

SPRINT 3: Gestión de tablas, índices y vistas

Descripción

En este sprint, se simula una situación empresarial en la que debes realizar diversas manipulaciones en las tablas de la base de datos. A su vez, tendrás que trabajar con índices y vistas. En esta actividad, continuarás trabajando con la base de datos que contiene información de una empresa dedicada a la venta de productos online. En esta tarea, empezarás a trabajar con información relacionada con tarjetas de crédito.

NIVEL 1

Ejercicio 1

Tu tarea es diseñar y crear una tabla llamada "credit_card" que almacene detalles cruciales sobre las tarjetas de crédito. La nueva tabla debe ser capaz de identificar de forma única cada tarjeta y establecer una relación adecuada con las otras dos tablas ("transaction" y "company"). Después de crear la tabla será necesario que ingreses la información del documento denominado "datos_introducir_credit". Recuerda mostrar el diagrama y realizar una breve descripción del mismo.

Solución:

Paso 1: Creo la tabla credit_card

```
USE transactions;

CREATE TABLE credit_card(
    id VARCHAR(10) NOT NULL PRIMARY KEY,
    iban VARCHAR(255) NOT NULL,
    pan VARCHAR(255),
    pin INT,
    cvv SMALLINT,
    expiring_date VARCHAR(255)
);

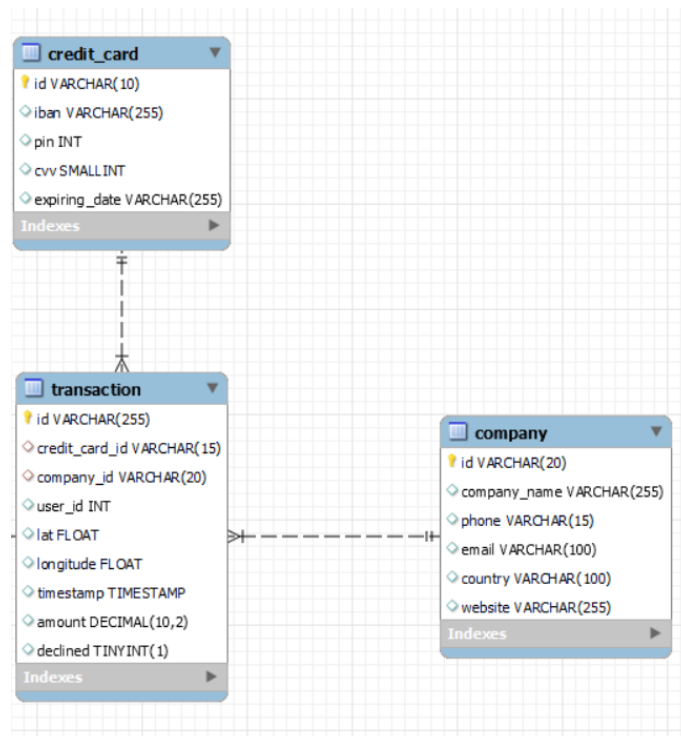
35  /*Creo una foreign key en la tabla de hechos (transaction)
36  con referencia a la primary key (id) de la nueva tabla (credit_card)*/
37
38  • ALTER TABLE transaction
39      ADD FOREIGN KEY fk_ccid(credit_card_id)
40      REFERENCES credit_card (id);
```

Paso 2: Cargo los dato, previamente he realizado la modificación de los permiso de MySQL para cargar datos de orígenes diferentes al establecido por defecto en el programa.

```
148 • LOAD DATA
149 INFILE 'D:/DIEGO/Desktop/BOOTCAMP/ESPECIALIDAD ANALISIS DE DATOS
150 /MySQL/Tarea S3/Base S3/datos_introducir_credit.sql'
151 INTO TABLE credit_card
152 FIELDS TERMINATED BY ","
153 ENCLOSED BY ""
154 LINES TERMINATED BY ";"
155 IGNORE 1 ROWS;
```

Descripción diagrama:

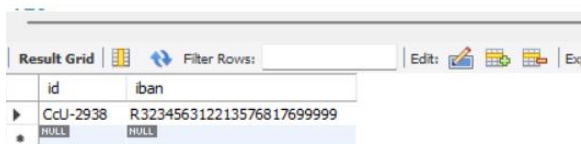
Se crea la tabla `credit_card` como dimensión de la tabla de hechos `transaction`, la relación que existe entre ellas es una relación de muchos a uno y el modelo corresponde a una estrella.



Ejercicio 2

El departamento de Recursos Humanos ha identificado un error en el número de cuenta del usuario con ID CcU-2938. La información que debe mostrarse para este registro es: R323456312213576817699999. Recuerda mostrar que el cambio se realizó.

```
165 • UPDATE credit_card
166     SET iban = R323456312213576817699999
167     WHERE id = 'CcU-2938';
168
169 • SELECT id, iban
170     FROM credit_card
171     WHERE id = 'CcU-2938';
```



id	iban
CcU-2938	R323456312213576817699999
NULL	NULL

Ejercicio 3

En la tabla "transaction" ingresa un nuevo usuario con la siguiente información:

Id	108B1D1D-5B23-A76C-55EF-C568E49A99DD
credit_card_id	CcU-9999
company_id	b-9999
user_id	9999
lato	829.999
longitud	-117.999
amunt	111.11
declined	0

```

185 • SHOW CREATE TABLE company;
186
187 • INSERT INTO company (id)
188   VALUE('b-9999');
189
190 • SHOW CREATE TABLE credit_card;
191
192 • ALTER TABLE credit_card
193   MODIFY iban VARCHAR(255) DEFAULT NULL;
194
195 • SHOW CREATE TABLE credit_card;
196
197 • INSERT INTO credit_card(id)
198   VALUE('CcU-9999');
199
200 • INSERT INTO transaction ( id, credit_card_id, company_id,
201                             user_id, lat, longitude,
202                             amount, declined)
203   VALUES ('10881D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999',
204           9999, 829.999, -117.999, 111.11, 0);
205
206 • SELECT *
207   FROM transaction
208   WHERE id = '10881D1D-5B23-A76C-55EF-C568E49A99DD';

```

Result Grid									
Filter Rows:									
Edit: Export/Import: Wrap Cell Content:									
id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined	
10881D1D-5B23-A76C-55EF-C568E49A99DD	CcU-9999	b-9999	9999	829.999	-117.999	NULL	111.11	0	
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	

Ejercicio 4

Desde recursos humanos te solicitan eliminar la columna "pan" de la tabla credit_card. Recuerda mostrar el cambio realizado.

```

218 • ALTER TABLE credit_card
219   DROP COLUMN pan;
220
221 • SELECT * FROM credit_card;

```

Result Grid					
Filter Rows:					
Edit: Export/Import:					
	id	iban	pin	cvv	expiring_date
▶	CcU-2938	R3234563122135768176999999	3257	984	10/30/22
	CcU-2945	DO26854763748537475216568689	9080	887	08/24/23
	CcU-2952	BG45IVQL52710525608255	4598	438	06/29/21
	CcU-2959	CR7242477244335841535	3583	667	02/24/23
	CcU-2966	BG72LKTQ15627628377363	4900	130	10/29/24

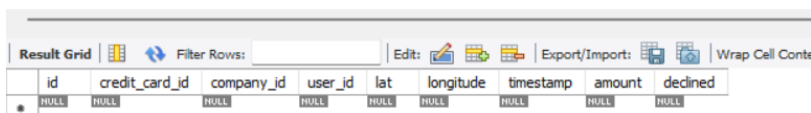
credit_card 8 x

NIVEL 2

Ejercicio 1

Elimina de la tabla transacción el registro con ID 02C6201E-D90A-1859-B4EE-88D2986D3B02 de la base de datos.

```
232 • DELETE FROM transaction
233 WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02';
234
235 • SELECT *
236 FROM transaction
237 WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02';
238
```



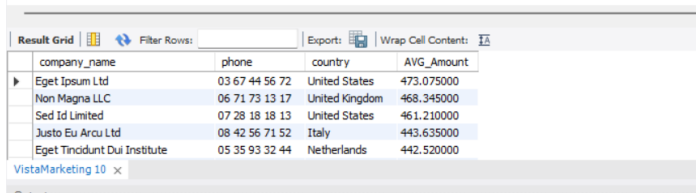
The screenshot shows a database interface with a toolbar and a result grid. The toolbar includes options for 'Result Grid', 'Filter Rows', 'Edit', 'Export/Import', and 'Wrap Cell Content'. The result grid has columns: id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amount, and declined. The first row contains null values for all columns.

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Ejercicio 2

La sección de marketing desea tener acceso a información específica para realizar análisis y estrategias efectivas. Se ha solicitado crear una vista que proporcione detalles clave sobre las compañías y sus transacciones. Será necesaria que crees una vista llamada VistaMarketing que contenga la siguiente información: Nombre de la compañía. Teléfono de contacto. País de residencia. Media de compra realizado por cada compañía. Presenta la vista creada, ordenando los datos de mayor a menor promedio de compra.

```
273 • CREATE VIEW VistaMarketing AS
274 SELECT company_name, phone, country, AVG(amount) AVG_Amount
275 FROM company c
276 INNER JOIN transaction t
277 ON c.id = t.company_id
278 GROUP BY company_name, phone, country
279 ORDER BY AVG_Amount DESC;
280
281 • SELECT *
282 FROM VistaMarketing;
283
```



The screenshot shows a database interface with a toolbar and a result grid. The toolbar includes options for 'Result Grid', 'Filter Rows', 'Export', and 'Wrap Cell Content'. The result grid has columns: company_name, phone, country, and AVG_Amount. The data is ordered by AVG_Amount in descending order.

company_name	phone	country	AVG_Amount
Eget Ipsum Ltd	03 67 44 56 72	United States	473.075000
Non Magna LLC	06 71 73 13 17	United Kingdom	468.345000
Sed Id Limited	07 28 18 18 13	United States	461.210000
Justo Eu Arcu Ltd	08 42 56 71 52	Italy	443.635000
Eget Tincidunt Dui Institute	05 35 93 32 44	Netherlands	442.520000

Ejercicio 3

Filtra la vista VistaMarketing para mostrar sólo las compañías que tienen su país de residencia en "Germany"

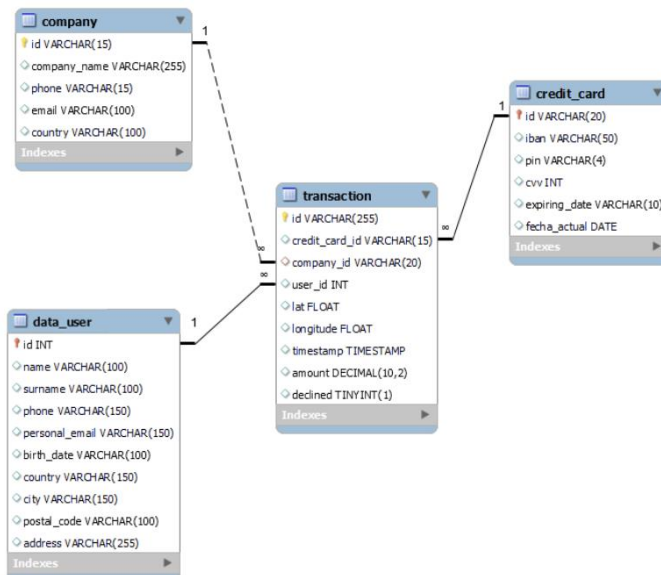
```
290 • SELECT *
291 FROM VistaMarketing
292 WHERE country = 'Germany';
293
```

	company_name	phone	country	AVG_Amount
▶	Aliquam PC	01 45 73 52 16	Germany	385.265000
	Ac Industries	09 34 65 40 60	Germany	289.645000
	Rutrum Non Inc.	02 66 31 61 09	Germany	266.900000
	Nunc Interdum Incorporated	05 18 15 48 13	Germany	244.025238
	Augue Foundation	06 88 43 15 63	Germany	240.800000
	Ac Fermentum Incorporated	06 85 56 52 33	Germany	206.465000
	Auctor Mauris Corp.	05 62 87 14 41	Germany	184.310000
	Convallis In Incorporated	06 66 57 29 50	Germany	156.730000

NIVEL 3

Ejercicio 1

La próxima semana tendrás una nueva reunión con los gerentes de marketing. Un compañero de tu equipo realizó modificaciones en la base de datos, pero no recuerda cómo las realizó. Te pide que le ayudes a dejar los comandos ejecutados para obtener el siguiente diagrama:

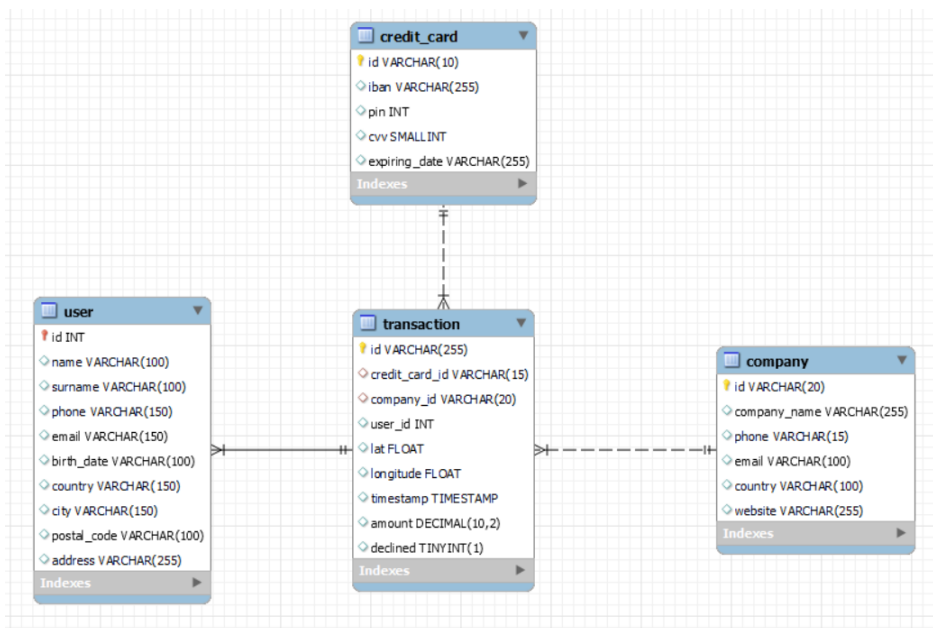


Paso 1: Se crea la tabla users con su PK y FK a la tabla de hechos.

```
1      -- Creamos la tabla user
2
3      • CREATE INDEX idx_user_id ON transaction(user_id);
4
5      • CREATE TABLE IF NOT EXISTS user (
6          id INT PRIMARY KEY,
7          name VARCHAR(100),
8          surname VARCHAR(100),
9          phone VARCHAR(150),
10         email VARCHAR(150),
11         birth_date VARCHAR(100),
12         country VARCHAR(150),
13         city VARCHAR(150),
14         postal_code VARCHAR(100),
15         address VARCHAR(255),
16         FOREIGN KEY(id) REFERENCES transaction(user_id)
17     );
```

Paso 2: Se insertan los datos partir del archivo csv: datos_introducir_user.sql

Diagrama final:



Ejercicio 2

La empresa también te solicita crear una vista llamada "InformeTecnico" que contenga la siguiente información:

ID de la transacción

Nombre del usuario/a

Apellido del usuario/a

IBAN de la tarjeta de crédito usada.

Nombre de la compañía de la transacción realizada.

Asegúrate de incluir información relevante de ambas tablas y utiliza alias para cambiar de nombre columnas según sea necesario.

Muestra los resultados de la vista, ordena los resultados de forma descendente en función de la variable ID de transacción.

```
315 • CREATE VIEW InformeTecnico AS
316 SELECT transaction.id AS Id_Transaction,
317        user.name AS Name_User,
318        user.country AS User_Country,
319        company.company_name AS Company,
320        company.country AS Country_Company,
321        company.website AS Web,
322        credit_card.iban AS iban,
323        transaction.amount AS Amount,
324        transaction.declined AS Declined
325
326 FROM transaction
327 JOIN user
328 ON transaction.user_id = user.id
329 JOIN credit_card
330 ON transaction.credit_card_id = credit_card.id
331 JOIN company
332 ON transaction.company_id = company.id
333 ORDER BY Id_Transaction DESC;
334
335 • SELECT * FROM InformeTecnico;
```

Result Grid									
Filter Rows:									
Export: Wrap Cell Contents:									
Id_Transaction	Name_User	User_Country	Company	Country_Company	Web	iban	Amount	Declined	
FE96CE47-8D59-381C-4E18-E3CA3D44E8FF	Kenyon	Canada	Magna A Neque Industries	Australia	https://whatsapp.com/group/9	DO26854763748537475216568689	480.13	1	
FE809ED4-2DB6-55AC-C915-929516E4646B	Molly	United Kingdom	Nunc Interdum Incorporated	Germany	https://wikipedia.org/en-us	SE2813123487163628531121	219.83	0	
FD9CBCCD-8E1E-8DA1-4606-7E3A6F3A5A65	Linus	Canada	Nunc Interdum Incorporated	Germany	https://wikipedia.org/en-us	KW9485332754781757886242955643	42.32	0	
FD89D51B-AE8D-77DC-E450-B8083FBD3187	Hilda	Canada	Malesuada PC	Ireland	https://yahoo.com/fr	LT053237077744561475	200.72	0	
FD2E8957-414B-BEEC-E9AD-59AA7A8A6290	Hedwig	Canada	Neque Tellus Imperdiet Corp.	Ireland	https://whatsapp.com/sub/cars	GE84848451582810541526	78.29	0	
FCE2AB9A-271D-2BDC-9E49-8DD92A373391	Hakeem	United Kingdom	Nunc Interdum Incorporated	Germany	https://wikipedia.org/en-us	MD1234119525145401270486	335.56	0	