Tarea S3.01 Manipulación de tablas

Alumna: Cristiane de Souza da Silva
05/2024

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.

1) Creación de la tabla credit_card

```
create table if not exists credit_card (
        id varchar(50),
        iban varchar(50),
        pan varchar(100),
        pin varchar(5),
        cvv int ,
        expiring_date varchar(20),
        primary key (id)

);

-- create a constraint  foreign key
```

100% 3:26

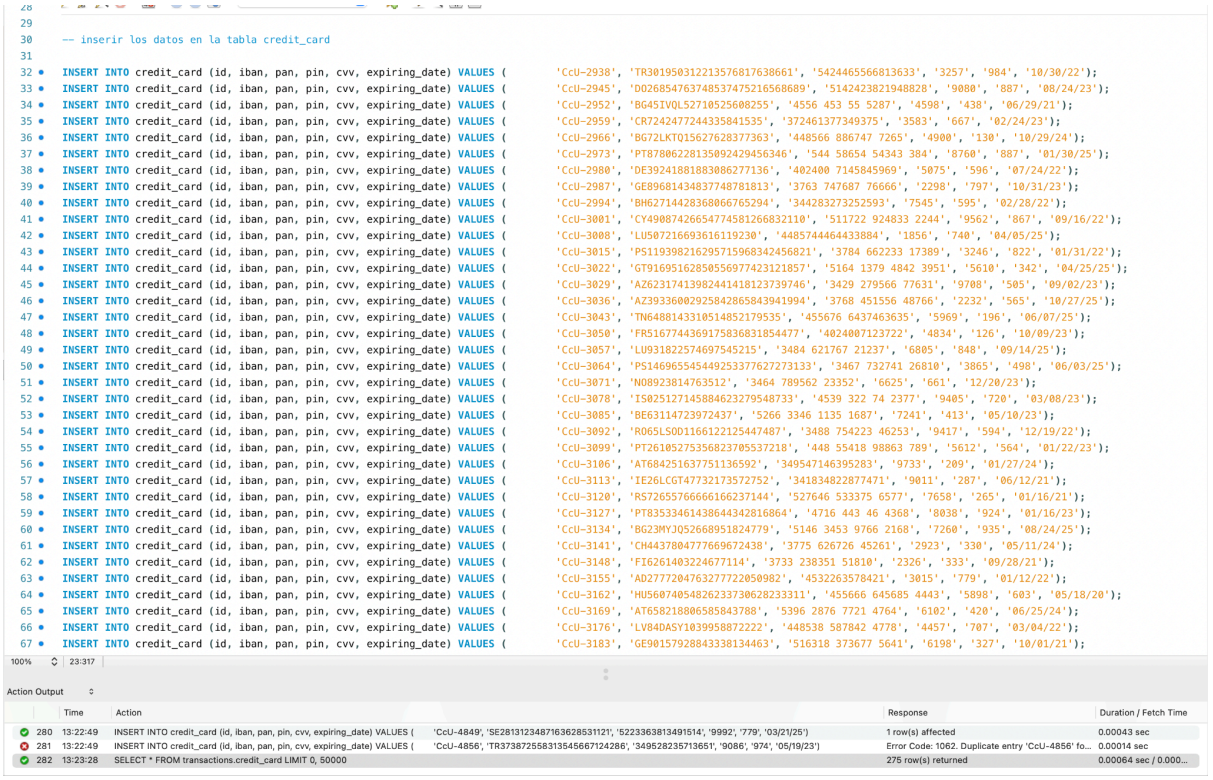
Action Output

	Time	Action			Response	Duration / Fetch Time
1	13:19:18	create table if not exists credit_card (	id varchar(50),	iban varchar(50),...	Error Code: 1046. No database selected Select the de...	0.00033 sec
2	13:19:28	create table if not exists credit_card (	id varchar(50),	iban varchar(50),...	Error Code: 1046. No database selected Select the de...	0.00023 sec
3	13:19:35	create table if not exists credit_card (	id varchar(50),	iban varchar(50),...	0 row(s) affected	0.0072 sec

La tabla credit_card fue generada a partir del script elaborado por mí. La tabla credit_card tiene seis columnas, dentro de ellas, una primary key (pk), **id.**

2) Inserción de los datos para la tabla credit_card

Los datos que se insertarán en la tabla provienen del archivo *'datos_introducir_credit.sql'* descargado en la página web del script 3 del curso de análisis de datos.



```
-- inserir los datos en la tabla credit_card

INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-2938', 'TR301950312213576817638661', '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-2952', 'BG45IVQL52710525608255', '4556 453 55 5287', '4598', '438', '06/29/21');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-2959', 'CR72424772443358415 35', '372461377349375', '3583', '667', '02/24/23');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-2966', 'BG72LKTQ15627628377363', '448566 886747 7265', '4900', '130', '10/29/24');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-2973', 'PT87806228135092429456346', '544 58654 54343 384', '8760', '887', '01/30/25');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-2980', 'DE39241881883086277136', '402400 7145845969', '5075', '596', '07/24/22');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-2987', 'GE89681434837748781813', '3763 747687 76666', '2298', '797', '10/31/23');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-2994', 'BH6271442836806676 5294', '344283273252593', '7545', '595', '02/28/22');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3001', 'CY490874266547745812668321 10', '511722 924833 2244', '9562', '867', '09/16/22');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3008', 'LU507216693616119230', '4485744464433884', '1856', '740', '04/05/25');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3015', 'PS11939821629571596834245682 1', '3784 662233 17389', '3246', '822', '01/31/22');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3022', 'GT91695162850556977423121857', '5164 1379 4842 3951', '5610', '342', '04/25/25');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3029', 'AZ623174139824414181 23739746', '3429 279566 77631', '9708', '505', '09/02/23');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3036', 'AZ393360029258428658439419 94', '3768 451556 48766', '2232', '565', '10/27/25');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3043', 'TN648814331051485217 9535', '455676 6437463635', '5969', '196', '06/07/25');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3050', 'FR5167744369175836831854477', '4024007123722', '4834', '126', '10/09/23');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3057', 'LU931822574697545215', '3484 621767 21237', '6805', '848', '09/14/25');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3064', 'PS14696554544925337762727 3133', '3467 732741 26810', '3865', '498', '06/03/25');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3071', 'NO8923814763512', '3464 789562 23352', '6625', '661', '12/20/23');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3078', 'IS025127145884623279548733', '4539 322 74 2377', '9405', '720', '03/08/23');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3085', 'BE63114723972437', '5266 3346 1135 1687', '7241', '413', '05/10/23');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3092', 'RO65LSOD1166122125447487', '3488 754223 46253', '9417', '594', '12/19/22');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3099', 'PT261052753568237055372 18', '448 55418 98863 789', '5612', '564', '01/22/23');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3106', 'AT6842516377511365 92', '3495471463952 83', '9733', '209', '01/27/24');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3113', 'IE26LCGT47732173572752', '341834822877471', '9011', '287', '06/12/21');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3120', 'RS7265576666616623 7144', '527646 533375 6577', '7658', '265', '01/16/21');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3127', 'PT835334614386443428 16864', '4716 443 46 4368', '8038', '924', '01/16/23');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3134', 'BG23MYJQ526689519 24779', '5146 3453 9766 2168', '7260', '935', '08/24/25');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3141', 'CH443780477766967 2438', '3775 626726 45261', '2923', '330', '05/11/24');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3148', 'FI6261403224677114', '3733 238351 51810', '2326', '333', '09/28/21');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3155', 'AD2777204763277722050982', '4532263578421', '3015', '779', '01/12/22');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3162', 'HU56074054826233730628233311', '455666 645685 4443', '5898', '603', '05/18/20');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3169', 'AT6582188065858437 88', '5396 2876 7721 4764', '6102', '420', '06/25/24');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3176', 'LV84DASY1039958872222', '448538 587842 4778', '4457', '707', '03/04/22');
INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES (  'CcU-3183', 'GE90157928843338134463', '516318 373677 5641', '6198', '327', '10/01/21');
```

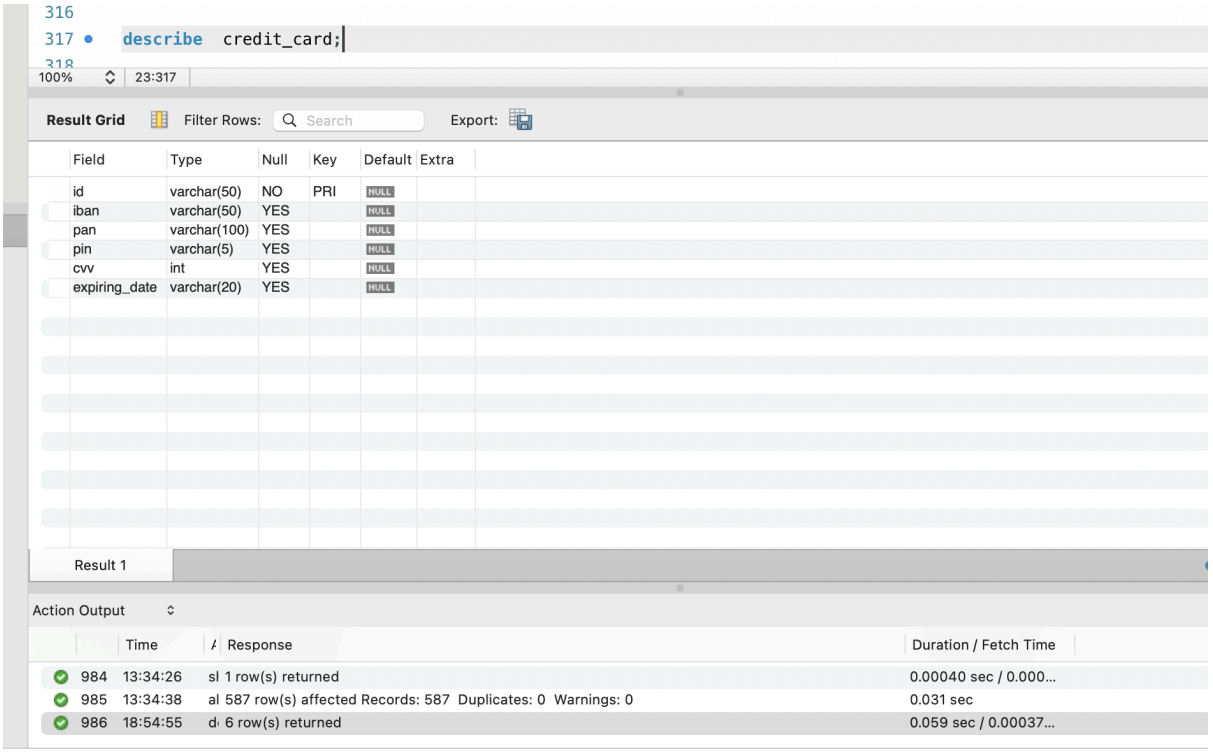
100% 23:317

Action Output

	Time	Action	Response	Duration / Fetch Time
280	13:22:49	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4849', 'SE2813123487163628531121', '52233638134915 14', '9992', '779', '03/21/25')	1 row(s) affected	0.00043 sec
281	13:22:49	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcU-4856', 'TR373872558313545667124286', '34952823571365 1', '9086', '974', '05/19/23')	Error Code: 1062. Duplicate entry 'CcU-4856' fo...	0.00014 sec
282	13:23:28	SELECT * FROM transactions.credit_card LIMIT 0, 50000	275 row(s) returned	0.00064 sec / 0.000...

MySQL solo acepta el formato ' YYYY-MM-DD ' como fecha. Por esta razón, la columna **expiring_date** se asignó como VARCHAR (carácter variable) para que se pudieran ingresar datos. Posteriormente, se cambiará el formato de la fecha.

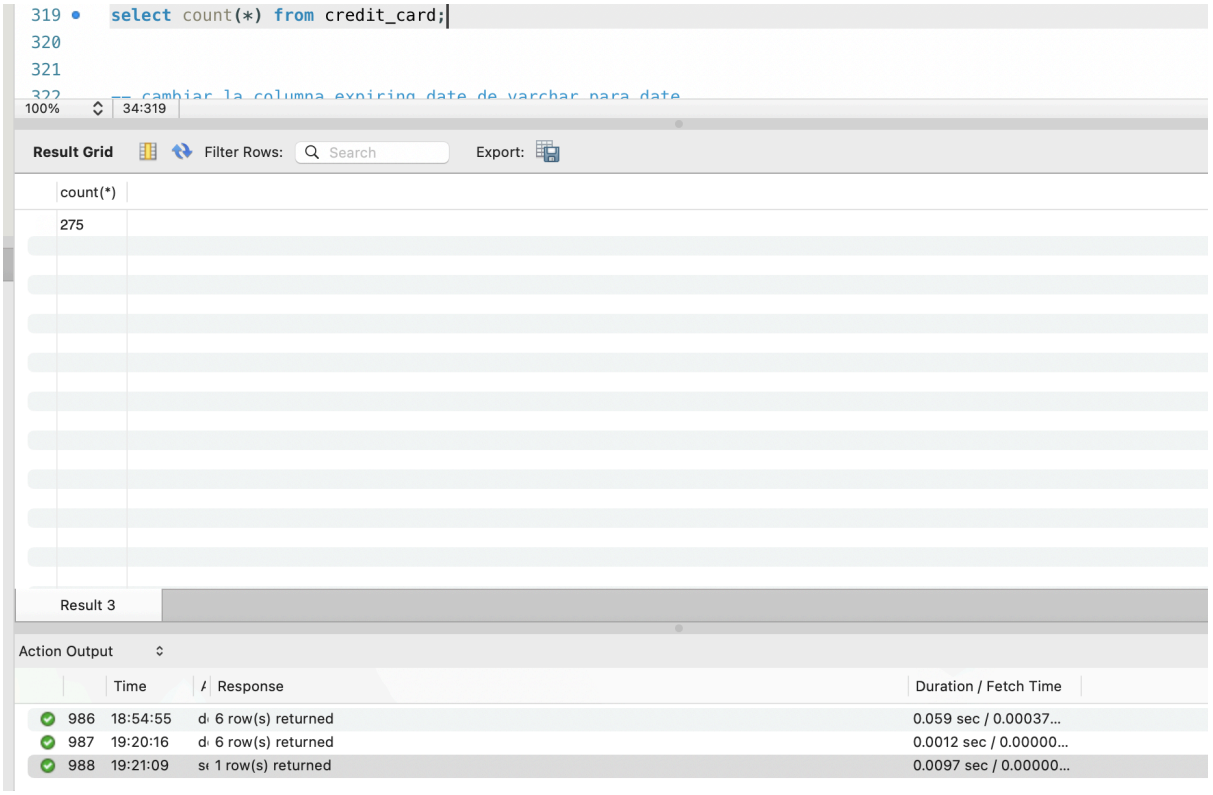
3) Descripción de la tabla tras inserción de los datos originales



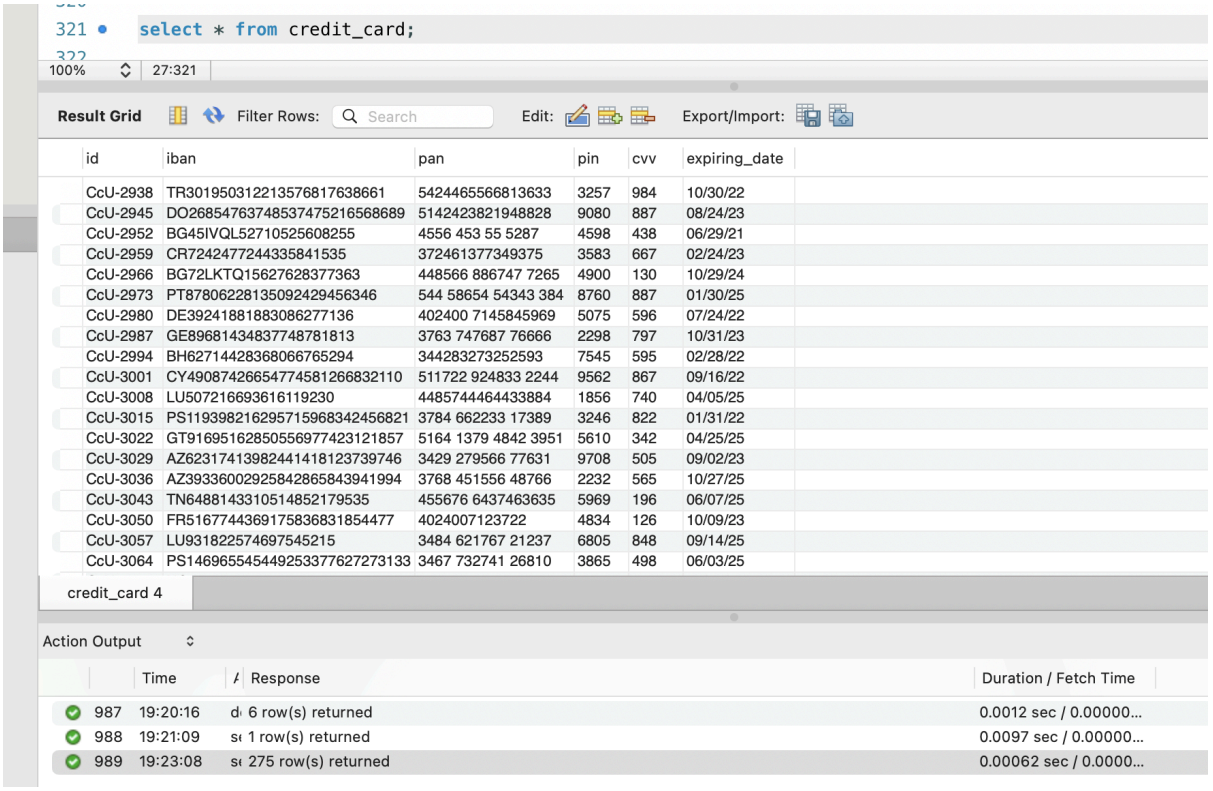
4) Creación de la constraint foreign key relacionada con la tabla **transaction**

La foreign key (fk) será creada para establecer una relación de 1 a N con la tabla **transaction**. La tabla **transaction** será alterada y la fk será añadida en ella. Esta relación de 1 (desde la tabla **credit_card**) a N (tabla **transaction**) significa que 1 tarjeta de crédito puede hacer varias transacciones, pero una transacción es hecha por una tarjeta de crédito.

5) Hay 275 filas distintas en la tabla credit_card.



6) Datos introducidos en la tabla **credit_card**



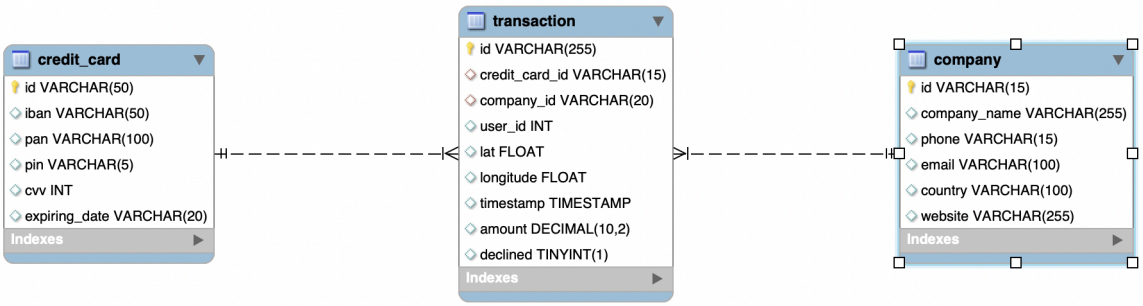
7) Diagrama **company-transaction-credit_card**

El diagrama creado en este ejercicio, **company-transaction-credit_card,** enseña la relación entre las tablas **credit_card**, **transaction** y **company**.

La tabla **credit_card**, creada en este ejercicio, tiene el campo **id** como primary key y es parte de la tabla **transaction** como foreign key (**credit_card_id**), estableciendo la relación entre ambas.

Esta relación es de 1 para N, o sea, una tarjeta de crédito puede hacer varias transacciones, pero una transacción es hecha por una única tarjeta de crédito.

De la misma manera, la tabla **company** tiene el campo **id** como primary key y es parte de la tabla **transaction** como foreign key (**company_id**), también estableciendo la relación entre ambas. Las tablas company y credit_card no están conectadas directamente. La relación es de 1 para N, o sea, una compañía puede hacer varias transacciones, pero una transacción es hecha por una única compañía.



El formato de la fecha será cambiado por ’ YYYY-MM-DD ‘. Para eso, utilicé la función **str_to_date** para cambio del formato mm/dd/yy para yyyy-mm-dd. A continuación, utilize la función date para cambiar el formato, lo que puede ser visto en la imagen abajo.

8) *Cambio de la columna "expiring_date" de VARCHAR para DATE*

```
330 •  describe credit_card;
```

100% 23:330

Result Grid Filter Rows: Search Export:

Field	Type	Null	Key	Default	Extra
id	varchar(50)	NO	PRI	NULL	
iban	varchar(50)	YES		NULL	
pan	varchar(100)	YES		NULL	
pin	varchar(5)	YES		NULL	
cvv	int	YES		NULL	
expiring_date	date	YES		NULL	

Result 5

Action Output

	Time	Response	Duration / Fetch Time
984	13:34:26	sl 1 row(s) returned	0.00040 sec / 0.000...
985	13:34:38	al 587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0	0.031 sec
986	18:54:55	d 6 row(s) returned	0.059 sec / 0.00037...
987	19:20:16	d 6 row(s) returned	0.0012 sec / 0.00000...
988	19:21:09	s 1 row(s) returned	0.0097 sec / 0.00000...
989	19:23:08	s 275 row(s) returned	0.00062 sec / 0.0000...
990	19:27:36	u 275 row(s) affected Rows matched: 275 Changed: 275 Warnings: 0	0.021 sec
991	19:27:50	al 275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0	0.030 sec
992	19:31:21	d 6 row(s) returned	0.0012 sec / 0.00001...

```
332 •  select * from credit_card;
333
```

100% 27:332

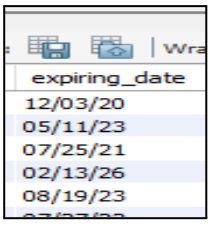
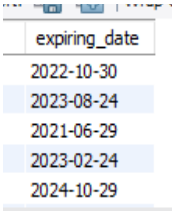
Result Grid Filter Rows: Search Edit: Export/Import:

id	iban	pan	pin	cvv	expiring_date
CcU-2938	TR301950312213576817638661	5424465566813633	3257	984	2022-10-30
CcU-2945	DO28547637485374752165686 89	5142423821948828	9080	887	2023-08-24
CcU-2952	BG45IVQL52710525608255	4556 453 55 5287	4598	438	2021-06-29
CcU-2959	CR7242477244335841535	372461377349375	3583	667	2023-02-24
CcU-2966	BG72LKTQ15627628377363	448566 886747 7265	4900	130	2024-10-29
CcU-2973	PT87806228135092429456346	544 58654 54343 384	8760	887	2025-01-30
CcU-2980	DE39241881883086277136	402400 7145845969	5075	596	2022-07-24
CcU-2987	GE89681434837748781813	3763 747687 76666	2298	797	2023-10-31
CcU-2994	BH62714428368066765294	344283273252593	7545	595	2022-02-28
CcU-3001	CY490874266547745812668 32110	511722 924833 2244	9562	867	2022-09-16
CcU-3008	LU507216693616119230	4485744464433884	1856	740	2025-04-05
CcU-3015	PS119398216295715968342456821	3784 662233 17389	3246	822	2022-01-31

credit_card 6

Action Output

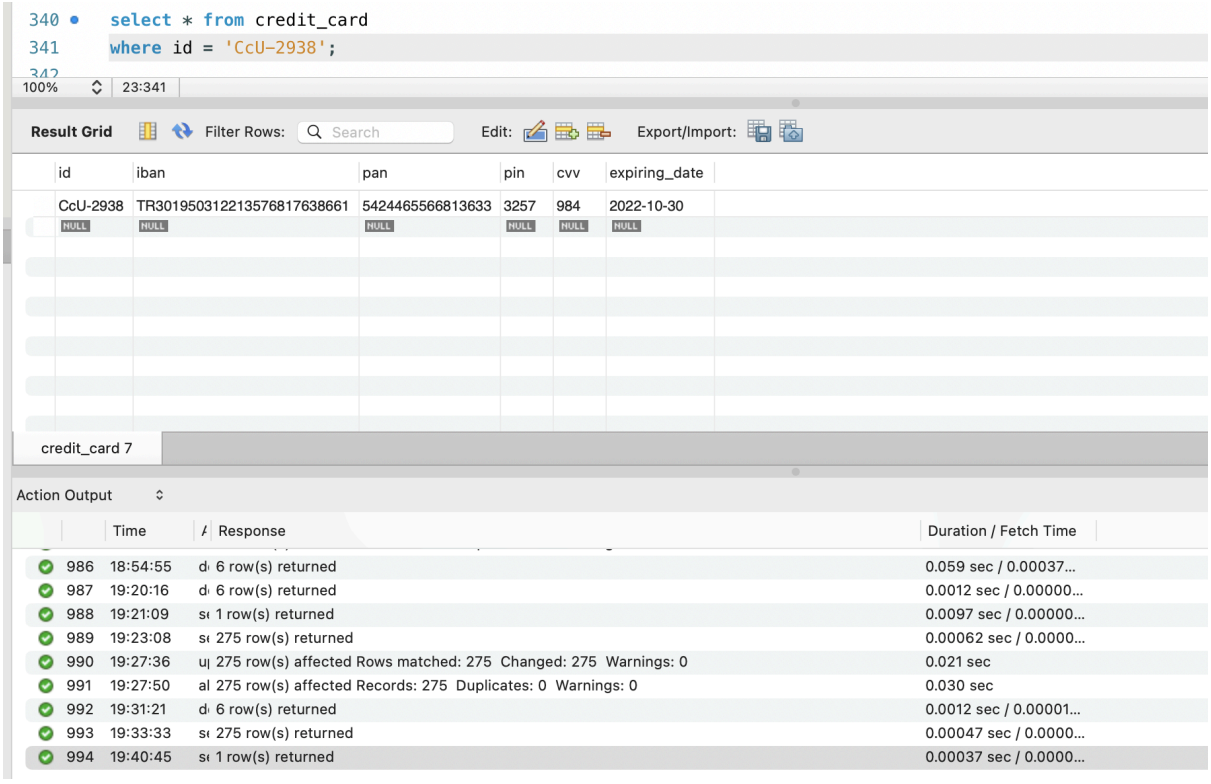
	Time	Response	Duration / Fetch Time
985	13:34:38	al 587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0	0.031 sec
986	18:54:55	d 6 row(s) returned	0.059 sec / 0.00037...
987	19:20:16	d 6 row(s) returned	0.0012 sec / 0.00000...
988	19:21:09	s 1 row(s) returned	0.0097 sec / 0.00000...
989	19:23:08	s 275 row(s) returned	0.00062 sec / 0.0000...
990	19:27:36	u 275 row(s) affected Rows matched: 275 Changed: 275 Warnings: 0	0.021 sec
991	19:27:50	al 275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0	0.030 sec
992	19:31:21	d 6 row(s) returned	0.0012 sec / 0.00001...
993	19:33:33	s 275 row(s) returned	0.00047 sec / 0.0000...

Antes (como VARCHAR)	Después (como DATE)
	

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.

1) Información sobre la tarjeta de ID CcU-2938 antes del cambio solicitado



2) Información sobre la tarjeta de ID CcU-2938 <u>después</u> del cambio solicitado

```
343 •  update credit_card   -- actualizando la tabla
344    set iban = 'R323456312213576817699999'   -- cambio de la cuenta del usuario
345    where id = 'CcU-2938';
346
347 •  select *    -- comprobando el cambio
348    from credit_card
349    where id = 'CcU-2938';
```

100% 23:349

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

id	iban	pan	pin	cvv	expiring_date
CcU-2938	R323456312213576817699999	5424465566813633	3257	984	2022-10-30
NULL	NULL	NULL	NULL	NULL	NULL

credit_card 8

Action Output

	Time	Response	Duration / Fetch Time
988	19:21:09	s 1 row(s) returned	0.0097 sec / 0.00000...
989	19:23:08	s 275 row(s) returned	0.00062 sec / 0.0000...
990	19:27:36	u 275 row(s) affected Rows matched: 275 Changed: 275 Warnings: 0	0.021 sec
991	19:27:50	al 275 row(s) affected Records: 275 Duplicates: 0 Warnings: 0	0.030 sec
992	19:31:21	d 6 row(s) returned	0.0012 sec / 0.00001...
993	19:33:33	s 275 row(s) returned	0.00047 sec / 0.0000...
994	19:40:45	s 1 row(s) returned	0.00037 sec / 0.0000...
995	19:44:04	u 1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.00097 sec
996	19:44:10	s 1 row(s) returned	0.00033 sec / 0.0000...

Ejercicio 3

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

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

"Una relación fk involucra una tabla *parent* que contiene los valores de la columna inicial y una tabla *child* con valores de columna que se refieren a los valores de la columna de la tabla *parent*." (citación: https://dev.mysql.com/doc/refman/8.0/en/create-table-foreign-keys.html)

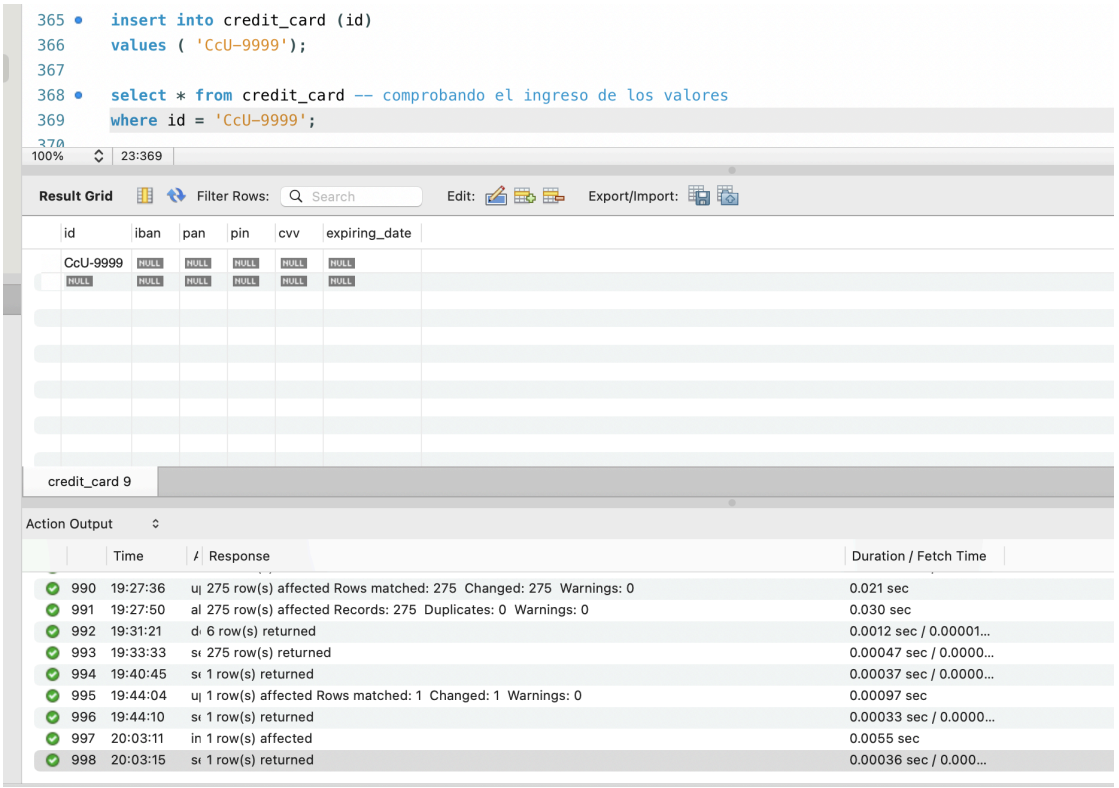
Una restricción de fk se define en la tabla *child*, como se hizo anteriormente en la tabla de **transacciones** en relación con las tablas **company** y **credit_card**.

Existe una relación establecida entre la **company** (*parent*) y la **transaction** (child), y las tablas credit_card (*parent*) y Transaction (child) debido a la restricción de su respectiva fk. Debido a esto, el cambio es realizado primero en las tablas **company** y **credit_card** para hacer la introducción de datos en sus respectivas columnas, como la pk. En otras palabras, si el intento de ingresar los datos se lleva a cabo primero en la tabla de transacciones (*child*), se produce un error en su ejecución.

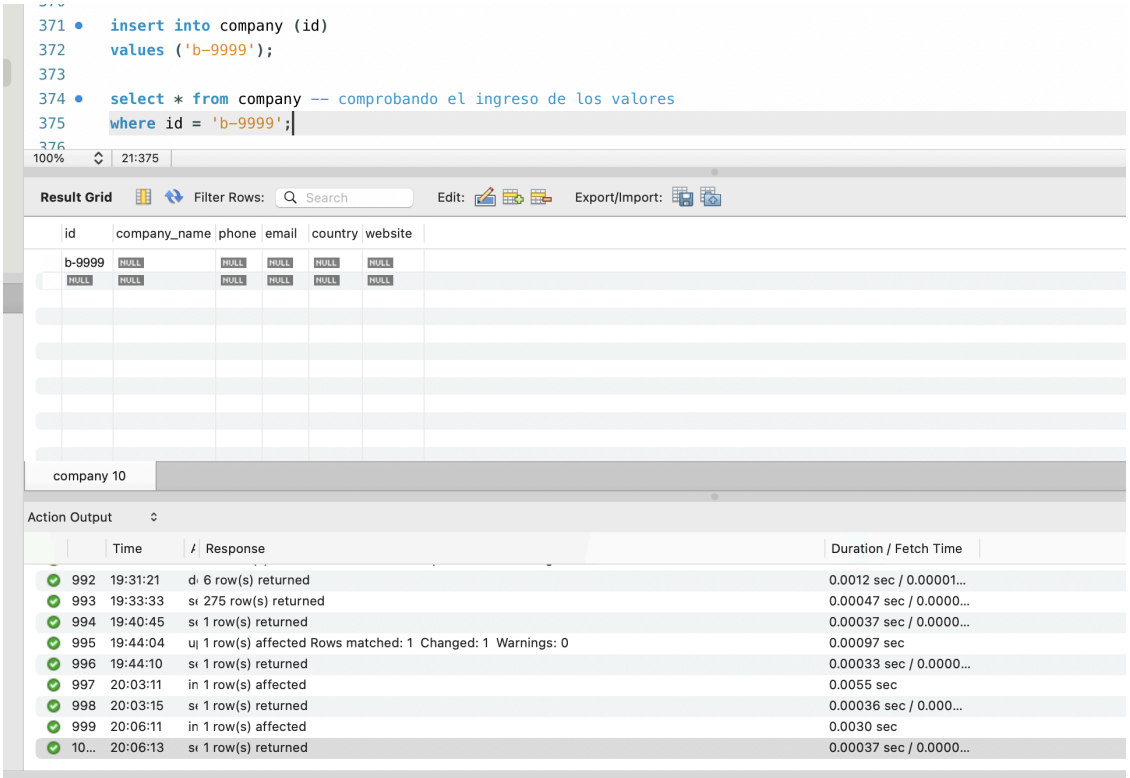
Antes de inserir los valores, voy a hacerlos en las tablas donde ellos son primary key:

Especialización en Análisis de Datos - IT ACADEMY - 2024

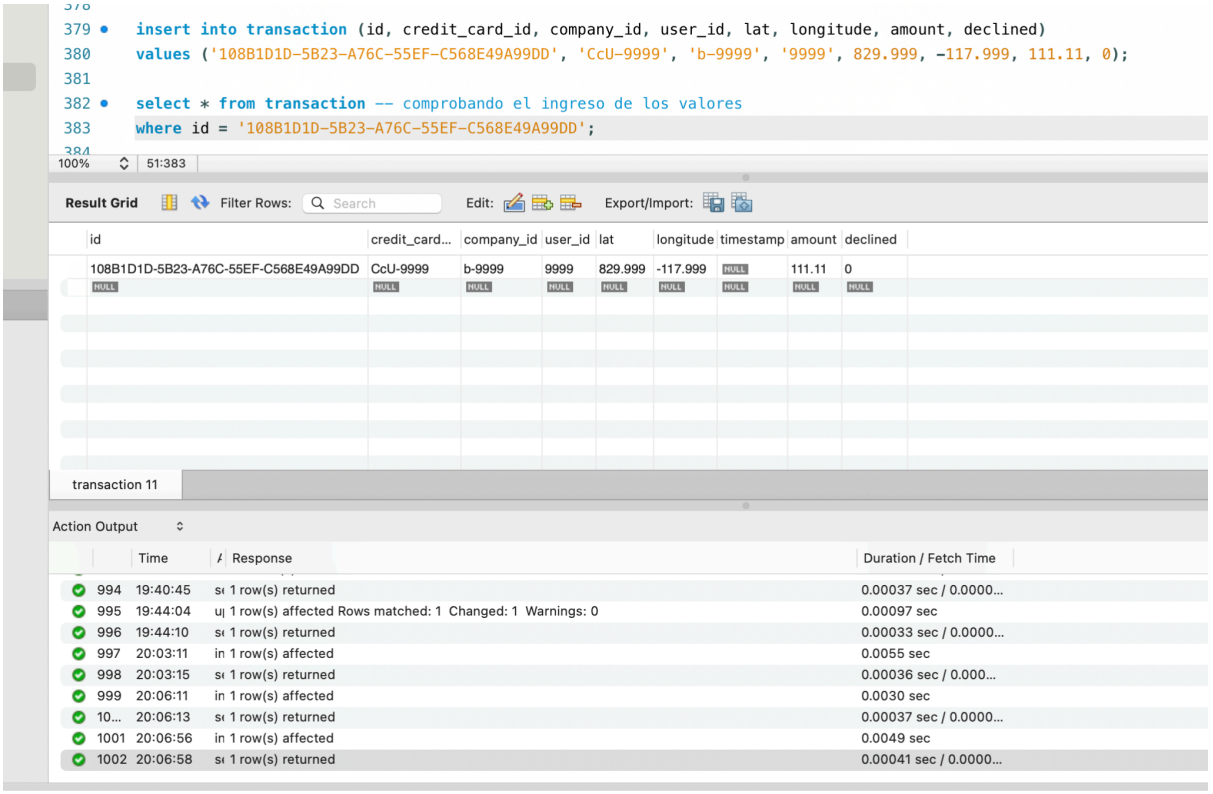
1) Inserción de la tarjeta de crédito CcU-99 en la tabla **credit_card**



2) Imagen 13 -inserción de la compañía b-9999 en la tabla **company**



3) Imagen 14 - inserción del nuevo usuario en la tabla **transaction.** La columna **timestamp** tiene el dato correspondiente como *NULL* porque no fue incluida en la inserción de datos.



```
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 -- comprobando el ingreso de los valores
where id = '108B1D1D-5B23-A76C-55EF-C568E49A99DD';
```

100% 51:383

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

id	credit_card...	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
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

transaction 11

Action Output

	Time	Response	Duration / Fetch Time
994	19:40:45	1 row(s) returned	0.00037 sec / 0.0000...
995	19:44:04	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.00097 sec
996	19:44:10	1 row(s) returned	0.00033 sec / 0.0000...
997	20:03:11	1 row(s) affected	0.0055 sec
998	20:03:15	1 row(s) returned	0.00036 sec / 0.000...
999	20:06:11	1 row(s) affected	0.0030 sec
10...	20:06:13	1 row(s) returned	0.00037 sec / 0.0000...
1001	20:06:56	1 row(s) affected	0.0049 sec
1002	20:06:58	1 row(s) returned	0.00041 sec / 0.0000...

Ejercicio 4

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

1) Tabla antes de la eliminación de la columna **pan**

```
394    -- verificar la tabla antes del cambio
395 •  select * from credit_card;
396
```

100% 27:395

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

id	iban	pan	pin	cvv	expiring_date
CcU-2938	R323456312213576817699999	5424465566813633	3257	984	2022-10-30
CcU-2945	DO2685476374853747521656868 9	5142423821948828	9080	887	2023-08-24
CcU-2952	BG45IVQL52710525608255	4556 453 55 5287	4598	438	2021-06-29
CcU-2959	CR7242477244335841535	372461377349375	3583	667	2023-02-24
CcU-2966	BG72LKTQ15627628377363	448566 886747 7265	4900	130	2024-10-29
CcU-2973	PT8780622813509242945634 6	544 58654 54343 384	8760	887	2025-01-30
CcU-2980	DE39241881883086277136	402400 7145845969	5075	596	2022-07-24
CcU-2987	GE896814348377487818 13	3763 747687 76666	2298	797	2023-10-31
CcU-2994	BH627144283680667652 94	344283273252593	7545	595	2022-02-28
CcU-3001	CY490874266547745812668321 10	511722 924833 2244	9562	867	2022-09-16
CcU-3008	LU507216693616119230	4485744464433884	1856	740	2025-04-05
CcU-3015	PS119398216295715968342456 821	3784 662233 17389	3246	822	2022-01-31
CcU-3022	GT916951628505569774231218 57	5164 1379 4842 3951	5610	342	2025-04-25
CcU-3029	AZ623174139824414181237397 46	3429 279566 77631	9708	505	2023-09-02
CcU-3036	AZ393360029258428656843941 994	3768 451556 48766	2232	565	2025-10-27
CcU-3043	TN648814331051485217953 5	455676 6437463635	5969	196	2025-06-07
CcU-3050	FR516774436917583683185447 7	4024007123722	4834	126	2023-10-09
CcU-3057	LU9318225746975452 15	3484 621767 21237	6805	848	2025-09-14
CcU-3064	PS146965545449253377627273 133	3467 732741 26810	3865	498	2025-06-03
CcU-3071	NO8923814763512	3464 789562 23352	6625	661	2023-12-20

credit_card 13

Action Output

	Time	Response	Duration / Fetch Time
996	19:44:10	s 1 row(s) returned	0.00033 sec / 0.0000...
997	20:03:11	in 1 row(s) affected	0.0055 sec
998	20:03:15	s 1 row(s) returned	0.00036 sec / 0.000...
999	20:06:11	in 1 row(s) affected	0.0030 sec
10...	20:06:13	s 1 row(s) returned	0.00037 sec / 0.0000...
1001	20:06:56	in 1 row(s) affected	0.0049 sec
1002	20:06:58	s 1 row(s) returned	0.00041 sec / 0.0000...
10...	20:08:32	s 1 row(s) returned	0.0021 sec / 0.00000...
10...	20:13:50	s 276 row(s) returned	0.00048 sec / 0.000...

2) Tabla <u>después</u> de la eliminación de la columna **pan**

```
397 •  alter table credit_card
398    drop column pan;
399
400
401 •  select * from  credit_card; --  Verificar el cambio mostrando la estructura de la tabla.
402
403
```

100% 29:401

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

id	iban	pin	cvv	expiring_date
CcU-2938	R323456312213576817699999	3257	984	2022-10-30
CcU-2945	DO2685476374853747521656868 9	9080	887	2023-08-24
CcU-2952	BG45IVQL52710525608255	4598	438	2021-06-29
CcU-2959	CR7242477244335841535	3583	667	2023-02-24
CcU-2966	BG72LKTQ15627628377363	4900	130	2024-10-29
CcU-2973	PT878062281350924294563 46	8760	887	2025-01-30
CcU-2980	DE39241881883086277136	5075	596	2022-07-24
CcU-2987	GE896814348377487818 13	2298	797	2023-10-31
CcU-2994	BH62714428368066765294	7545	595	2022-02-28
CcU-3001	CY490874266547745812668321 10	9562	867	2022-09-16
CcU-3008	LU507216693616119230	1856	740	2025-04-05

credit_card 14

Action Output

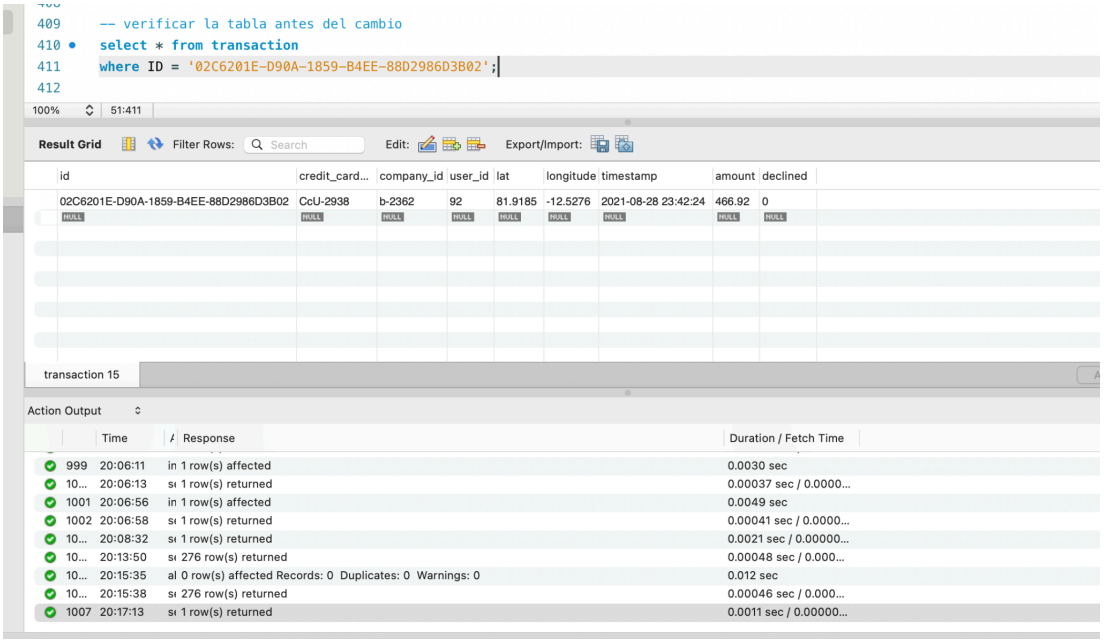
	Time	Response	Duration / Fetch Time
998	20:03:15	s 1 row(s) returned	0.00036 sec / 0.000...
999	20:06:11	in 1 row(s) affected	0.0030 sec
10...	20:06:13	s 1 row(s) returned	0.00037 sec / 0.0000...
1001	20:06:56	in 1 row(s) affected	0.0049 sec
1002	20:06:58	s 1 row(s) returned	0.00041 sec / 0.0000...
10...	20:08:32	s 1 row(s) returned	0.0021 sec / 0.00000...
10...	20:13:50	s 276 row(s) returned	0.00048 sec / 0.000...
10...	20:15:35	al 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.012 sec
10...	20:15:38	s 276 row(s) returned	0.00046 sec / 0.000...

Nivel 2

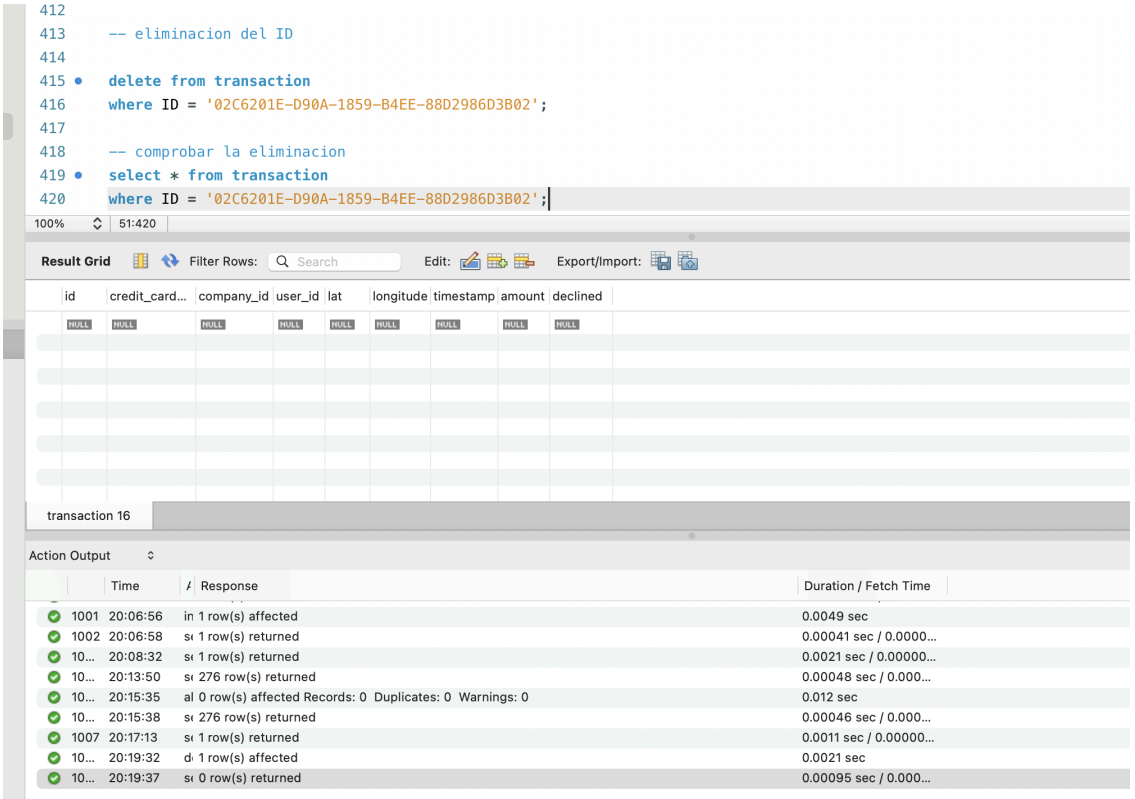
Ejercicio 1

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

1) Registro con ID antes de su eliminación



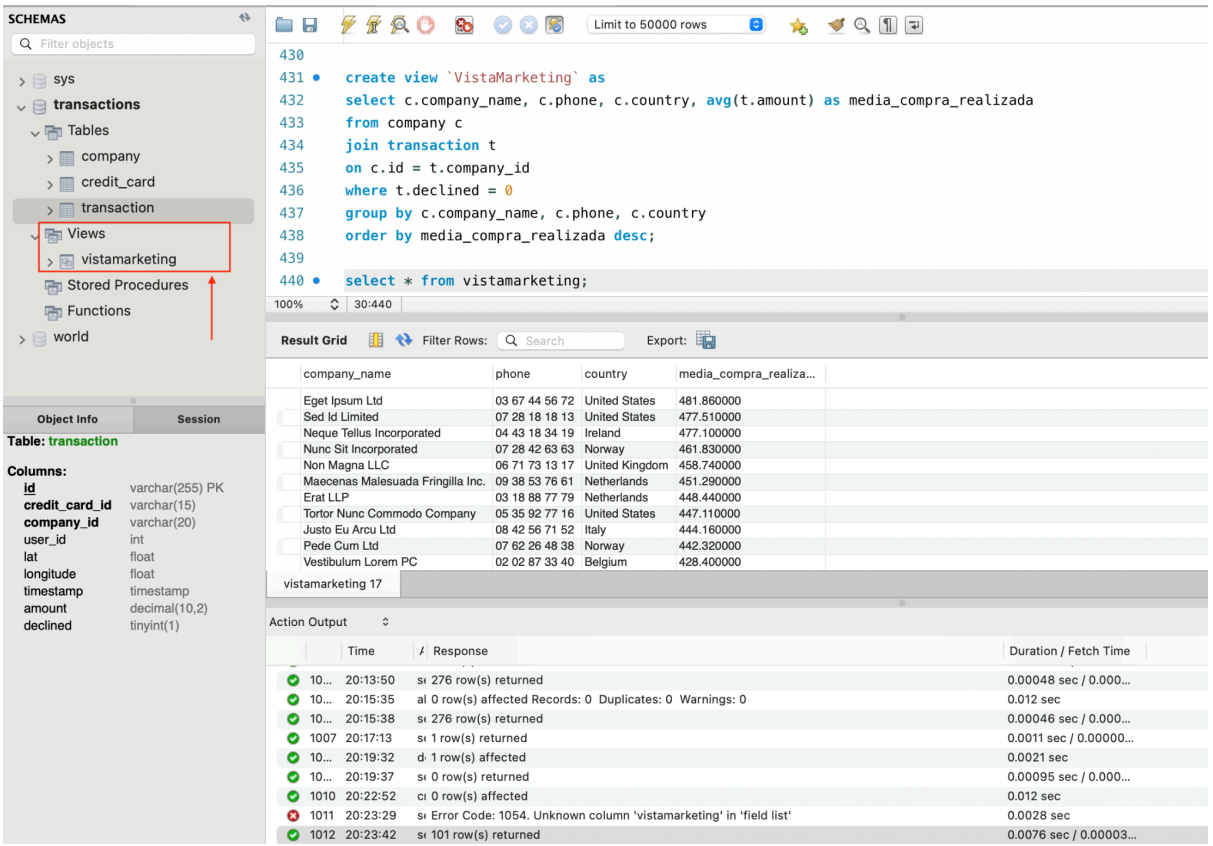
2) Registro con ID después de su eliminación



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 realizada por cada compañía. Presenta la vista creada, ordenando los datos de mayor a menor promedio de compra.

1) Creación de la vista vistamarketing



Ejercicio 3

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

2) VistaMArketing con filtro 'Germany'

```
443    -- Ejercicio 3
444    -- Filtra la vista VistaMarketing para mostrar sólo las compañías que tienen su país de residencia en "Germany"
445
446 •  select *
447    from vistamarketing
448    where country = 'Germany';
449
```

100% | 27:448

Result Grid | Filter Rows: Search | Export:

company_name	phone	country	media_compra_realiza...
Ac Industries	09 34 65 40 60	Germany	396.150000
Auctor Mauris Corp.	05 62 87 14 41	Germany	308.990000
Ac Fermentum Incorporated	06 85 56 52 33	Germany	293.570000
Aliquam PC	01 45 73 52 16	Germany	280.340000
Rutrum Non Inc.	02 66 31 61 09	Germany	266.900000
Nunc Interdum Incorporated	05 18 15 48 13	Germany	242.947692
Convallis In Incorporated	06 66 57 29 50	Germany	60.990000
Augue Foundation	06 88 43 15 63	Germany	15.050000

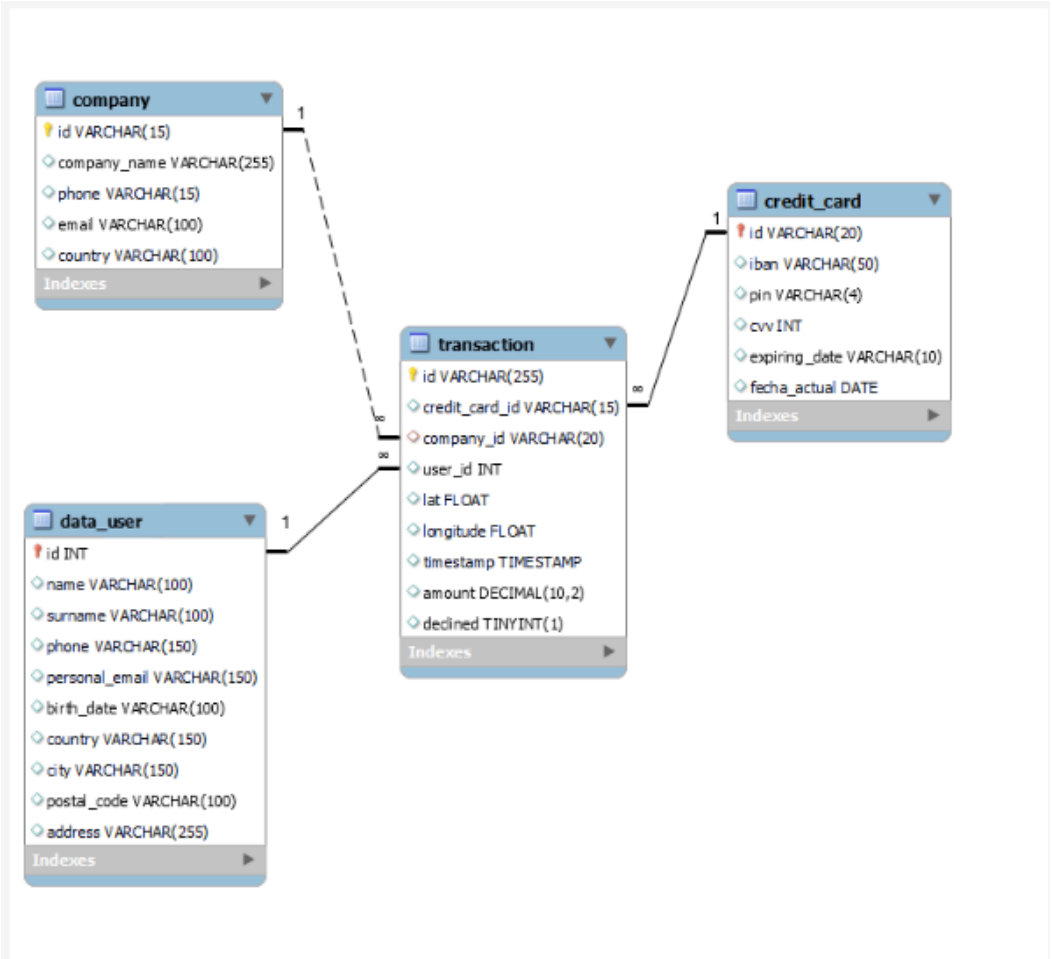
vistamarketing 18

Action Output

	Time	Response	Duration / Fetch Time
10...	20:15:35	al 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.012 sec
10...	20:15:38	s 276 row(s) returned	0.00046 sec / 0.000...
1007	20:17:13	s 1 row(s) returned	0.0011 sec / 0.00000...
10...	20:19:32	d 1 row(s) affected	0.0021 sec
10...	20:19:37	s 0 row(s) returned	0.00095 sec / 0.000...
1010	20:22:52	c 0 row(s) affected	0.012 sec
1011	20:23:29	s Error Code: 1054. Unknown column 'vistamarketing' in 'field list'	0.0028 sec
1012	20:23:42	s 101 row(s) returned	0.0076 sec / 0.00003...
1013	20:26:30	s 8 row(s) returned	0.0029 sec / 0.00001...

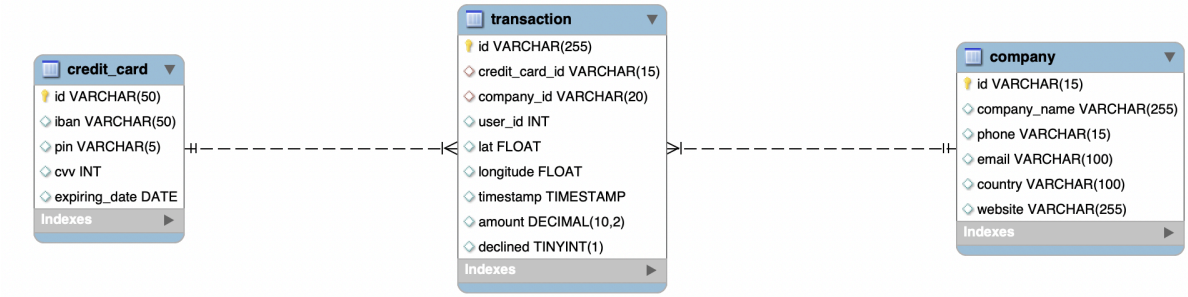
Nivel 3

La próxima semana tendré 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:



Especialización en Análisis de Datos - IT ACADEMY - 2024

1) Modelo Anterior



credit_card	
Columna	***datatype***
id	VARCHAR (50)
iban	VARCHAR (50)
pin	VARCHAR (5)
cvv	INT
expiring_date	DATE

transaction	
Columna	***datatype***
id	VARCHAR (255)
credit_card_id	VARCHAR (15)
company_id	VARCHAR (20)
user_id	INT
lat	FLOAT
longitude	FLOAT
timestamp	TIMESTAMP
amount	DECIMAL(10,2)
declined	TINYINT(1)

company	
Columna	***datatype***
id	VARCHAR (15)
company_name	VARCHAR (255)
phone	VARCHAR (15)
email	VARCHAR (100)
country	VARCHAR (100)
website	VARCHAR (255)

2) Creación de la tabla **user**

El script de la tabla proviene del archivo *'estructura_datos_user.sql'* descargado en la página web del script 3 del curso de análisis de datos.

```
457        -- Creamos la tabla user
458
459 •   CREATE INDEX idx_user_id ON transaction(user_id);
460
461 •  CREATE TABLE IF NOT EXISTS user (
462            id INT PRIMARY KEY,
463            name VARCHAR(100),
464            surname VARCHAR(100),
465            phone VARCHAR(150),
466            email VARCHAR(150),
467            birth_date VARCHAR(100),
468            country VARCHAR(150),
469            city VARCHAR(150),
470            postal_code VARCHAR(100),
471            address VARCHAR(255),
472            FOREIGN KEY(id) REFERENCES transaction(user_id)
473        );
474
475
```

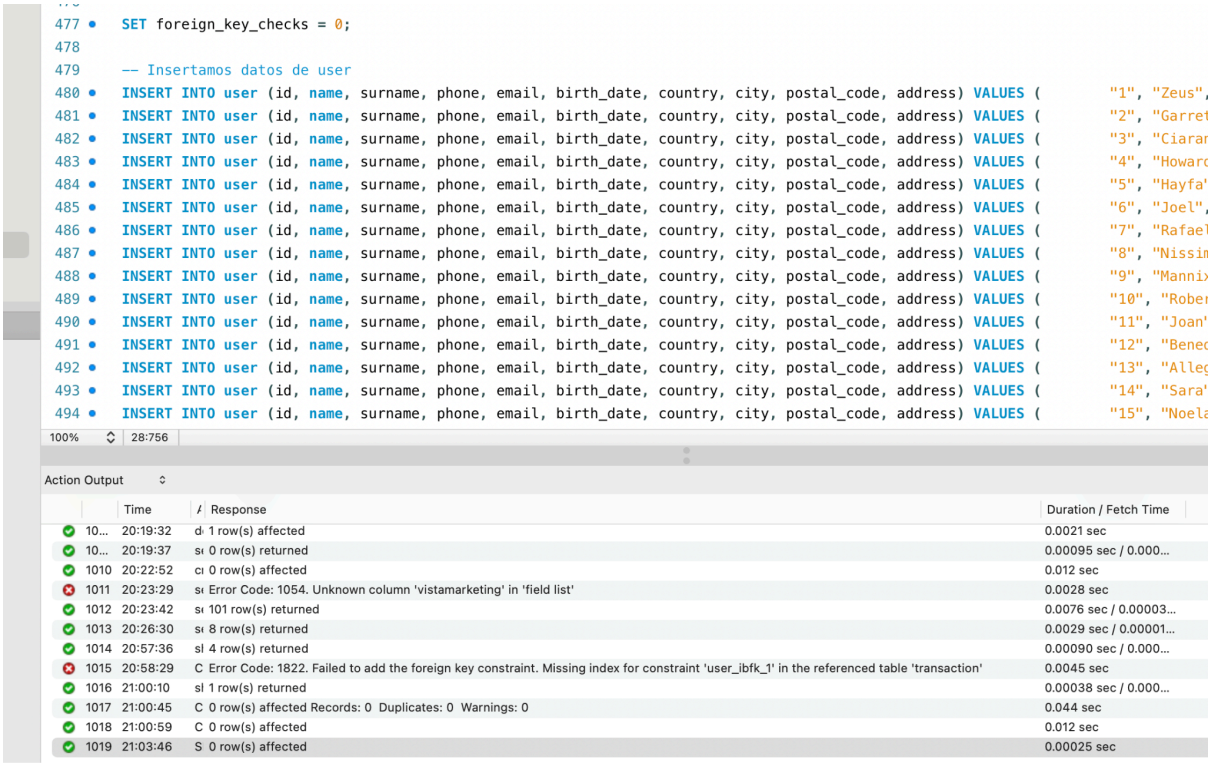
100% 7:473

Action Output

	Time	Response	Duration / Fetch Time
1007	20:17:13	s 1 row(s) returned	0.0011 sec / 0.00000...
10...	20:19:32	d 1 row(s) affected	0.0021 sec
10...	20:19:37	s 0 row(s) returned	0.00095 sec / 0.000...
1010	20:22:52	c 0 row(s) affected	0.012 sec
1011	20:23:29	s Error Code: 1054. Unknown column 'vistamarketing' in 'field list'	0.0028 sec
1012	20:23:42	s 101 row(s) returned	0.0076 sec / 0.00003...
1013	20:26:30	s 8 row(s) returned	0.0029 sec / 0.00001...
1014	20:57:36	s 4 row(s) returned	0.00090 sec / 0.000...
1015	20:58:29	C Error Code: 1822. Failed to add the foreign key constraint. Missing index for constraint 'user_ibfk_1' in the referenced table 'transaction'	0.0045 sec
1016	21:00:10	s 1 row(s) returned	0.00038 sec / 0.000...
1017	21:00:45	C 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.044 sec
1018	21:00:59	C 0 row(s) affected	0.012 sec

3) Inserción de los datos en la tabla **user**

Los datos que se insertarán en la tabla provienen del archivo *'datos_introducir_user(1).sql'* descargado en la página web del script 3 del curso de análisis de datos.



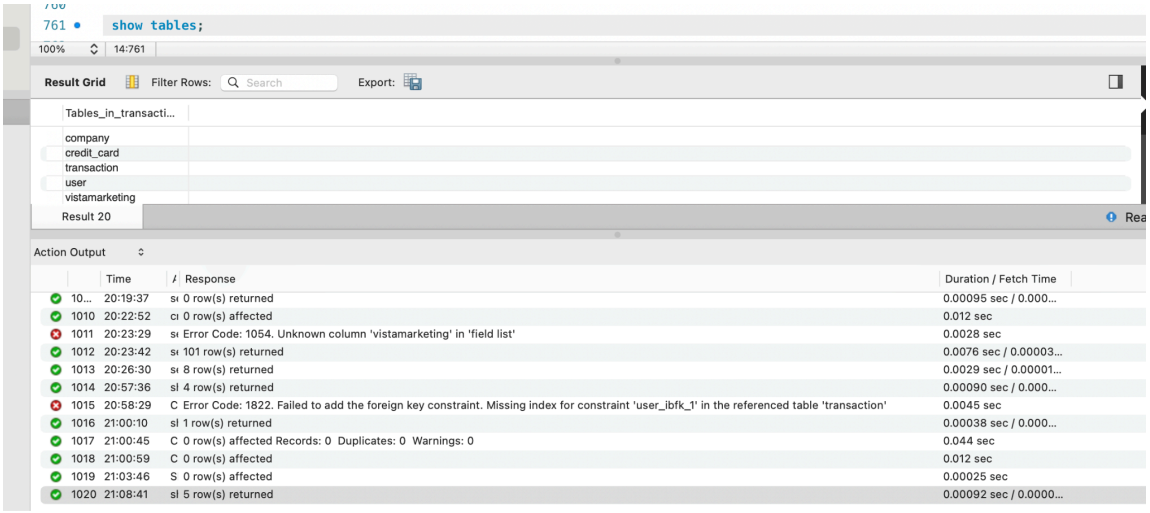
```
477 •  SET foreign_key_checks = 0;
478
479    -- Insertamos datos de user
480 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "1", "Zeus",
481 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "2", "Garret
482 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "3", "Ciarar
483 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "4", "Howarc
484 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "5", "Hayfa'
485 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "6", "Joel",
486 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "7", "Rafael
487 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "8", "Nissim
488 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (        "9", "Mannix
489 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "10", "Rober
490 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "11", "Joan'
491 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "12", "Benec
492 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "13", "Alleg
493 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "14", "Sara'
494 •  INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES (       "15", "Noela
```

100% 28:756

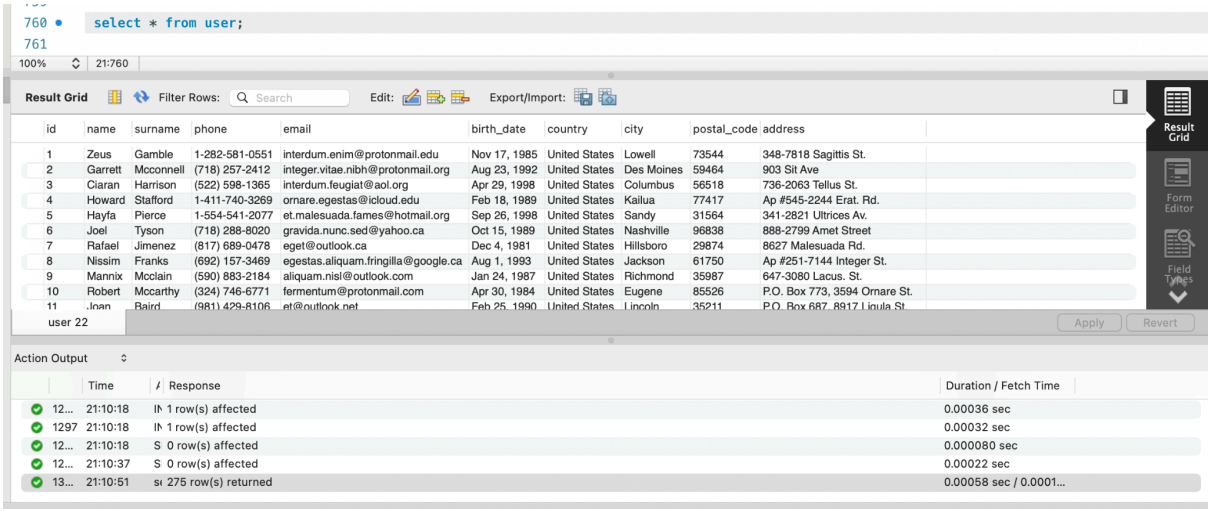
Action Output

	Time	Response	Duration / Fetch Time
✓ 10...	20:19:32	d 1 row(s) affected	0.0021 sec
✓ 10...	20:19:37	s 0 row(s) returned	0.00095 sec / 0.000...
✓ 1010	20:22:52	c 0 row(s) affected	0.012 sec
✗ 1011	20:23:29	s Error Code: 1054. Unknown column 'vistamarketing' in 'field list'	0.0028 sec
✓ 1012	20:23:42	s 101 row(s) returned	0.0076 sec / 0.00003...
✓ 1013	20:26:30	s 8 row(s) returned	0.0029 sec / 0.00001...
✓ 1014	20:57:36	s 4 row(s) returned	0.00090 sec / 0.000...
✗ 1015	20:58:29	C Error Code: 1822. Failed to add the foreign key constraint. Missing index for constraint 'user_ibfk_1' in the referenced table 'transaction'	0.0045 sec
✓ 1016	21:00:10	s 1 row(s) returned	0.00038 sec / 0.000...
✓ 1017	21:00:45	C 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.044 sec
✓ 1018	21:00:59	C 0 row(s) affected	0.012 sec
✓ 1019	21:03:46	S 0 row(s) affected	0.00025 sec

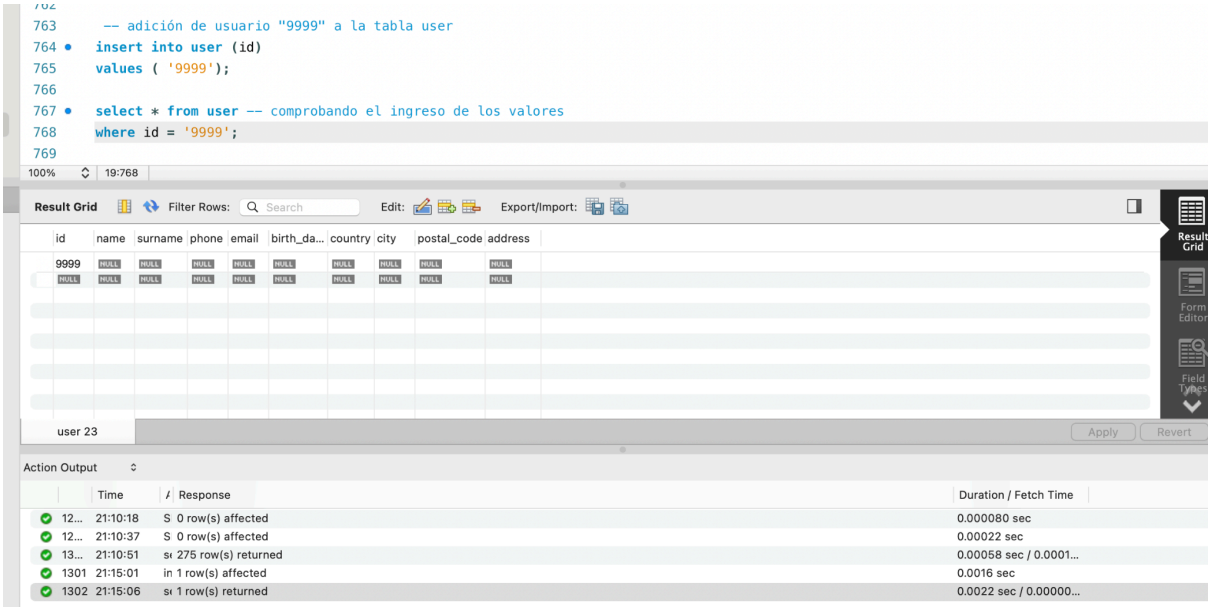
4) Tabla user en el database **transactions**



5) Tabla user con 275 filas



6) Adición de usuario "9999" en la tabla **user.**

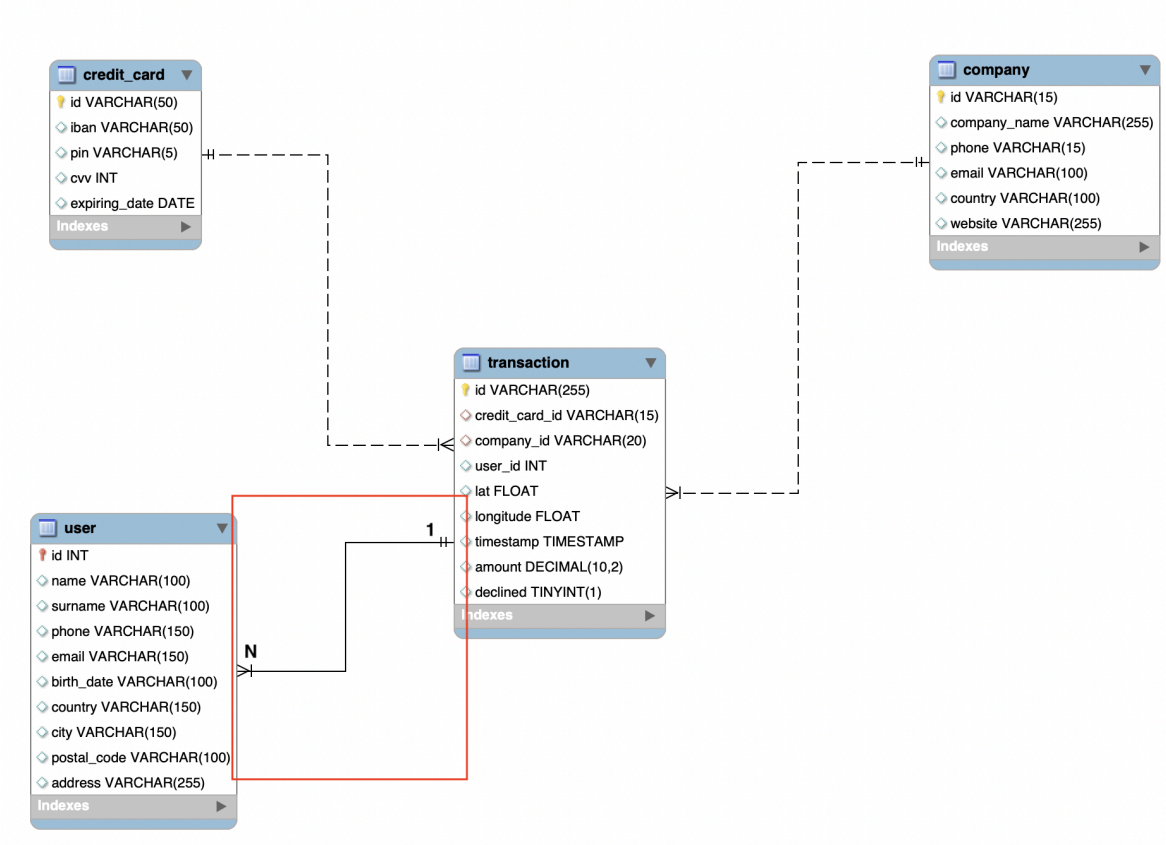


7) Elaboración del diagrama

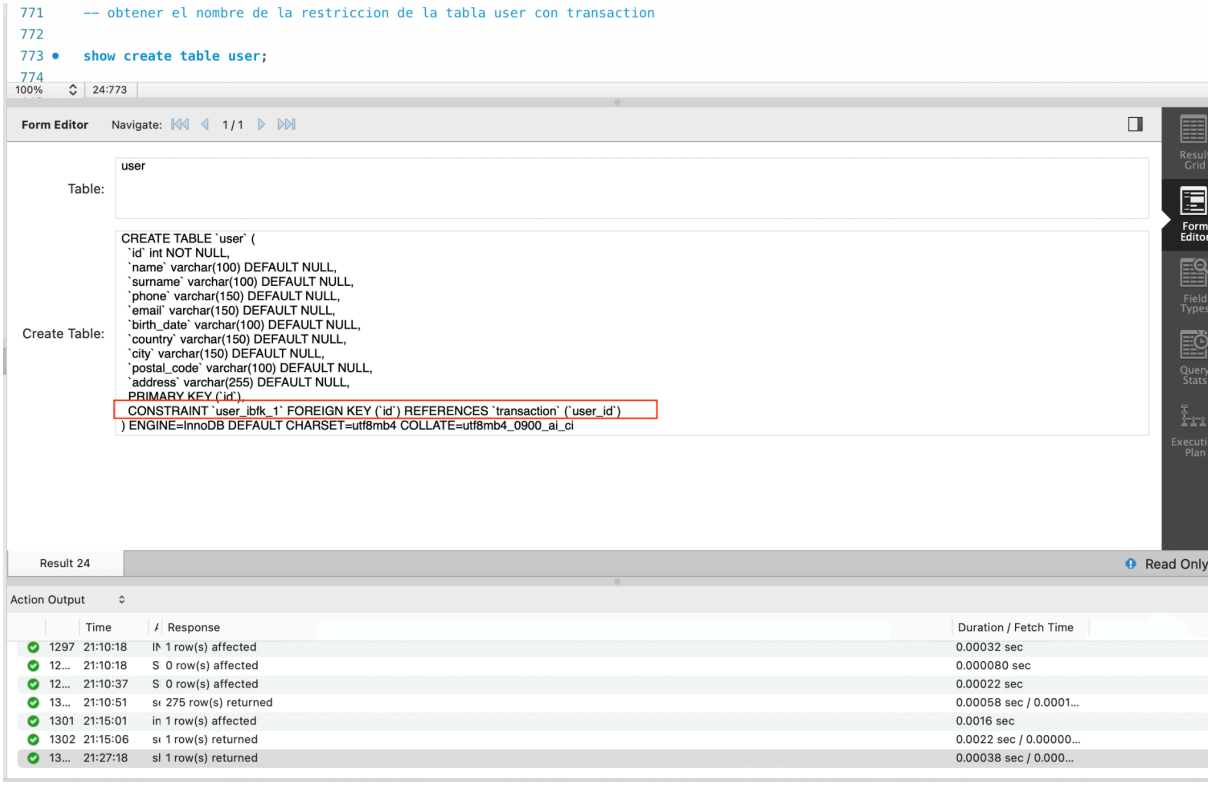
El script para la creación de la tabla user fue obtenido desde la página web del sprint 3. Allí ya había una relación entre las tablas **user** y **transaction** a través de la *Foreign Key* **id (user)** y **user_id(transaction)**. Sin embargo, de esta manera el **id** de la tabla **user** es utilizado como *primary key* y *foreign key* y la relación se establece como 1 para N , desde **transaction** para **user**. Es decir, habría una transacción realizada por varios usuarios y esto no es correcto. Lo que queremos es una transacción hecha por un usuario, pero un usuario puede hacer varias transacciones.

Para cambiar esto, la *Foreign Key (fk)* user_id será establecida en la tabla **transaction (*child*)** al revés de la tabla **user (*parent*).**

8) Diagrama antes del cambio de la relación 1-N entre las tablas **transaction** y **user**.

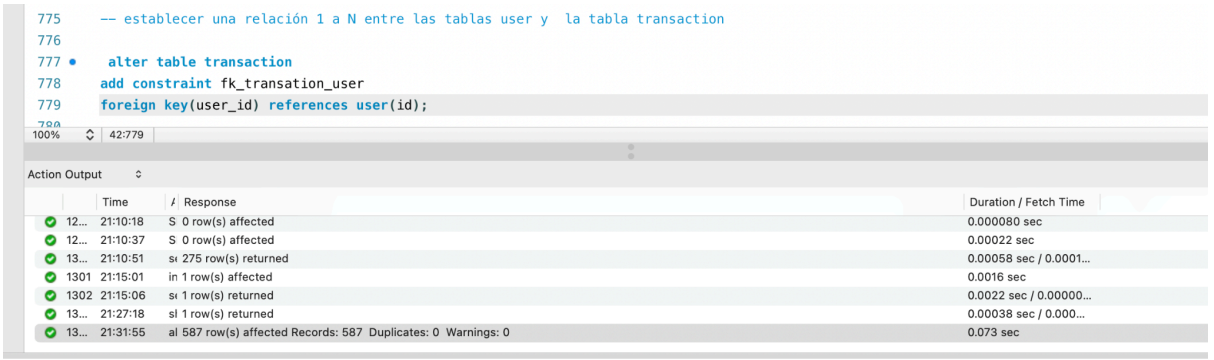


9) Averiguar el nombre del fk de la tabla **user**.



El nombre de la restricción de la tabla **user** es **‘user_ibfk_1’**

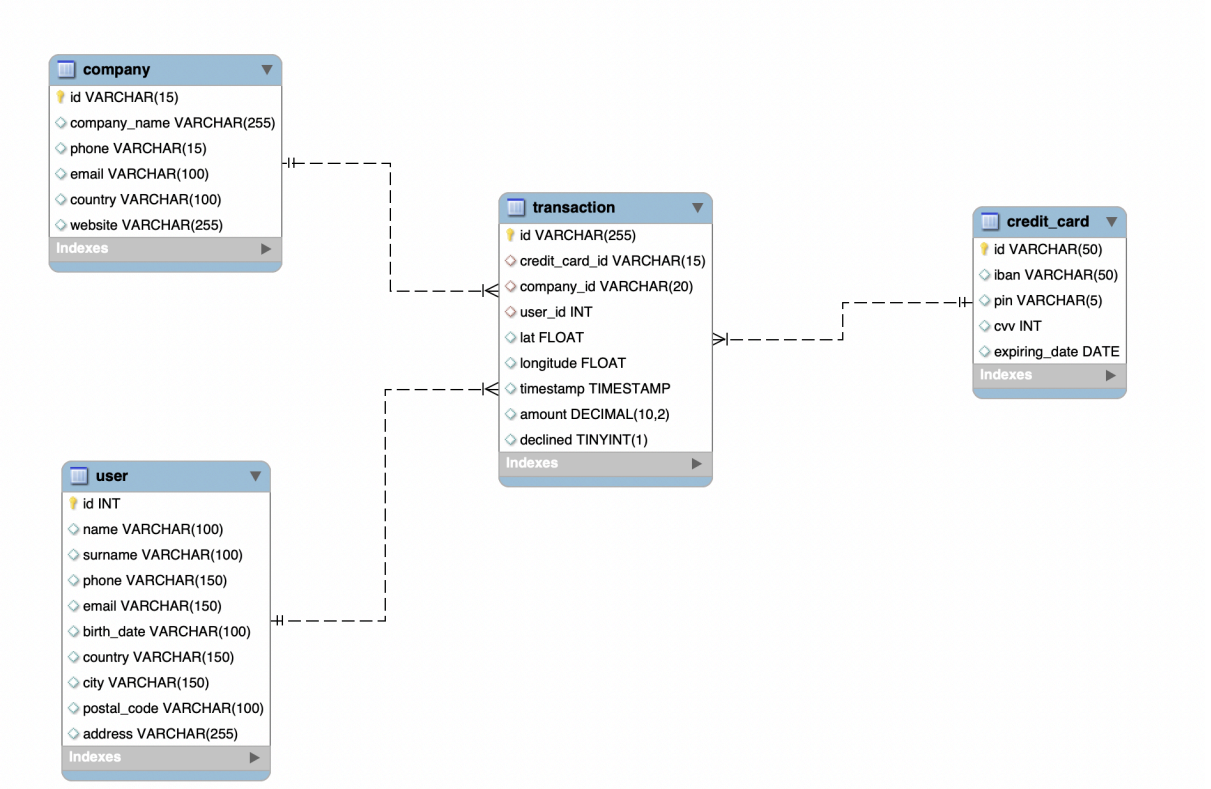
10) Establecer una relación 1 a N entre las tablas **user** y la tabla **transaction**



11) Eliminación de la FK ‘user_ibfk_1’.

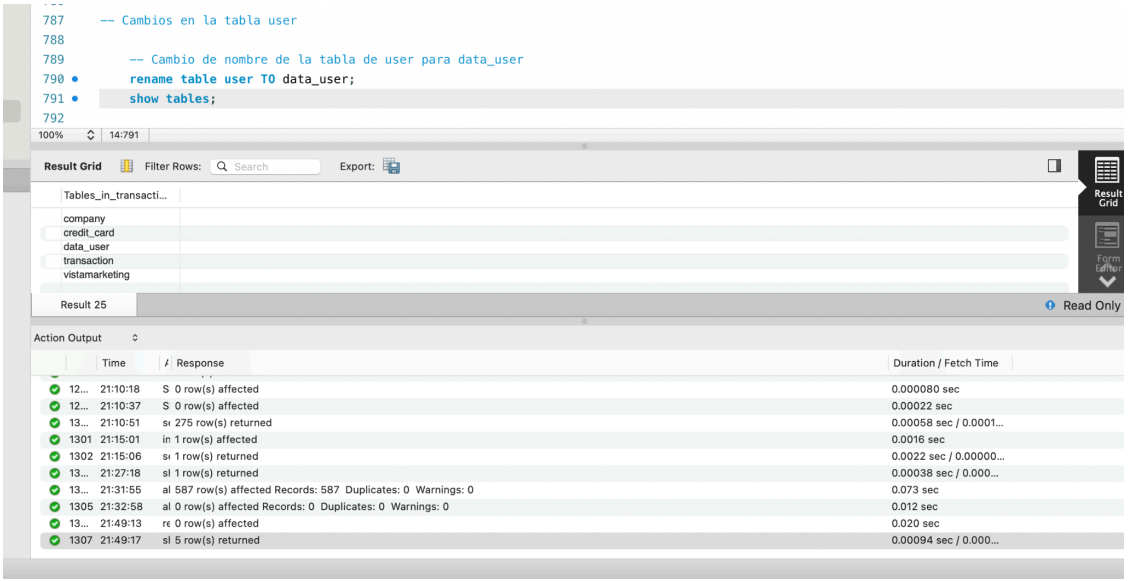
Ahora tenemos un diagrama donde las tablas **company**, **credit_card** y **user** tienen una relación 1-N con la tabla **transaction**.

A continuación, la imagen del diagrama después de la eliminación de la restricción foreign key de la tabla user

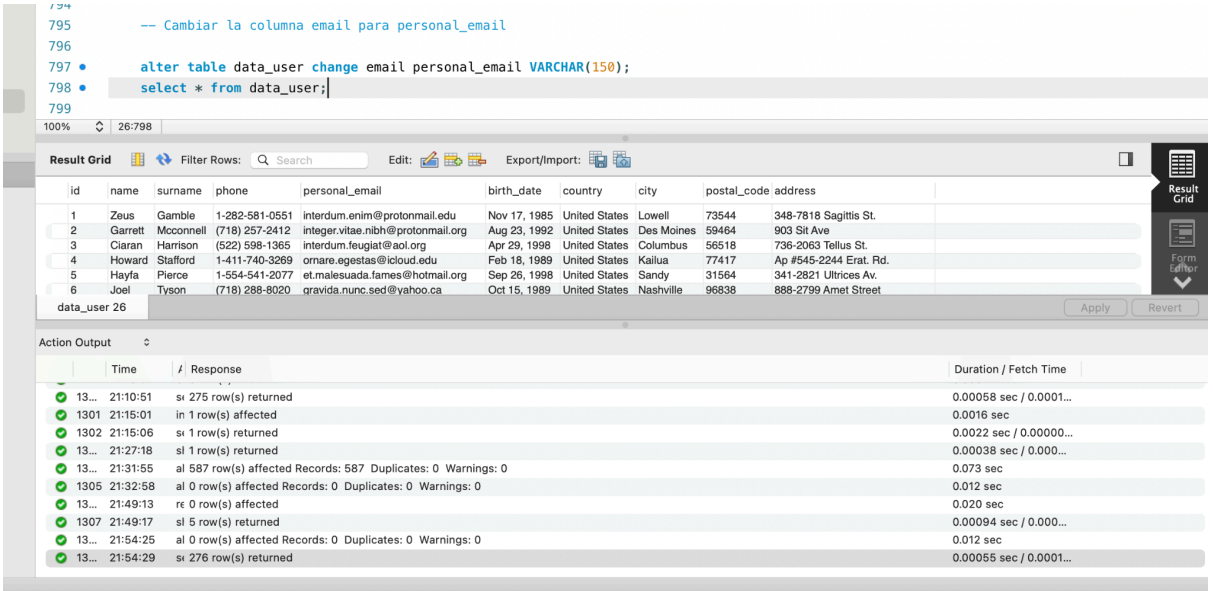


12) Cambios en la tabla **user**

a) Cambio de nombre de la tabla ‘user’ para ‘**data_user’**



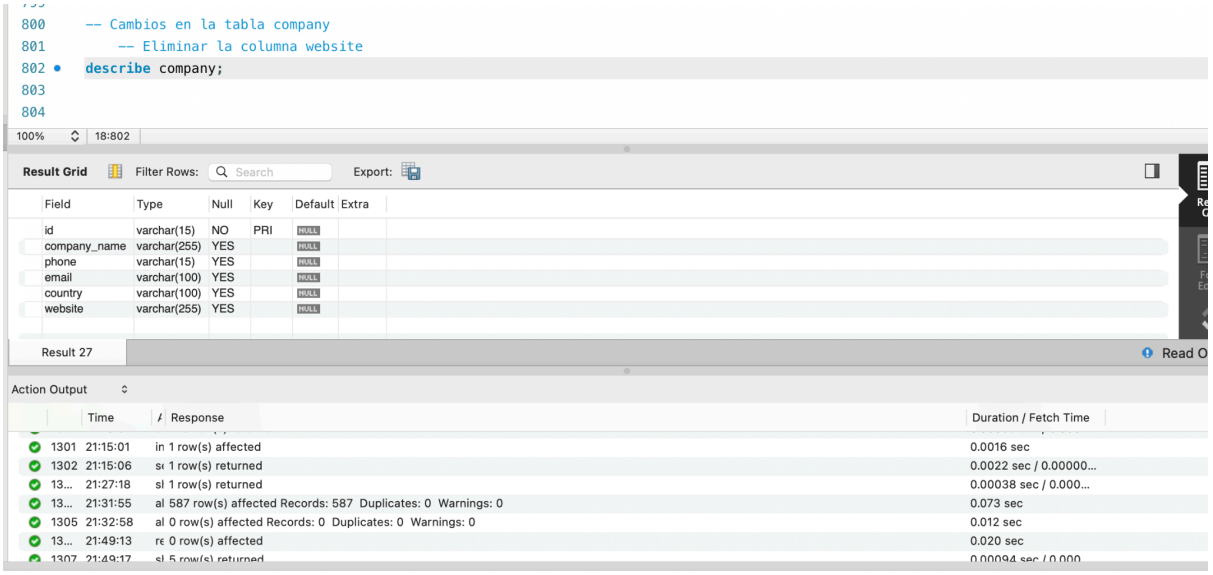
b) Cambiar la columna ‘email’ para ‘**personal_email’**



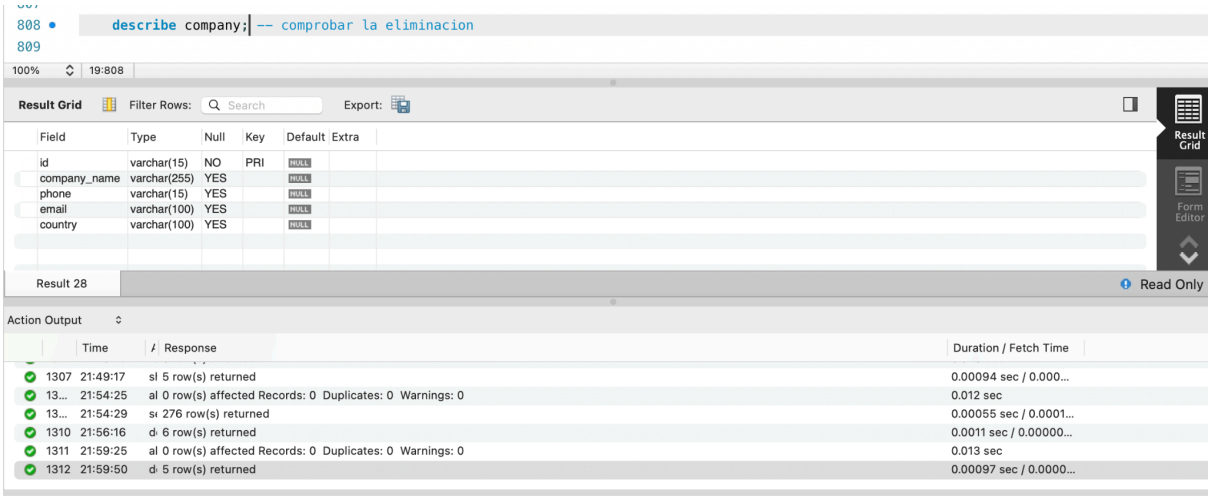
13) Cambios en la tabla **company**

a) Eliminar la columna **website**

Descripción de la tabla **company** con la columna 'website'



b) Eliminación de la columna 'website'

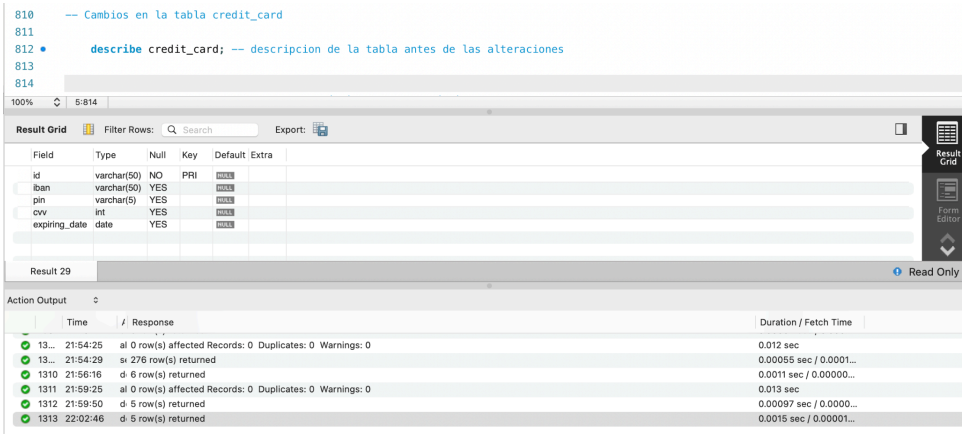


14) Cambios en la tabla **transaction**

Ningún cambio.

15) Cambios en la tabla **credit_card**

Descripción de la tabla antes de las alteraciones



a) Cambio de la columna id de VARCHAR(50) para VARCHAR(20)

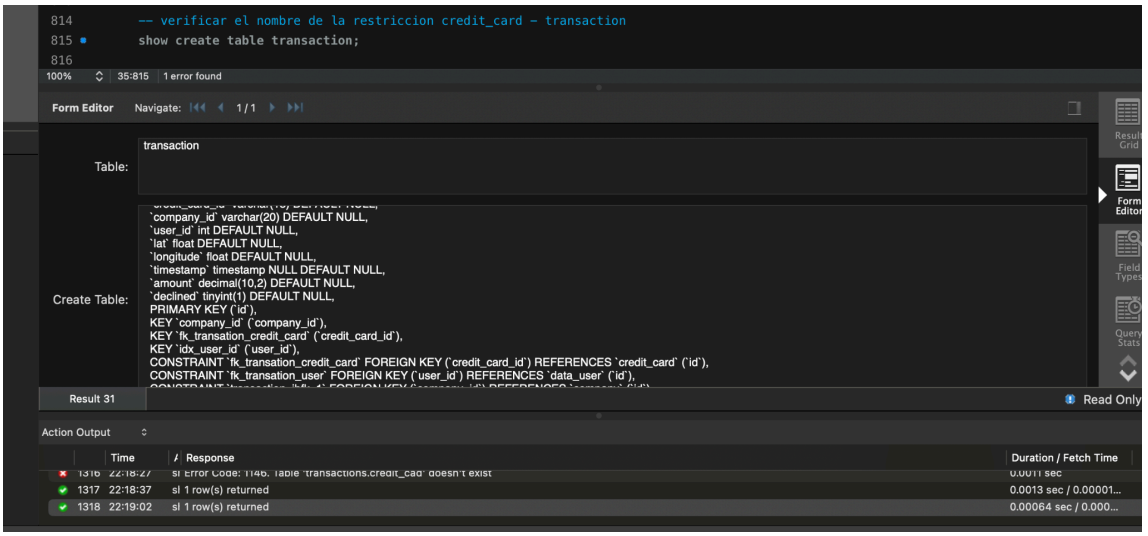
Al intentar cambiar el tipo de datos de una clave primaria que es la clave extranjera en la tabla de transacciones, se me impidió realizar esta acción debido a la restricción FK.

Se dio el siguiente error:

Error Code: 1833. Cannot change column 'id': used in a foreign key constraint 'fk_transation_credit_card' of table 'transactions.transaction'

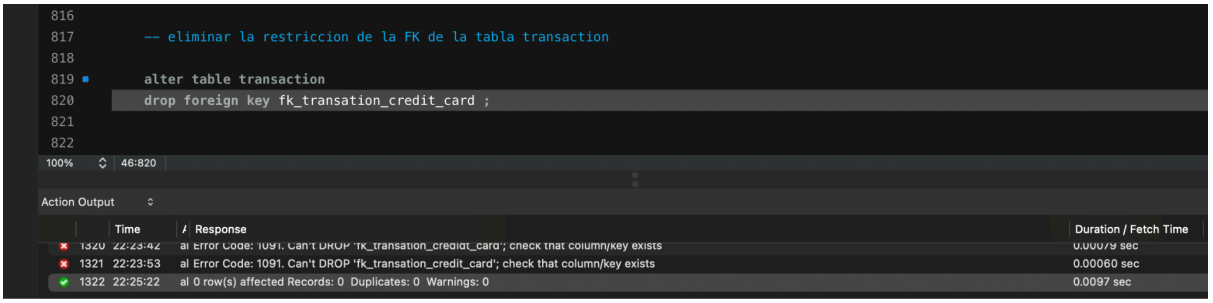
Eliminaré la restricción y después de las modificaciones, la agregaré nuevamente

Primero, verificaré el nombre de la restricción fk **credit_card - transaction**



La restricción se llama ‘ fk_transation_credidt_card’.

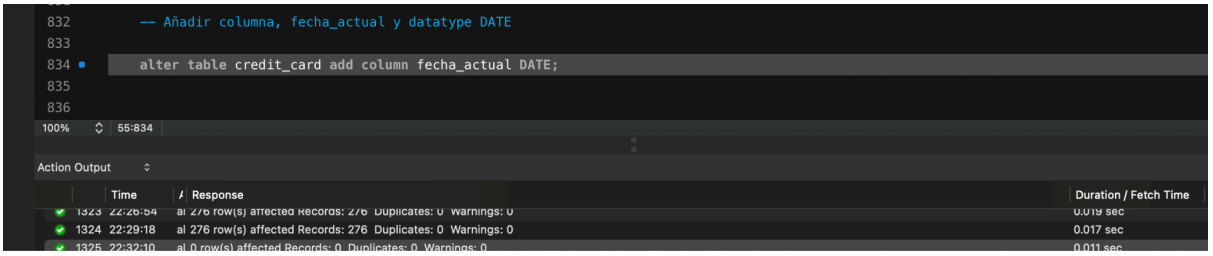
A continuación, la eliminación de la restricción:



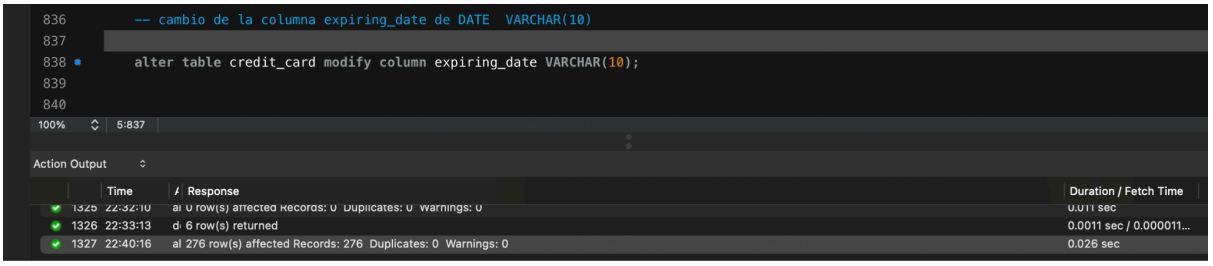
b) Cambio de la columna id de VARCHAR(50) para VARCHAR(20)

c) Cambio de la columna pin de VARCHAR(5) para VARCHAR(4)

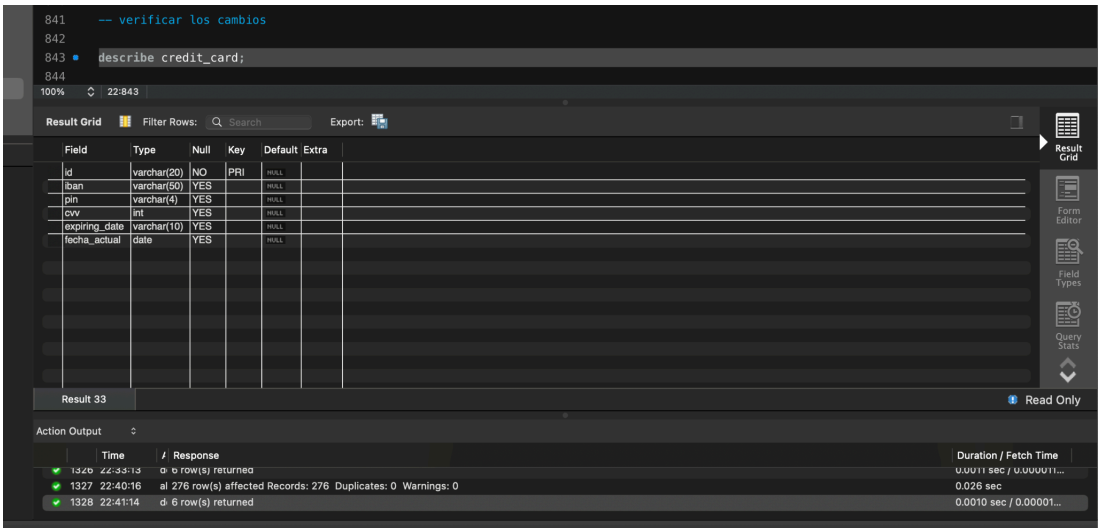
d) Añadir columna, fecha_actual y *datatype* DATE



e) Cambio expiring_date para datatype VARCHAR(10)

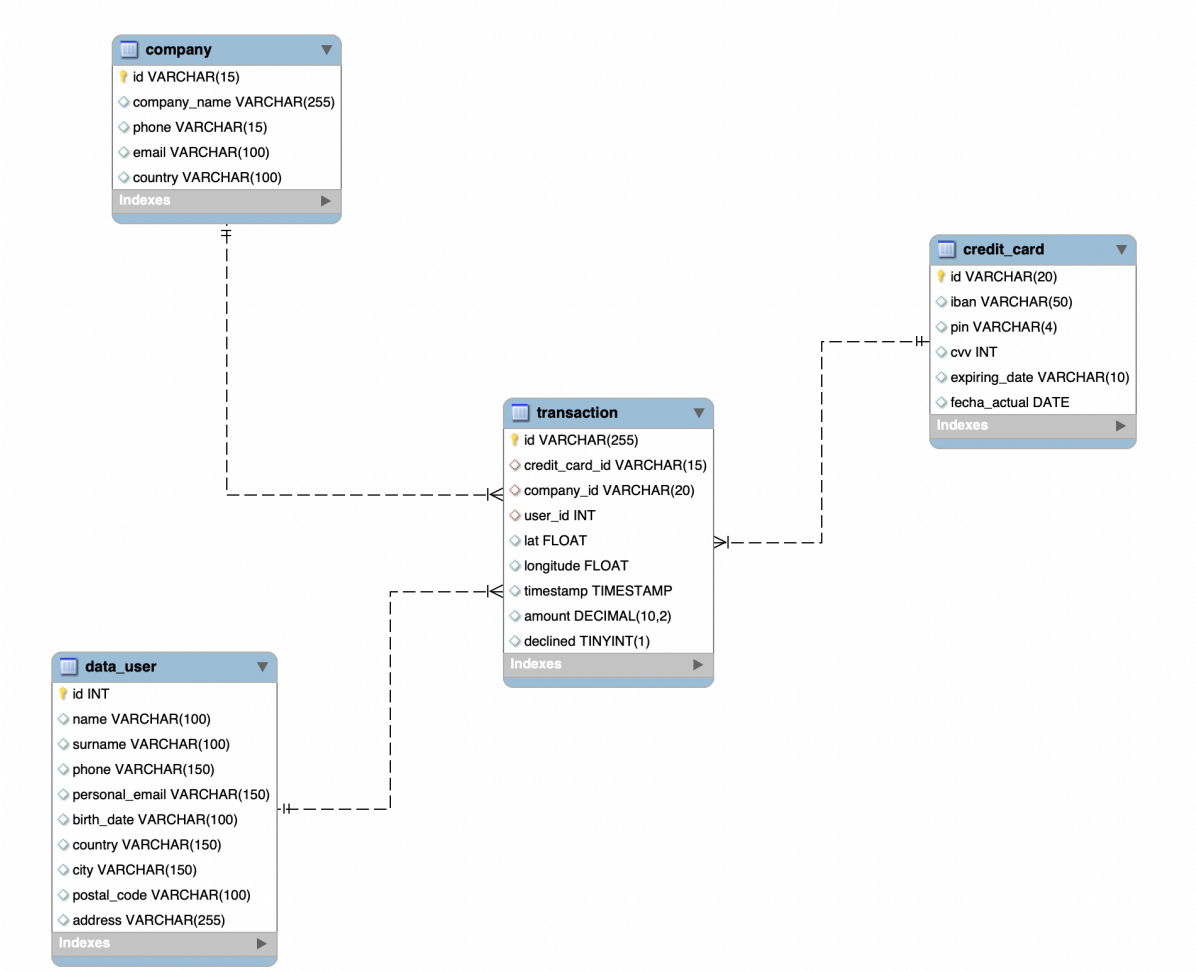


Verificar los cambios:



Después de todos los cambios, volveré con la restricción fk credit_card- transaction

16) Modelo Final



Ejercicio 2

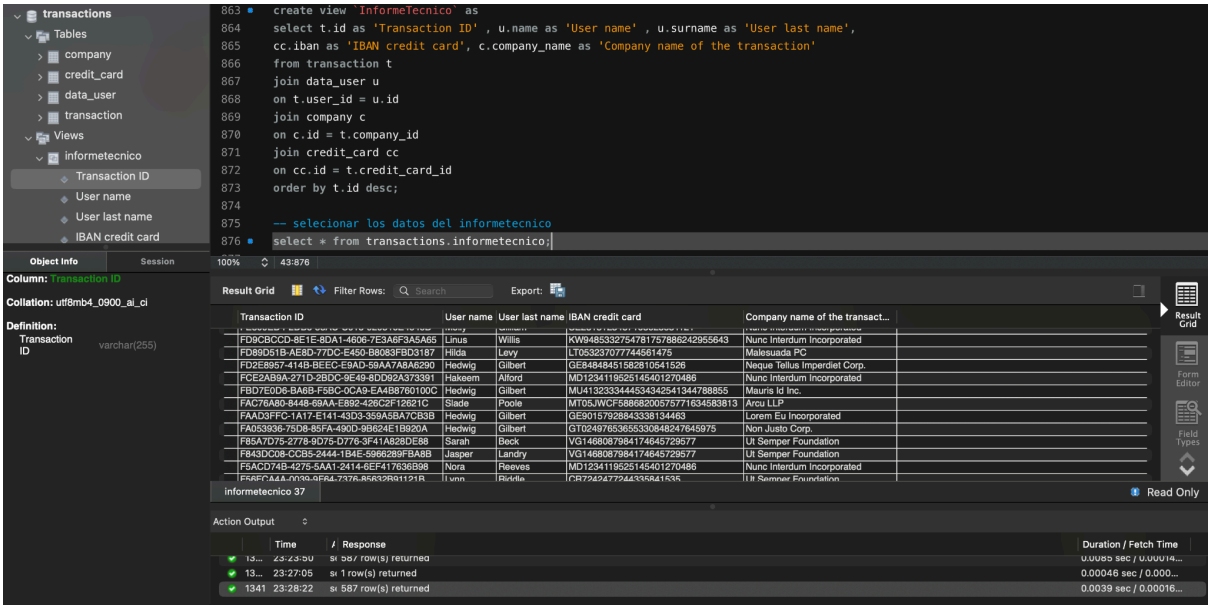
La empresa también te solicita crear una vista llamada "InformeTecnico" que contenga la siguiente información:

- **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.**

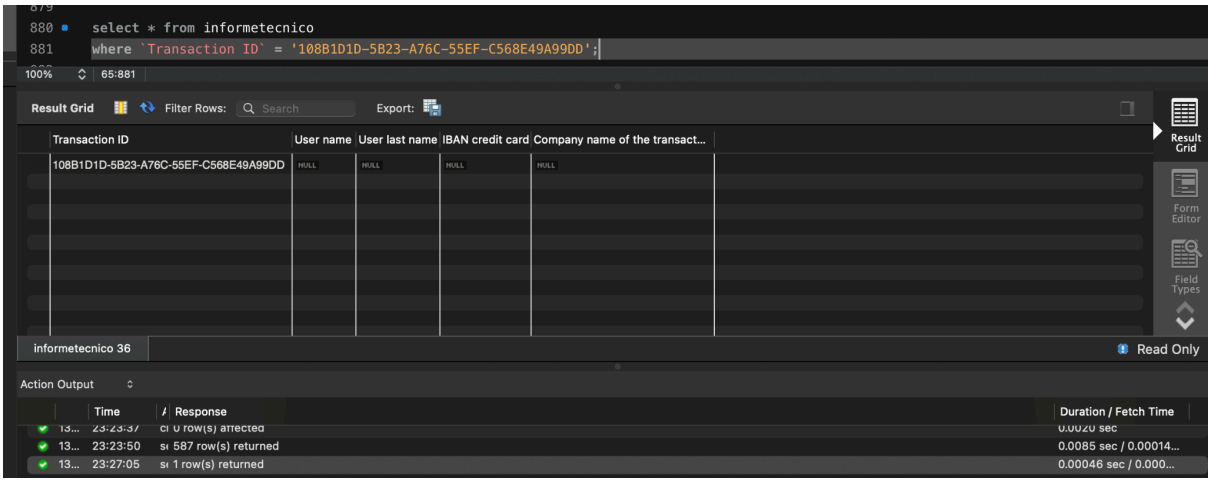
Muestra los resultados de la vista, ordena los resultados de forma descendente en función de la variable ID de transacción.

1) Elaboración del script de la vista informetecnico

.



2) Verificar el id de la transacción añadido en el ejercicio 3, nivel 1



Hasta que los recursos humanos informen sobre los datos del usuario 999, de la tarjeta de crédito CcU-9999 y de la compañía b-9999, los datos de la transaction '108B1D1D-5B23-A76C-55EF-C568E49A99DD' quedaran incompletos.