

Sprint 3

NIVEL 1 ★

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

Paso 1:

Crear la "database" si no existe (no es el caso) y seleccionarla para operar sobre ella.

```
-- Creamos el database si no existe y operamos sobre ella
create database if not exists transactions;
use transactions;
```

Paso 2:

Crear la estructura de la tabla con sus distintos campos y tipos de data para almacenar en ella:

- ID: Seteado como **Primary Key** de la tabla. Sirve para identificar una tarjeta de crédito inequívocamente, por lo que sus datos son únicos. "Varchar (20)" para almacenar múltiples caracteres y limitada para ahorrar espacio.
- IBAN: Muestra el número de cuenta asociado a cada ID. "Varchar (50)" para almacenar múltiples caracteres y limitada para ahorrar espacio.
- PAN: Muestra el número de tarjeta asociado a cada ID. "Varchar (100)" para almacenar múltiples caracteres y limitada para ahorrar espacio.
- PIN: Muestra el PIN de la cuenta asociado a cada ID. "Varchar (4)" para ahorrar espacio. Podría setearse como "smallint", pero es un dato con el que no se van a hacer operaciones.
- CVV: Muestra el código de seguridad de la tarjeta asociado a cada ID. Se podría guardar como "Varchar (3)" para más eficiencia, ya que no se van a hacer operaciones con esos números, pero en futuros ejercicios se requiere en "int".
- Expiring_Date: Muestra la fecha de caducidad de la tarjeta asociada a cada ID. Se guarda como "Varchar (20)" porque para guardarlo como "date", o cualquier formato de fechas, es necesario cambiar el formato de los datos a introducir a "aa/mm/dd" (actualmente en mm/dd/aa). El costo de hacer esto es muy elevado para luego no hacer operaciones con dichas fechas.

```
-- Creamos la tabla "credit_card"
• ○ create table if not exists credit_card (
    id varchar(20) primary key,
    iban varchar(50),
    pan varchar(100),
    pin varchar(4),
    cvv int,
    expiring_date varchar(20)
);
```

Paso 3:

Se establece la relación con la tabla “transaction” añadiendo un “constraint” de Foreign Key en esta última, a través del campo “credit_card_id”, para relacionarse con la Primary Key (ID) de la nueva tabla “credit_card”.

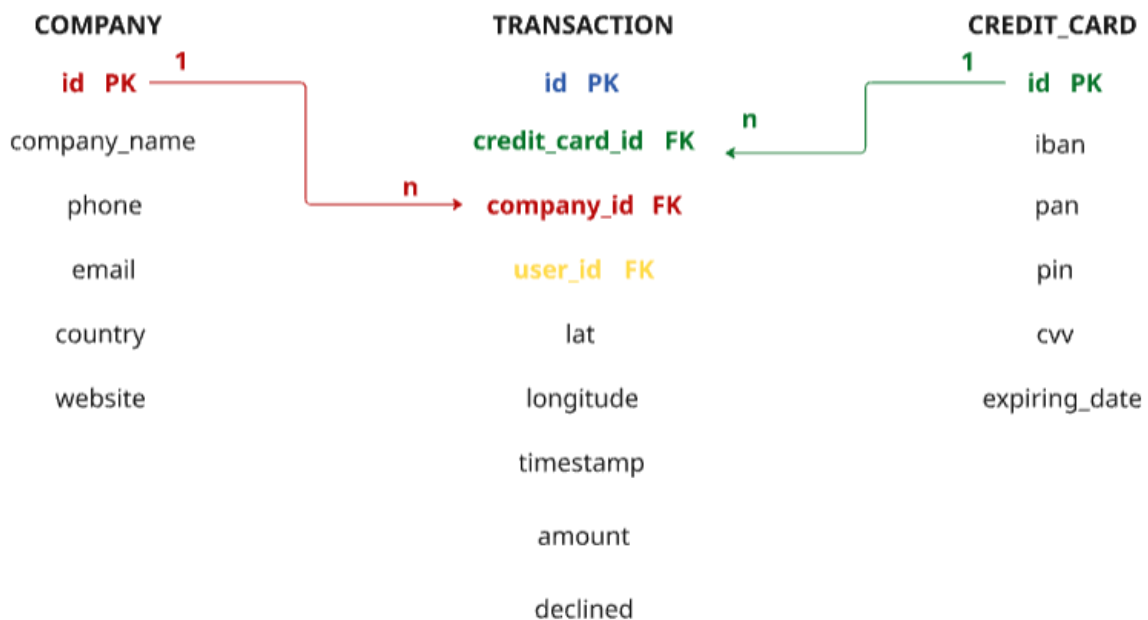
```
-- Añadimos la relación de transaction con credit_card
alter table transactions.transaction
add constraint fk_credit_card foreign key (credit_card_id) references credit_card(id);
```

Paso 4:

Se introducen los datos del archivo “datos_introducir_credit”.

la estructura final del “database” quedaría en algo como esto:

TRANSACTIONS








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.

Antes:

```
20 • select transactions.credit_card.id,  
21     transactions.credit_card.iban  
22     from transactions.credit_card  
23     where transactions.credit_card.id = "CcU-2938";  
24
```

<

Result Grid   Filter Rows: Edit:   

	id	iban
▶	CcU-2938	TR301950312213576817638661
•	NULL	NULL






Query:

```
update transactions.credit_card  
set iban = "R323456312213576817699999"  
where transactions.credit_card.id = "CcU-2938";
```

Después:

```
24 • select transactions.credit_card.id,  
25     transactions.credit_card.iban  
26     from transactions.credit_card  
27     where transactions.credit_card.id = "CcU-2938";  
28
```

<

Result Grid   Filter Rows: Edit:    E

	id	iban
▶	CcU-2938	R323456312213576817699999
•	NULL	NULL

Antes de añadir este nuevo registro, hay que introducir el id de la tarjeta de crédito y el de la compañía en sus respectivas tablas. Esto se debe a que la tabla "transaction" cuenta con constraints en forma de Foreign Keys que están vinculadas a las Primary Keys de las tablas de dimensiones ("credit_card" y "company"). Por lo tanto, estos datos deben existir previamente en las tablas de dimensiones antes de que puedan ser utilizados para crear registros en la tabla "transaction".

**Se han creado los ID en ambas tablas, pero en un futuro habría que actualizar el resto de datos asociados.*

Query:

Resultado:

[illegible]

Ejercicio 4 | Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit_card. Record a mostrar el canvi realitzat.

Antes:

	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
	CcU-2973	PT87806228135092429456346	544 58654 54343 384	8760	887	01/30/25
	CcU-2980	DE39241881883086277136	402400 7145845969	5075	596	07/24/22
	CcU-2987	GE89681434837748781813	3763 747687 76666	2298	797	10/31/23
	CcU-2994	BH62714428368066765294	344283273252593	7545	595	02/28/22
	CcU-3001	CY49087426654774581266832110	511722 924833 2244	9562	867	09/16/22
	CcU-3008	LU507216693616119230	4485744464433884	1856	740	04/05/25

Query:

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

Después:

Result Grid		Filter Rows:		Edit:		Export/Im
	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	
	CcU-2973	PT87806228135092429456346	8760	887	01/30/25	
	CcU-2980	DE39241881883086277136	5075	596	07/24/22	
	CcU-2987	GE89681434837748781813	2298	797	10/31/23	
	CcU-2994	BH62714428368066765294	7545	595	02/28/22	
	CcU-3001	CY49087426654774581266832110	9562	867	09/16/22	
	CcU-3008	LU507216693616119230	1856	740	04/05/25	

NIVEL 2 ★★

Ejercicio 1 | Elimina de la taula transaction el registre amb ID 02C6201E-D90A-1859-B4EE-88D2986D3B02 de la base de dades.

Antes:

```
46 • select *
47   from transactions.transaction
48   where transactions.transaction.id = "02C6201E-D90A-1859-B4EE-88D2986D3B02";
```

Result Grid

	id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
▶	02C6201E-D90A-1859-B4EE-88D2986D3B02	CcJ-2938	b-2362	92	81.9185	-12.5276	2021-08-28 23:42:24	466.92	0
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Query:

```
-- Request
delete transactions.transaction
from transactions.transaction
where transactions.transaction.id = "02C6201E-D90A-1859-B4EE-88D2986D3B02";
```

Después:

```
61 • select *
62   from transactions.transaction
63   where transactions.transaction.id = "02C6201E-D90A-1859-B4EE-88D2986D3B02";
```

Result Grid

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

transaction 31 ×

Output

Action Output

#	Time	Action	Message
✓ 1	12:21:20	select * from transactions.transaction where transactions.transaction.id = "02C6201E-D90A-1859-B4EE-88D2...	0 row(s) returned

Ejercicio 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.

Query:

```
1 • CREATE VIEW `VistaMarketing` AS
2   select transactions.company.company_name,
3   transactions.company.phone,
4   transactions.company.country,
5   round(avg(transactions.transaction.amount),2) as media_de_compra
6   from transactions.company
7   join transactions.transaction
8   on transactions.company.id = transactions.transaction.company_id
9   where transactions.transaction.declined = 0 -- Eliminamos los registros rechazados
10  group by transactions.transaction.company_id -- Agrupamos por ID en caso de compañías con el mismo nombre
11  order by media_de_compra desc;
```

Resultado:

```
78 • select *
79   from transactions.vistamarketing;
80
```

company_name	phone	country	media_de_compra
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
Maecenas Malesuada Fringilla Inc.	09 38 53 76 61	Netherlands	451.29
Erat LLP	03 18 88 77 79	Netherlands	448.44
Tortor Nunc Commodo Company	05 35 92 77 16	United States	447.11
Justo Eu Arcu Ltd	08 42 56 71 52	Italy	444.16
Pede Cum Ltd	07 62 26 48 38	Norway	442.32
Vestibulum Lorem PC	02 02 87 33 40	Belgium	428.40
Mauris Institute	05 29 60 36 87	Sweden	427.71

vistamarketing 33 x

Output

Action Output

#	Time	Action	Message
1	12:25:02	select * from transactions.vistamarketing LIMIT 0, 1000	101 row(s) returned

Ejercicio 3 | Filtra la vista VistaMarketing per a mostrar només les companyies que tenen el seu país de residència en "Germany"

Query:

```
select *  
from transactions.vistamarketing  
where transactions.vistamarketing.country = "Germany";
```

Resultado:

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
company_name	phone	country	media_de_compra
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
Nunc Interdum Incorporated	05 18 15 48 13	Germany	242.95
Convallis In Incorporated	06 66 57 29 50	Germany	60.99
Augue Foundation	06 88 43 15 63	Germany	15.05

vistamarketing 34 x

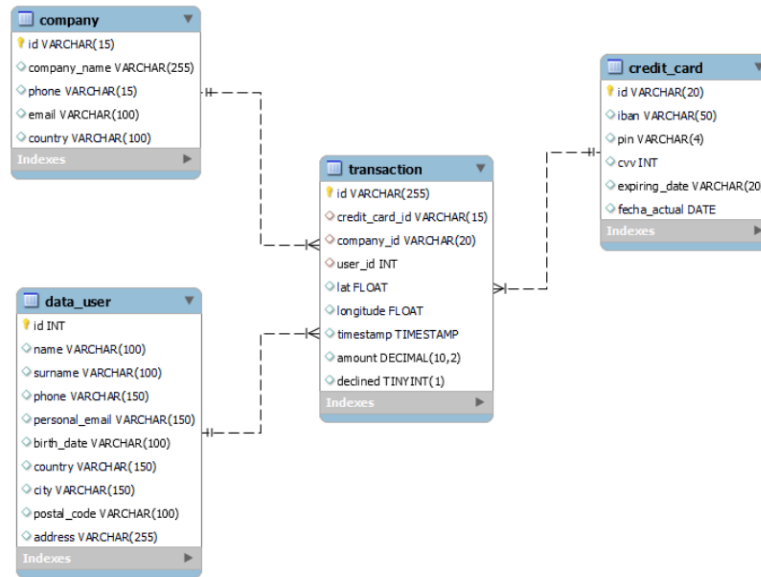
Output

Action Output

#	Time	Action	Message
1	12:27:25	select * from transactions.vistamarketing where transactions.vistamarketing.country = "Germany" LIMIT 0, 1000	8 row(s) returned

NIVEL 3 ★★★★★

Ejercicio 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:



* Suponiendo que la estructura de la database está exactamente como debería tras haber realizado los ejercicios previos

- Paso 1 | Cambios relacionados con la tabla “user”:

- Creación de la tabla user:

```
91 • use transactions;
92 • CREATE INDEX idx_user_id ON transaction(user_id);
93 • CREATE TABLE IF NOT EXISTS user (
94     id INT PRIMARY KEY,
95     name VARCHAR(100),
96     surname VARCHAR(100),
97     phone VARCHAR(150),
98     email VARCHAR(150),
99     birth_date VARCHAR(100),
100    country VARCHAR(150),
101    city VARCHAR(150),
102    postal_code VARCHAR(100),
103    address VARCHAR(255)
104 );
```

* Se ha eliminado la línea “FOREIGN KEY(id) REFERENCES transaction(user_id)” con respecto al archivo original. la FK ha de ser introducida en la tabla “transaction”.

- Se añaden los datos a la tabla del archivo “datos_introducir_user”.

- Se añade el id de usuario del “nivel 1 | ejercicio 3” para evitar futuros errores a la hora de crear constraints con la tabla “transaction”:

```
107 • insert into transactions.user(id)
108     values('9999');
```

- Se renombra la columna “email” a “personal_email”, como en el diagrama:

```
110 • alter table transactions.user
111     rename column email to personal_email;
```

- Se renombra la tabla “user” a “data_user” como en el diagrama:

```
114 • rename table transactions.user to data_user;
```

- Paso 2 | Cambios en la tabla “company”:

- Se elimina la columna “website”:

```
118 • alter table transactions.company
119     drop column website;
```

- Paso 3 | Cambios en la tabla “credit_card”:

- Se crea la columna “fecha_actual”:

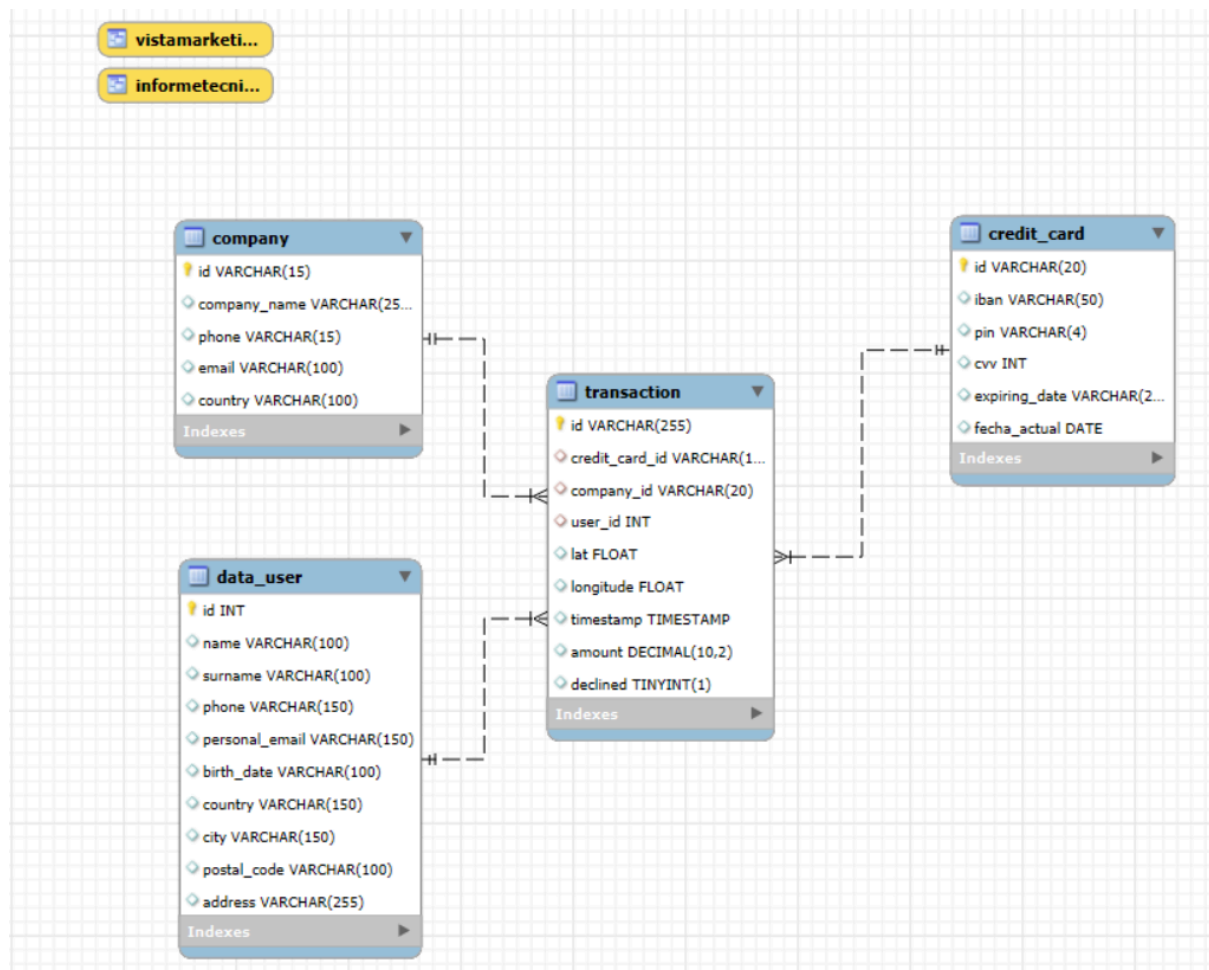
```
123 • alter table transactions.credit_card
124     add column fecha_actual date;
```

- Paso 4 | Cambios en la tabla “transaction”:

- Se añade el constraint para relacionar la Primary Key de la tabla “user” (ID) con la Foreign Key de la tabla “transaction” (user_id):

```
125 • alter table transactions.transaction
126     add constraint fk_user foreign key (user_id) references user(id);
```

Después de todos los cambios, es esquema de la database queda de la siguiente forma:



Ejercicio 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.

Además de los datos que se piden, se han incluido también las siguientes columnas para mejorar la eficiencia de la "View":

- Fecha de la transacción.
- User ID: Por si hay distintos clientes con el mismo nombre y apellidos.
- Ingresos de la transacción: El gasto del usuario en la transacción.
- Media de gasto del usuario: Solo en transacciones válidas (no rechazadas).
- Transacción rechazada: Columna personalizada que valida si la transacción ha sido rechazada o no.

Query:

```
1 • CREATE VIEW `informetecnico` AS
2   with media_gasto_usuario_tabla as
3   (
4       select round(avg(transactions.transaction.amount),2) as media_gasto_usuario,
5       transactions.transaction.user_id
6       from transactions.transaction
7       where transactions.transaction.declined = 0
8       group by transactions.transaction.user_id
9   )
10  select transactions.transaction.id as id_transacción,
11         date(transactions.transaction.timestamp) as fecha_transacción,
12         transactions.company.company_name as compañía,
13         transactions.user.id as user_id,
14         transactions.user.name as nombre_usuario,
15         transactions.user.surname as apellido_usuario,
16         transactions.credit_card.iban,
17         transactions.transaction.amount as ingresos_de_la_transacción,
18         media_gasto_usuario_tabla.media_gasto_usuario,
19
20         case
21             when transactions.transaction.declined = 0 then "NO"
22             when transactions.transaction.declined = 1 then "SÍ"
23         end as transacción_rechazada
24  from transactions.transaction
25  join transactions.user
26  on transactions.transaction.user_id = transactions.user.id
27  join transactions.credit_card
28  on transactions.transaction.credit_card_id = transactions.credit_card.id
29  join transactions.company
30  on transactions.transaction.company_id = transactions.company.id
31  join media_gasto_usuario_tabla
32  on transactions.transaction.user_id = media_gasto_usuario_tabla.user_id;
```

Resultado (filtrado como especifica el enunciado):

164 • select *
165 from transactions.infometecnico
166 order by id_transacción desc;
167

Result Grid

Filter Rows:

Exports

Wrap Cell Content:

	id_transacción	fecha_transacción	compañía	user_id	nombre_usuario	apellido_usuario	iban
▶	FE96CE47-BD59-381C-4E18-E3CA3D44E8FF	2021-06-15	Magna A Neque Industries	275	Kenyon	Hartman	DO26854763748537475216568689
	FE809ED4-2DB6-55AC-C915-929516E46468	2021-11-09	Nunc Interdum Incorporated	185	Molly	Gilliam	SE2813123487163628531121
	FD9CBCCD-8E1E-8DA1-4606-7E3A6F3A5A65	2021-06-13	Nunc Interdum Incorporated	223	Linus	Willis	KW9485332754781757886242955643
	FD89D51B-AE8D-77DC-E450-B8083FBD3187	2022-03-16	Malesuada PC	250	Hilda	Levy	LT053237077744561475

informetecnico 37

×

Output

Action Output

#	Time	Action	Message
✓ 1	12:59:19	select * from transactions.infometecnico order by id_transacción desc LIMIT 0, 1000	587 row(s) returned

ingresos_de_la_transacción	media_gasto_usuario	transacción_rechazada
480.13	236.78	SÍ
219.83	219.83	NO
42.32	270.78	NO
200.72	331.73	NO

Message
587 row(s) returned