

company_name	phone	country	transaction_average
Ac Industries	09 34 65 40 60	Germany	396.15
Auctor Mauris Corp.	05 62 87 14 41	Germany	308.99
Ac Fermentum Incorporated	06 85 56 52 33	Germany	293.57
Aliquam PC	01 45 73 52 16	Germany	280.34
Rutrum Non Inc.	02 66 31 61 09	Germany	266.90
Nunc Interdum Incorporated	05 18 15 48 13	Germany	242.95

vistamarketing 26 x

Output

Action Output

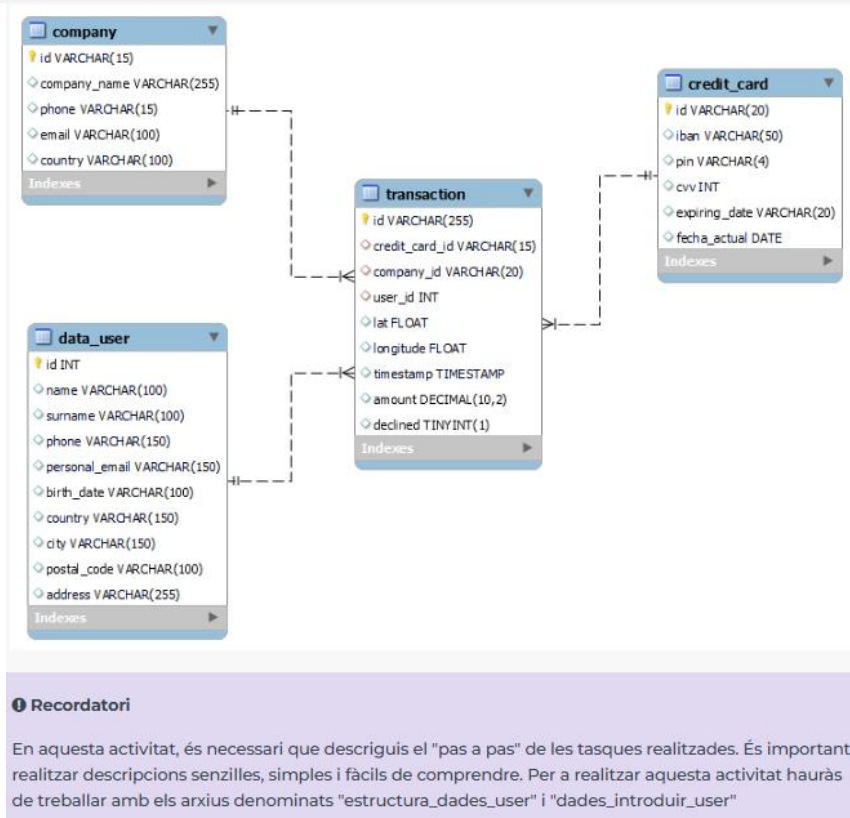
#	Time	Action	Message
✓ 1	21:39:49	SELECT * FROM vistamarketing WHERE country = 'Germany'	8 row(s) returned

Nivel 3

Ejercicio 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:



Observamos una a una cada tabla:

- Transaction:
 - o Sin cambios significativos.
- Credit_card:
 - o Crear campo fecha_actual.
- Company:
 - o Eliminar campo website.
- User:
 - o Cambiar nombres: tabla user, campo email.

Observamos las relaciones entre las tablas:

- La estructura de la tabla user que presenta el ejercicio, existe una relación inversa a la esperada, ya que indica que la columna user.id es una clave foránea que hace referencia a la columna transaction.user_id.

Por tanto, para corregir los errores del compañero, se debería hacer lo siguiente:

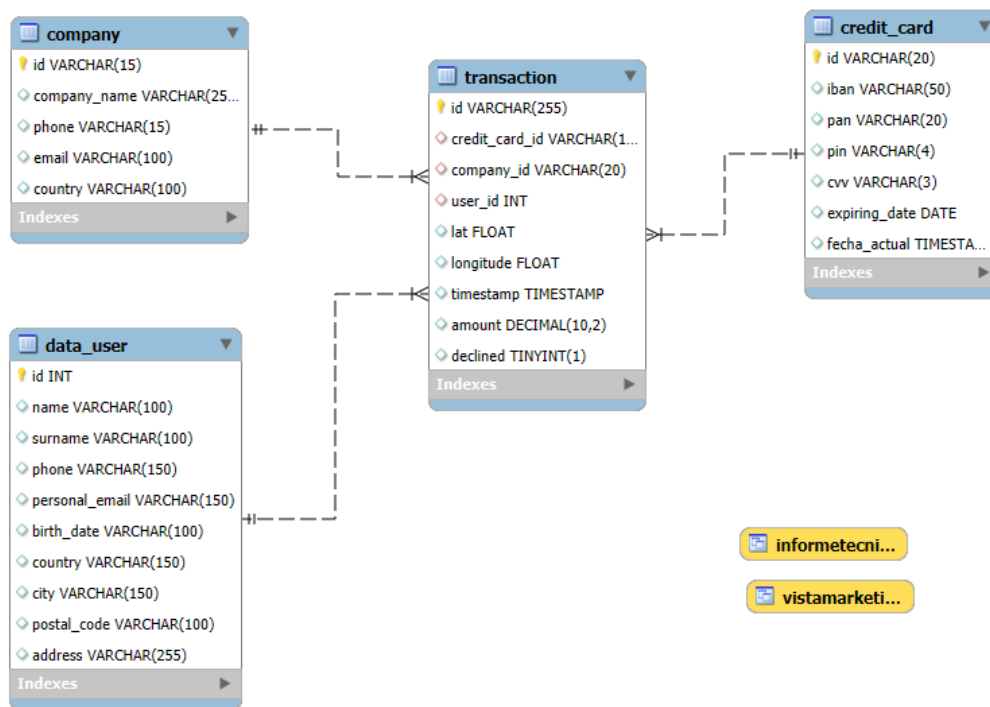
```

164      -- 1. Añadir columna credit_card.fecha_actual:
165 • ALTER TABLE credit_card
166 ADD fecha_actual TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP;
167      -- de esta forma se actualizará automáticamente al hacer uso de ella
168      -- 2. Eliminar columna company.website:
169 • ALTER TABLE company
170 DROP COLUMN website;
171      -- 3. Modificar nombres tabla user:
172 • ALTER TABLE user RENAME data_user;
173 • ALTER TABLE data_user RENAME COLUMN email TO personal_email;
174      -- 4. Modificar relación data_user - transaction:
175 • ALTER TABLE transaction
176 ADD CONSTRAINT fk_transaction_user_id FOREIGN KEY (user_id) REFERENCES data_user(id);
177 • ALTER TABLE data_user
178 DROP FOREIGN KEY data_user_ibfk_1;

```

Nota: no se cambió el tipo de dato credit_card.cvv a INT, ya que sería más correcto mantenerlo como VARCHAR.

El modelo final queda así:



Ejercicio 2 –

L'empresa també et sol·licita 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.

Se creó la vista InformeTecnico:

```

191      -- Solo con transacciones aceptadas:
192      -- t.id, du.name, u.surname, cc.iban, c.company_name
193      -- Al cargar tabla user tengo que añadir la FK en transactions
194 • SET foreign_key_checks = 0;
195 • ALTER TABLE transaction
196 ADD FOREIGN KEY (user_id) REFERENCES user(id);
197 • SET foreign_key_checks = 1;
198      -- Crear la vista
199 • CREATE VIEW InformeTecnico AS
200 SELECT t.id AS transaction_id, du.name AS user_name, du.surname AS user_surname, cc.iban, c.company_name
201 FROM transaction AS t
202 JOIN data_user AS du
203 ON du.id = t.user_id
204 JOIN company AS c
205 ON c.id = t.company_id
206 JOIN credit_card AS cc
207 ON cc.id = t.credit_card_id
208 WHERE t.declined = 0
209 ORDER BY transaction_id ASC;
210      -- Compruebo:
211 • SELECT * FROM transactions.infometechnico;

```

transaction_id	user_name	user_surname	iban	company_name
0466A42E-47CF-8D24-FD01-C0B689713128	William	Benjamin	MD1234119525145401270486	Nunc Interdum Incorporated
063FBA79-99EC-66FB-29F7-25726D1764A5	Kenyon	Hartman	GE89681434837748781813	Amet Nulla Donec Corporation
0668296C-CD89-A883-768C-2E4C4F8C8AE	Chloe	Keith	BA542358041365401657	Non Institute
06CD9AA5-9B42-D684-DDDD-A5E394FEB999	Lynn	Riddle	CR7242477244335841535	Ut Semper Foundation
0A476ED9-0C13-1962-F87B-D3563924B539	Sasha	Emerson	PK1842602717357586645371	Nunc Interdum Incorporated
1017AA59-3D5F-7A4C-1992-D151A8D1FA0A	Ocean	Nelson	CY04215371490518737133975175	Non Institute

informetecnico 13 x

Output

Action Output

#	Time	Action	Message
✓ 1320	12:43:50	CREATE VIEW InformeTecnico AS SELECT t.id AS transaction_id, du.name AS user_name, du.surname AS user_surname, cc.iban, c.company_name FR...	0 row(s) affected
✗ 1321	12:43:54	SELECT * FROM transactions.infometechnico2	Error Code: 1146. Table 'transactions.infometechnico2' doesn't exist
✓ 1322	12:43:59	SELECT * FROM transactions.infometechnico	499 row(s) returned