Level 1

Exercise 1

Your task is to design and create a table called "credit_card" that stores crucial details about credit cards. The new table must be able to uniquely identify each card and establish an appropriate relationship with the other two tables ("transaction" and "company"). After creating the table, you will need to enter the information in the document called "credit_input_data". Remember to show the diagram and provide a brief description of it.

```
CREATE TABLE IF NOT EXISTS credit_card

(

id varchar(15) primary key,

iban varchar(255),

pan varchar(255),

pin int,

cvv int,

expiring_date char(15)

);
```

Create the table credit_card

```sql
-- Level 1
-- Exercise 1
-- Your task is to design and create a table called "credit_card" that stores crucial details about credit cards.
-- The new table must be able to uniquely identify each card and establish an appropriate relationship with the other two
--  tables ("transaction" and "company"). After creating the table, you will need to enter the information in the document called "credit_input_data".
-- Remember to show the diagram and provide a brief description of it.

CREATE TABLE IF NOT EXISTS credit_card
(
id varchar(15) primary key,
iban varchar(255),
pan varchar(255),
pin int,
cvv int,
expiring_date char(15)
);
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
210332	15:49:16	select * from company	100 row(s) returned	0.000 sec / 0.000 sec
210333	15:52:23	CREATE TABLE IF NOT EXISTS credit_card (id varchar(15) primary key, iban varchar(255), pan v...	0 row(s) affected	0.047 sec

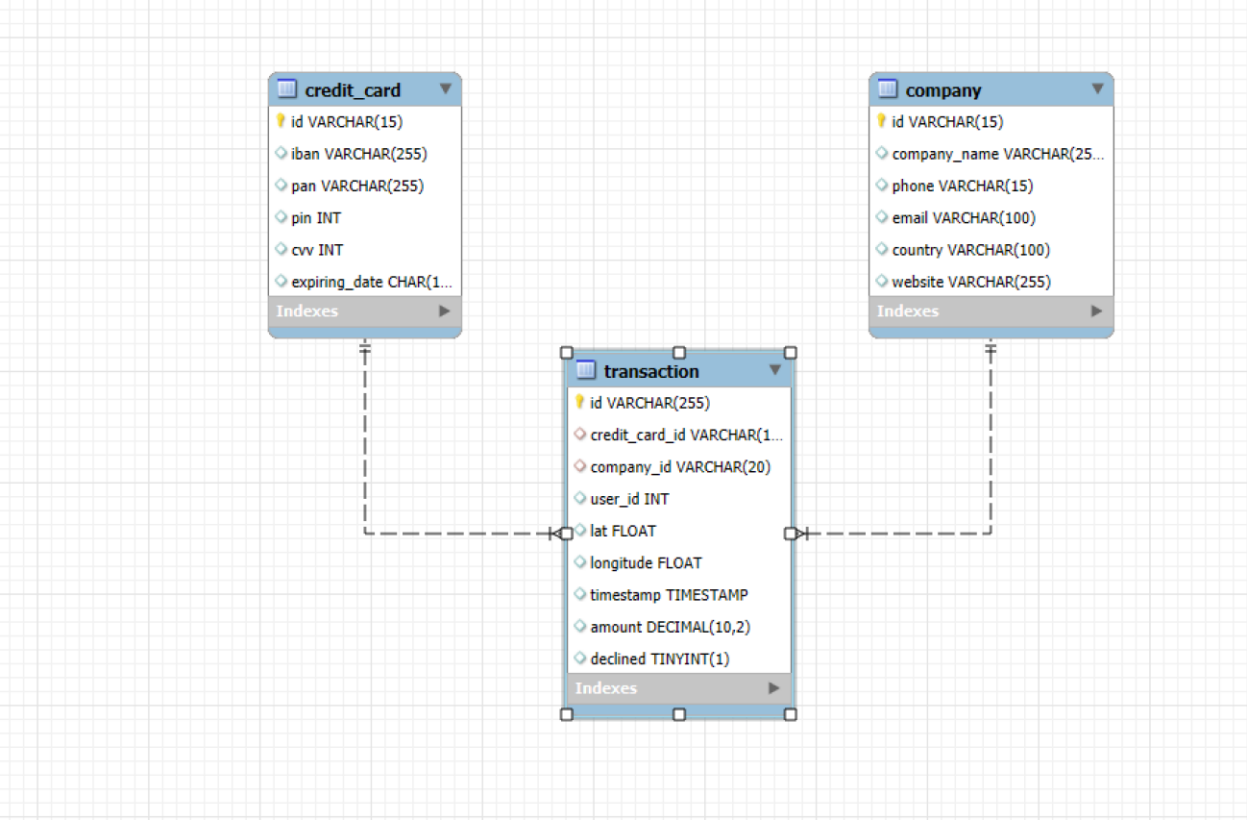
Add the foreign key to the table transaction it refers the table credit_card

```sql
-- Level 1
-- Exercise 1
-- Your task is to design and create a table called "credit_card" that stores crucial details about credit cards.
-- The new table must be able to uniquely identify each card and establish an appropriate relationship with the other two
--  tables ("transaction" and "company"). After creating the table, you will need to enter the information in the document called "credit_input_data".
-- Remember to show the diagram and provide a brief description of it.

CREATE TABLE IF NOT EXISTS credit_card
(
id varchar(15) primary key,
iban varchar(255),
pan varchar(255),
pin int,
cvv int,
expiring_date char(15)
);

select * from credit_card;

alter table transaction
add constraint fk_credit_card_id
FOREIGN KEY (credit_card_id)
REFERENCES credit_card(id);
```

Output

Action Output

#	Time	Action	Message	Duration / Fetch
215337	16:00:19	select * from credit_card	5000 row(s) returned	0.000 sec / 0.015
215338	16:00:40	alter table transaction add constraint fk_credit_card_id FOREIGN KEY (credit_card_id) REFERENC...	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings: 0	3.266 sec


credit_card
id VARCHAR(15)
iban VARCHAR(255)
pan VARCHAR(255)
pin INT
cvv INT
expiring_date CHAR(1...
Indexes
company
id VARCHAR(15)
company_name VARCHAR(25...
phone VARCHAR(15)
email VARCHAR(100)
country VARCHAR(100)
website VARCHAR(255)
Indexes
transaction
id VARCHAR(255)
credit_card_id VARCHAR(1...
company_id VARCHAR(20)
user_id INT
lat FLOAT
longitude FLOAT
timestamp TIMESTAMP
amount DECIMAL(10,2)
declined TINYINT(1)
Indexes

Credit_card table:

This table stores the credit card information.The Id field is the primary key. transaction.credit_card_id is a foreign key.It references credit_card.id.This table has no foreign key or reference to the company table.

column	type	description
Id	Varchar(15)	**Primary Key** — uniquely identifies each credit card
iban	Varchar(255)	International Bank Account Number
pan	Varchar(255)	Primary Account Number
pin	int	Personal Identification Number
cvv	int	Card security code
Expiring_date	char	Expiration date

Exercise 2

The Human Resources department has identified an error in the account number associated with the credit card with ID CcU-2938. The information that should be displayed for this record is: TR323456312213576817699999. Remember to show that the change was made.

```
select *
from credit_card;

select * from
credit_card
where id = 'CcU-2938';

update credit_card
set iban = 'TR323456312213576817699999'
where  id = 'CcU-2938';
select * from
credit_card
where id = 'CcU-2938';
```

```
27
28      -- Exercise 2
29      -- The Human Resources department has identified an error in the account number associated with the credit card with ID CcU-2938.
30      -- he information that should be displayed for this record is: TR323456312213576817699999. Remember to show that the change was made.
31 •    select *
32      from credit_card;
33
34 •    select * from
35      credit_card
36      where id = 'CcU-2938';
37
38 •    update credit_card
39      set iban = 'TR323456312213576817699999'
40      where  id = 'CcU-2938';
41 •    select * from
42      credit_card
```

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

id	iban	pan	pin	cvv	expiring_date
CcU-2938	TR301950312213576817638661	5424465566813633	3257	984	10/30/22
NULL	NULL	NULL	NULL	NULL	NULL

```
26
27
28      -- Exercise 2
29      -- The Human Resources department has identified an error in the account number associated with the credit card with ID CcU-2938.
30      -- he information that should be displayed for this record is: TR323456312213576817699999. Remember to show that the change was made.
31 •    select *
32      from credit_card;
33
34 •    select * from
35      credit_card
36      where id = 'CcU-2938';
37
38 •    update credit_card
39      set iban = 'TR323456312213576817699999'
40      where  id = 'CcU-2938';
41 •    select * from
42      credit_card
43      where id = 'CcU-2938';
```

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

id	iban	pan	pin	cvv	expiring_date
CcU-2938	TR323456312213576817699999	5424465566813633	3257	984	10/30/22
NULL	NULL	NULL	NULL	NULL	NULL

Exercise 3

In the "transaction" table, a new user is entered with the following information:

Id	108B1D1D-5B23-A76C-55EF-C568E49A99DD
credit_card_id	CcU-9999
company_id	b-9999
user_id	9999
Latin	829,999
length	-117,999
amount	111.11
declined	0

```
insert into
company(id)
values('b-9999');


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'
);


insert into
 credit_card(id)
 values('CcU-9999');
```

select * from transaction

where id ='108B1D1D-5B23-A76C-55EF-C568E49A99DD';

```
-- user_id  9999
-- Latin    829,999
-- length   -117,999
-- amount   111.11
-- declined 0

insert into
company(id)
values('b-9999');

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'
);

insert into
  credit_card(id)
  values('CcU-9999');
  select * from transaction
  where id ='108B1D1D-5B23-A76C-55EF-C568E49A99DD';
```

Action Output

#	Time	Action	Message	Duration / Fetch
215345	16:13:49	insert into credit_card(id) values('CcU-9999')	1 row(s) affected	0.016 sec
215346	16:13:55	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amou...	1 row(s) affected	0.015 sec

```
-- length   -117,999
-- amount   111.11
-- declined 0

insert into
company(id)
values('b-9999');

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'
);

insert into
  credit_card(id)
  values('CcU-9999');
  select * from transaction
  where id ='108B1D1D-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
108B1D1D-5B23-A76C-55EF-C568E49A99DD	CcU-9999	b-9999	9999	829.999	-117.999	NULL	111.11	0

Action Output

#	Time	Action	Message	Duration / Fetch
215346	16:13:55	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp, amou...	1 row(s) affected	0.015 sec
215347	16:15:38	select * from transaction where id ='108B1D1D-5B23-A76C-55EF-C568E49A99DD'	1 row(s) returned	0.000 sec / 0.000 sec

Exercise 4

Human resources has asked you to delete the "pan" column from the credit_card table. Remember to show the change made.

alter table credit_card

drop column pan;

select * from credit_card;

```
77
78
79
80
81
82
83
84
85      -- Exercise 4
86      -- Human resources has asked you to delete the "pan" column from the credit_card table. Remember to show the change made.
87
88  •   alter table credit_card
89      drop column pan;
90
91
92  •   select * from credit_card;
93
94
```

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

id	iban	pin	cvv	expiring_date
CcS-4857	XX4857591835292505850771	1819	467	09/27/25
CcS-4858	XX8581768137002436094025	3964	817	12/28/28
CcS-4859	XX7826930491423553609370	4983	277	11/26/26
CcS-4860	XX5559590368835304645299	6876	661	07/27/27
CcS-4861	XX2035182877195191627307	5710	398	04/25/26

Level 2

Exercise 1

Delete the record with ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD from the transaction table of the database.

select *

from transaction;

delete from transaction

where id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD ';

select *

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

```
-- Level 2
-- Exercise 1
-- Remove from the transaction table the record with ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD from the database.
select *
from transaction;

delete from transaction
where id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD ';

select *
from transaction where id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD ';


-- Exercise 2
-- The marketing department wants to have access to specific information to perform effective analysis and strategies.
--  It has been requested to create a view that provides key details about the companies and their transactions.
--  You will need to create a view called VistaMarketing that contains the following information:
-- Company name. Contact phone number. Country of residence. Average purchase made by each company.
-- Present the view created, ordering the data from highest to lowest average purchase.
create view VistaMarketing as
select
```

Action Output

	Time	Action	Message	Duration / Fetch
15350	16:21:30	select * from transaction	100001 row(s) returned	0.016 sec / 0.297 sec
15351	16:21:39	delete from transaction where id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD '	0 row(s) affected	0.000 sec

```sql
-- Level 2
-- Exercise 1
-- Remove from the transaction table the record with ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD from the database.
select *
from transaction;

delete from transaction
where id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD ';

select *
from transaction where id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD ';
```

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

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

transaction 12 | Apply

Output

Action Output

#	Time	Action	Message	Duration / Fetch
215351	16:21:39	delete from transaction where id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD '	0 row(s) affected	0.000 sec
215352	16:22:34	select * from transaction where id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD '	0 row(s) returned	0.000 sec / 0.000 se

Exercise 2

The marketing department wants to have access to specific information to perform effective analysis and strategies. It has been requested to create a view that provides key details about the companies and their transactions. You will need to create a view called VistaMarketing that contains the following information: Company name. Contact phone number. Country of residence. Average purchase made by each company. Present the view created, ordering the data from highest to lowest average purchase.

```
create view VistaMarketing as
select
com.company_name,com.phone,com.country,
round(avg(tran.amount),2) as Average_sales
from company as com
inner join transaction as tran on
com.id= tran.company_id
where tran.declined =0
group by com.id;
select * from VistaMarketing
order by Average_sales DESC;
```

```
115
116     -- Exercise 2
117     -- The marketing department wants to have access to specific information to perform effective analysis and strategies.
118     --  It has been requested to create a view that provides key details about the companies and their transactions.
119     --  You will need to create a view called VistaMarketing that contains the following information:
120     -- Company name. Contact phone number. Country of residence. Average purchase made by each company.
121     -- Present the view created, ordering the data from highest to lowest average purchase.
122 •   create view VistaMarketing as
123     select
124     com.company_name,com.phone,com.country,
125     round(avg(tran.amount),2) as Average_sales
126     from company as com
127     inner join transaction as tran on
128     com.id= tran.company_id
129     where tran.declined =0
130     group by com.id;
131 •   select * from VistaMarketing
132     order by Average_sales DESC;
```

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

company_name	phone	country	Average_sales
Ac Fermentum Incorporated	06 85 56 52 33	Germany	284.91
Pretium Neque Corp.	07 77 48 55 28	Australia	275.58
Urna Convallis Associates	06 01 24 77 04	United States	273.57
At Associates	09 56 61 10 65	New Zealand	272.74
Metus Vitae Associates	08 25 44 40 66	Australia	270.05

VistaMarketing 13 ×

Result Grid

Read Only

Output

Action Output

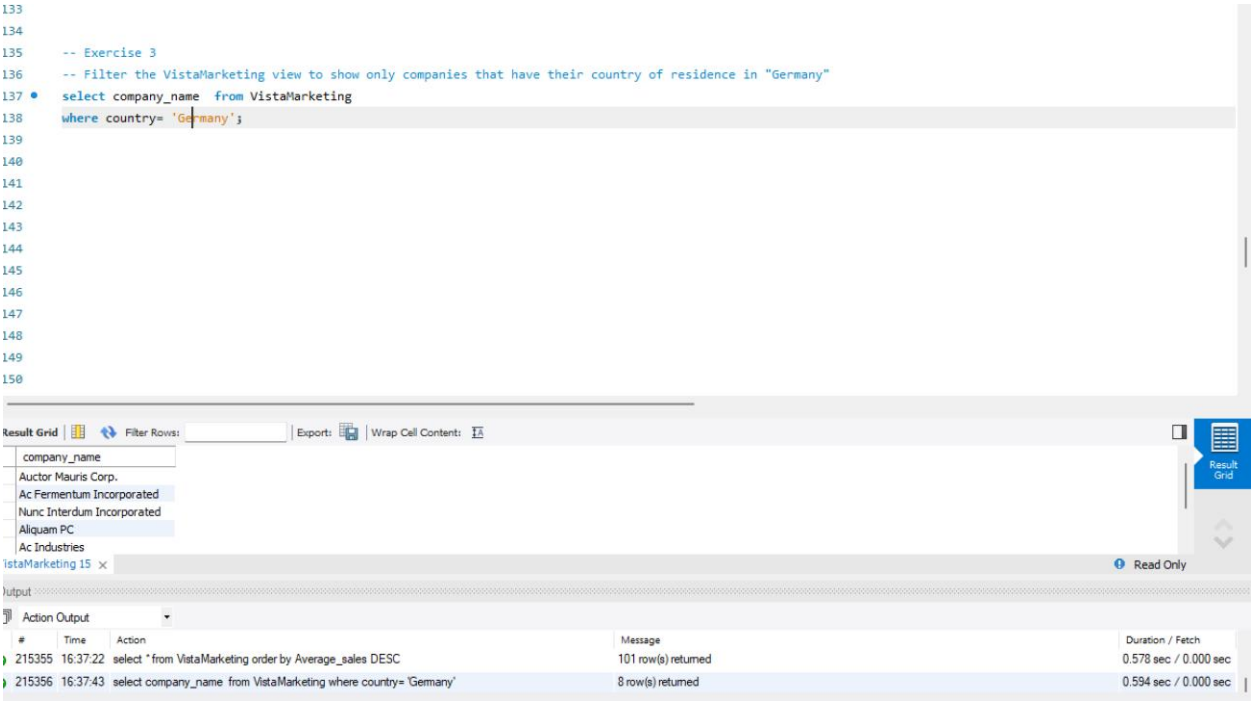
#	Time	Action	Message	Duration / Fetch
215353	16:29:41	create view VistaMarketing as select com.company_name,com.phone,com.country, round(avg(tran...	0 row(s) affected	0.032 sec
215354	16:29:56	select * from VistaMarketing order by Average_sales DESC	101 row(s) returned	0.641 sec / 0.000 sec

Exercise 3

Filter the VistaMarketing view to show only companies that have their country of residence in "Germany"

select company_name from VistaMarketing

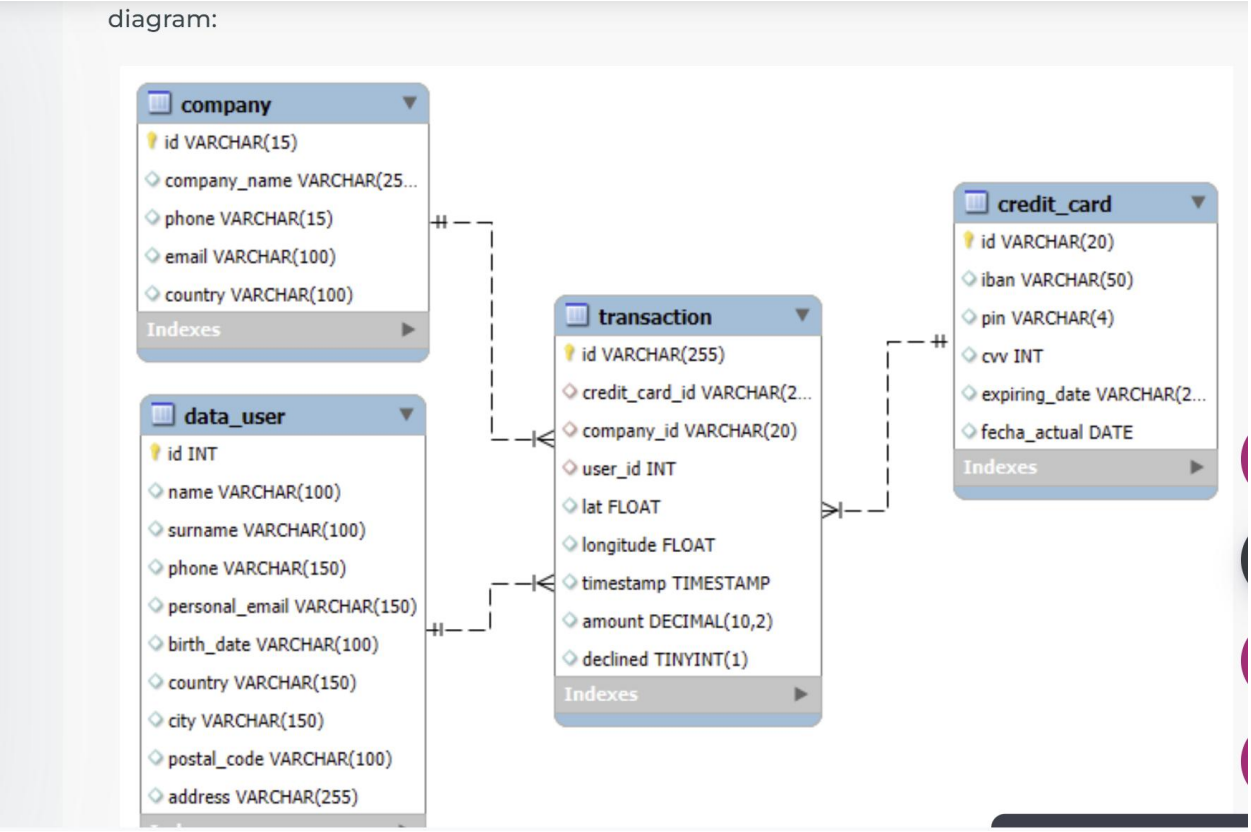
where country= 'Germany';



Level 3

Exercise 1

Next week you will have another meeting with the marketing managers. A colleague of your team made modifications to the database, but he does not remember how he did them. He asks you to help him leave the commands executed to obtain the following diagram:



-- create table user

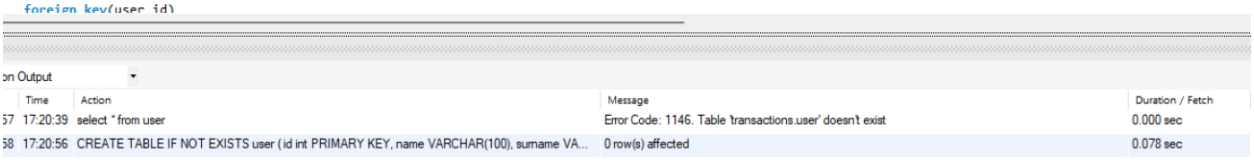
CREATE TABLE IF NOT EXISTS user (

```
    id int PRIMARY KEY,
    name VARCHAR(100),
    surname VARCHAR(100),
    phone VARCHAR(150),
    email VARCHAR(150),
    birth_date VARCHAR(100),
    country VARCHAR(150),
    city VARCHAR(150),
    postal_code VARCHAR(100),
    address VARCHAR(255)
);
select * from user;
select * from transaction;

SELECT DISTINCT user_id
FROM transaction
WHERE user_id IS NOT NULL
AND user_id NOT IN (SELECT id FROM user);

insert into
user(id)
values('9999');

-- add the foreign key to the transaction table and it refer the user table
alter table transaction
add constraint fk_user_id1
```

```
FOREIGN KEY(user_id)
REFERENCES user(id);

-- delete the website column in the companytable
select * from company;
alter table company
drop column website;


-- modify the  datatype of the column in the  credit_card table
alter table credit_card
modify column pin varchar(4),
modify column expiring_date varchar(25);


-- add the column(fecha_actual) in the credit_card table
alter table credit_card
add fecha_actual date default(current_date);

-- change the column name of the table
alter table user
change  email personal_email varchar(150);

-- change the table name
rename table user to  data_user;
```

First step create the table user

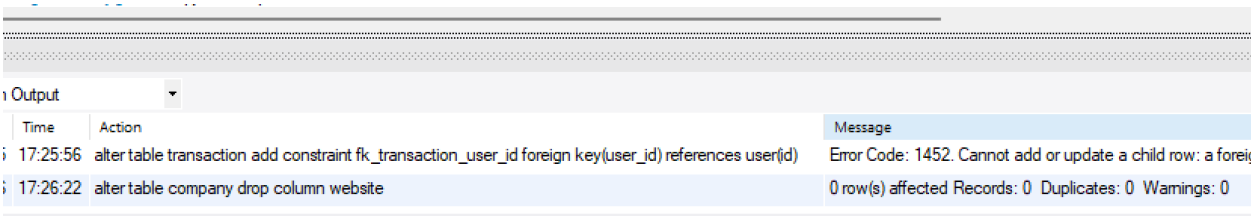
```sql
-- Level 3
-- Next week you will have another meeting with the marketing managers.
-- A colleague of your team made modifications to the database, but he does not remember how he did them.
-- He asks you to help him leave the commands executed to obtain the following diagram:

-- create table user
CREATE TABLE IF NOT EXISTS user (
    id int PRIMARY KEY,
    name VARCHAR(100),
    surname VARCHAR(100),
    phone VARCHAR(150),
    email VARCHAR(150),
    birth_date VARCHAR(100),
    country VARCHAR(150),
    city VARCHAR(150),
    postal_code VARCHAR(100),
    address VARCHAR(255)
);
select * from user;
select * from transaction;

alter table transaction
add constraint fk_transaction_user_id
foreign key(user_id)
```



	Time	Action	Message	Duration / Fetch
57	17:20:39	select * from user	Error Code: 1146. Table 'transactions.user' doesn't exist	0.000 sec
58	17:20:56	CREATE TABLE IF NOT EXISTS user (id int PRIMARY KEY, name VARCHAR(100), surname VA...	0 row(s) affected	0.078 sec

Here Drop the column website in the company table

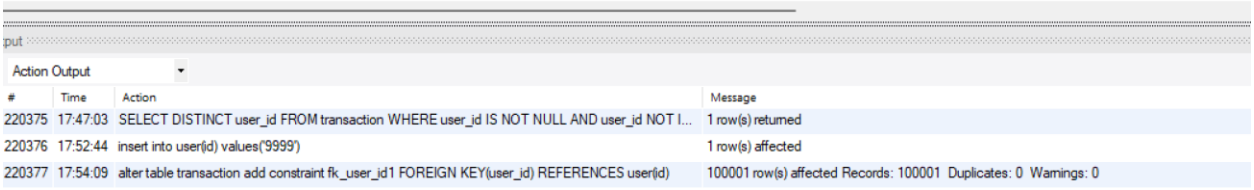
```sql
-- delete the website column in the companytable
select * from company;
alter table company
drop column website;
```



Time	Action	Message
17:25:56	alter table transaction add constraint fk_transaction_user_id foreign key(user_id) references user(id)	Error Code: 1452. Cannot add or update a child row: a forei
17:26:22	alter table company drop column website	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

Add the foreign constraint to the table transaction references the user table

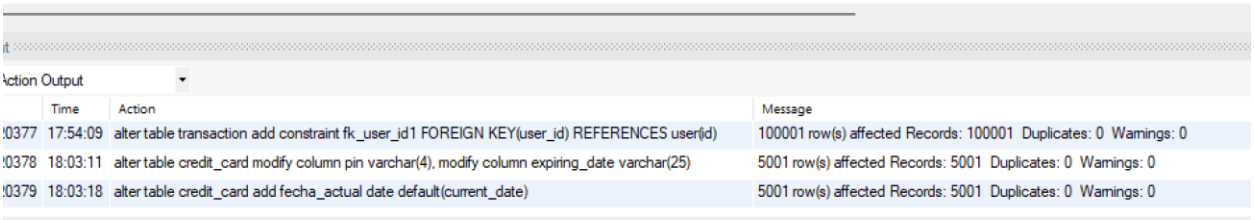
```sql
-- add the foreign key to the transaction table and it refer the user table
alter table transaction
add constraint fk_user_id1
FOREIGN KEY(user_id)
REFERENCES user(id);

-- delete the website column in the companytable
select * from company;
alter table company
drop column website;



alter table credit_card
modify column pin varchar(4);

alter table credit_card
modify column expiring_date varchar(25);



alter table credit_card
```



#	Time	Action	Message
220375	17:47:03	SELECT DISTINCT user_id FROM transaction WHERE user_id IS NOT NULL AND user_id NOT I...	1 row(s) returned
220376	17:52:44	insert into user(id) values('9999')	1 row(s) affected
220377	17:54:09	alter table transaction add constraint fk_user_id1 FOREIGN KEY(user_id) REFERENCES user(id)	100001 row(s) affected Records: 100001 Duplicates: 0 Warnings: 0

Add the column facha_actual (get the current date)in the credit_card table

```sql
-- modify the  datatype of the column in the  credit_card table
alter table credit_card
modify column pin varchar(4),
modify column expiring_date varchar(25);


-- add the column(fecha_actual) in the credit_card table
alter table credit_card
add fecha_actual date default(current_date);
```

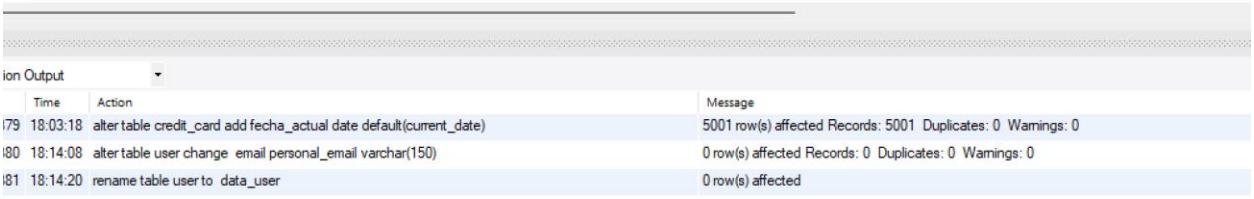


Action Output

	Time	Action	Message
0377	17:54:09	alter table transaction add constraint fk_user_id1 FOREIGN KEY(user_id) REFERENCES user(id)	100001 row(s) affected Records: 100001 Duplicates: 0 Warnings: 0
0378	18:03:11	alter table credit_card modify column pin varchar(4), modify column expiring_date varchar(25)	5001 row(s) affected Records: 5001 Duplicates: 0 Warnings: 0
0379	18:03:18	alter table credit_card add fecha_actual date default(current_date)	5001 row(s) affected Records: 5001 Duplicates: 0 Warnings: 0

Change name of the table user

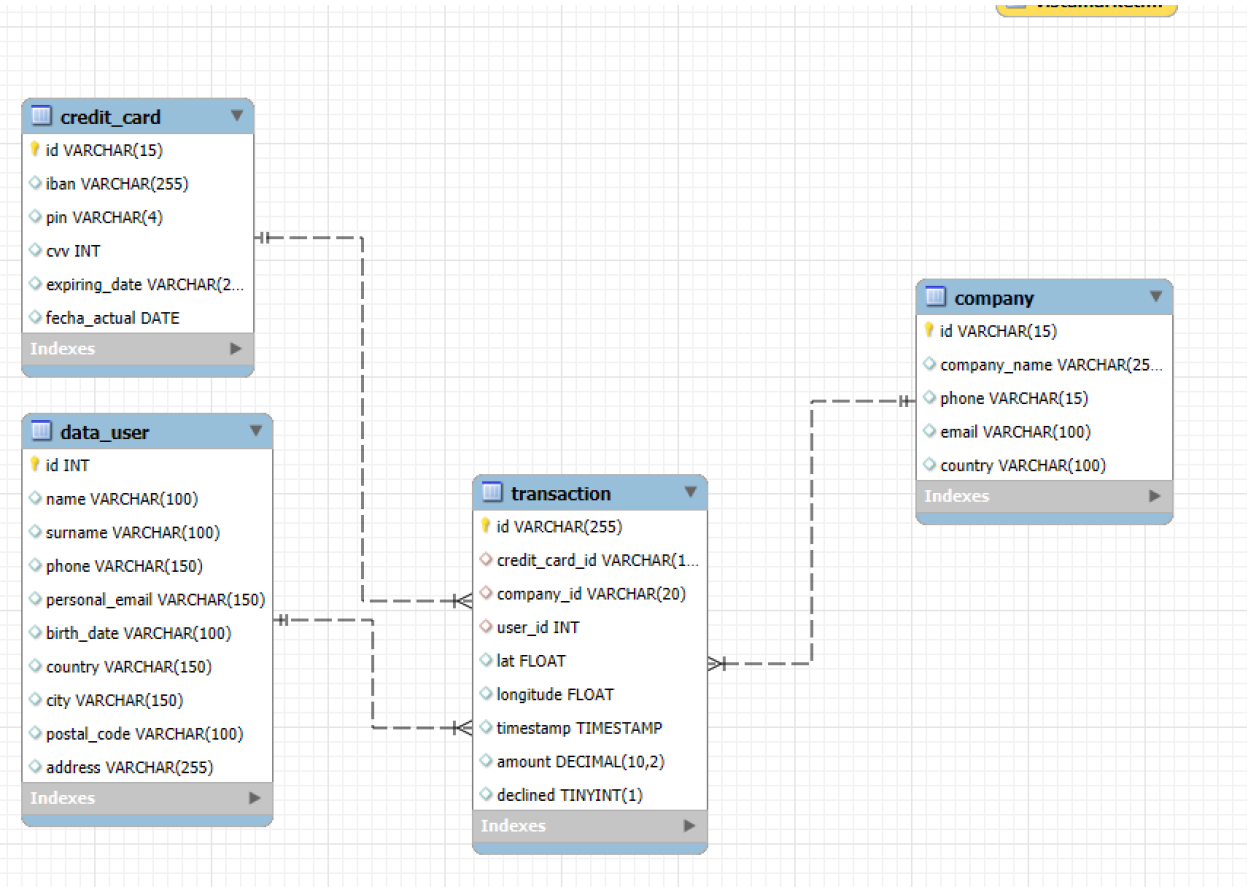
```sql
-- change the column name of the table
alter table user
change  email personal_email varchar(150);

-- change the table name
rename table user to  data_user;
```



ion Output

	Time	Action	Message
79	18:03:18	alter table credit_card add fecha_actual date default(current_date)	5001 row(s) affected Records: 5001 Duplicates: 0 Warnings: 0
80	18:14:08	alter table user change email personal_email varchar(150)	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0
81	18:14:20	rename table user to data_user	0 row(s) affected

ER Diagram


credit_card
id VARCHAR(15)
iban VARCHAR(255)
pin VARCHAR(4)
cvv INT
expiring_date VARCHAR(2...
fecha_actual DATE
Indexes
company
id VARCHAR(15)
company_name VARCHAR(25...
phone VARCHAR(15)
email VARCHAR(100)
country VARCHAR(100)
Indexes
data_user
id INT
name VARCHAR(100)
surname VARCHAR(100)
phone VARCHAR(150)
personal_email VARCHAR(150)
birth_date VARCHAR(100)
country VARCHAR(150)
city VARCHAR(150)
postal_code VARCHAR(100)
address VARCHAR(255)
Indexes
transaction
id VARCHAR(255)
credit_card_id VARCHAR(1...
company_id VARCHAR(20)
user_id INT
lat FLOAT
longitude FLOAT
timestamp TIMESTAMP
amount DECIMAL(10,2)
declined TINYINT(1)
Indexes

Exercise 2

The company also asks you to create a view called "TechnicalReport" that contains the following information:

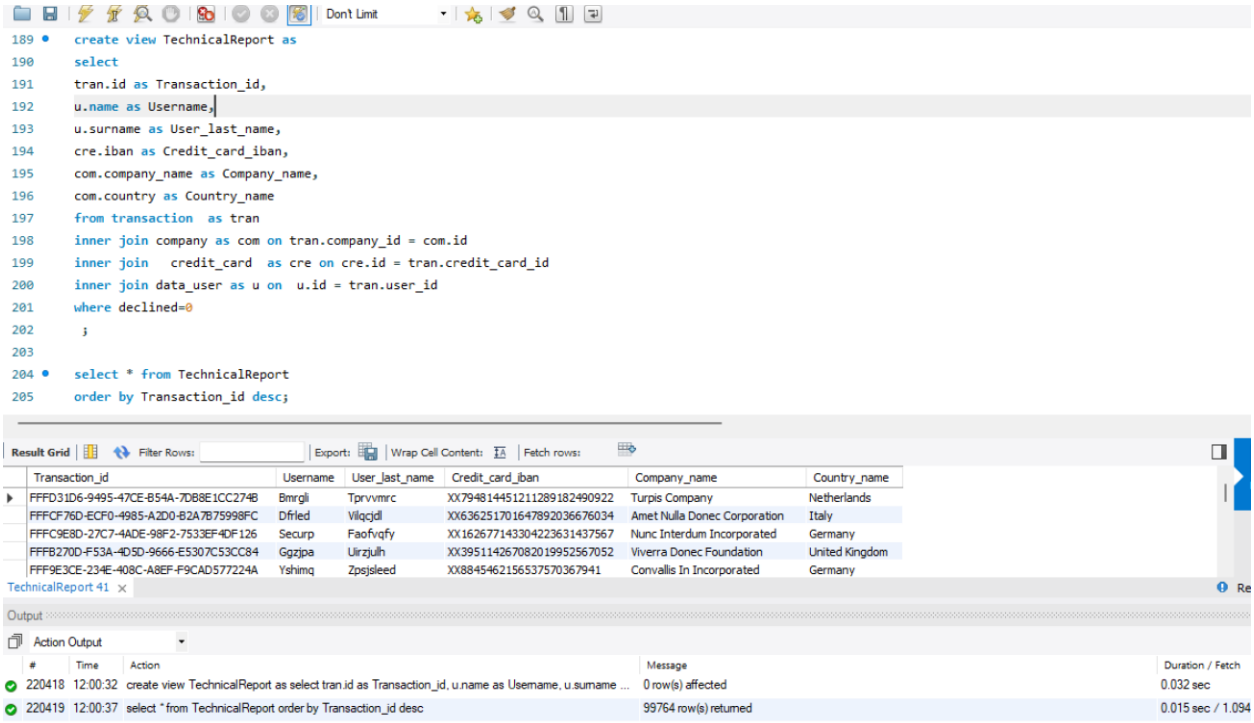
- Transaction ID
- Username
- User's last name
- IBAN of the credit card used.
- Name of the company of the transaction carried out.
- Be sure to include relevant information from tables you will be familiar with, and use aliases to rename columns as needed.

Displays the results of the view, sorts the results in descending order based on the transaction ID variable.

```
create view TechnicalReport as
select
tran.id as Transaction_id,
u.name as Username,
u.surname as User_last_name,
cre.iban as Credit_card_iban,
com.company_name as Company_name,
com.country as Country_name
from transaction  as tran
inner join company as com on tran.company_id = com.id
inner join   credit_card  as cre on cre.id = tran.credit_card_id
inner join data_user as u on  u.id = tran.user_id
where declined=0
 ;
```

select * from TechnicalReport

order by Transaction_id desc;



```
create view TechnicalReport as
select
tran.id as Transaction_id,
u.name as Username,
u.surname as User_last_name,
cre.iban as Credit_card_iban,
com.company_name as Company_name,
com.country as Country_name
from transaction  as tran
inner join company as com on tran.company_id = com.id
inner join   credit_card  as cre on cre.id = tran.credit_card_id
inner join data_user as u on  u.id = tran.user_id
where declined=0
 ;

select * from TechnicalReport
order by Transaction_id desc;
```

Result Grid

Transaction_id	Username	User_last_name	Credit_card_iban	Company_name	Country_name
FFFD31D6-9495-47CE-B54A-7DB8E1CC274B	Bmrgli	Tprvvmrc	XX7948144512112891824909 22	Turpis Company	Netherlands
FFFCF76D-ECF0-4985-A2D0-B2A7B75998FC	Dfrled	Vilqcjdl	XX6362517016478920366760 34	Amet Nulla Donec Corporation	Italy
FFFC9E8D-27C7-4ADE-98F2-7533EF4DF126	Securp	Faofvqfy	XX1626771433042236314375 67	Nunc Interdum Incorporated	Germany
FFFB270D-F53A-4D5D-9666-E5307C53CC84	Ggzjpa	Uirzjulh	XX3951142670820199525670 52	Viverra Donec Foundation	United Kingdom
FFF9E3CE-234E-408C-A8EF-F9CAD577224A	Yshimq	Zpsjsleed	XX8845462156537570367941	Convallis In Incorporated	Germany

TechnicalReport 41

Output

Action Output

#	Time	Action	Message	Duration / Fetch
220418	12:00:32	create view TechnicalReport as select tran.id as Transaction_id, u.name as Username, u.surname ...	0 row(s) affected	0.032 sec
220419	12:00:37	select * from TechnicalReport order by Transaction_id desc	99764 row(s) returned	0.015 sec / 1.094