

Descripció sprint

En aquest sprint, es simula una situació empresarial en la qual has de realitzar diverses manipulacions en les taules de la base de dades. Al seu torn, hauràs de treballar amb índexs i vistes. En aquesta activitat, continuaràs treballant amb la base de dades que conté informació d'una empresa dedicada a la venda de productes en línia. En aquesta tasca, començaràs a treballar amb informació relacionada amb targetes de crèdit.

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 ingessis la informació del document denominat "dades_introduir_credit". Recorda mostrar el diagrama i realitzar una breu descripció d'aquest.

```

1
2 • ○ CREATE TABLE IF NOT EXISTS credit_card (
3     id VARCHAR(100) PRIMARY KEY,
4     iban VARCHAR(50) NOT NULL,
5     pan VARCHAR(100),
6     pin INT ,
7     cvv SMALLINT ,
8     expiring_date VARCHAR(10)
9 );
10

```

4

Para la realización de este ejercicio, se ha procedido a la creación de una tabla mediante el comando **CREATE TABLE**.

Se ha decidido acotar los valores VARCHAR con el fin de ahorrar memoria, estableciendo un rango de 50-100 para los primeros valores. Para la fecha, como la longitud es invariable, se ha establecido de 10 (XX/XX/XXXX).

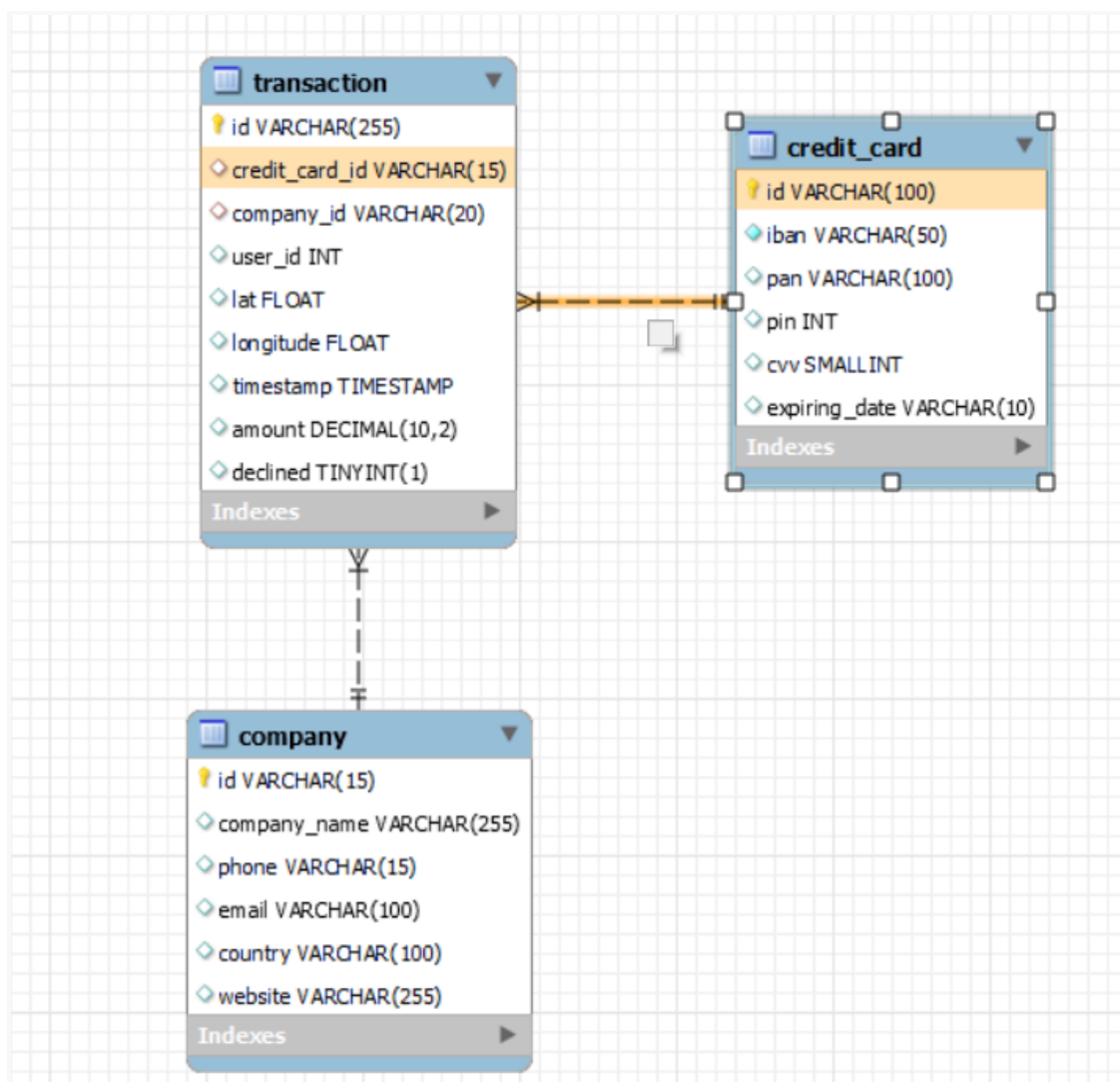
“CVV” al ser un número corto, se ha decidido establecerlo como smallint con el fin de ahorrar memoria.

```

1
2 • ALTER TABLE transaction
3   ADD CONSTRAINT fk_credit_card
4   FOREIGN KEY (credit_card_id)
5   REFERENCES credit_card(id);
6

```

Una vez creada la tabla, se ha establecido la relación con la tabla “*transaction*” para relacionarlas mediante el *id* de *credit_card*. Se establece la foreign key para poder obtener la cardinalidad a la hora de crear el diagrama de relaciones mediante el reverse engineer.



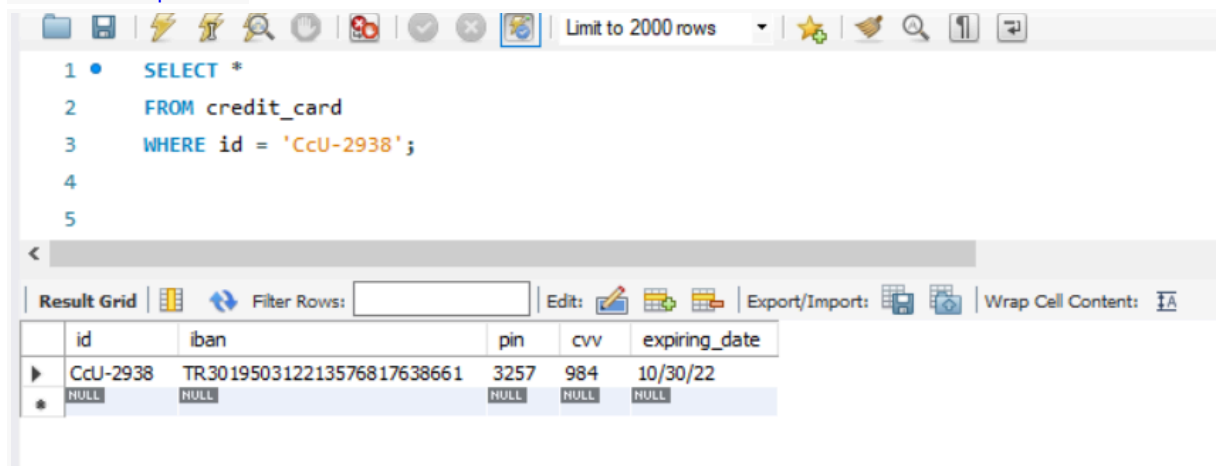
Una vez establecida la relación, el diagrama resultante se muestra arriba. Se pueden observar las cardinalidades 1-N.

- Exercici 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.

En primer lugar, se hace un select filtrando el id mencionado. Se identifica el Id y se procede a cambiar la información de este mediante el comando update. Por último, se observa que se vuelve a hacer un select para observar los cambios realizados.

Resultado previo:



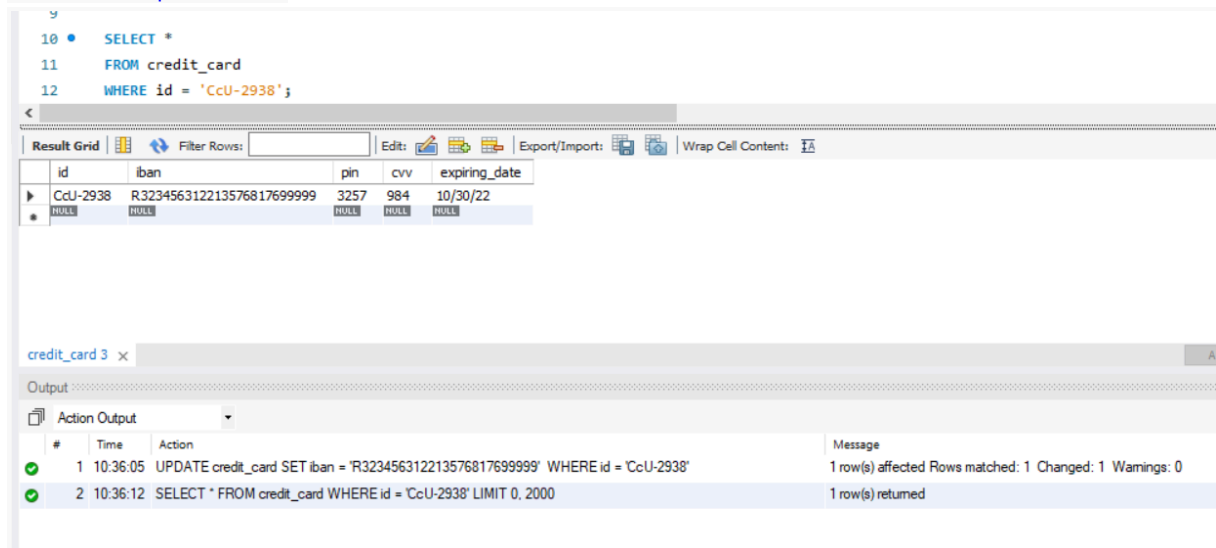
The screenshot shows a database query tool interface. At the top, there's a toolbar with various icons. Below it, a SQL query is entered in a text area:

```
1 SELECT *
2 FROM credit_card
3 WHERE id = 'CcU-2938';
4
5
```

Below the query editor, there's a "Result Grid" section. It includes a "Filter Rows:" input field, an "Edit:" button, an "Export/Import:" button, and a "Wrap Cell Content:" checkbox. The results are displayed in a table:

	id	iban	pin	cvv	expiring_date
▶	CcU-2938	TR301950312213576817638661	3257	984	10/30/22
*	NULL	NULL	NULL	NULL	NULL

Resultado posterior:



The screenshot shows the same database query tool interface. The SQL query is now:

```
10 SELECT *
11 FROM credit_card
12 WHERE id = 'CcU-2938';
```

The "Result Grid" section shows the updated results:

	id	iban	pin	cvv	expiring_date
▶	CcU-2938	R323456312213576817699999	3257	984	10/30/22
*	NULL	NULL	NULL	NULL	NULL

Below the result grid, there's an "Output" section. It shows a list of actions performed:

#	Time	Action	Message
1	10:36:05	UPDATE credit_card SET iban = 'R323456312213576817699999' WHERE id = 'CcU-2938'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
2	10:36:12	SELECT * FROM credit_card WHERE id = 'CcU-2938' LIMIT 0, 2000	1 row(s) returned

- Exercici 3

En la taula "transaction" ingressa un nou usuari 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

Para este ejercicio, no se pueden introducir nuevos datos en la tabla *transaction*, ya que el programa te devuelve un error de foreign key. Esto es debido a que el programa no encuentra la FK con la otra tabla (*credit_card* ni *company*). Por este motivo, se procede a insertar un nuevo registro en la tabla *credit_card* y en la tabla *company* mediante el comando INSERT INTO. AL solo tener información del ID, las demás columnas se dejan en NULL o en 0 (en el caso del iban ya que no puede ser null debido a su naturaleza):

```
INSERT INTO credit_card
VALUES ('CcU-9999', 0, null, null, null, null);
```

```
INSERT INTO company
VALUES ('b-9999', null, null, null, null, null);
```

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

Una vez introducido el nuevo registro en la tabla *credit_card*, ya se pueden introducir la nueva información en la tabla de transacción mediante el comando INSERT INTO. En la consigna a seguir, no se da información sobre el timestamp, por lo que procede a introducirlo como null.

Una vez introducido los nuevos valores, se procede al SELECT, y se puede observar los cambios realizados en las tablas *credit_card* y *transaction*:

```

24 • select *
25   from credit_Card;
26

```

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

	id	iban	pin	cvv	expiring_date
▶	CcU-9999	0	NULL	NULL	NULL
	CcU-4856	TR373872558313545667124286	9086	974	05/19/23
	CcU-4849	SE2813123487163628531121	9992	779	03/21/25
	CcU-4842	SA2156708581957118818229	4655	750	11/11/24
	CcU-4835	PT34592171131763200132583	1149	680	01/08/24
	CcU-4828	BG11ILMJ30149367569464	2789	942	09/04/22
	CcU-4821	LT253147505686466784	9000	867	07/15/20
	CcU-4814	MR4845282437847152280636374	8790	124	12/19/20
	CcU-4807	LB19298318715580851625676971	8596	626	04/07/22
	CcU-4800	SI97824334522161436	3745	886	05/23/20

credit_Card 14 x

Output

Action Output

#	Time	Action	Message
1	11:17:22	select * from credit_Card LIMIT 0, 2000	276 row(s) returned

```

24 • select *
25   from transaction;
26

```

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

	id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
▶	108B1D1D-5B23-A76C-55EF-C568E49A99DD	CcU-9999	b-9999	9999	829.999	-117.999	NULL	111.11	0
	9A58B3E0-1E27-94E2-6487-9A63F07BC833	CcU-4856	b-2278	196	20.2527	-82.4442	2021-08-11 02:02:29	380.79	0
	14CAE5B5-8FB1-3E4A-4C85-0EA4167534F4	CcU-4849	b-2302	189	-53.6202	93.0533	2021-12-31 00:29:42	388.04	0
	158A3ACB-541C-DBCC-658D-6373CC67BF1C	CcU-4849	b-2302	183	42.5424	-170.347	2022-03-08 05:02:19	240.29	0
	1822A630-4B6E-DABC-2576-6EC107FA15C2	CcU-4849	b-2278	194	-32.3809	-0.822218	2021-09-16 23:56:49	140.31	0
	18C4E2DD-1E4C-F35E-2198-C660BD81DC25	CcU-4849	b-2302	177	68.0133	91.4839	2021-09-24 18:55:25	237.04	0
	241827A6-E87C-6DA3-5AC9-BBCFD42B7D2A	CcU-4849	b-2302	190	-12.1172	-175.322	2021-08-04 04:44:35	487.64	0
	35F284A2-5028-6DB8-A631-78B7981716C2	CcU-4849	b-2278	192	-41.6613	-178.036	2022-01-10 19:27:25	167.36	0
	3A3E00C1-C191-02EF-AD8E-42C9DA644E4A	CcU-4849	b-2302	175	0.340013	174.033	2021-11-10 19:14:50	451.66	0
	3B37BE91-420F-4361-238A-3C5A42B998F7	CcU-4849	b-2278	191	7.17413	-0.998084	2022-03-02 02:21:42	154.94	0

transaction 15 x

Output

Action Output

#	Time	Action	Message
1	11:17:22	select * from credit_Card LIMIT 0, 2000	276 row(s) returned
2	11:18:17	select * from transaction LIMIT 0, 2000	587 row(s) returned

- Exercici 4

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

Para la eliminación de una columna de nuestra base de datos, se utiliza el comando ALTER y DROP:

```

26
27 • ALTER TABLE credit_Card
28 DROP COLUMN pan;
29

```

Se puede observar que ahora en la tabla credit_Card no aparece la columna pan:

	id	iban	pin	cvv	expiring_date
▶	CcU-2938	R323456312213576817699999	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

	id	iban	pan	pin	cvv	expiring_date
▶	CcU-2938	R323456312213576817699999	5424465566813633	3257	984	10/30/22
	CcU-2945	DO26854763748537475216568689	5142423821948828	9080	887	08/24/23
	CcU-2952	BG45IVQL52710525608255	4556 453 55 5287	4598	438	06/29/21
	CcU-2959	CR7242477244335841535	372461377349375	3583	667	02/24/23
	CcU-2966	BG72LKTQ15627628377363	448566 886747 7265	4900	130	10/29/24

Nivell 2

Exercici 1

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

En primer lugar, se procede a identificar el registro que se va a eliminar:

```

25 • select *
26 from transaction;
27

```

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
02C6201E-D90A-1859-B4EE-88D2986D3B02	CcU-2938	b-2362	92	81.9185	-12.5276	2021-08-28 23:42:24	466.92	0
0466A42E-47CF-8D24-FD01-C08689713128	CcU-4219	b-2302	170	-43.9695	-117.525	2021-07-26 07:29:18	49.53	0
063FBA79-99EC-66FB-29F7-25726D1764A5	CcU-2987	b-2250	275	-81.2227	-129.05	2022-01-06 21:25:27	92.61	0

Para su eliminación, se utiliza el comando delete , utilizando como filtro el comando where para especificar el id que se quiere eliminar:

```

28
29 • delete from transaction
30 where id = '02C6201E-D90A-1859-B4EE-88D2986D3B02';
31
32

```

Result Grid									
Filter Rows: <input type="text"/>									
Edit: Export/Import: Wrap Cell Content:									
id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined	
0466A42E-47CF-8D24-FD01-C0B689713128	CcU-4219	b-2302	170	-43.9695	-117.525	2021-07-26 07:29:18	49.53	0	
063FBA79-99EC-66FB-29F7-25726D1764A5	CcU-2987	b-2250	275	-81.2227	-129.05	2022-01-06 21:25:27	92.61	0	
0668296C-CDB9-A883-76BC-2E4C44F8C8AE	CcU-3743	b-2618	265	-34.3593	-100.556	2022-01-26 02:07:14	394.18	0	

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.

Para la creación de una vista con diferentes tablas, se procede a utilizar JOINS con estas mismas. Se utiliza el comando CREATE VIEW . Para seleccionar la mediana de las ventas, se procede a hacer un round a dos decimales, ya que la moneda euro / dólar consta únicamente de dos decimales.

Por motivos estéticos, se establece un alias a la mediana de las ventas como "avg". Se ordena por la mediana de ventas de mayor a menor.

Por otro lado, se utiliza la condición WHERE para seleccionar aquellas transacciones que han generado ingresos/ventas. Se entiende que toda transacción que genera ingreso conlleva tener el declined = 0 . (Es decir, la transacción NO ha sido cancelada)

```

41 • CREATE VIEW VistaMarketing AS
42 SELECT c.company_name, c.phone, c.country, round(avg(amount),2) AS avg
43 FROM company c
44 JOIN transaction AS t ON c.id = t.company_id
45 WHERE declined = 0
46 GROUP BY c.company_name , c.phone, c.country
47 ORDER BY round(avg(amount),2) DESC;

```

49 • SELECT *

50 FROM vistamarketing;

51

<

Result Grid

Filter Rows:

Export:

Wrap Cell Content:

	company_name	phone	country	avg
▶	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

vistamarketing 16 x

Output

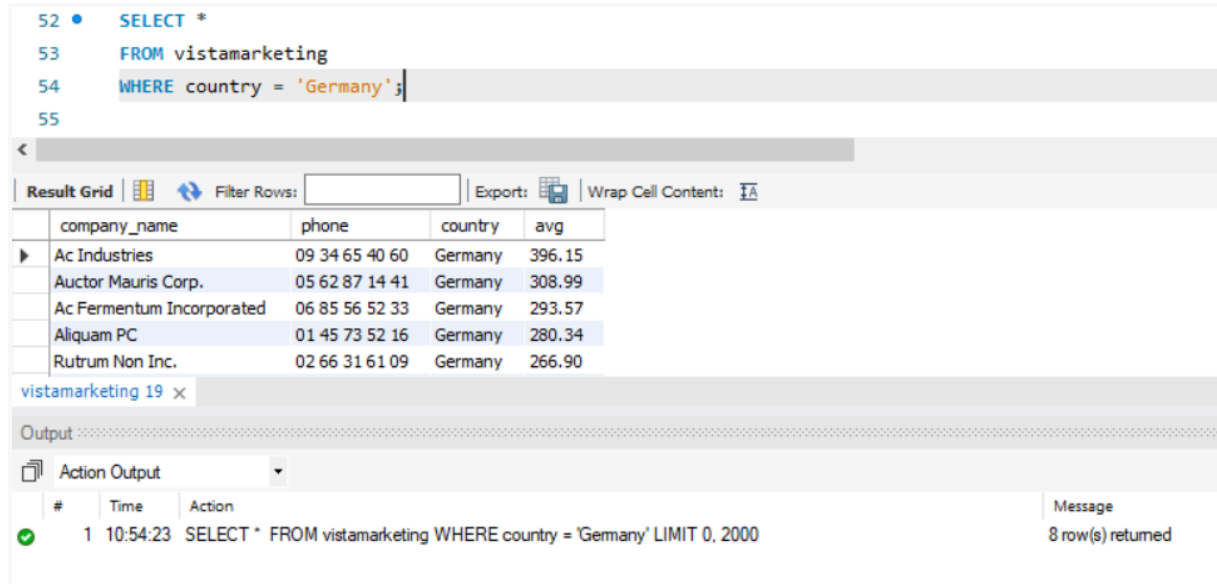
Action Output

#	Time	Action	Message
✓ 1	10:52:49	CREATE VIEW VistaMarketing AS SELECT c.company_name, c.phone, c.country, round(avg(amount),2) AS avg...	0 row(s) affected
✓ 2	10:52:54	SELECT * FROM vistamarketing LIMIT 0, 2000	101 row(s) returned

Exercici 3

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

Para hacer el filtro de un único país, se utiliza el comando de condición where:



```

52 • SELECT *
53 FROM vistamarketing
54 WHERE country = 'Germany';
55

```

company_name	phone	country	avg
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

vistamarketing 19 x

Output

Action Output

#	Time	Action	Message
1	10:54:23	SELECT * FROM vistamarketing WHERE country = 'Germany' LIMIT 0, 2000	8 row(s) returned

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.

Para este ejercicio, es imprescindible alterar la tabla transaction, y establecer correctamente la foreign key para establecer una buena relación con la tabla user. Si no se procede a este previo paso, la cardinalidad que se establece es incorrecta (es la invertida). Por ello, se utiliza el ALTER TABLE y se establece el constraint de FK. Así, a la hora de crear el diagrama mediante el reverse engineer, se puede observar que la cardinalidad es la correcta, de 1 a N.

```

48 • ALTER TABLE transaction
49 ADD CONSTRAINT fk_card
50 FOREIGN KEY (user_id)
51 REFERENCES user(id);
52

```

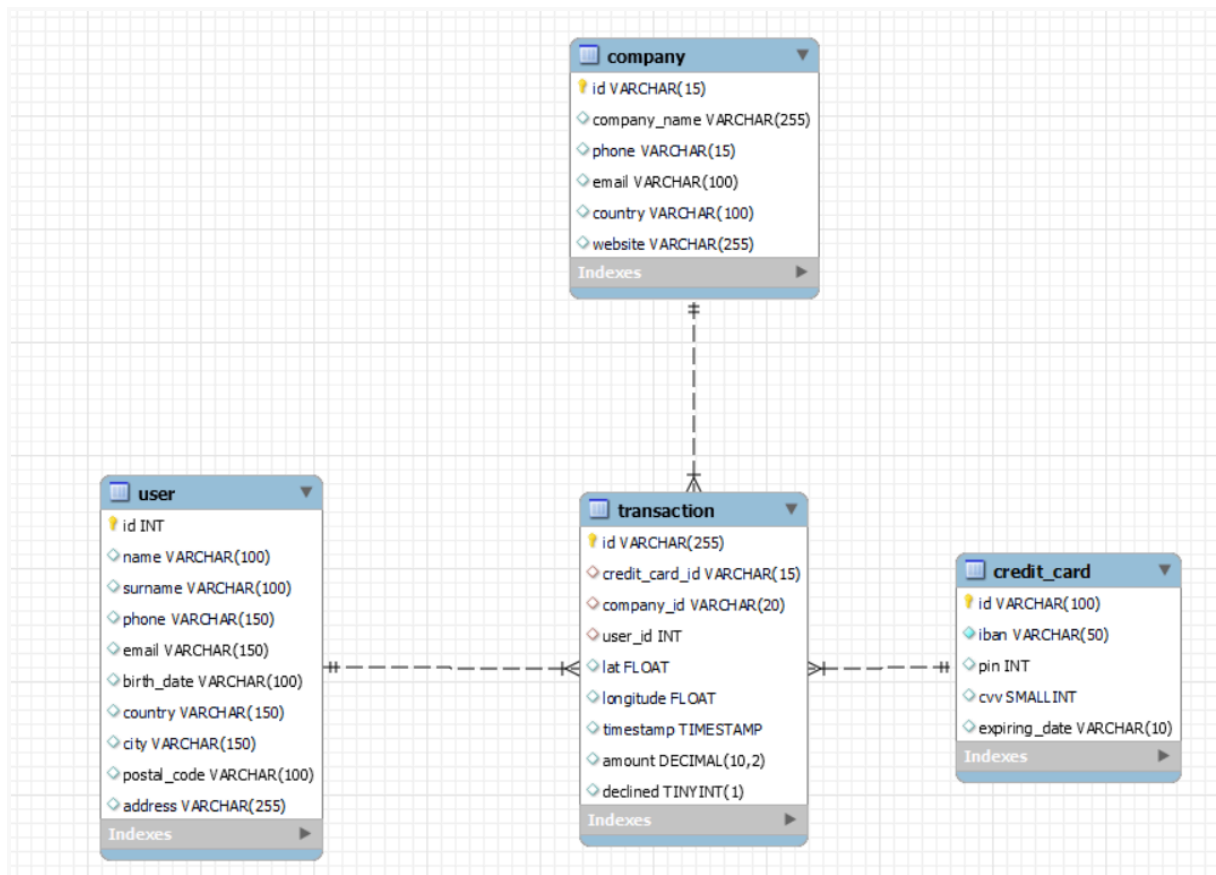
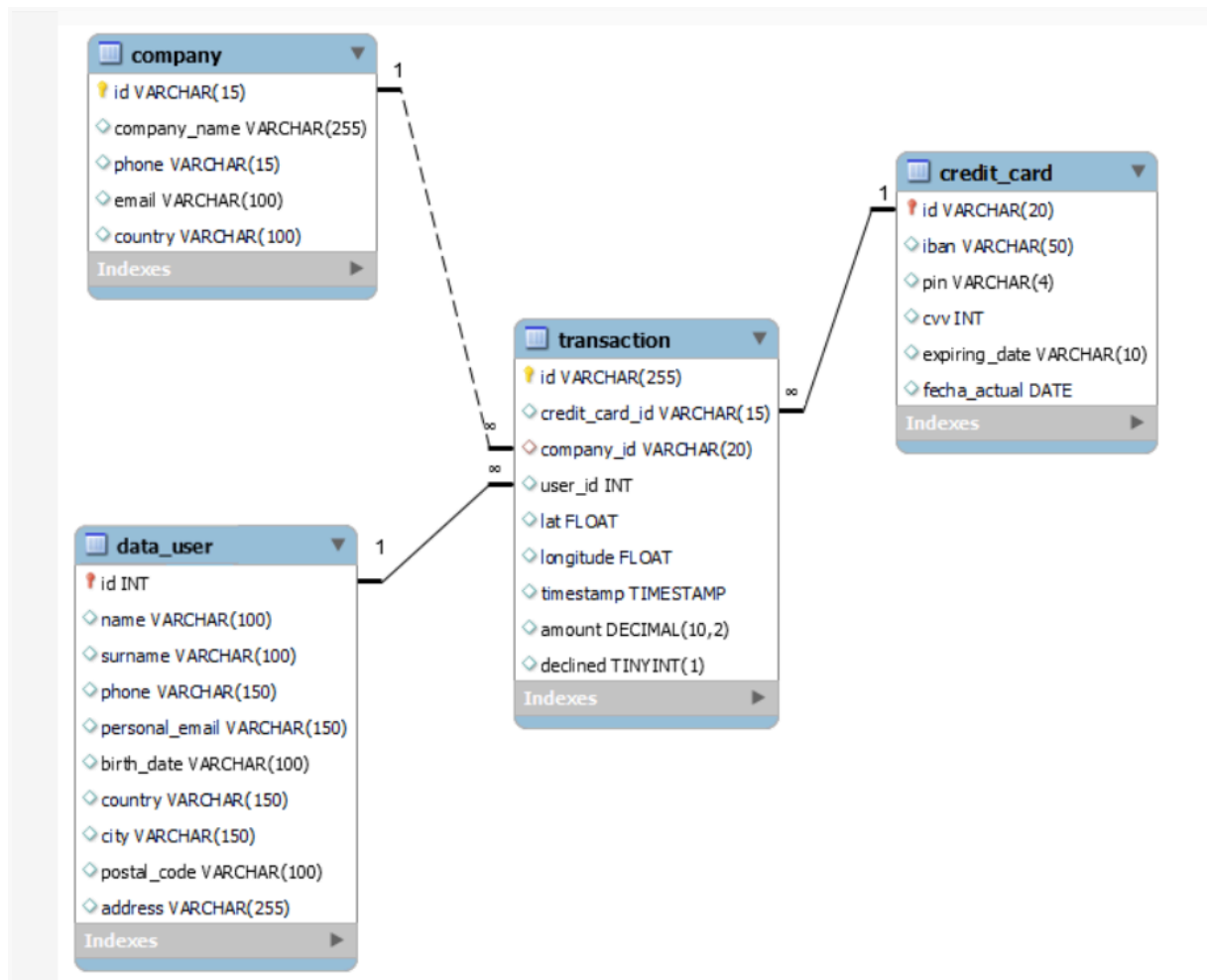


Diagrama objetivo:



Se pueden observar ciertas diferencias en el diagrama. Por ello, se procede a la realización de ciertos cambios mediante comandos para conseguir tener un diagrama homogéneo.

- Mediante este comando , lo que conseguimos es crear una relación fuerte entre las tablas:

```
alter table transaction drop primary key ,  
add primary key ( id, credit_card_id, user_id);
```

- Se renombra la tabla user a data_user para asemejarse al diagrama deseado:

```
rename table user to data_user;
```

- Se renombra la columna del email:

```
ALTER TABLE data_user rename column email to personal_email;
```

- Se borra la columna website ya que no esta presente en el diagrama deseado.

```
ALTER table company DROP column website;
```

- Se modifica la longitud de la columna id.

```
ALTER TABLE credit_card modify column id varchar(20);
```

- Se modifica la longitud de la columna pin.

```
ALTER TABLE credit_card modify column pin varchar(4);
```

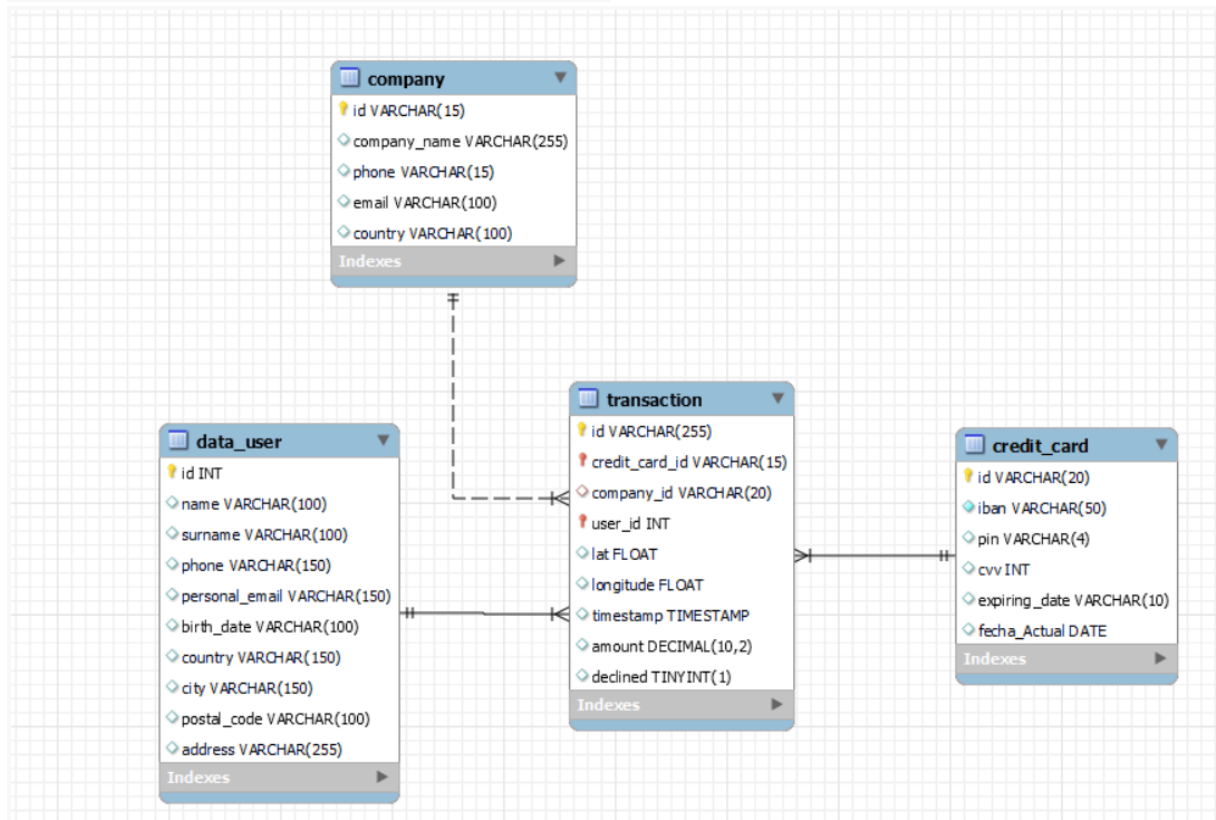
- Se modifica el tipo de dato de la columna cvv.

```
ALTER TABLE credit_card modify column cvv INT;
```

- Se agrega una nueva columna llamada fecha_actual.

```
ALTER TABLE credit_card  
ADD COLUMN fecha_Actual DATE ;
```

Diagrama resultante de las modificaciones:



Exercici 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.

Para la creación de la view, se procede a usar el comando CREATE VIEW , incorporando joins para conectar las respectivas tablas. No se utilizan alias ya que no se denota una necesidad estricta. La información mostrada es la relevante : id, iban, nombre y apellido y el nombre de la compañía:

```

CREATE VIEW InformeTecnico AS
SELECT t.id, c.iban , u.name, u.surname, co.company_name
FROM transaction t
JOIN user AS u ON u.id = t.user_id
JOIN credit_card AS c ON c.id = t.credit_card_id
join company AS co ON co.id = t.company_id
GROUP BY t.id, c.iban , u.name, u.surname, co.company_name
ORDER BY t.id DESC;

```

65 • SELECT *

66 FROM InformeTecnico;

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

	id	iban	name	surname	company_name
	FE96CE47-8D59-381C-4E18-E3CA3D44E8FF	DO26854763748537475216568689	Kenyon	Hartman	Magna A Neque Industries
	FE809ED4-2DB6-55AC-C915-929516E4646B	SE2813123487163628531121	Molly	Gilliam	Nunc Interdum Incorporated
	FD9CBCCD-8E1E-8DA1-4606-7E3A6F3A5A65	KW9485332754781757886242955643	Linus	Willis	Nunc Interdum Incorporated
	FD89D51B-AE8D-77DC-E450-88083FBD3187	LT053237077744561475	Hilda	Levy	Malesuada PC
	FD2E8957-414B-BEEC-E9AD-59AA7A8A6290	GE84848451582810541526	Hedwig	Gilbert	Neque Tellus Imperdiet Corp.
	FCE2AB9A-271D-2BDC-9E49-8DD92A373391	MD1234119525145401270486	Hakeem	Alford	Nunc Interdum Incorporated
	FBD7E0D6-BA6B-F5BC-OCA9-EA4B8760100C	MU4132333444534342541344788855	Hedwig	Gilbert	Mauris Id Inc.
	FAC76A80-8448-69AA-E892-426C2F12621C	MT05JWCF58868200575771634583813	Slade	Poole	Arcu LLP
	FAAD3FFC-1A17-E141-43D3-359A5BA7CB3B	GE90157928843338134463	Hedwig	Gilbert	Lorem Eu Incorporated
	FA053936-75D8-85FA-490D-9B624E1B920A	GT02497653655330848247645975	Hedwig	Gilbert	Non Justo Corp.

InformeTecnico 9 x

Output

Action Output

#	Time	Action	Message
1	10:38:46	SELECT * FROM credit_card LIMIT 0, 2000	276 row(s) returned
2	10:39:13	SELECT * FROM InformeTecnico LIMIT 0, 2000	586 row(s) returned