Tasca S3.01. Manipulació de taules
Descripció
En aquest sprint se simula una situació empresarial en la qual hauràs de realitzar diverses manipulacions a les taules d'una base de dades. A més, treballaràs amb índexs i vistes per optimitzar consultes i organitzar la informació.
Continuaràs treballant amb la base de dades que conté informació d'un marketplace, un entorn similar a Amazon on diverses empreses venen els seus productes a través d'un canal en línia. En aquesta activitat, començaràs a treballar amb dades relacionades amb targetes de crèdit.
Afegeix les taules al model segons correspongui:
Nivell 1: Taula "credit_card"
Nivell 3 : Taula "user"

Nivell 1
Exercici 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.

Per tal de crear la taula 'credit_card' he utilitzat el codi de la captura inferior. Donat que l'id s'ha definit com a PRIMARY KEY, ja s'identifica cadascuna de les targetes de manera única.

Un cop s'ha inserit la informació a partir dels documents sql proporcionats, que hi havia al document sense cap error, i tal i com es veu a la captura inferior, he creat una columna per a transformar les dates en el format correcte de MySQL, he borrat la columna original i he posat el nom que tenia l'original a la nova columna (expired_date). A més, per tal de crear l'associació amb les altres dues taules, he creat una foreign key (prèviament he creat un índex tot i que no era necessari ja que es fa de manera automàtica si no hi és) a la taula transaction amb la columna credit_card_id. Ho he referenciat amb la taula credit_card la columna id.

```sql
ALTER TABLE credit_card ADD COLUMN new_date DATE;
SET SQL_SAFE_UPDATES = 0;
UPDATE credit_card
set new_date = str_to_date(expiring_date, '%m/%d/%Y');
SET SQL_SAFE_UPDATES = 1;

ALTER TABLE credit_card DROP COLUMN expiring_date;
ALTER TABLE credit_card RENAME COLUMN new_date TO expiring_date;

-- CREATE INDEX idx_credit_card_id
-- ON transaction (credit_card_id); -- not needed since when creating the foreign key then it will be created

-- ALTER TABLE transaction
-- DROP FOREIGN KEY fk_credit_card_id;

ALTER TABLE transaction
ADD CONSTRAINT fk_credit_card_id
FOREIGN KEY (credit_card_id) REFERENCES credit_card(id)
ON UPDATE CASCADE; -- whenever the primary key will be updated in the parent, the value in the child would also be updated.
```

100% 39:39

Action Output

	Time	Action	Response	Duration / F
11...	12:49:20	ALTER TABLE credit_card ADD COLUMN new_date DATE	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.011 sec
11...	12:49:20	SET SQL_SAFE_UPDATES = 0	0 row(s) affected	0.00079 sec
11...	12:49:20	UPDATE credit_card set new_date = str_to_date(expiring_date, '%m/%d/%Y')	5000 row(s) affected Rows matched: 5000 Changed: 5000 Warni...	0.080 sec
11...	12:49:20	SET SQL_SAFE_UPDATES = 1	0 row(s) affected	0.00024 sec
11...	12:49:20	ALTER TABLE credit_card DROP COLUMN expiring_date	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.0032 sec
11...	12:49:20	ALTER TABLE credit_card RENAME COLUMN new_date TO expiring_date	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.0029 sec
11...	12:49:20	ALTER TABLE transaction ADD CONSTRAINT fk_credit_card_id FOREIGN KEY (credit_card_id) REFERENCES...	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings:...	0.565 sec
11...	12:49:49	ALTER TABLE transaction DROP FOREIGN KEY fk_credit_card_id	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.0073 sec
11...	12:49:55	ALTER TABLE transaction ADD CONSTRAINT fk_credit_card_id FOREIGN KEY (credit_card_id) REFERENCES...	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings:...	0.595 sec

El resultat d'aquests canvis es veu en el següent diagrama. En ell s'observen les diferents columnes per a cada taula, amb relacions de la taula transaccions amb les taules company i credit_card. Així que en la bbdd el que veiem és informació sobre les transaccions que s'han realitzat, de quines compnayies han estat així com amb quina targeta.

Exercici 2

El departament de Recursos Humans ha identificat un error en el número de compte associat a la targeta de crèdit amb ID CcU-2938. La informació que ha de mostrar-se per a aquest registre és: TR323456312213576817699999. Recorda mostrar que el canvi es va realitzar.

```
/*
Exercici 2
El departament de Recursos Humans ha identificat un error en el número de compte associat a la targeta de crèdit amb ID CcU-2938.
La informació que ha de mostrar-se per a aquest registre és: TR323456312213576817699999. Recorda mostrar que el canvi es va realitzar.*/

UPDATE credit_card
SET iban = 'TR323456312213576817699999'
WHERE id = 'CcU-2938';

SELECT id, iban
FROM credit_card
WHERE id = 'CcU-2938';
```

100% 8:50 1 error found

Result Grid | Filter Rows: Search | Edit: | Export/Import:

id	iban
CcU-2938	TR323456312213576817699999
NULL	NULL

credit_card 1 | Apply | Revert

Action Output

	Time	Action	Response	Duration / Fetch Time
1	13:34:42	SELECT * FROM transactions.credit_card LIMIT 0, 50000	5000 row(s) returned	0.0016 sec / 0.0067 s...
2	13:36:19	UPDATE credit_card SET iban = 'TR323456312213576817699999' WHERE id = 'CcU-2938'	1 row(s) affected Rows matched: 1 Changed: 1 Warni...	0.0026 sec
3	13:36:19	SELECT id, iban FROM credit_card WHERE id = 'CcU-2938' LIMIT 0, 50000	1 row(s) returned	0.0017 sec / 0.00001...

Exercici 3

En la taula "transaction" ingressa una nova transacció amb la següent informació:

Id 108B1D1D-5B23-A76C-55EF-C568E49A99DD
credit_card_id CcU-9999
company_id b-9999
user_id 9999
lat 829.999
longitude -117.999
amount 111.11
declined 0

```
INSERT INTO credit_card (id) -- If a value is defined as a foreign key, it first needs to be inserted into the parent table
VALUES
('CcU-9999');

INSERT INTO company (id)
VALUES
('b-9999');

INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined)
VALUES
('108B1D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', '9999', '829.999', '-117.999', 111.11, 0);

SELECT *
FROM transaction
WHERE credit_card_id = 'CcU-9999';
```

100% 35:87

Result Grid | Filter Rows: Search | Edit: | Export/Import:

id	credit_card...	company_id	user_id	lat	longitude	timestamp	amount	declined
1	CcU-9999	b-9999	9999	829.999	-117.999	NULL	111.11	0
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

transaction 2 | Apply | Revert

Action Output

	Time	Action	Response	Duration / Fet
11...	12:53:36	INSERT INTO credit_card (id) -- If a value is defined as a foreign key, it first needs to be inserted into the par...	1 row(s) affected	0.0018 sec
11...	12:53:36	INSERT INTO company (id) VALUES ('b-9999')	1 row(s) affected	0.00086 sec
11...	12:53:36	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES...	1 row(s) affected	0.0011 sec
11...	12:53:36	SELECT * FROM transaction WHERE credit_card_id = 'CcU-9999' LIMIT 0, 50000	1 row(s) returned	0.00080 sec /

Exercici 4

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

```
/* Exercici 4
Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Recorda mostrar el canvi realitzat.
*/

ALTER TABLE credit_card
DROP COLUMN pan;

SELECT *
FROM credit_card
LIMIT 1;
```

Result Grid

id	iban	pin	cvv	expiring_date
CcS-4857	XX4857591835292505850771	1819	467	2025-09-27

credit_card 3

	Time	Action	Response	Duration / Fetch Time
5	21:34:12	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VALUES...	1 row(s) affected	0.00086 sec
6	21:35:58	SELECT credit_card_id FROM transaction WHERE credit_card_id = 'CcU-9999' LIMIT 0, 50000	1 row(s) returned	0.0011 sec / 0.00000...
7	21:36:12	SELECT * FROM transaction WHERE credit_card_id = 'CcU-9999' LIMIT 0, 50000	1 row(s) returned	0.0013 sec / 0.00001...
8	21:39:05	ALTER TABLE credit_card DROP COLUMN pan	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.016 sec
9	21:39:06	SELECT * FROM credit_card LIMIT 1	1 row(s) returned	0.00086 sec / 0.000...

Nivell 2

Exercici 1

Elimina de la taula transaction el registre amb ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD de la base de dades.

```
/*Nivell 2
Exercici 1
Elimina de la taula transaction el registre amb ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD de la base de dades.*/

SELECT id
FROM transaction
WHERE id='000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'; -- to check that it exists

DELETE from transaction
WHERE id='000447FE-B650-4DCF-85DE-C7ED0EE1CAAD';

SELECT id
FROM transaction
WHERE id='000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'; -- to check that it was deleted
```

Result Grid

id
000447FE-B650-4DCF-85DE-C7ED0EE1CAAD

transaction 1 | transaction 2

	Time	Action	Response	Duration / Fetch Time
1	07:55:00	SELECT id FROM transaction WHERE id='000447FE-B650-4DCF-85DE-C7ED0EE1CAAD' LIMIT 0, 50000	1 row(s) returned	0.0082 sec / 0.00001...
2	07:55:00	DELETE from transaction WHERE id='000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'	1 row(s) affected	0.0033 sec
3	07:55:00	SELECT id FROM transaction WHERE id='000447FE-B650-4DCF-85DE-C7ED0EE1CAAD' LIMIT 0, 50000	0 row(s) returned	0.00045 sec / 0.000...

Exercici 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.

A la captura superior, veiem a schema com s'ha creat la vista (esquema de l'esquerra a l'apartat view) i també s'ha comprovat que s'hagi creat bé amb una query. També trobem, en forma de comentari, com donar-li accés a la vista a un usuari, important per controlar l'accés a les dades.

Exercici 3

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

Nivell 3

Exercici 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:

Primer de tot he mirat i executat les comandes dels documents "estructura_dades_user" i "dades_introduir_user".

Primerament he vist que la columna 'Id' a data_user ja està definit com a primary key al crear la taula. Alxí que les accions que he fet, les queries es poden veure a la captura de sota, són:

1. Canviar el tipus de dades de la columna id de la taula data_user de VARCHAR a INT per així que sigui el mateix que user_id de la taula transaction.
2. Afegit a la taula user l'id '9999' donat que apareix a la taula transaction.
3. Definir 'user_id' de la taula 'transaction' com a clau forànea 'fk_user_id' associada amb la clau primària 'id' de la taula user.
4. Canviar el nom de la taula 'user' a 'data_user'
5. De la taula 'company' eliminar la columna 'website'
6. De la taula 'credit_card' modificar el tipus de dada per a 'id', 'pin', 'cvv', i 'expiring_date', així com inserir la columna 'fecha_actual' en format data.
7. De la taula 'transaction' modificar el nombre de caràcters disponibles a la columna credit_card_id.

```
-- 3. Definir fk, taula transaction amb pk a user
ALTER TABLE transaction
ADD CONSTRAINT fk_user_id
FOREIGN KEY (user_id) REFERENCES user(id)
ON UPDATE CASCADE; -- whenever the primary key will be updated in the parent, the value in the child would also be updated.

-- 4. Canviar nom taula user
ALTER TABLE user RENAME TO data_user;

-- 5. De la taula 'company' eliminar la columna 'website'
ALTER TABLE company
DROP COLUMN website;

SELECT *
FROM company
LIMIT 1;
```

Per al següent punt primer he hagut d'eliminar la FK de la taula transaction i al finalitzar l'he tornat a crear.

```
-- 6. De la taula credit_card, modificar tipus de dades i afegir columna fecha_actual
-- Primer he d'eliminar la fk_credit_card_id per a canviar el tipus d'id perquè sinó dóna error.

ALTER TABLE transaction
DROP CONSTRAINT fk_credit_card_id;

ALTER TABLE credit_card
    MODIFY COLUMN id VARCHAR(20),
    MODIFY COLUMN pin VARCHAR(4),
    MODIFY COLUMN cvv INT,
    MODIFY COLUMN expiring_date VARCHAR(20),
    ADD COLUMN fecha_actual DATE;

SELECT *
FROM credit_card
LIMIT 1;
```

Result Grid

id	iban	pin	cvv	expiring_date	fecha_actual
CcS-4857	XX4857591835292505850771	1819	467	2025-09-27	NULL
NULL	NULL	NULL	NULL	NULL	NULL

Action Output

	Time	Action	Response	Duration / F
11...	13:07:01	ALTER TABLE transaction DROP CONSTRAINT fk_credit_card_id	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.015 sec
11...	13:07:01	ALTER TABLE credit_card MODIFY COLUMN id VARCHAR(20), MODIFY COLUMN pin VARCHAR(4), MODIF...	5001 row(s) affected Records: 5001 Duplicates: 0 Warnings: 0	0.033 sec
11...	13:07:01	SELECT * FROM credit_card LIMIT 1	1 row(s) returned	0.00022 sec
11...	13:07:01	ALTER TABLE transaction ADD CONSTRAINT fk_credit_card_id FOREIGN KEY (credit_card_id) REFERENCES...	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings:...	0.691 sec

```
-- Primer he d'eliminar la fk_credit_card_id per a canviar el tipus d'id perquè sinó dóna error.

ALTER TABLE transaction
DROP CONSTRAINT fk_credit_card_id;

ALTER TABLE credit_card
    MODIFY COLUMN id VARCHAR(20),
    MODIFY COLUMN pin VARCHAR(4),
    MODIFY COLUMN cvv INT,
    MODIFY COLUMN expiring_date VARCHAR(20),
    MODIFY COLUMN iban VARCHAR(50),
    ADD COLUMN fecha_actual DATE;
```

```
SELECT *
FROM credit_card
LIMIT 1;

ALTER TABLE transaction
ADD CONSTRAINT fk_credit_card_id
FOREIGN KEY (credit_card_id) REFERENCES credit_card(id)
ON UPDATE CASCADE;

-- 7. De la taula 'transaction' modificar el nombre de caràcters disponibles a la columna credit_card_id.
ALTER TABLE transaction MODIFY COLUMN credit_card_id VARCHAR(20);
```

100% 1:214

Action Output

	Time	Action	Response	Duration / Fet
11...	13:07:01	ALTER TABLE credit_card MODIFY COLUMN id VARCHAR(20), MODIFY COLUMN pin VARCHAR(4), MODIF...	5001 row(s) affected Records: 5001 Duplicates: 0 Warnings: 0	0.033 sec
11...	13:07:01	SELECT * FROM credit_card LIMIT 1	1 row(s) returned	0.00022 sec /
11...	13:07:01	ALTER TABLE transaction ADD CONSTRAINT fk_credit_card_id FOREIGN KEY (credit_card_id) REFERENCES...	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings:...	0.691 sec
11...	13:09:07	ALTER TABLE transaction MODIFY COLUMN credit_card_id VARCHAR(20)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.015 sec

Amb tots aquests canvis queda el diagrama que ha creat el company:

Exercici 2

L'empresa també us demana 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.
- Assegureu-vos d'incloure informació rellevant de les taules que coneixereu i utilitzeu àlies per canviar de nom columnes segons calgui.

Mostra els resultats de la vista, ordena els resultats de forma descendent en funció de la variable ID de transacció.

```sql
-- DROP VIEW informetecnico;

CREATE VIEW InformeTecnico AS
SELECT t.id AS id_transaccio,
    du.name AS nom_usuari,
    du.surname AS cognom_usuari,
    cc.iban,
    cc.expiring_date AS data_caducitat_targeta, -- afegit com a informació rellevant
    c.company_name AS nom_empresa,
    t.declined AS estat_compra -- afegit com a informació rellevant
FROM transaction t
JOIN data_user du
    ON t.user_id = du.id
JOIN credit_card cc
    ON cc.id = t.credit_card_id
JOIN company c
    ON c.id = t.company_id;

SELECT *
FROM informetecnico
ORDER BY id_transaccio DESC;
```

100% 9:244

Result Grid Filter Rows: Search Export: Fetch rows:

id_transaccio	nom_usuari	cognom_usuari	iban	data_caducitat_targeta	nom_empresa	estat_compra
FFFD31D6-9495-47CE-B54A-7DB8E1CC274B	Bmrgli	Tprvvmrc	XX79481445121128918249092 2	2029-07-25	Turpis Company	0
FFFCF76D-ECF0-4985-A2D0-B2A7B75998FC	Dfried	Vilqcjdl	XX6362517016478920366760 34	2028-01-28	Amet Nulla Donec Corporation	0
FFFC9E8D-27C7-4ADE-98F2-7533EF4DF126	Securp	Faofvqfy	XX16267714330422363143756 7	2027-10-28	Nunc Interdum Incorporated	0
FFFB270D-F53A-4D5D-9666-E5307C53CC84	Ggzjpa	Ulrzjulh	XX39511426708201995256705 2	2028-04-28	Viverra Donec Foundation	0

informetecnico 24 Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
50...	14:03:15	DROP VIEW informetecnico	0 row(s) affected	0.0034 sec
50...	14:03:23	CREATE VIEW InformeTecnico AS SELECT t.id AS id_transaccio, du.name AS nom_usuari, du.surname AS cognom_usuari, cc.iban, cc.expiring_date AS data_caducitat_targeta, -- afe...	0 row(s) affected	0.0047 sec
50...	14:03:23	SELECT * FROM informetecnico ORDER BY id_transaccio DESC LIMIT 0, 50000	50000 row(s) returned	0.0059 sec / 0.133 sec