

Nivel 1 - Ejercicio 1

Cree la tabla `credit_card` y defina el tipo de datos para cada campo. Además importe los datos para completar la tabla y vinculela con la tabla `transaction`.

```

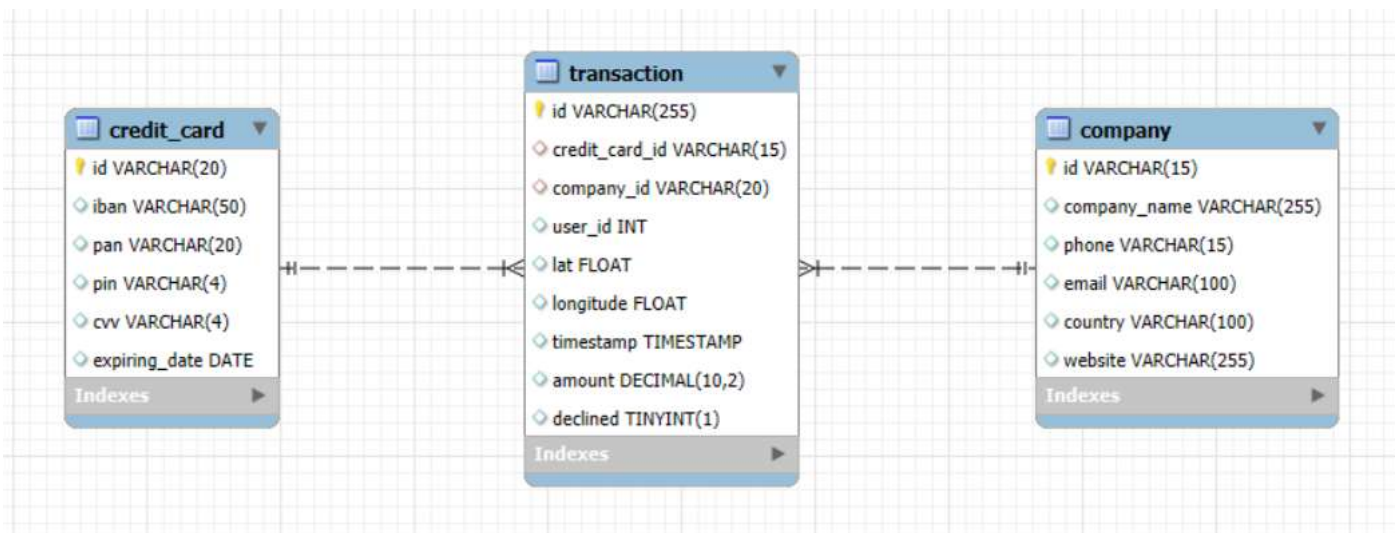
1  # Nivel 1 - Ejercicio 1
2
3  •  USE transactions;
4
5  •  CREATE TABLE credit_card (
6      id VARCHAR(20) PRIMARY KEY,
7      iban VARCHAR(50),
8      pan VARCHAR(20),
9      pin VARCHAR(4),
10     cvv VARCHAR(4),
11     expiring_date VARCHAR(20));
12
13     -- Insertamos datos de credit_card
14
15  •  UPDATE credit_card
16     SET expiring_date = DATE_FORMAT(STR_TO_DATE(expiring_date, '%m/%d/%y'), '%Y-%m-%d');
17
18  •  ALTER TABLE credit_card
19     MODIFY COLUMN expiring_date DATE;
20
21  •  ALTER TABLE transaction
22     ADD FOREIGN KEY (credit_card_id) REFERENCES credit_card(id);
23
24  •  SELECT *
25     FROM credit_card;
26

```

Result Grid						
Filter Rows:						
Edit:						
Export/Import:						
Wrap Cell Content:						
	id	iban	pin	cvv	expiring_date	fecha_actual
▶	CcU-2938	R323456312213576817699999	3257	984	2022-10-30	2025-02-26
	CcU-2945	DO26854763748537475216568689	9080	887	2023-08-24	2025-02-26
	CcU-2952	BG45IVQL52710525608255	4598	438	2021-06-29	2025-02-26
	CcU-2959	CR7242477244335841535	3583	667	2023-02-24	2025-02-26
	CcU-2966	BG72LKTQ15627628377363	4900	130	2024-10-29	2025-02-26
	CcU-2973	PT370000000000000000000000	0300	007	2025-01-01	2025-02-26

credit_card 1 x

Output			
Action Output			
#	Time	Action	Message
5	15:34:55	SELECT * FROM transactions.credit_card LIMIT 0, 50000	275 row(s) returned
6	15:36:14	SELECT * FROM credit_card LIMIT 0, 50000	275 row(s) returned



Nivel 1 - 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.

```

24  # Nivel 1 - Ejercicio 2
25
26  • UPDATE credit_card
27    SET iban = 'R323456312213576817699999'
28    WHERE id = 'CcU-2938';
29
30  • SELECT *
31    FROM credit_card
32    WHERE id = 'CcU-2938';
33
  
```

Result Grid						
Filter Rows:						
Edit:						
Export/Import:						
Wrap Cell Content:						
	id	iban	pan	pin	cvv	expiring_date
▶	CcU-2938	R323456312213576817699999	5424465566813633	3257	984	2022-10-30
*	NULL	NULL	NULL	NULL	NULL	NULL

credit_card 4 ×						
Output						
Action Output						
#	Time	Action	Message			
✓ 319	15:30:16	UPDATE credit_card SET iban = 'R323456312213576817699999' WHERE id = 'CcU-2938'	1 row(s) affected			
✓ 320	15:30:21	SELECT * FROM credit_card WHERE id = 'CcU-2938' LIMIT 0, 50000	1 row(s) returned			

Nivel 1 - Ejercicio 3

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

```
37 #Nivel 1 - Ejercicio 3
38
39 • INSERT INTO credit_card (id)
40   VALUES ('CcU-9999');
41
42 • INSERT INTO company (id)
43   values ('b-9999');
44
45 • INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined)
46   VALUES ('10881D1D-5B23-A76C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', '9999', 829.999, -117.999, 111.11, 0);
47
48 • SELECT *
49   FROM transaction
50  WHERE id = '10881D1D-5B23-A76C-55EF-C568E49A99DD';
```

Result Grid

	id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
▶	10881D1D-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 15 x

Output

Action Output

#	Time	Action	Message
✓ 346	16:27:30	INSERT INTO credit_card (id) VALUES ('CcU-9999')	1 row(s) affected
✓ 347	16:27:33	INSERT INTO company (id) values ('b-9999')	1 row(s) affected
✓ 348	16:27:36	INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, amount, declined) VA...	1 row(s) affected
✓ 349	16:27:44	SELECT * FROM transaction WHERE id = '10881D1D-5B23-A76C-55EF-C568E49A99DD' LIMIT 0, 50...	1 row(s) returned

Nivel 1 - Ejercicio 4

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

```
52 #Nivel 1 - Ejercicio 4
53
54 • ALTER table credit_card
55   DROP COLUMN pan;
56
57 • SELECT *
58   FROM credit_card
59   LIMIT 1;
60
```

Result Grid

	id	iban	pin	cvv	expiring_date
▶	CcU-2938	R323456312213576817699999	3257	984	2022-10-30
*	NULL	NULL	NULL	NULL	NULL

Nivel 2 - Ejercicio 1

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

```
61      #Nivel 2 - Ejercicio 1
62
63 •    DELETE FROM transaction
64      WHERE id='02C6201E-D90A-1859-B4EE-88D2986D3B02';
65
66 •    SELECT *
67      FROM transaction
68      WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02';
```

Result Grid									
Filter Rows: <input type="text"/>									
Edit: Export/Import: Wrap									
	id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Nivel 2 - Ejercicio 2

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.

```
70      #Nivel 2 - Ejercicio 2
71
72 •    CREATE VIEW `vistamarketing` AS
73      SELECT company.company_name, company.phone, company.country, ROUND(AVG(transaction.amount), 2) AS AVGtransaction
74      FROM company
75      LEFT JOIN transaction ON company.id=transaction.company_id
76      WHERE declined = 0
77      GROUP BY company.company_name, company.phone, company.country
78      ORDER BY AVGtransaction DESC;
79
80 •    SELECT * FROM transactions.vistamarketing;
81
```

Result Grid			
Filter Rows: <input type="text"/>			
Exports: Wrap Cell Contents:			
	company_name	phone	country
▶	Eget Ipsum Ltd	03 67 44 56 72	United States
	Sed Id Limited	07 28 18 18 13	United States
	Neque Tellus Incorporated	04 43 18 34 19	Ireland
	Nunc Sit Incorporated	07 28 42 63 63	Norway
	Non Magna LLC	06 71 73 13 17	United Kingdom
	Maecenas Malesuada Fringilla Inc.	09 38 53 76 61	Netherlands
	Erat LLP	03 18 88 77 79	Netherlands
	Tortor Nunc Commodo Company	05 35 92 77 16	United States
	Justo Eu Arcu Ltd	08 42 56 71 52	Italy
	Pede Cum Ltd	07 62 26 48 38	Norway

vistamarketing 19 x

Output

Action Output

#	Time	Action	Message
357	16:30:45	CREATE VIEW `vistamarketing` AS SELECT company company_name, company phone, company.cou...	0 row(s) affected
358	16:30:48	SELECT * FROM transactions.vistamarketing LIMIT 0, 50000	101 row(s) returned

Nivel 2 - Ejercicio 3

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

```
82      #Nivel 2 - Ejercicio 3
83
84      SELECT *
85      FROM transactions.vistamarketing
86      WHERE country='Germany';
87
```

Result Grid				
Filter Rows: <input type="text"/>				
Export:				
Wrap Cell Content:				
	company_name	phone	country	AVGtransaction
▶	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 22 x				
Output				
Action Output				
#	Time	Action	Message	
✓ 360	16:33:50	SELECT * FROM transaction WHERE id = '02C6201E-D90A-1859-B4EE-88D2986D3B02' LIMIT 0, 500...	0 row(s) returned	
✓ 361	16:35:17	SELECT * FROM transactions.vistamarketing WHERE country='Germany' LIMIT 0, 50000	8 row(s) returned	

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: ...

```
1      -- Creamos la tabla user
2
3  •   CREATE INDEX idx_user_id ON transaction(user_id);
4
5  •   CREATE TABLE IF NOT EXISTS user (
6      id INT PRIMARY KEY,
7      name VARCHAR(100),
8      surname VARCHAR(100),
9      phone VARCHAR(150),
10     email VARCHAR(150),
11     birth_date VARCHAR(100),
12     country VARCHAR(150),
13     city VARCHAR(150),
14     postal_code VARCHAR(100),
15     address VARCHAR(255),
16     FOREIGN KEY(id) REFERENCES transaction(user_id)
17 );
18
19
20
21
22
```

Output			
Action Output			
#	Time	Action	Message
367	16:39:02	CREATE INDEX idx_user_id ON transaction(user_id)	Error Code: 1061. Dup
368	16:39:04	CREATE TABLE IF NOT EXISTS user (id INT PRIMARY KEY, name VARCHAR(100), ...	0 row(s) affected

```

1  -- Insertamos datos de user
2  * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "1", "Zeus", "Gambli
3  * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "2", "Garrett", "Ric
4  * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "3", "Ciaran", "Hari
5  * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "4", "Howard", "Sta
6  * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "5", "Hayfa", "Pieri
7  * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "6", "Joel", "Tyson"
8  * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "7", "Rafael", "Tim
9  * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "8", "Nissim", "Frai
10 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "9", "Hannis", "McC
11 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "10", "Robert", "Ric
12 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "11", "Joan", "Bairn
13 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "12", "Benedict", "B
14 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "13", "Allegra", "Si
15 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "14", "Sara", "Flym
16 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "15", "Noelani", "Pi
17 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "16", "Eric", "Roth"
18 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "17", "Bruce", "Gil
19 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "18", "Russell", "D
20 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "19", "Nicholas", "
21 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "20", "Kelsey", "Har
22 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "21", "Hall", "Reev
23 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "22", "Allistair", "
24 * INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) VALUES ( "23", "Kelsie", "Be

```

Output

Action Output

#	Time	Action	Message
643	16:40:52	INSERT INTO user (id, name, surname, phone, email, birth_date, country, city, postal_code, address) V...	1 row(s) affected
644	16:40:52	SET foreign_key_checks = 1	0 row(s) affected

88 #Nivel 3 - Ejercicio 1

89

```

90 * ALTER TABLE company
91 DROP COLUMN website;
92 * ALTER TABLE user
93 RENAME COLUMN email TO personal_email;
94 * ALTER TABLE user
95 RENAME data_user;
96 * ALTER TABLE credit_card
97 ADD COLUMN fecha_actual DATE;
98 * UPDATE credit_card
99 SET fecha_actual = CURDATE();

```

Output

Action Output

#	Time	Action	Message
647	16:42:35	ALTER TABLE company DROP COLUMN website	0 row(s) affected Re
648	16:42:46	ALTER TABLE user RENAME COLUMN email TO personal_email	0 row(s) affected Re
649	16:42:50	ALTER TABLE user RENAME data_user	0 row(s) affected
650	16:42:52	ALTER TABLE credit_card ADD COLUMN fecha_actual DATE	0 row(s) affected Re
651	16:42:55	UPDATE credit_card SET fecha_actual = CURDATE()	276 row(s) affected


```

101 * show create table data_user;
102 * ALTER TABLE data_user
103 DROP CONSTRAINT data_user_ibfk_1;
104 * INSERT INTO data_user (id)
105 values ('9999');
106 * ALTER TABLE transaction
107 ADD FOREIGN KEY (user_id) REFERENCES data_user(id);
108 * SELECT * FROM data_user;

```

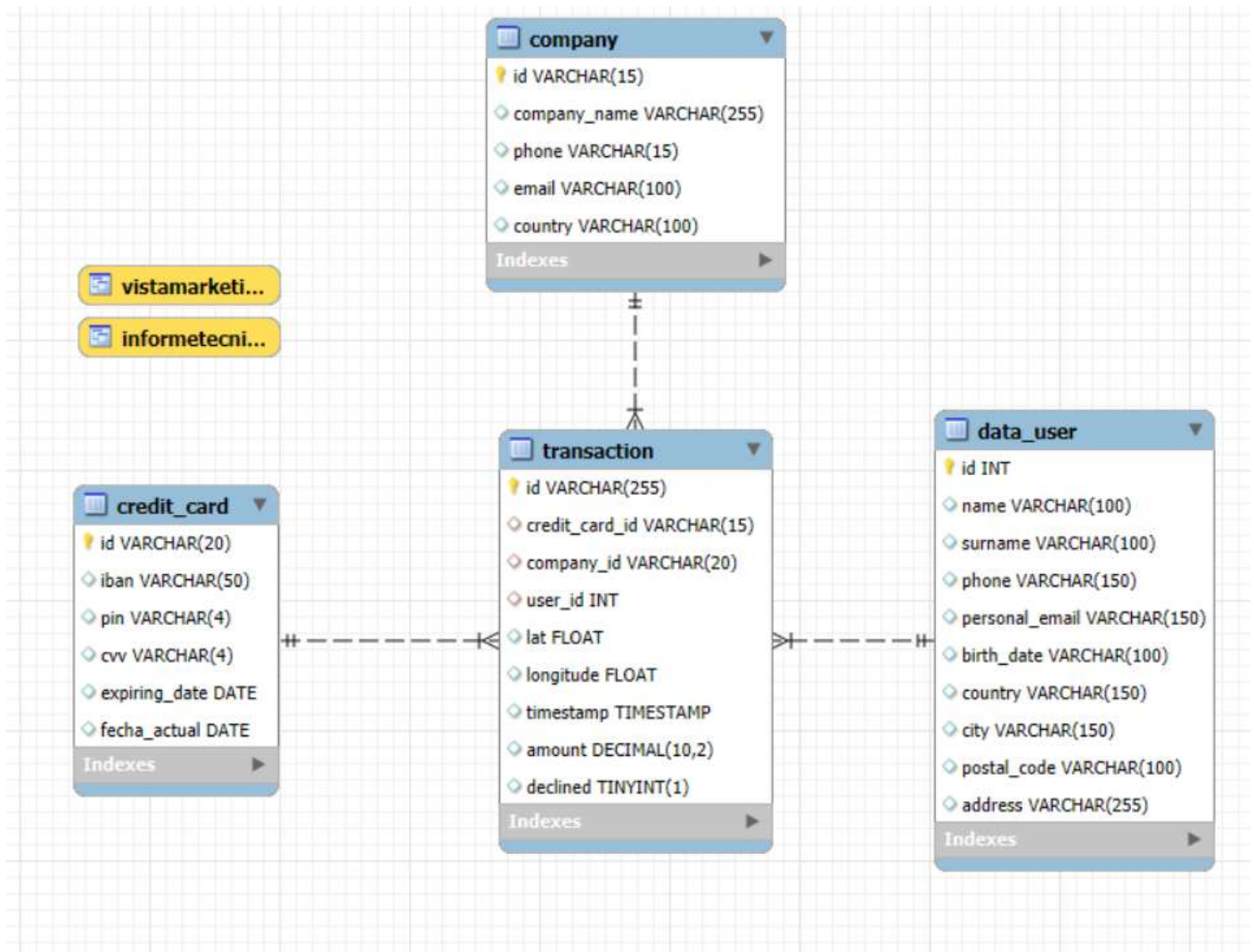
Result Grid										
	id	name	surname	phone	personal_email	birth_date	country	city	postal_code	address
1	Zeus	Gamble		1-282-581-0551	interdum.enim@protonmail.edu	Nov 17, 1985	United States	Lowell	73544	348-7818 Sagittis St.
2	Garrett	Mcconnell		(718) 257-2412	integer.vitae.nibh@protonmail.org	Aug 23, 1992	United States	Des Moines	59464	903 Sit Ave
3	Ciaran	Harrison		(522) 598-1365	interdum.feugiat@aol.org	Apr 29, 1998	United States	Columbus	56518	736-2063 Tellus St.
4	Howard	Stafford		1-411-740-3269	omare.egestas@icloud.edu	Feb 18, 1989	United States	Kailua	77417	Ap #545-2244 Erat. Rd.
5	Hayfa	Pierce		1-554-541-2077	et.malesuada.fames@hotmail.org	Sep 26, 1998	United States	Sandy	31564	341-2821 Ultrices Av.
6	Joel	Tyson		(718) 288-8020	gravida.nunc.sed@yahoo.ca	Oct 15, 1989	United States	Nashville	96838	888-2799 Amet Street
7	Rafael	Jimenez		(817) 689-0478	eget@outlook.ca	Dec 4, 1981	United States	Hillsboro	29874	8627 Malesuada Rd.
8	Nissim	Franks		(692) 157-3469	egestas.aliquam.fringilla@google.ca	Aug 1, 1993	United States	Jackson	61750	Ap #251-7144 Integer St.
9	Moshe	Medina		(500) 883-0104	phas.vestibulum@protonmail.org	Jan 24, 1983	United States	Richmond	39127	547 3080 Integer St.

data_user 1 x

Output

Action Output

#	Time	Action	Message
2	16:55:05	ALTER TABLE transaction ADD FOREIGN KEY (user_id) REFERENCES data_user(id)	587 row(s) affected Records: 587 Duplicates: 0 Warnings: 0
3	16:55:57	SELECT * FROM data_user LIMIT 0, 50000	276 row(s) returned



Nivel 3 - 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.

```
110 #Nivel 3 - Ejercicio 2
111
112 • CREATE VIEW `informetecnico` AS
113 SELECT transaction.id AS TransactionID, data_user.name AS UserName, data_user.surname AS UserSurname,
114 credit_card.iban AS Creditcardiban, company.company_name AS Companyname
115 FROM data_user
116 LEFT JOIN transaction ON data_user.id=transaction.user_id
117 RIGHT JOIN company ON transaction.company_id=company.id
118 RIGHT JOIN credit_card ON transaction.credit_card_id=credit_card.id;
119
120 • SELECT *
121 FROM transactions.informetecnico
122 ORDER BY TransactionID DESC;
```

TransactionID	UserName	UserSurname	Creditcardiban	Companyname
FE96CE47-BD59-381C-4E18-E3CA3D44E8FF	Kenyon	Hartman	DO26854763748537475216568689	Magna A Neque Industries
FE809ED4-2DB6-55AC-C915-929516E4646B	Molly	Gilliam	SE2813123487163628531121	Nunc Interdum Incorporated
FD9CBCCD-8E1E-8DA1-4606-7E3A6F3A5A65	Linus	Willis	KW9485332754781757886242955643	Nunc Interdum Incorporated
FD89D51B-AE8D-77DC-E450-B8083FBD3187	Hilda	Levy	LT053237077744561475	Malesuada PC
FD2E8957-414B-BEEC-E9AD-59AA7A8A6290	Hedwig	Gilbert	GE84848451582810541526	Neque Tellus Imperdiet Corp.
FCE2AB9A-271D-2BDC-9E49-8DD92A373391	Hakeem	Alford	MD1234119525145401270486	Nunc Interdum Incorporated
FBD7E0D6-BA6B-F5BC-0CA9-EA4B8760100C	Hedwig	Gilbert	MU4132333444534342541344788855	Mauris Id Inc.

informetecnico 2 x

Output

Action Output

#	Time	Action	Message
8	17:03:14	CREATE VIEW `informetecnico` AS SELECT transaction.id AS TransactionID, data_user.name AS User...	0 row(s) affected
9	17:03:17	SELECT * FROM transactions.informetecnico ORDER BY TransactionID DESC LIMIT 0, 50000	587 row(s) returned