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.

-- Tengo que eliminar la tabla credit_card para tener los pantallazos que faltan
-- Para esto, antes tengo que eliminar la foreign_key que relaciona credit_card con transaction.
este es el codigo:

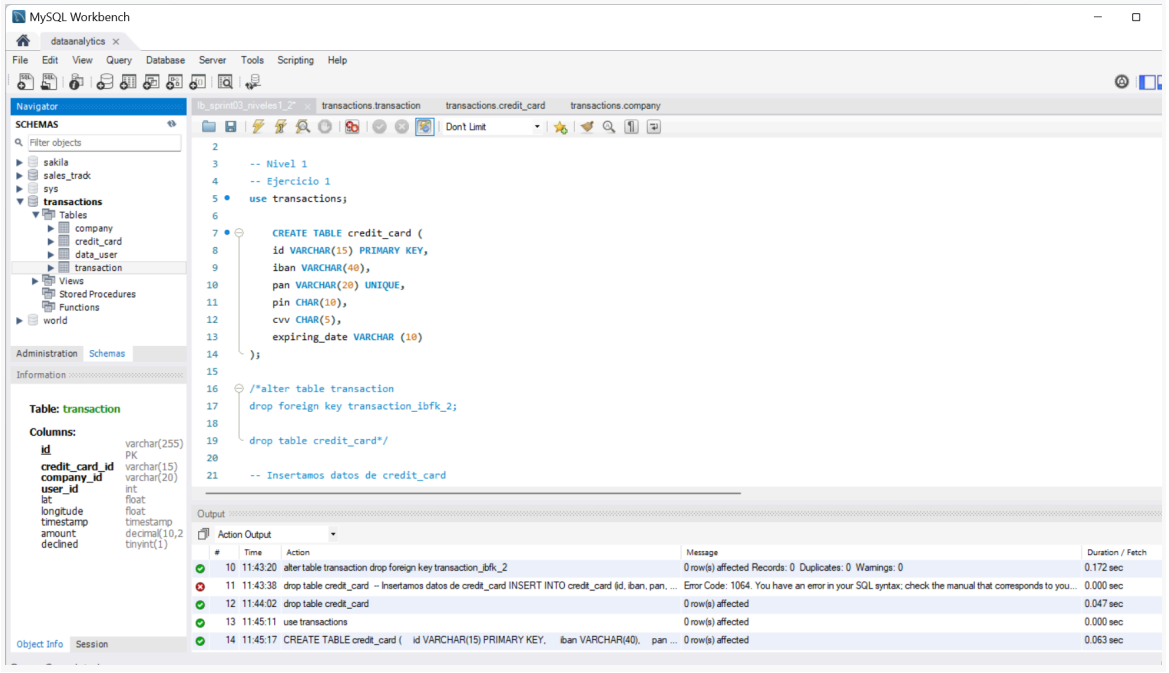
```
alter table transaction
drop foreign key transaction_ibfk_2;
```

-- Luego elimino la tabla credit_card. Script abajo:

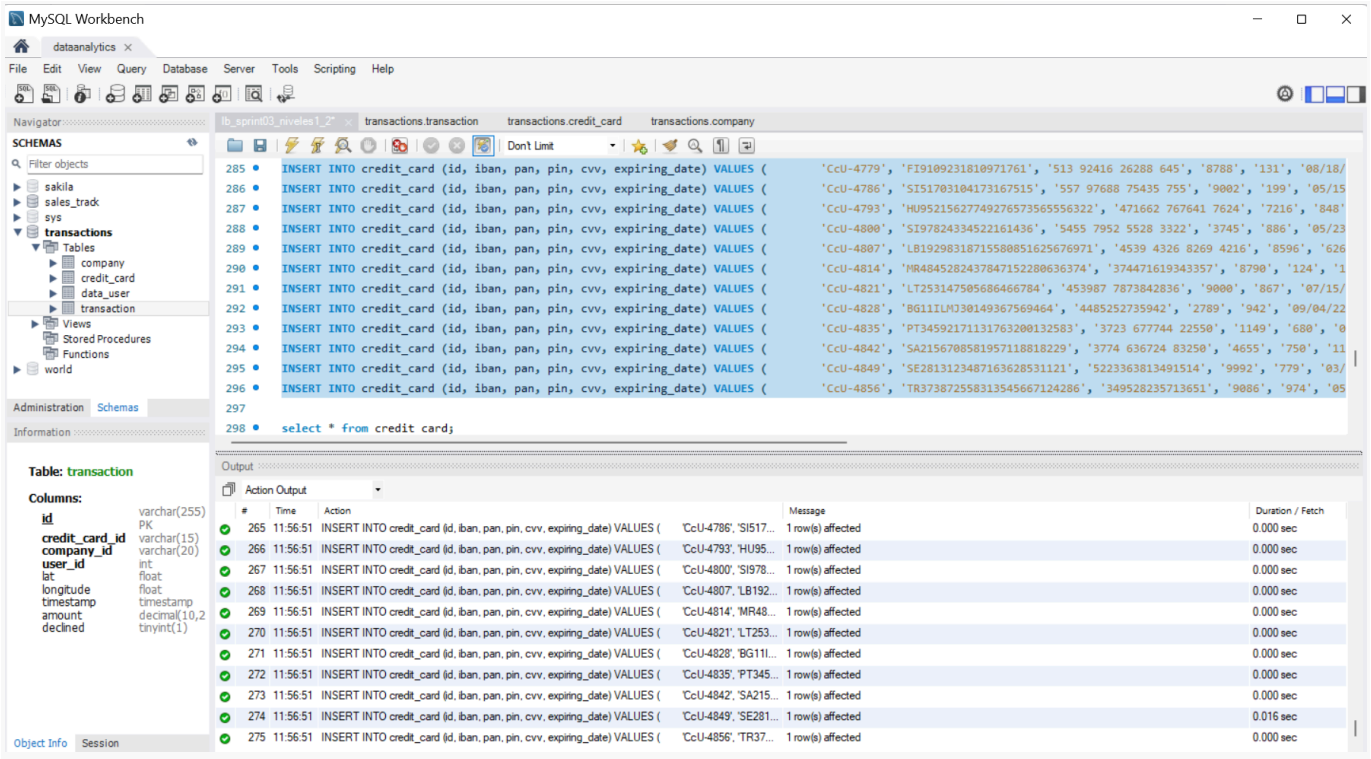
```
drop table credit_card
```

-- Vuelvo a crear la tabla credit_card dentro de database transactions

```
use transactions;

CREATE TABLE credit_card (
id VARCHAR(15) PRIMARY KEY,
iban VARCHAR(40),
pan VARCHAR(20) UNIQUE,
pin CHAR(10),
cvv CHAR(5),
expiring_date VARCHAR (10)
);
```



```
-- Insertamos datos de credit_card
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date)
VALUES (	'CcU-2938', 'TR3019503122135768176386 61', '5424465566813633', '3257', '984', '10/30/22');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date)
VALUES (	'CcU-2945', 'DO26854763748537475216568689', '5142423821948828', '9080', '887',
'08/24/23');
*
*
*
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date)
VALUES (	'CcU-4842', 'SA2156708581957118818229', '3774 636724 83250', '4655', '750', '11/11/24');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date)
VALUES (	'CcU-4849', 'SE2813123487163628531121', '5223363813491514', '9992', '779', '03/21/25');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date)
 VALUES (	'CcU-4856', 'TR373872558313545667124286', '349528235713651', '9086', '974',
'05/19/23');
```



-- Intento modificar la tabla transaction para añadir el id de la table credit_card como Foreign Key con este script:

```
ALTER TABLE transaction
ADD FOREIGN KEY (credit_card_id) REFERENCES credit_card(id);
```

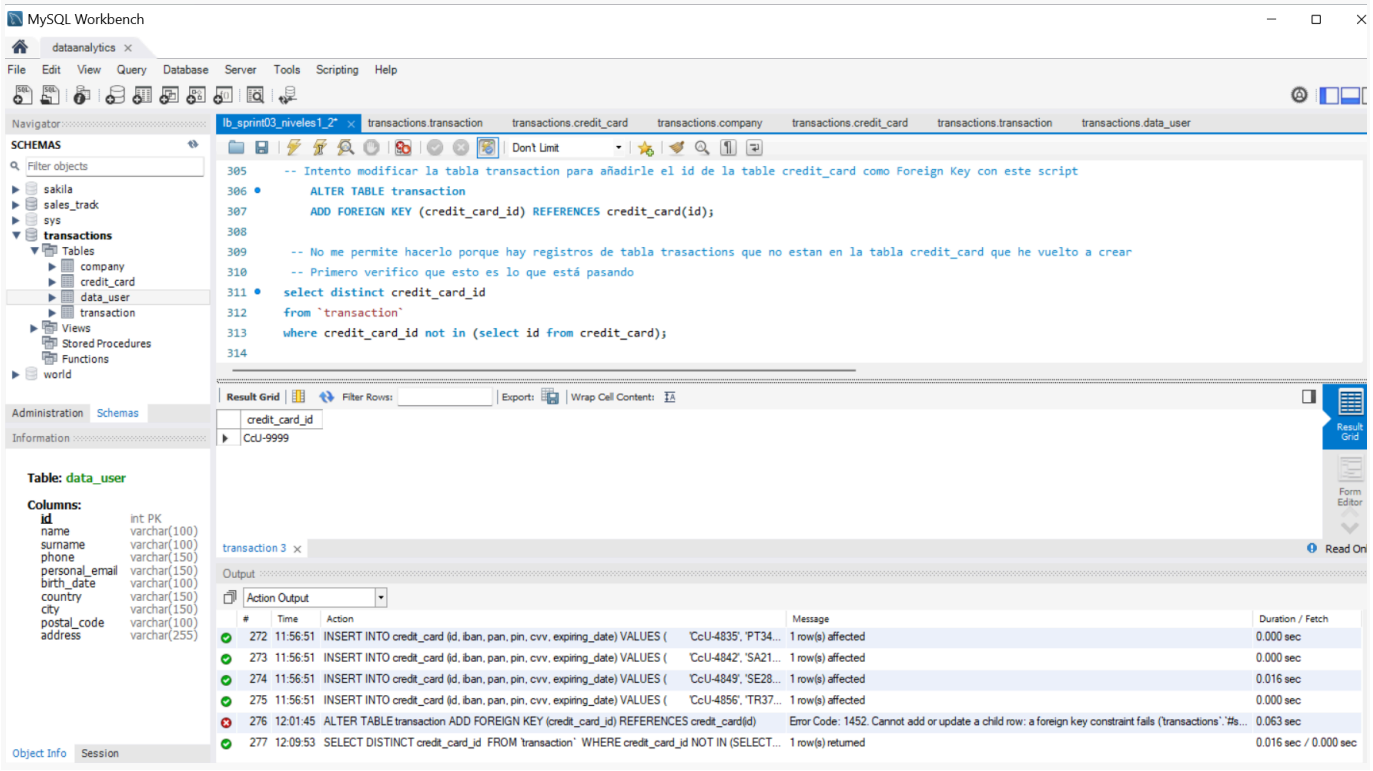
-- No me permite hacerlo. No da: Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`transactions`.`#sql-15a0_e4`, CONSTRAINT `transaction_ibfk_2` FOREIGN KEY (`credit_card_id`) REFERENCES `credit_card` (`id`))

Parece que hay registros de tabla transaction que no están en la tabla credit_card que he vuelto a crear

-- Primero verifico si es esto es lo que está pasando

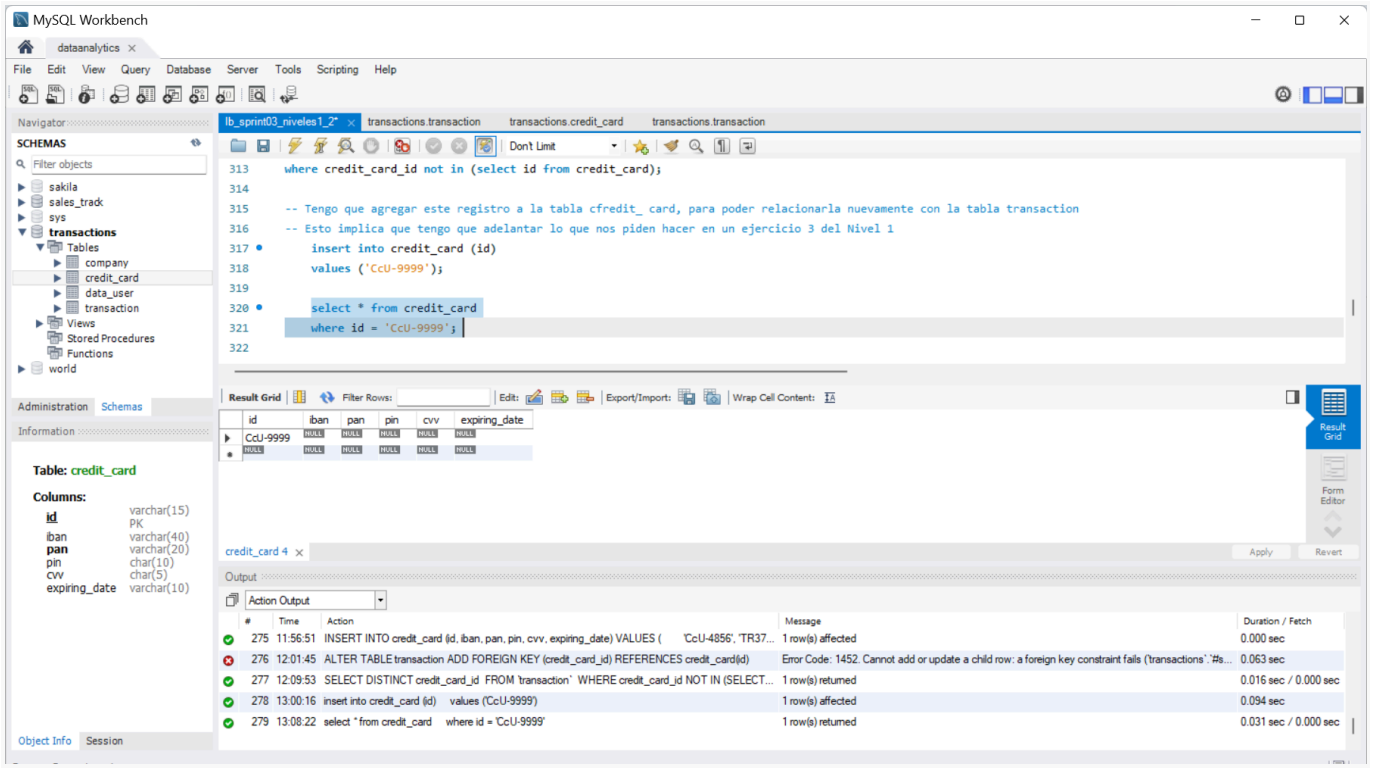
```
select distinct credit_card_id
from `transaction`
where credit_card_id not in (select id from credit_card);
```

Sí hay un registro que no está en credit_card, ya que fue creada nuevamente. Es el registro: CcU-9999



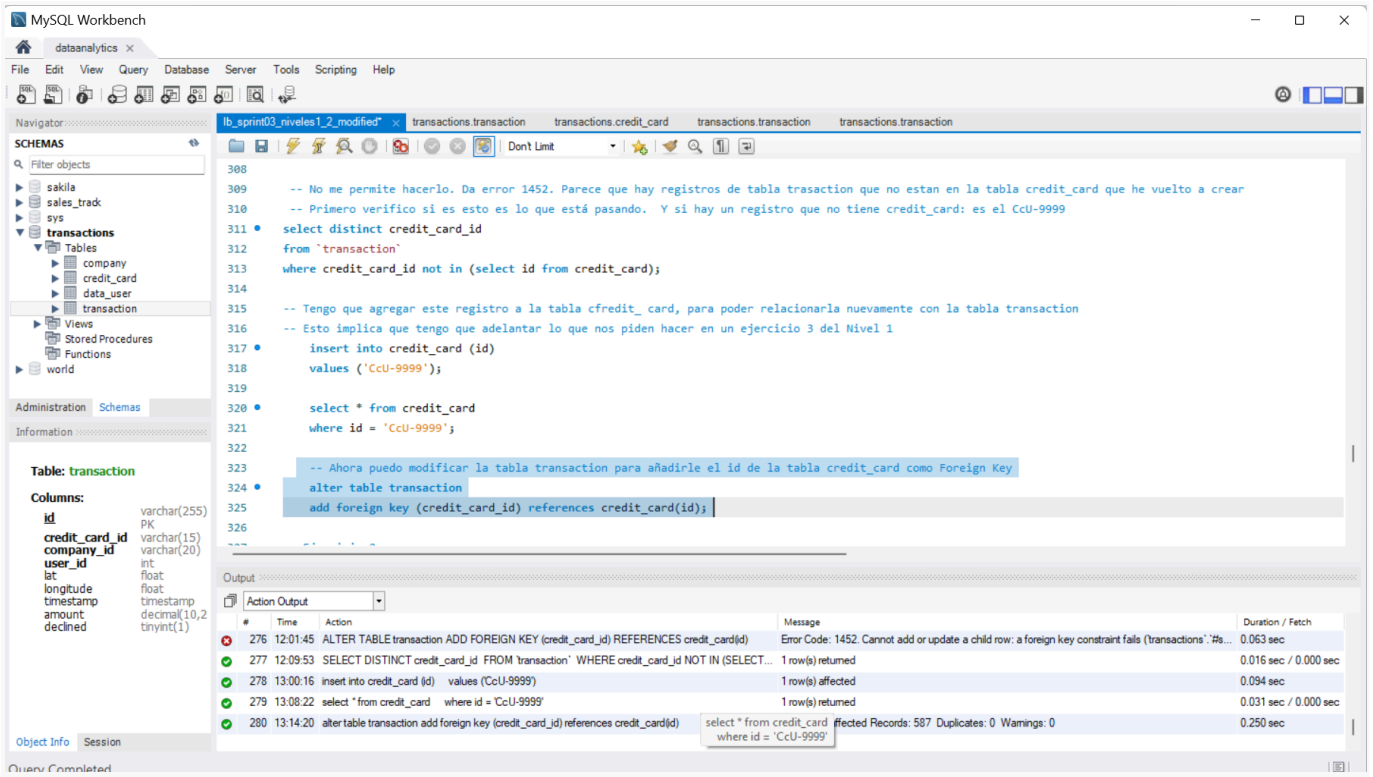
Tengo que agregar este registro a la tabla credit_ card, antes de poder relacionarla nuevamente con la tabla transaction
Esto implica que tengo que adelantar lo que nos piden hacer en un ejercicio 3 del Nivel 1

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

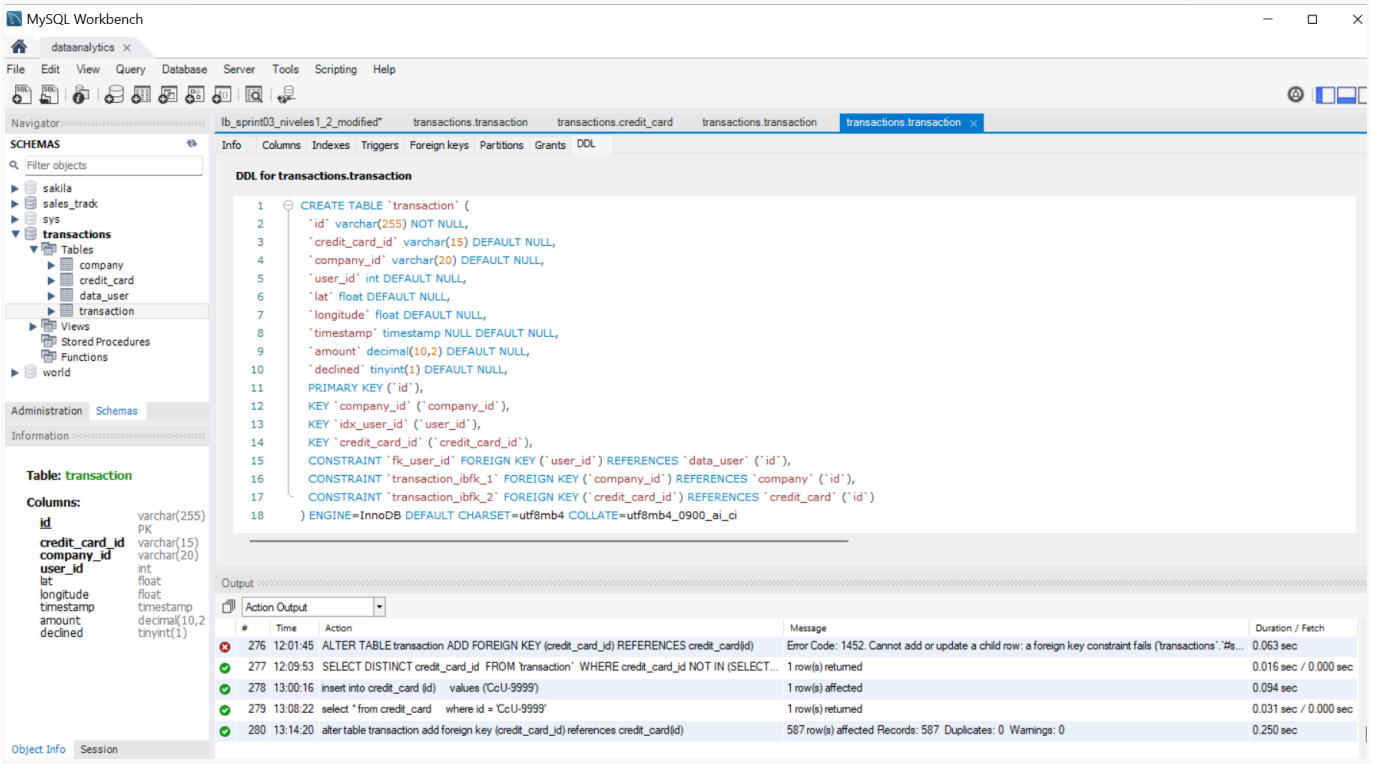


Ahora puedo modificar la tabla transaction para añadirle el id de la tabla credit_card como Foreign Key

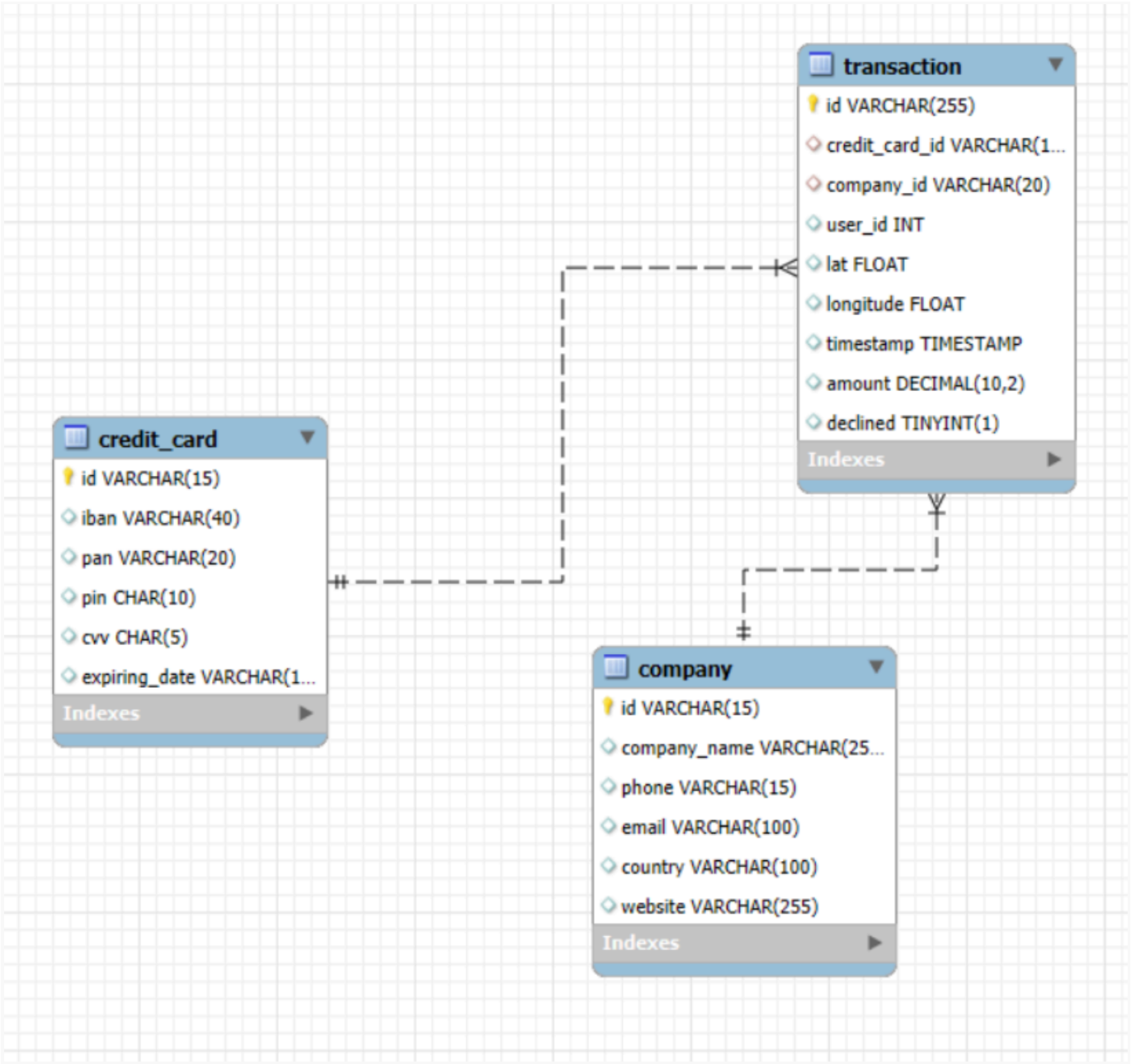
```
alter table transaction
add foreign key (credit_card_id) references credit_card(id);
```



Así queda la estructura de la tabla transaction



Y este es el diagrama entidad relación:



- 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.

```
299
300     -- Este es el iban erroneo (antes del cambio):
301 •   select * from credit_card
302     where id = 'Ccu-2938';
303
```

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

id	iban	pan	pin	cvv	expiring_date
CcU-2938	TR3019503122135768176386 61	5424465566813633	3257	984	10/30/22
NULL	NULL	NULL	NULL	NULL	NULL

```
304     -- Modifico el iban para que sea igual a R323456312213576817699999
305 •   update credit_card
306     set iban = 'R323456312213576817699999'
307     WHERE id = 'CcU-2938';
308
309     -- Este es el iban correcto tras el cambio
310 •   select * from credit_card
311     where id = 'Ccu-2938';
```

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

id	iban	pan	pin	cvv	expiring_date
CcU-2938	R323456312213576817699999	5424465566813633	3257	984	10/30/22
NULL	NULL	NULL	NULL	NULL	NULL

- Ejercicio 3

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

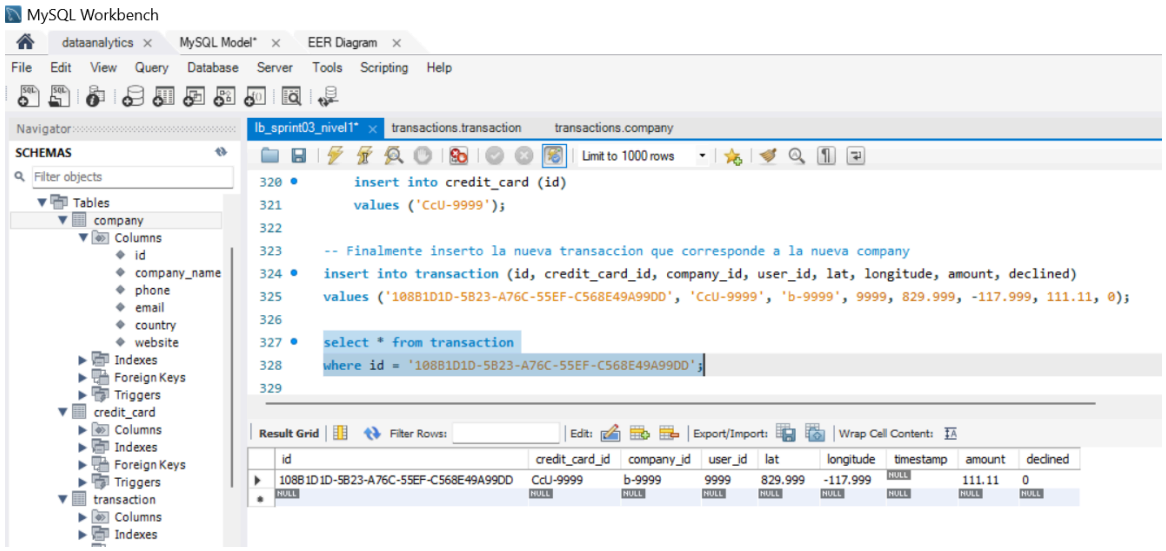
Id	108B1D1D-5B23-A76C-55EF-C568E49A99DD
credit_card_id	CcU-9999
company_id	b-9999
user_id	9999
lato	829.999
longitud	-117.999
amunt	111.11
declined	0

-- Inicialmente, este script no funciona porque tengo que agregar la info del nuevo usuario tanto en tabla company como en tabla credit_card

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);

-- Primero tengo que registrar a la nueva compañia b-9999 en la tabla company
insert into company (id)
values ('b-9999');
-- Luego registro la tarjeta de crédito en la tabla credit_card
insert into credit_card (id)
values ('CcU-9999');
-- Finalmente inserto la nueva transaccion que corresponde a la nueva company

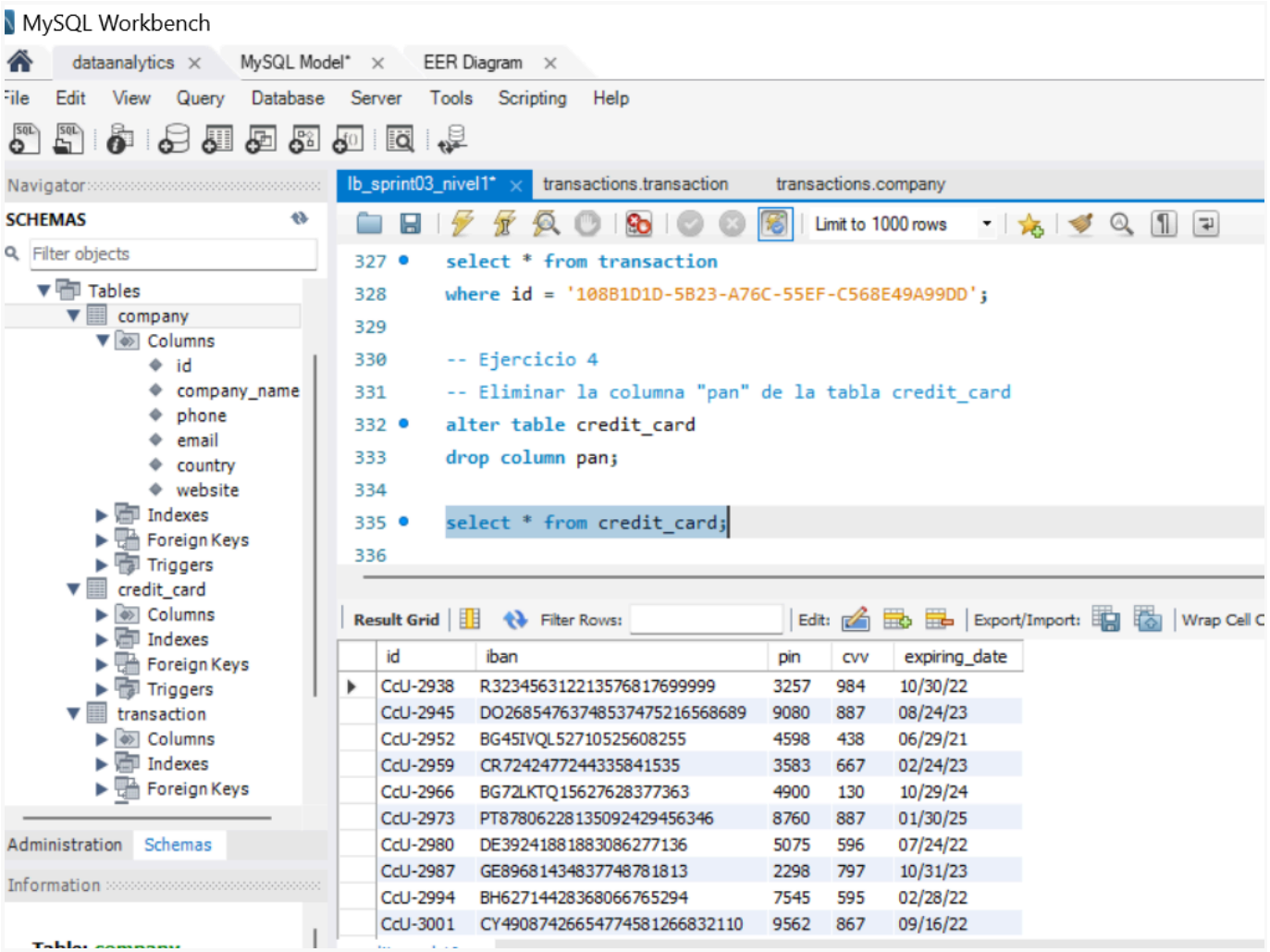
insert into transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined)
values ('108B1D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', 9999, 829.999, -117.999, 111.11, 0);



- Ejercicio 4

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

```
alter table credit_card
drop column pan;
```

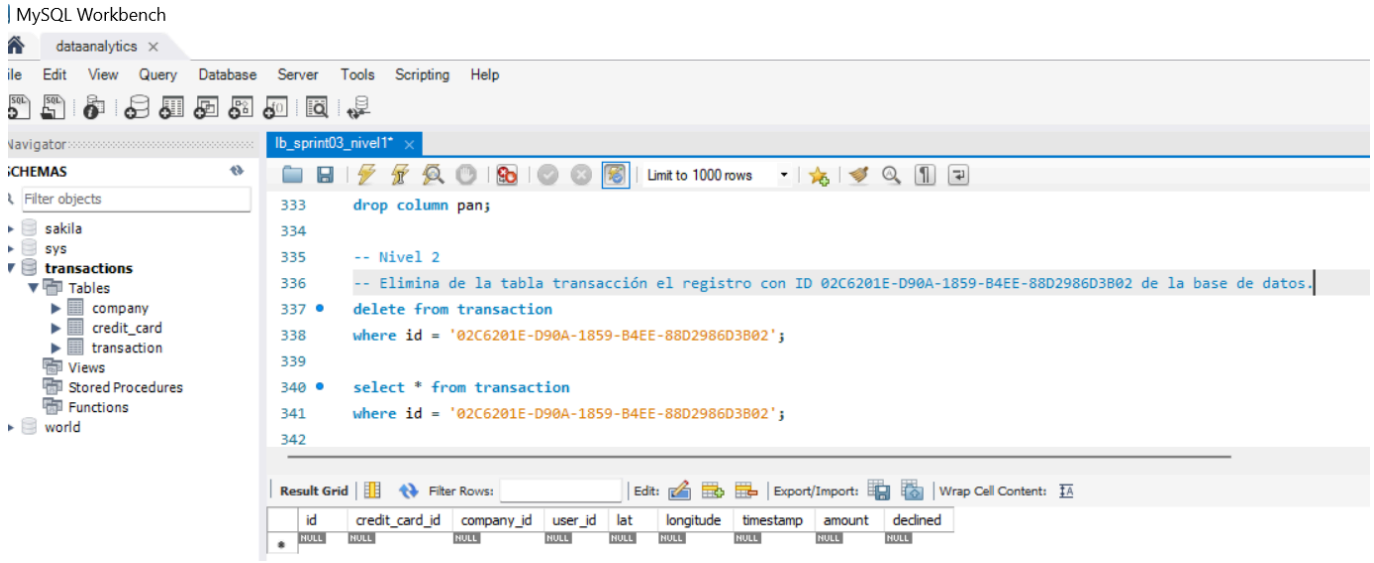


Nivel 2

Ejercicio 1

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

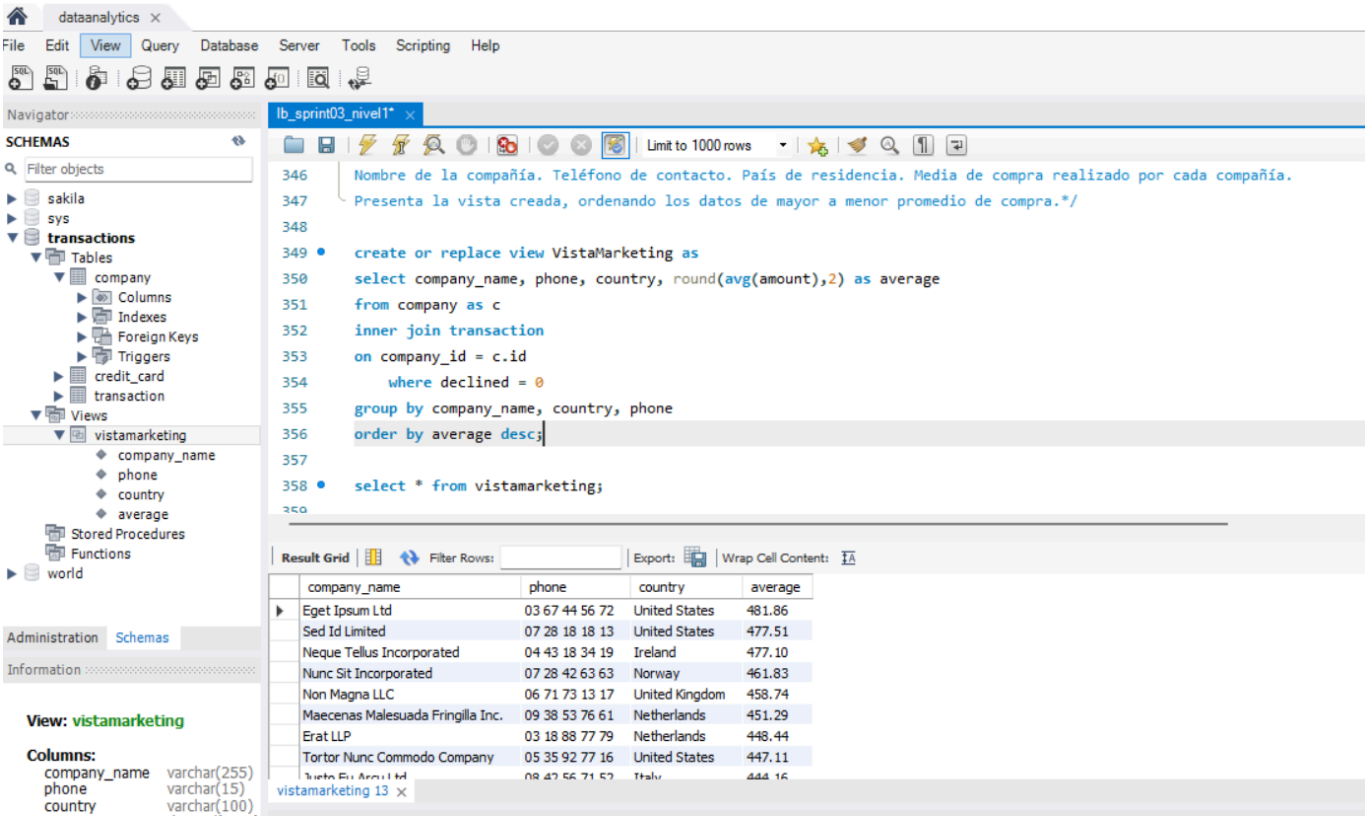
```
delete from transaction
where id = '02C6201E-D90A-1859-B4EE-88D2986D3B02';
```



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.

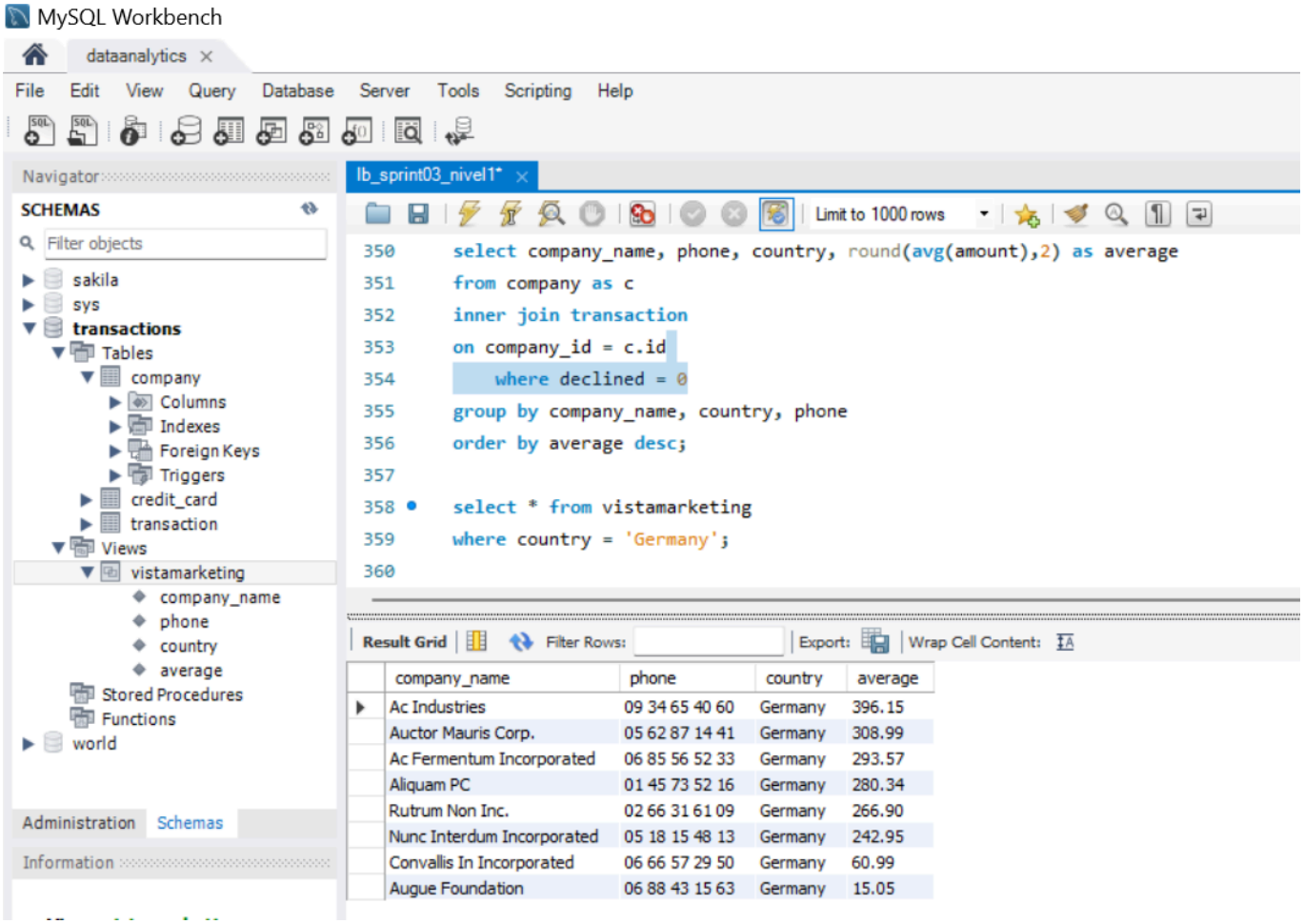
```
create or replace view VistaMarketing as
select company_name, phone, country, round(avg(amount),2) as average
from company as c
inner join transaction
on company_id = c.id
        where declined = 0
group by company_name, country, phone
order by average desc;

select * from vistamarketing;
```



Ejercicio 3

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

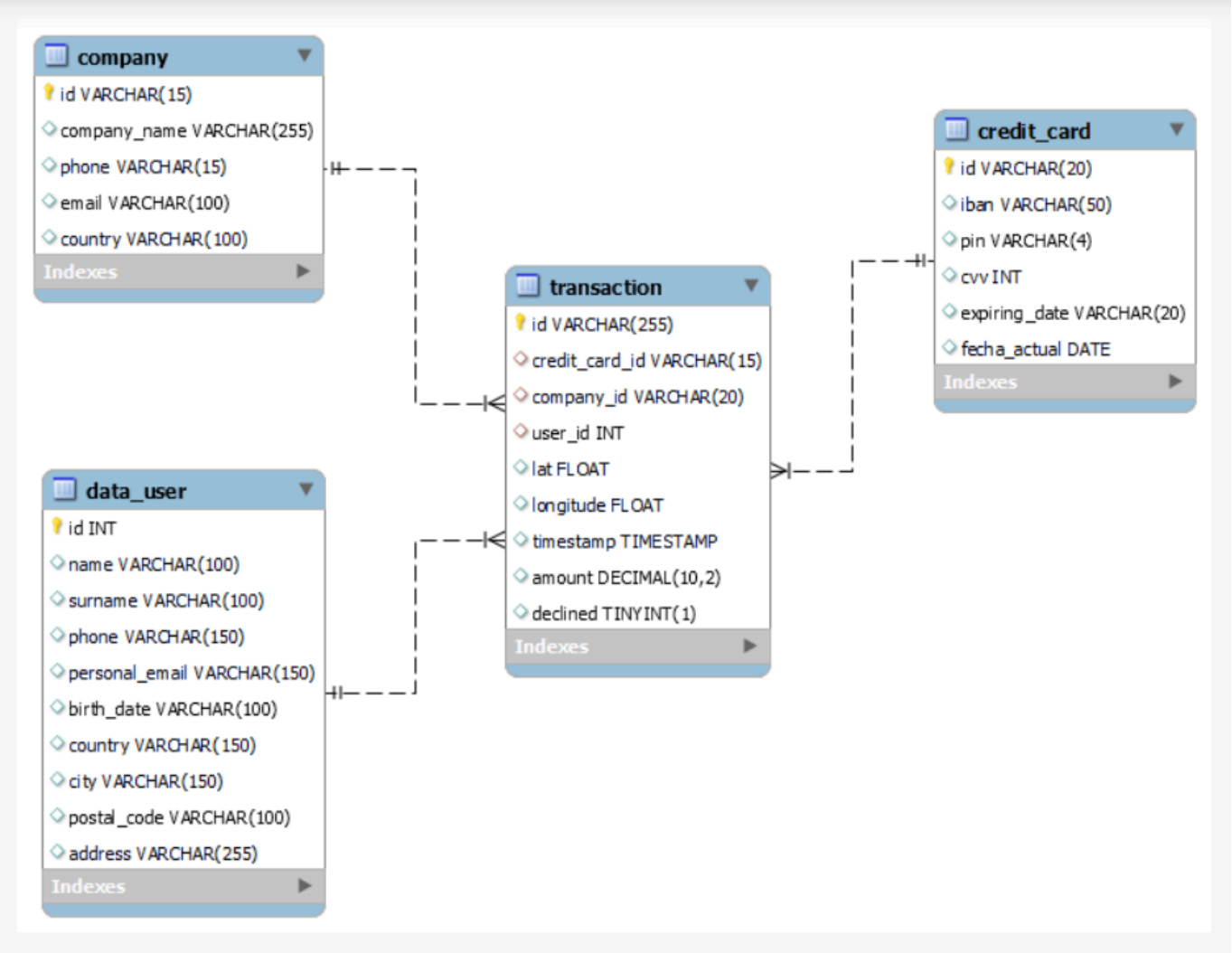
```
select * from vistamarketing
where country = 'Germany';
```



Nivel 3

Ejercicio 1

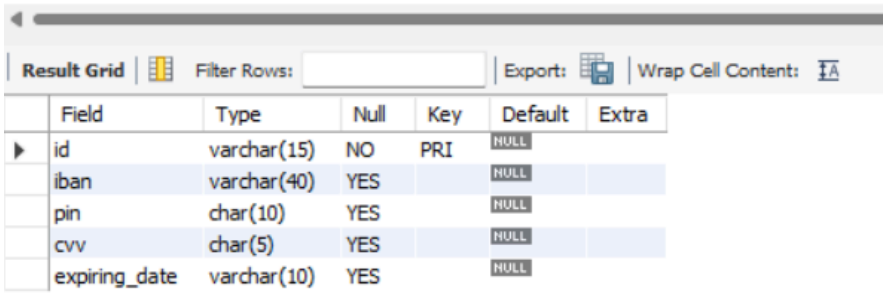
La próxima semana tendrás una nueva reunión con los gerentes de marketing. Un compañero de tu equipo realizó modificaciones en la base de datos, pero no recuerda cómo las realizó. Te pide que le ayudes a dejar los comandos ejecutados para obtener el siguiente diagrama:



Recordatorio;
En esta actividad, es necesario que describas el "paso a paso" de las tareas realizadas. Es importante realizar descripciones sencillas, simples y fáciles de comprender. Para realizar esta actividad deberás trabajar con los archivos denominados "estructura_datos_user" y "datos_introducir_user"

Primero,voy a modificar la tabla credit_card
Campos de tabla credit_card ANTES de la modificación

```
301
302     -- Muestro campos de tabla credit_card
303 •   show columns from credit_card;
304
305
```



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

Field	Type	Null	Key	Default	Extra
id	varchar(15)	NO	PRI	NULL	
iban	varchar(40)	YES		NULL	
pin	char(10)	YES		NULL	
cvv	char(5)	YES		NULL	
expiring_date	varchar(10)	YES		NULL	

Modifico campos id, iban, expiring_date de la tabla credit_card.

```
alter table credit_card
modify id varchar(20),
modify iban varchar(50),
modify expiring_date varchar(20);
```

Modifico campos pin, cvv de la tabla credit_card.

```
alter table credit_card
modify pin varchar(4),
modify cvv int;
```

Agrego campo fecha_actual a la tabla credit_card:

```
alter table credit_card add column fecha_actual date;
show columns from credit_card;
```

Campos de tabla credit_card DESPUES de las modificaciones:

MySQL Workbench

dataanalytics

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- sakila
- sys
- **transactions**
  - Tables
    - company
    - credit_card
    - transaction
    - user
  - Views
    - vistamarketing
  - Stored Procedures
  - Functions
- world

Administration Schemas

Information

No object selected

lb_sprint03_nivel1* lb_sprint03_nivel3*

Limit to 1000 rows

```
305  -- Modifico campos id, iban, expiring_date de la tabla credit_card.
306  alter table credit_card
307  modify id varchar(20),
308  modify iban varchar(50),
309  modify expiring_date varchar(20);
310
311  -- Modifico campos pin, cvv de la tabla credit_card.
312  alter table credit_card
313  modify pin varchar(4),
314  modify cvv int;
315
316  show columns from credit_card;
317
318  -- Agrego campo fecha_actual a la tabla credit_card
319  alter table credit_card add column fecha_actual date;
320
321  show columns from credit_card;
```

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

Field	Type	Null	Key	Default	Extra
id	varchar(20)	NO	PRI	NULL	
iban	varchar(50)	YES		NULL	
pin	varchar(4)	YES		NULL	
cvv	int	YES		NULL	
expiring_date	varchar(20)	YES		NULL	
fecha_actual	date	YES		NULL	

Result 5

Eliminar el campo website de tabla company.

Tabla company **ANTES de la eliminación del campo website**:

```
318  -- Elimino el campo website de la tabla company
319  show columns from company;
320
```

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

Field	Type	Null	Key	Default	Extra
id	varchar(15)	NO	PRI	NULL	
company_name	varchar(255)	YES		NULL	
phone	varchar(15)	YES		NULL	
email	varchar(100)	YES		NULL	
country	varchar(100)	YES		NULL	
website	varchar(255)	YES		NULL	

Tabla company **DESPUES de la eliminacion del campo website:**

```
alter table company
drop column website;
```

```
320
321 •  show columns from company;
322
```

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

Field	Type	Null	Key	Default	Extra
id	varchar(15)	NO	PRI	NULL	
company_name	varchar(255)	YES		NULL	
phone	varchar(15)	YES		NULL	
email	varchar(100)	YES		NULL	
country	varchar(100)	YES		NULL	

Estructura de la tabla users

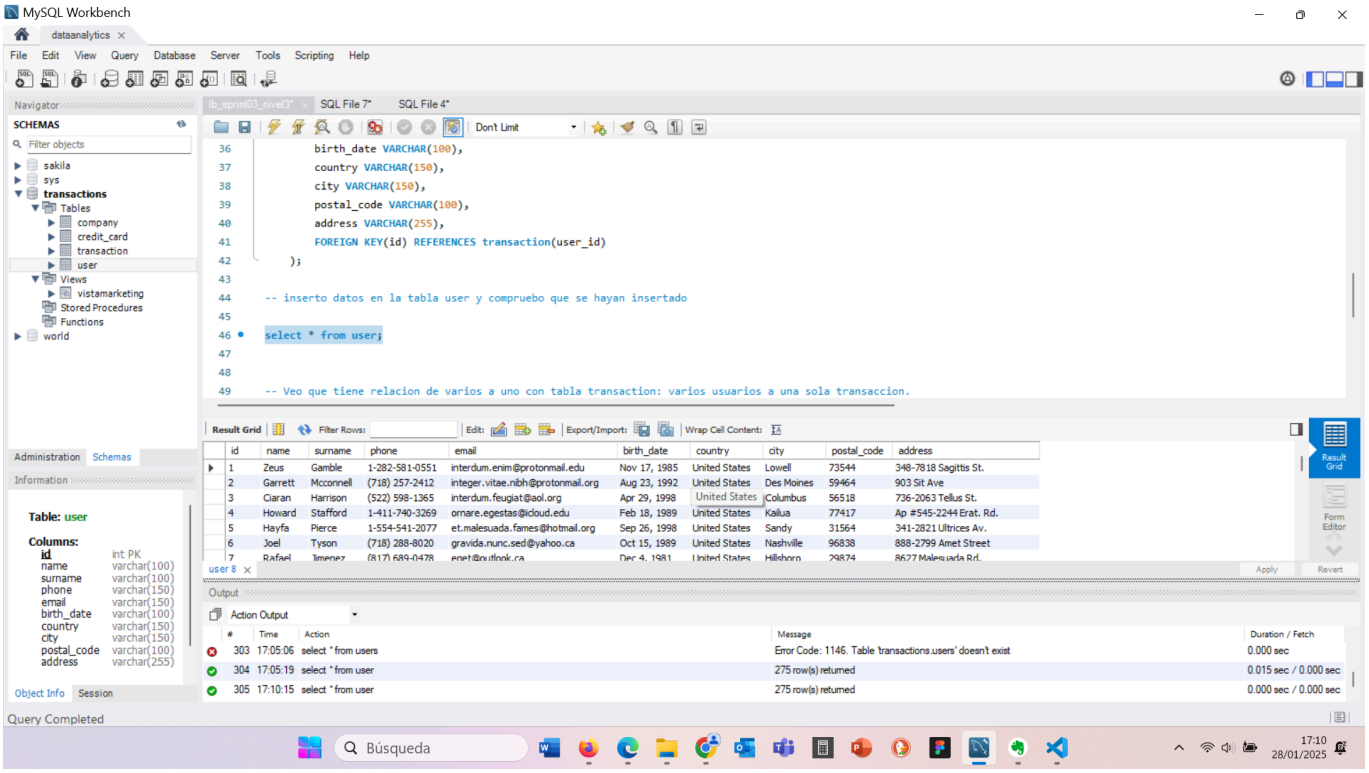
Primero, creo el index idx_user_id en la tabla transaction

```
CREATE INDEX idx_user_id ON transaction(user_id);
```

Creo la tabla users con el script del archivo estructura_datos_user.sql.

```
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),
    FOREIGN KEY(id) REFERENCES transaction(user_id)
);
```

Inserto datos del archivo en la tabla user y compruebo que se hayan insertado

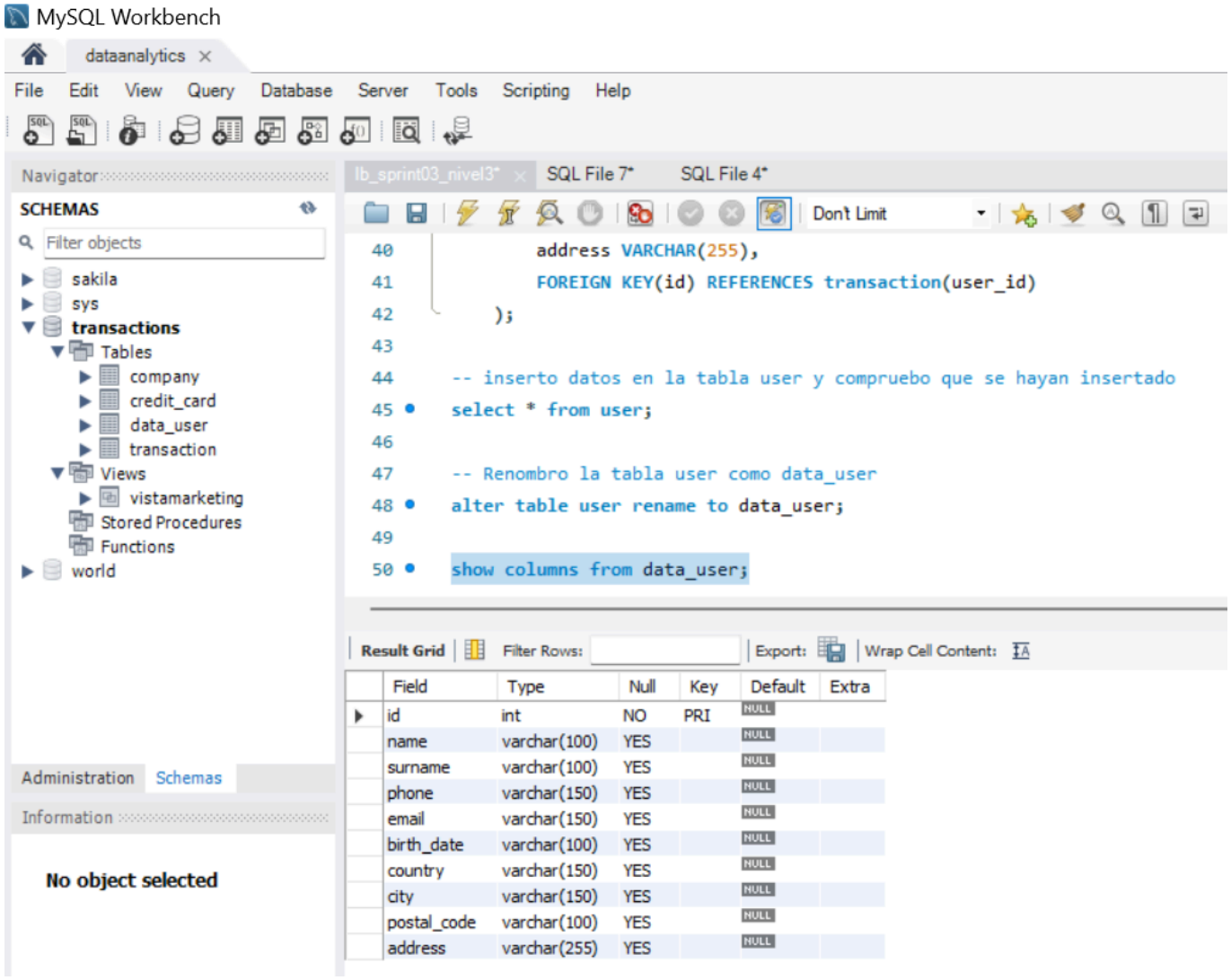


Renombro la tabla user como data_user

alter table user rename to data_user;

Ahora tengo que cambiar el nombre del campo email a personal_email en la tabla data_user

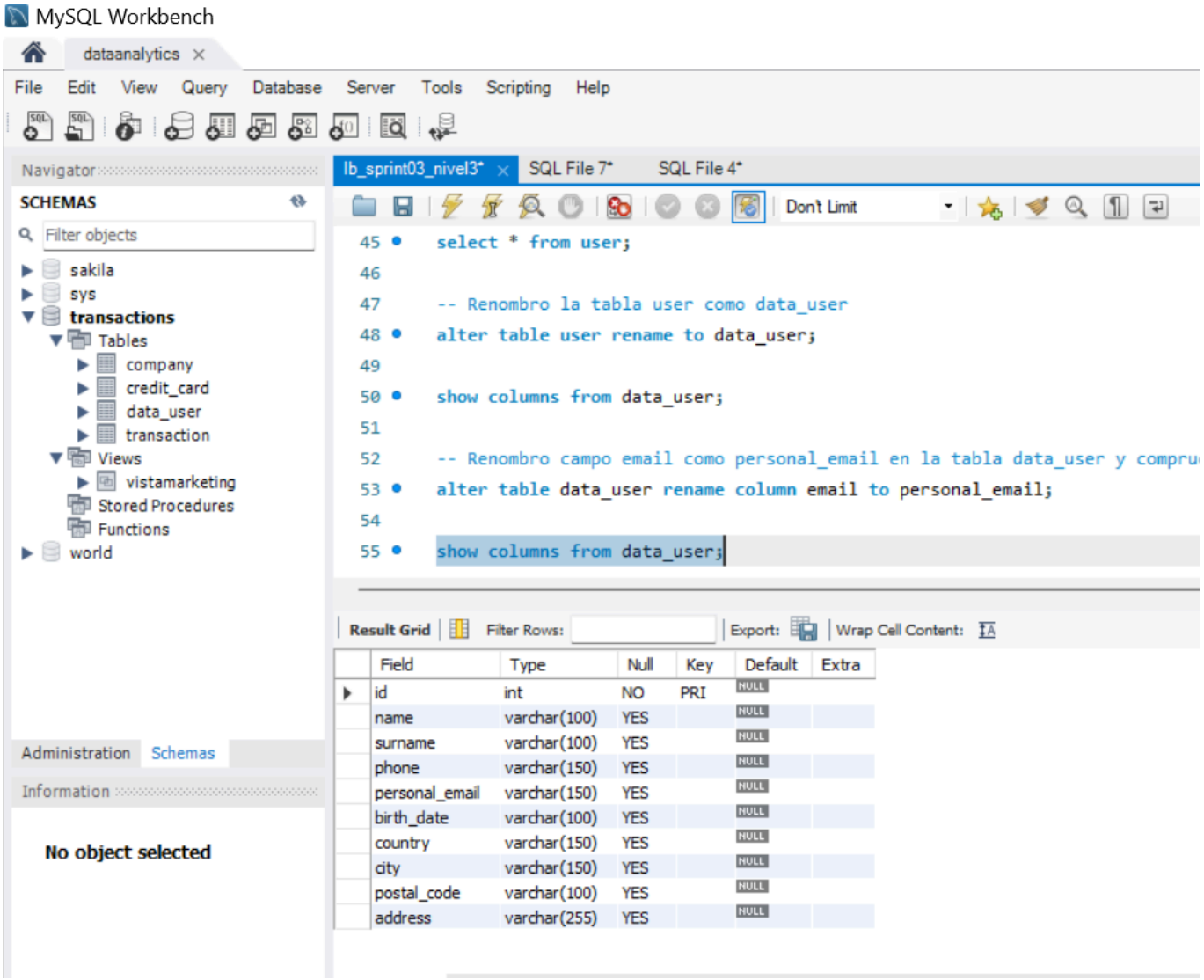
Campos de data_user ANTES de modificar nombre de campo email



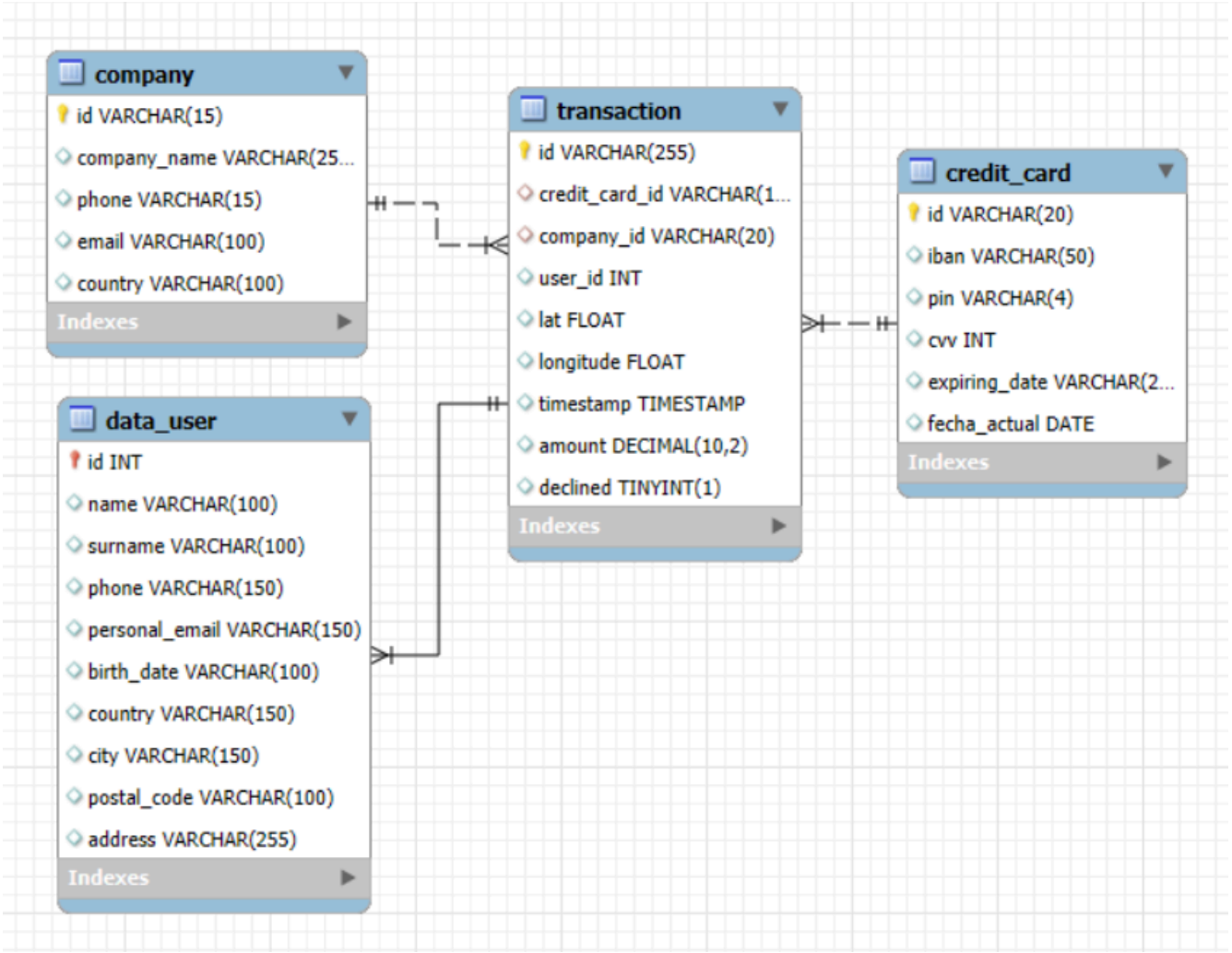
Renombro campo email como personal_email en la tabla data_user

```
alter table data_user rename column email to personal_email;
```

Campos de data_user DESPUES de modificar nombre de campo email



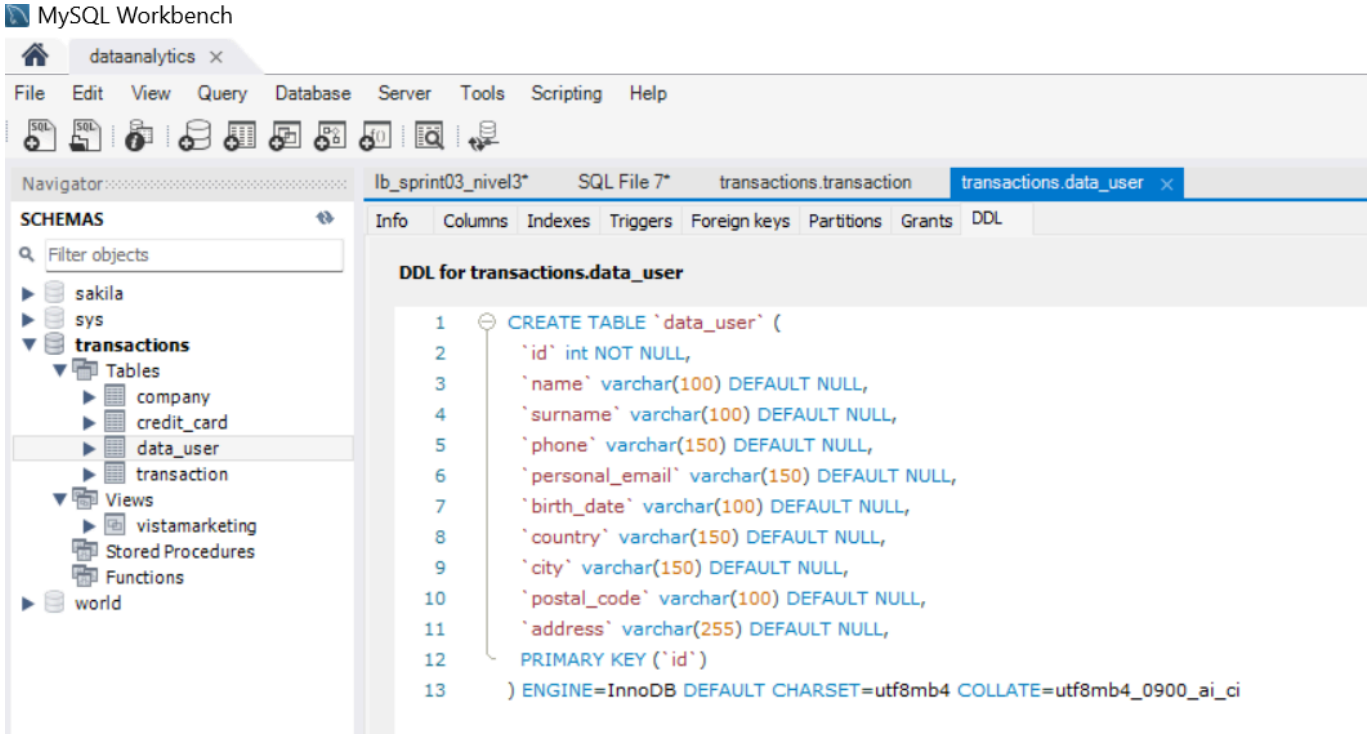
Veo el diagrama entidad relación de las tablas . Es incorrecto. La relación entre la tabla data_user y transaction no puede ser de varias a una. Una sola transacción no puede tener varios usuarios.
Tenemos que eliminar la foreign key que está en la tabla data_user.

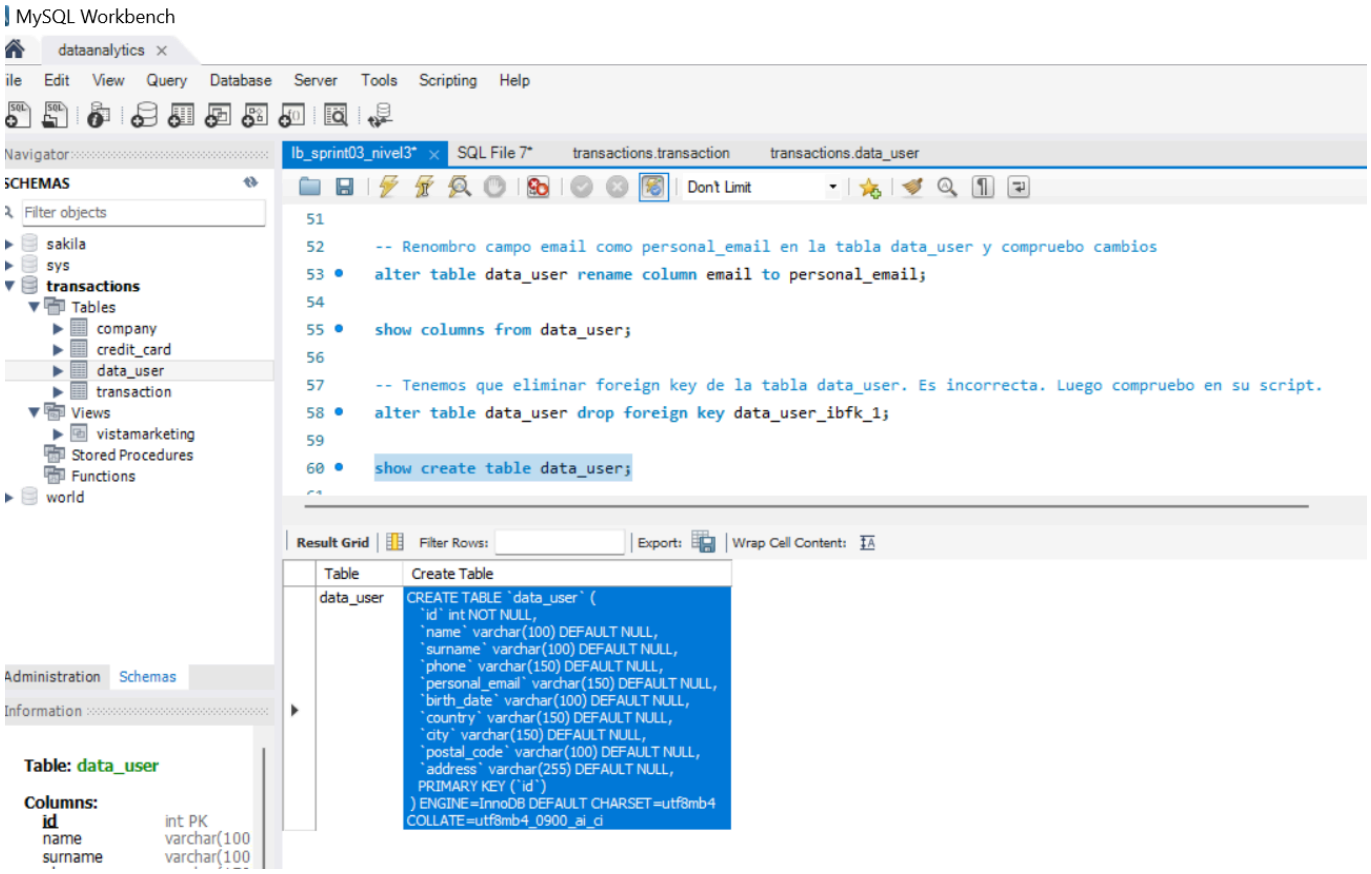


-- Tenemos que eliminar foreign key de la tabla data_user. Es incorrecta.

```
alter table data_user drop foreign key data_user_ibfk_1;
```

Luego compruebo en el script de la tabla data_user que efectivamente se eliminó esta foreign key.





Voy a crear la foreign key que relaciona a data_user con transaction en transaction

Modifico la table transaction para añadirle el id de la tabla data_user como Foreign Key

```
alter table transaction
add constraint fk_user_id
foreign key (user_id) references data_user(id);
```

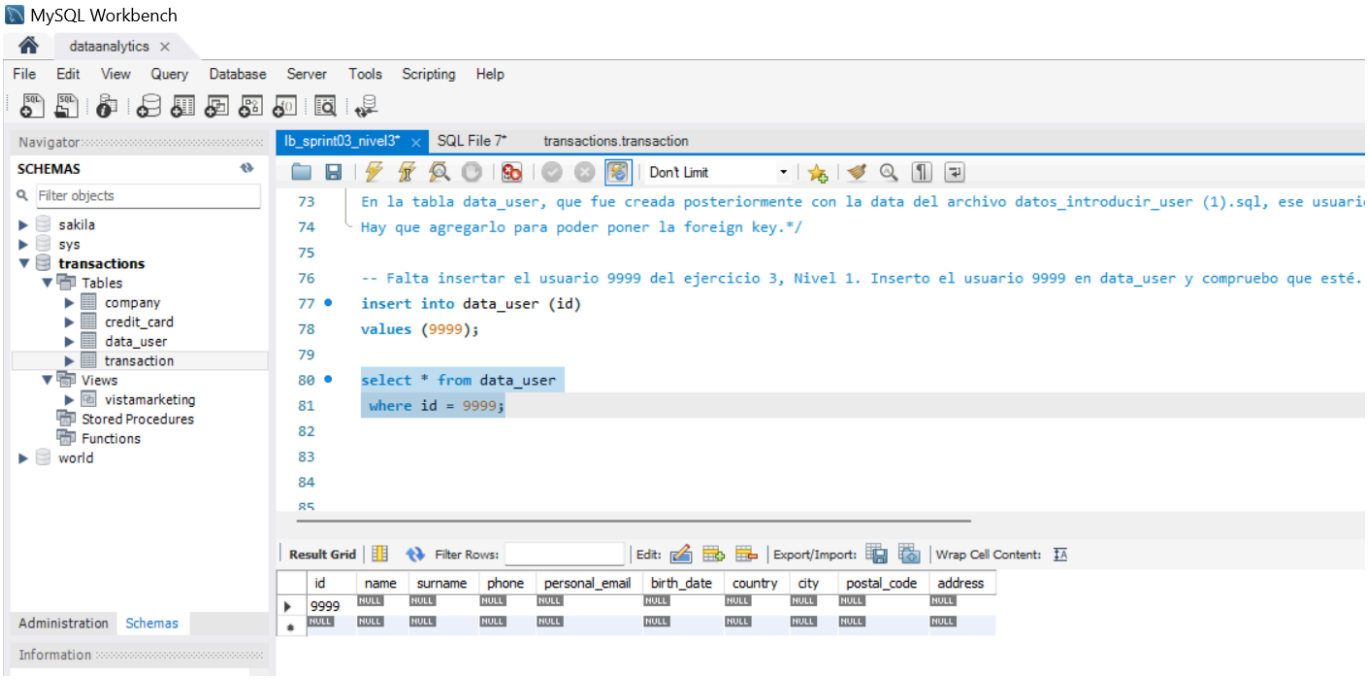
Mensaje de error:
Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (`transactions`.`#sql-15a0_6e`, CONSTRAINT `fk_user_id` FOREIGN KEY (`user_id`) REFERENCES `data_user` (`id`))

No se puede agregar la clave foránea porque algunos valores en la columna `user_id` de la tabla `transaction` no coinciden con ningún valor en la columna `id` de la tabla `data_user`.

Esto se debe a que hay valores en el campo user_id de la tabla transaction que no coinciden con el id de la tabla data_user. En la tabla transaction existe el usuario 9999 que agregamos en un ejercicio anterior. En la tabla data_user, que fue creada posteriormente con la data del archivo datos_introducir_user (1).sql, ese usuario no está. Hay que agregarlo para poder poner la foreign key en la tabla transaction.

Insertamos el usuario 9999 del ejercicio 3, Nivel 1. Inserto el usuario 9999 en data_user y compruebo que esté.

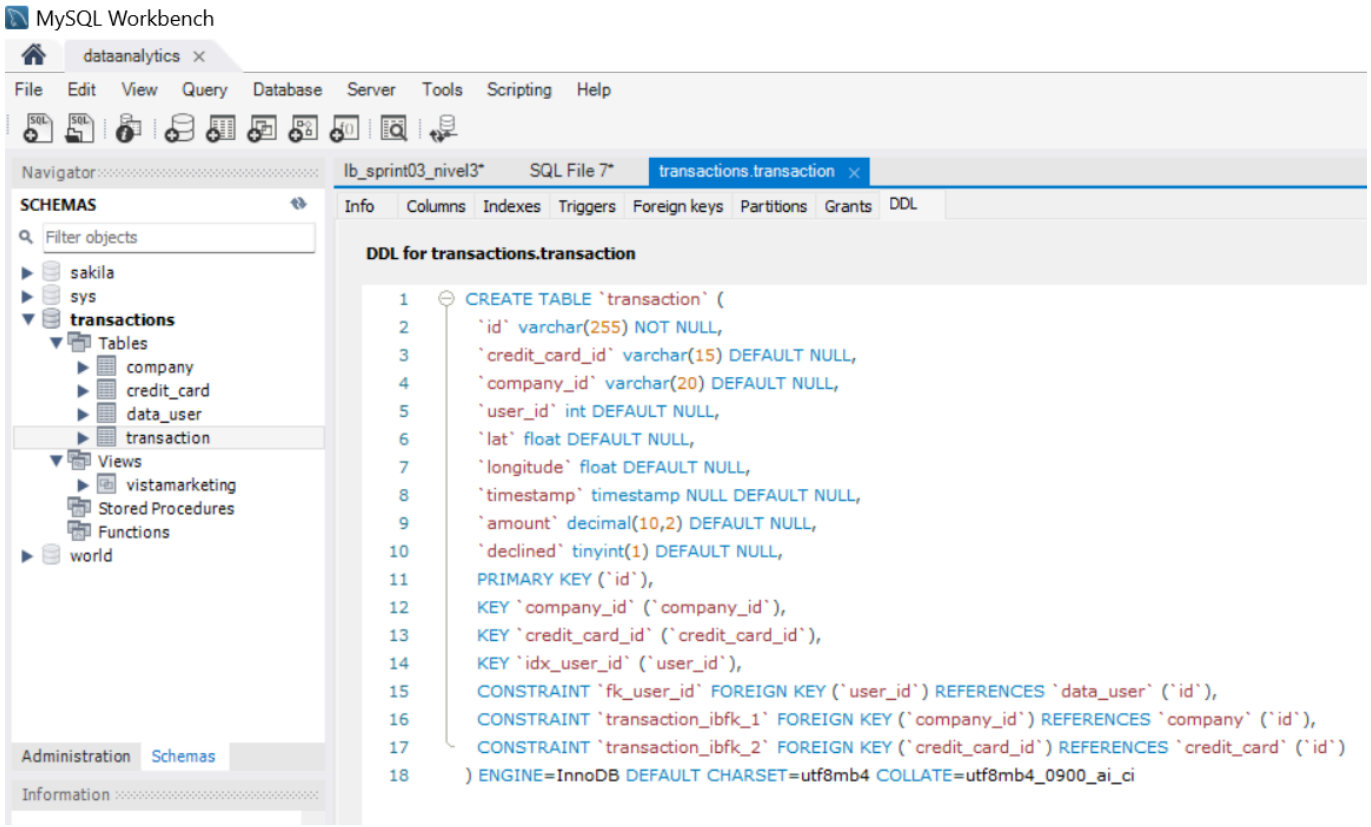
```
insert into data_user (id)
values (9999);

select * from data_user
where id = 9999;
```



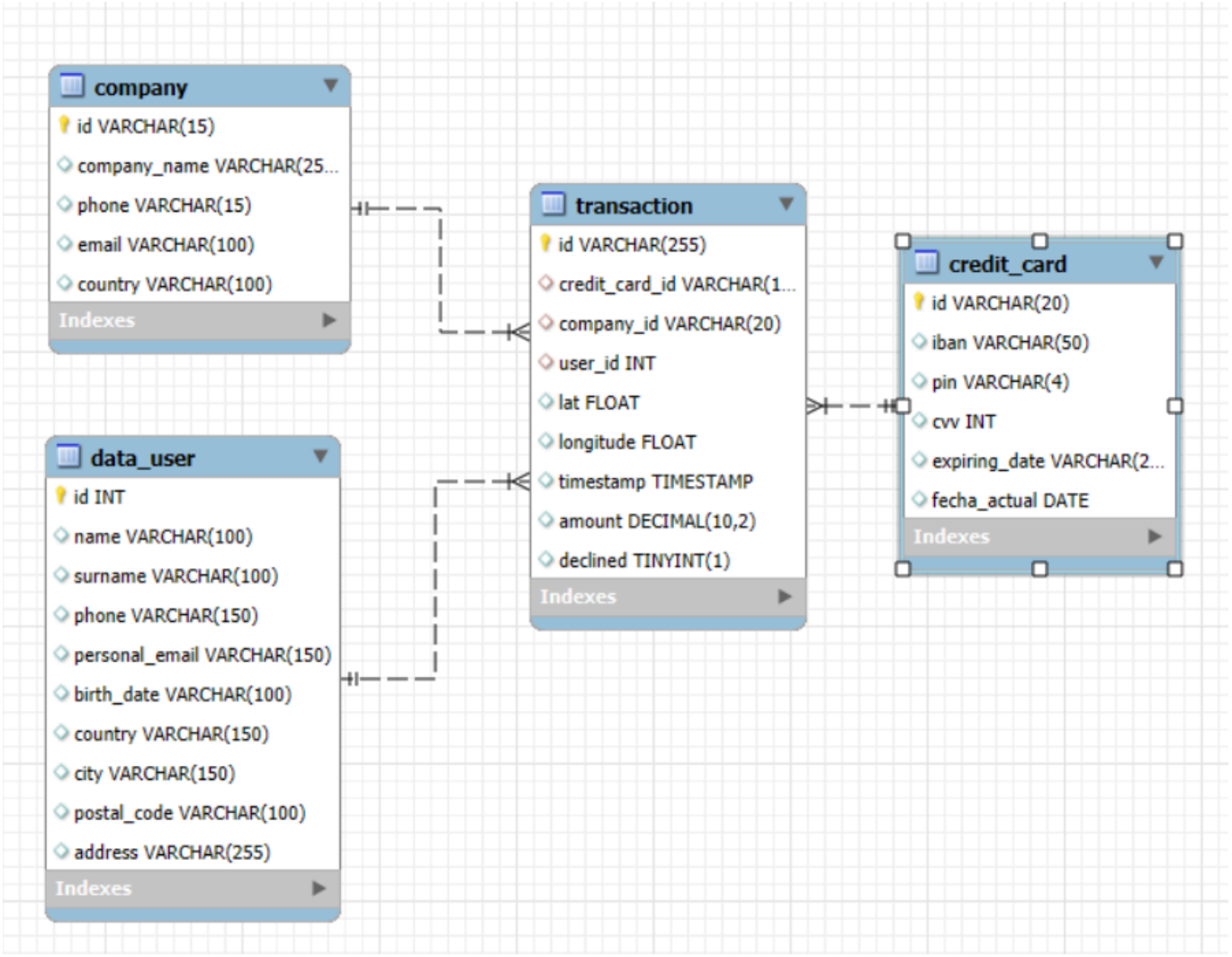
Modifico la table transaction para añadirle el id de la tabla data_user como Foreign Key

```
alter table transaction
add constraint fk_user_id
foreign key (user_id) references data_user(id);
```

Reviso en el script y si se ha creado.



Voy a ver el diagrama entidad relación para comprobar que todo esté OK



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.

```
create view informetecnico as
select t.id as transaction_id, du.name , du.surname, cc.iban, c.company_name
from transaction as t
join company as c
on c.id = t.company_id
join data_user as du
on du.id = t.user_id
join credit_card as cc
on cc.id = t.credit_card_id
order by t.id desc;
```