

**José Messias Garcia da Silva Ferreira**  
SPRINT 3 - NIVEL 1

## **Exercici 1**

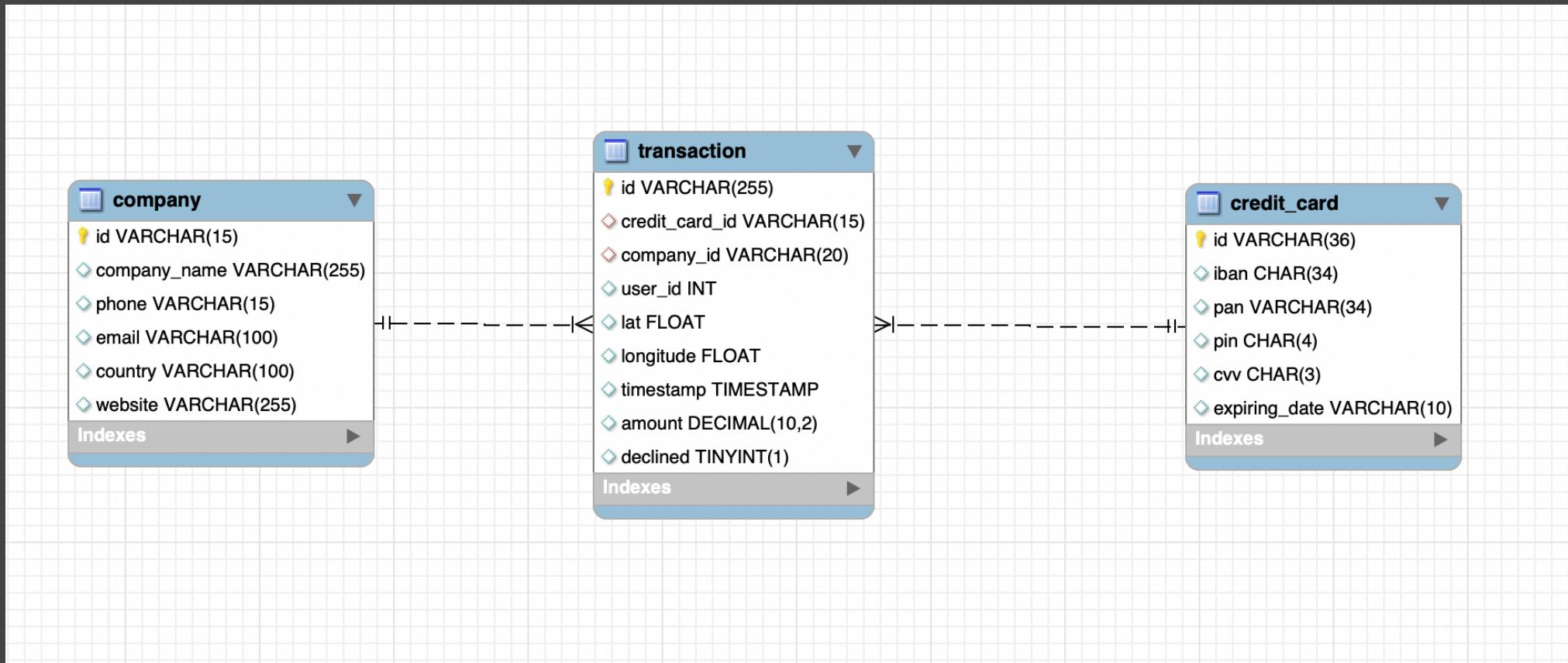
La teva tasca és dissenyar i crear una taula anomenada "credit\_card" que emmagatzemi detalls crucials sobre les targetes de crèdit. La nova taula ha de ser capaç d'identificar de manera única cada targeta i establir una relació adequada amb les altres dues taules ("transaction" i "company"). Després 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.

Primero de todo, he creado la estructura de la tabla, donde he puesto el **ID** como *primary key* de la tabla **credit\_card**, con formato **VARCHAR(36)**. Después defini los demás campos: **format** como **CHAR(34)**, **PAN** como **VARCHAR(34)**, **PIN** como **CHAR(4)** y **CVV** como **CHAR(3)**.

Al principio elegí **expiring\_date** como **VARCHAR**, pero de esta manera sería imposible realizar cálculos con fechas en ese formato.

Luego utilicé la sentencia **ALTER TABLE ... ADD FOREIGN KEY** para añadir la clave foránea en la tabla **transaction**, vinculándola con la *primary key* **id** de la tabla **credit\_card**, creando así una relación entre las tres tablas, donde la tabla **transaction** es la **tabla principal**.

Para evitar problemas al calcular las fechas con el formato **VARCHAR**, creé una columna temporal y copié en ella los datos de la antigua columna, utilizando la función **STR\_TO\_DATE**. De esta forma, transformé esa columna temporal en la nueva columna con formato **DATE**. Posteriormente eliminé la columna **expiring\_date** original y renombré la nueva con este nombre. Así, ahora tenemos la columna **expiring\_date** en formato **DATE**, que es el correcto para trabajar con fechas.



```

13
14  -- CREAR DATA_STRUCTURE
15
16 • ⊖ CREATE TABLE credit_card (
17      id VARCHAR(36) PRIMARY KEY,
18      iban CHAR(34),
19      pan VARCHAR(34),
20      pin CHAR(4),
21      cvv CHAR(3),
22      expiring_date VARCHAR(10)
23 );
24
25  -- CHECK DATA
26
27 •   SELECT * FROM credit_card;
28 •   SELECT * FROM transaction;
29 •   DESCRIBE credit_card;
30
31  -- ALTER FOREIGN KEY OF TRANSACTION, ADDING CREDIT_CARD_ID AS FOREIGN KEY.
32
33 •   ALTER TABLE transaction
34     ADD FOREIGN KEY(credit_card_id) REFERENCES credit_card(id);
35
36  -- ADD TEMPORARY COLUMN
37
38 •   ALTER TABLE credit_card
39     ADD COLUMN expiring_date_tmp DATE;

```

100% ◇ 19:18

Action Output ◇

	Time	Action	Response	Duration / Fetch Time
✖ 200218	06:38:04	DESCRIBE transactions	Error Code: 1146. Table 'transactions.transactions' doesn't exist	0.0027 sec
✓ 200219	06:38:17	DESCRIBE transaction	9 row(s) returned	0.0017 sec / 0.00001...
✓ 200220	06:38:32	CREATE TABLE credit_card ( id VARCHAR(36) PRIMARY KEY, iban CHAR(34), pan VAR...	0 row(s) affected	0.0085 sec

```

13
14    -- CREAR DATA_STRUCTURE
15
16 • CREATE TABLE credit_card (
17     id VARCHAR(36) PRIMARY KEY,
18     iban CHAR(34),
19     pan VARCHAR(34),
20     pin CHAR(4),
21     cvv CHAR(3),
22     expiring_date VARCHAR(10)
23 );
24
25    -- CHECK DATA
26
27 • SELECT * FROM credit_card;
28 • SELECT * FROM transaction;
29 • DESCRIBE credit card;

```

100% 16:27

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

id	iban	pan	pin	cvv	expiring_date
CcS-4857	XX4857591835292505850771	2314242385113924	1819	467	09/27/25
CcS-4858	XX8581768137002436094025	6582720299715533	3964	817	12/28/28
CcS-4859	XX7826930491423553609370	8861684536289642	4983	277	11/26/26
CcS-4860	XX5559590368835304645299	2481155515498456	6876	661	07/27/27
CcS-4861	XX2035182877195191627307	1308930301149557	5710	398	04/25/26
CcS-4862	XX4774721462463645409758	6715617009807829	4042	174	11/27/26
CcS-4863	XX1476829664245046207111	3140879819451394	5969	449	12/27/29
CcS-4864	XX8380298893385731196159	5793672133649114	8481	139	02/28/26

credit\_card 5 Apply

Action Output

	Time	Action	Response	Duration / Fetch Time
✓ 205220	06:39:46	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcS-9580', 'XX78125888...', ...)	1 row(s) affected	0.0018 sec
✓ 205221	06:39:46	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcS-9581', 'XX915670516...', ...)	1 row(s) affected	0.0021 sec
✓ 205222	06:39:56	SELECT * FROM credit_card	5000 row(s) returned	0.00070 sec / 0.0047...

```

25  -- CHECK DATA
26
27 •  SELECT * FROM credit_card;
28 •  SELECT * FROM transaction;
29 •  DESCRIBE credit_card;
30
31  -- ALTER FOREIGN KEY OF TRANSACTION, ADDING CREDIT_CARD_ID AS FOREIGN KEY.
32
33 •  ALTER TABLE transaction
34    ADD FOREIGN KEY(credit_card_id) REFERENCES credit_card(id);
35
36  -- ADD TEMPORARY COLUMN
37
38 •  ALTER TABLE credit_card
39    ADD COLUMN expiring_date_tmp DATE;
40
41  -- CONVERT VARCHAR TO DATE USING THE FUNCTION STR_TO_DATE ON THE NEW COLUMN EXPIRING_DATE
42
43 •  UPDATE credit_card
44    SET expiring_date_tmp = STR_TO_DATE(expiring_date, '%m/%d/%y');
45
46 •  DESCRIBE credit_card;
47
48  -- DELETE OLD COLUMN AFTER CHANGE DATATYPE
49
50 •  ALTER TABLE credit_card DROP COLUMN expiring_date;

```

100% | 20:33 |

#### Action Output

	Time	Action	Response	Duration / Fetch Time
✓ 205221	06:39:46	INSERT INTO credit_card (id, iban, pan, pin, cvv, expiring_date) VALUES ('CcS-9581', 'XX915670516...', ...)	1 row(s) affected	0.0021 sec
✓ 205222	06:39:56	SELECT * FROM credit_card	5000 row(s) returned	0.00070 sec / 0.0047...
✓ 205223	06:40:47	ALTER TABLE transaction ADD FOREIGN KEY(credit_card_id) REFERENCES credit_card(id)	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings: 0	1.370 sec

```

25 -- CHECK DATA
26
27 • SELECT * FROM credit_card;
28 • SELECT * FROM transaction;
29 • DESCRIBE credit_card;
30
31 -- ALTER FOREIGN KEY OF TRANSACTION, ADDING CREDIT_CARD_ID AS FOREIGN KEY.
32
33 • ALTER TABLE transaction
34 ADD FOREIGN KEY(credit_card_id) REFERENCES credit_card(id);
35
36 -- ADD TEMPORARY COLUMN
37
38 • ALTER TABLE credit_card
39 ADD COLUMN expiring_date_tmp DATE;
40
41 -- CONVERT VARCHAR TO DATE USING THE FUNCTION STR_TO_DATE ON THE NEW COLUMN EXPIRING_DATE
42
43 • UPDATE credit_card
44 SET expiring_date_tmp = STR_TO_DATE(expiring_date, '%m/%d/%y');
45
46 • DESCRIBE credit_card;
47
48 -- DELETE OLD COLUMN AFTER CHANGE DATATYPE
49
50 • ALTER TABLE credit_card DROP COLUMN expiring_date;
51

```

100% ◊ | 19:38 |

Action Output ◊

	Time	Action	Response	Duration / Fetch Time
✓ 205222	06:39:56	SELECT * FROM credit_card	5000 row(s) returned	0.00070 sec / 0.0047...
✓ 205223	06:40:47	ALTER TABLE transaction ADD FOREIGN KEY(credit_card_id) REFERENCES credit_card(id)	100000 row(s) affected Records: 100000 Duplicates: 0 Warnings: 0	1.370 sec
✓ 205224	06:41:15	ALTER TABLE credit_card ADD COLUMN expiring_date_tmp DATE	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.018 sec

```
26 •
27 •
28 • SELECT * FROM transaction;
29 • DESCRIBE credit_card;
30
31 -- ALTER FOREIGN KEY OF TRANSACTION, ADDING CREDIT_CARD_ID AS FOREIGN KEY.
32
33 • ALTER TABLE transaction
34   ADD FOREIGN KEY(credit_card_id) REFERENCES credit_card(id);
35
36 -- ADD TEMPORARY COLUMN
37
38 • ALTER TABLE credit_card
39   ADD COLUMN expiring_date_tmp DATE;
40
41 -- CONVERT VARCHAR TO DATE USING THE FUNCTION STR_TO_DATE ON THE NEW COLUMN EXPIRING_DATE
42
43 • UPDATE credit_card
44   SET expiring_date_tmp = STR_TO_DATE(expiring_date, '%m/%d/%y');
45
46 • DESCRIBE credit_card;
47
48 -- DELETE OLD COLUMN AFTER CHANGE DATATYPE
49
50 • ALTER TABLE credit_card DROP COLUMN expiring_date;
51
52 -- RENAME NEW COLUMN AND DATA TYPE TO OLD NAME
53
54 □ ALTER TABLE credit_card
100% 11:43
```

```

36 -- ADD TEMPORARY COLUMN
37
38 • ALTER TABLE credit_card
39   ADD COLUMN expiring_date_tmp DATE;
40
41 -- CONVERT VARCHAR TO DATE USING THE FUNCTION STR_TO_DATE ON THE NEW COLUMN EXPIRING_DATE
42
43 • UPDATE credit_card
44   SET expiring_date_tmp = STR_TO_DATE(expiring_date, '%m/%d/%y');
45
46 • DESCRIBE credit_card;
47
48 -- DELETE OLD COLUMN AFTER CHANGE DATATYPE
49
50 • ALTER TABLE credit_card DROP COLUMN expiring_date;
51
52 -- RENAME NEW COLUMN AND DATA TYPE TO OLD NAME

```

100% | 8:46

**Result Grid** Filter Rows:  Search Export:

Field	Type	Null	Key	Default	Extra
id	varchar(36)	NO	PRI	NULL	
iban	char(34)	YES		NULL	
pan	varchar(34)	YES		NULL	
pin	char(4)	YES		NULL	
cvv	char(3)	YES		NULL	
expiring_date	varchar(10)	YES		NULL	
expiring_date_tmp	date	YES		NULL	

Result 6

Action Output

	Time	Action	Response	Duration / Fetch Time
205224	06:41:15	ALTER TABLE credit_card ADD COLUMN expiring_date_tmp DATE	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.018 sec
205225	06:41:34	UPDATE credit_card SET expiring_date_tmp = STR_TO_DATE(expiring_date, '%m/%d/%y')	5000 row(s) affected Rows matched: 5000 Changed: 5000 Warnings: 0	0.130 sec
205226	06:41:48	DESCRIBE credit_card	7 row(s) returned	0.0016 sec / 0.00001...

```

39    ADD COLUMN expiring_date_tmp DATE;
40
41    -- CONVERT VARCHAR TO DATE USING THE FUNCTION STR_TO_DATE ON THE NEW COLUMN EXPIRING_DATE
42
43 • UPDATE credit_card
44     SET expiring_date_tmp = STR_TO_DATE(expiring_date, '%m/%d/%y');
45
46 • DESCRIBE credit_card;
47
48    -- DELETE OLD COLUMN AFTER CHANGE DATATYPE
49
50 • ALTER TABLE credit_card DROP COLUMN expiring_date;
51
52    -- RENAME NEW COLUMN AND DATA TYPE TO OLD NAME
53
54 • ALTER TABLE credit_card
55     CHANGE COLUMN expiring_date_tmp expiring_date DATE;
56
57    -- CHECK DATA
58
59 • DESCRIBE credit_card;
60 • SELECT * FROM credit_card;
61
62 /* 
63 EXERCISE 2 - LEVEL 1 - SPRINT 3
64 The Human Resources department has identified an error in the account number associated with the credit card with ID CcU-2938.
65 The information that must be displayed for this record is: TR323456312213576817600000 Remember to show that the change was made

```

100% 36:50 Action Output

	Time	Action	Response	Duration / Fetch Time
✓ 205225	06:41:34	UPDATE credit_card SET expiring_date_tmp = STR_TO_DATE(expiring_date, '%m/%d/%y')	5000 row(s) affected Rows matched: 5000 Changed: 5000 Warnings: 0	0.130 sec
✓ 205226	06:41:48	DESCRIBE credit_card	7 row(s) returned	0.0016 sec / 0.00001...
✓ 205227	06:42:05	ALTER TABLE credit_card DROP COLUMN expiring_date	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.015 sec

```

41 -- CONVERT VARCHAR TO DATE USING THE FUNCTION STR_TO_DATE ON THE NEW COLUMN EXPIRING_DATE
42
43 • UPDATE credit_card
44 SET expiring_date_tmp = STR_TO_DATE(expiring_date, '%m/%d/%y');
45
46 • DESCRIBE credit_card;
47
48 -- DELETE OLD COLUMN AFTER CHANGE DATATYPE
49
50 • ALTER TABLE credit_card DROP COLUMN expiring_date;
51
52 • DESCRIBE credit_card;
53
54 -- RENAME NEW COLUMN AND DATA TYPE TO OLD NAME
55
56 • ALTER TABLE credit_card
57 CHANGE COLUMN expiring_date tmp_expiring_date DATE;

```

100% 22:52

Result Grid    Filter Rows:  Search    Export:

Field	Type	Null	Key	Default	Extra
id	varchar(36)	NO	PRI	NULL	
iban	char(34)	YES		NULL	
pan	varchar(34)	YES		NULL	
pin	char(4)	YES		NULL	
cvv	char(3)	YES		NULL	
expiring_date_tmp	date	YES		NULL	

Result 7    Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
✓ 205226	06:41:48	DESCRIBE credit_card	7 row(s) returned	0.0016 sec / 0.00001...
✓ 205227	06:42:05	ALTER TABLE credit_card DROP COLUMN expiring_date	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.015 sec
✓ 205228	06:42:26	DESCRIBE credit_card	6 row(s) returned	0.0017 sec / 0.00001...

```

47
48  -- DELETE OLD COLUMN AFTER CHANGE DATATYPE
49
50 • ALTER TABLE credit_card DROP COLUMN expiring_date;
51
52 • DESCRIBE credit_card;
53
54  -- RENAME NEW COLUMN AND DATA TYPE TO OLD NAME
55
56 • ALTER TABLE credit_card
57    CHANGE COLUMN expiring_date_tmp expiring_date DATE;
58
59  -- CHECK DATA
60
61 • DESCRIBE credit_card;
62 • SELECT * FROM credit_card;
63
64 /*
65 EXERCISE 2 - LEVEL 1 - SPRINT 3
66 The Human Resources department has identified an error in the account number associated with the credit card with ID CcU-2938.
67 The information that must be displayed for this record is: TR323456312213576817699999. Remember to show that the change was made.
68 */
69
70 • SELECT *
71   FROM credit_card
72   WHERE id = 'CcU-2938';

```

100% ▾ | 19:56 |

Action Output ▾

	Time	Action	Response	Duration / Fetch Time
✓ 205227	06:42:05	ALTER TABLE credit_card DROP COLUMN expiring_date	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.015 sec
✓ 205228	06:42:26	DESCRIBE credit_card	6 row(s) returned	0.0017 sec / 0.00001...
✓ 205229	06:42:44	ALTER TABLE credit_card CHANGE COLUMN expiring_date_tmp expiring_date DATE	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.0091 sec

```

51
52 • DESCRIBE credit_card;
53
54 -- RENAME NEW COLUMN AND DATA TYPE TO OLD NAME
55
56 • ALTER TABLE credit_card
57   CHANGE COLUMN expiring_date_tmp expiring_date DATE;
58
59 -- CHECK DATA
60
61 • DESCRIBE credit_card;
62 • SELECT * FROM credit_card;
63
64 /* EXERCISE 2 - LEVEL 1 - SPRINT 3
65 The Human Resources department has identified an error in the account number associated with the credit card with ID CcU-2938.
66 The information that must be displayed for this record is: TR323456312213576817699999. Remember to show that the change was made.

```

100% 16:61

**Result Grid** Filter Rows: Search: Export:

Field	Type	Null	Key	Default	Extra
id	varchar(36)	NO	PRI	NULL	
iban	char(34)	YES		NULL	
pan	varchar(34)	YES		NULL	
pin	char(4)	YES		NULL	
cvv	char(3)	YES		NULL	
expiring_date	date	YES		NULL	

Result 8 Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
✓ 205228	06:42:26	DESCRIBE credit_card	6 row(s) returned	0.0017 sec / 0.00001...
✓ 205229	06:42:44	ALTER TABLE credit_card CHANGE COLUMN expiring_date_tmp expiring_date DATE	0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.0091 sec
✓ 205230	06:43:03	DESCRIBE credit_card	6 row(s) returned	0.0017 sec / 0.00001...

## Exercici 2

El departament de Recursos Humans ha identificat un error en el número de compte associat a la targeta de crèdit amb ID CcU-2938. La informació que ha de mostrarse per a aquest registre és: TR323456312213576817699999. Recorda mostrar que el canvi es va realitzar.

En este ejercicio he utilizado la sentencia **UPDATE** con el objetivo de modificar el número de cuenta (**iban**) en el registro cuyo **id** corresponde a "**CcU-2938**". Esto permite cambiar el valor de una o varias columnas en filas específicas de una tabla, siempre que se cumple la condición definida en la cláusula **WHERE**. Después ejecuté la instrucción **SELECT** para mostrar el valor actualizado en la tabla.

The screenshot shows the MySQL Workbench interface. The code editor contains the following SQL script:

```
57 •
58 •
59
60 • /* EXERCISE 2 - LEVEL 1 - SPRINT 3
61 • The Human Resources department has identified an error in the account number associated with the credit card with ID CcU-2938.
62 • The information that must be displayed for this record is: TR323456312213576817699999. Remember to show that the change was made.
63 • */
64
65
66 • SELECT *
67 FROM credit_card
68 WHERE id = 'CcU-2938';
69
70 • UPDATE credit_card SET iban = 'TR323456312213576817699999' WHERE id = 'CcU-2938';
71
72 • SELECT *
73 FROM credit_card
74 WHERE id = 'CcU-2938';
75
76 • /*
77 • EXERCISE 3 - LEVEL 1 - SPRINT 3
78 • In the 'transaction' table, insert a new user with the following information:
79 • id: 10881D1D-5B23-A76C-55EF-C568E49A99DD, credit_card_id: CcU-9999, company_id: b-9999, user_id: 9999, lat: 829.999,
80 • longitude: -117.999, amount: 111.11, declined: 0
81 • */
```

The results grid shows the current state of the **credit\_card** table:

id	iban	pin	cvv	expiring_date	fecha_actual
CcU-2938	TR323456312213576817699999	3257	984	2022-10-30	2025-09-25
HULL	HULL	HULL	HULL	HULL	HULL

Action Output:

Time	Action	Response	Duration / Fetch Time
26 11:01:12	DESCRIBE credit_card	6 row(s) returned	0.0039 sec / 0.00001...
27 11:03:48	SELECT * FROM credit_card WHERE id = 'CcU-2938'	1 row(s) returned	0.00050 sec / 0.0000...

### Exercici 3

En la taula "transaction" ingressa un nou usuari amb la següent informació:

Para evitar discrepancias, he creado **datos ficticios** de esta misma compañía en las tablas **credit\_card** y **company**. De esta forma, al utilizar **JOINS** por ejemplo, **no tendremos valores NULL** en esta transacción.

The screenshot shows a MySQL Workbench interface. The SQL editor contains the following code:

```
79
80     longitude: -117.999, amount: 111.11, declined: 0
81 */
82
83 • USE transactions;
84
85 • INSERT INTO transaction (id, credit_card_id, company_id, user_id, lat, longitude, timestamp,amount, declined)
86     VALUES ('108B1D1D-5B23-A6C-55EF-C568E49A99DD', 'CcU-9999', 'b-9999', 9999, 829.999, -117.999, (CURRENT_TIMESTAMP), 111.11, 0);
87
88 • INSERT INTO credit_card (id, iban, pin, cvv, expiring_date) VALUES ('CcU-9999', 'TR301950312213576817638661', '3257', '984', '2022-10-30');
89
90 • INSERT INTO company (id, company_name, phone, email, country, website) VALUES ('b-9999', 'Mondego Incorporated', '06 84 33 15 97', 'mondego@hotmail.net', 'Rio de Janeiro', 'Brazil');
91
92 • SELECT *
93     FROM transaction t
94     JOIN company c ON t.company_id = c.id
95     WHERE user_id = 9999;
96
97 • SELECT *
98     FROM transaction
99     WHERE user_id = 9999;
100
101    /*
```

The results grid displays the following data:

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined	id	company_name	phone	email	country
108B1D1D-5B23-A76C-55EF-C568E49A99DD	CcU-9999	b-9999	9999	829.999	-117.999	2025-09-24 16:02:52	111.11	0	b-9999	Mondego Incorporated	06 84 33 15 97	mondego...	Rio de Janeiro

Action Output:

Time	Action	Response	Duration / Fetch Time
28 11:17:59	SELECT DISTINCT country FROM company	16 row(s) returned	0.0015 sec / 0.00006...
29 11:32:20	SELECT * FROM transaction t JOIN company c ON t.company_id = c.id WHERE user_id = 9999	1 row(s) returned	0.00099 sec / 0.000...

#### Exercici 4

Des de recursos humans et sol·liciten eliminar la columna "pan" de la taula credit\_card. Recorda mostrar el canvi realitzat.

```
101  /*
102   EXERCISE 4 - LEVEL 1 - SPRINT 3
103   Human Resources has requested that you remove the "pan" column from the credit_card table. Remember to show the change made.
104 */
105
106 • ALTER TABLE credit_card DROP pan;
107 • SHOW COLUMNS FROM credit_card;
108
109
110
111
112
113
114
115
116
117
118
119
120
```

100% 1:119

Result Grid Filter Rows: Search Export:

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 20 Read Only

Action Output

Time	Action	Response	Duration / Fetch Time
29 11:32:20	SELECT * FROM transaction t JOIN company c ON t.company_id = c.id WHERE user_id = 9999	1 row(s) returned	0.00099 sec / 0.000...
30 11:34:08	SHOW COLUMNS FROM credit_card	6 row(s) returned	0.0019 sec / 0.00001...

**SPRINT 3 - NIVEL 2**

## Exercici 1

Elimina de la taula transaction el registre amb ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD de la base de dades.

The screenshot shows the MySQL Workbench interface. The SQL editor window contains the following code:

```
110 | EXERCISE 1 - LEVEL 2 - SPRINT 3
111 | Delete from the transaction table the record with ID 000447FE-B650-4DCF-85DE-C7ED0EE1CAAD from the database.
112 |
113
114 • SELECT *
115   FROM transaction
116   WHERE id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD';
117
118 • DELETE FROM transaction
119   WHERE id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD';
120
121
122
123
124
125
126
127
128
129
```

The Result Grid shows the transaction table with one row of data:

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

Action Output shows the history of actions:

Time	Action	Response	Duration / Fetch Time
30 11:34:08	SHOW COLUMNS FROM credit_card	6 row(s) returned	0.0019 sec / 0.00001...
31 11:36:04	SELECT * FROM transaction WHERE id = '000447FE-B650-4DCF-85DE-C7ED0EE1CAAD'	0 row(s) returned	0.00085 sec / 0.000...

## Exercici 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 creis 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.

```
-- TABLE REQUIRED BY MARKETING TEAM

142 • SELECT c.company_name, c.phone, c.country, ROUND(AVG(t.amount),2) AS avg_sales
143 FROM company c
144 JOIN transaction t ON c.id = t.company_id
145 WHERE t.declined = 0
146 GROUP BY c.company_name, c.phone, c.country
147 ORDER BY avg_sales DESC;
148
149 -- REQUIRED CREATE VIEW
150
151 • CREATE VIEW VistaMarketing AS
152 SELECT c.company_name, c.phone, c.country, ROUND(AVG(t.amount),2) AS avg_sales
153 FROM company c
154 JOIN transaction t ON c.id = t.company_id
155 WHERE t.declined = 0
156 GROUP BY c.company_name, c.phone, c.country
157 ORDER BY avg_sales DESC;
158
159 • SELECT *
160 FROM transactions.vistamarketing;
161
```

100% 1:161

Result Grid Filter Rows: Search Export:

company_name	phone	country	avg_sales
Ac Fermentum Incorporated	06 85 56 52 33	Germany	284.91
Pretium Neque Corp.	07 77 48 55 28	Australia	275.58
Urna Convallis Associates	06 01 24 77 04	United States	273.57
At Associates	09 58 61 10 65	New Zealand	272.74
Metus Vitae Associates	08 25 44 40 66	Australia	270.05
Aliquam Diam Limited	00 70 04 47 40	Lithuania	260.00

Result Grid Read Only

Vertical Output vistamarketing 22

Action Output

Time	Action	Response	Duration / Fetch Time
31 11:36:04	SELECT * FROM transaction WHERE id = '00044/FE-B650-4DCF-85DE-C7EDUEETCAAD'	0 row(s) returned	0.000085 sec / 0.000...
32 11:36:49	SELECT * FROM transactions.vistamarketing	101 row(s) returned	0.303 sec / 0.000021...

### Exercici 3

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

The screenshot shows a database interface with a query editor and a results grid.

**Query Editor:**

```
/*
EXERCISE 3 - LEVEL 2 - SPRINT 3
Filter the VistaMarketing view to show only the companies whose country of residence is 'Germany'
*/
171
172 • SELECT *
173   FROM transactions.vistamarketing
174 WHERE country = 'Germany';
175
176
177
178
179
180
181
182
183
184
185
186
187
```

**Result Grid:**

company_name	phone	country	avg_sales
Ac Fermentum Incorporated	06 85 56 52 33	Germany	284.91
Nunc Interdum Incorporated	05 18 15 48 13	Germany	259.32
Convallis In Incorporated	06 66 57 29 50	Germany	257.69
Ac Industries	09 34 65 40 60	Germany	255.17
Rutrum Non Inc.	02 66 31 61 09	Germany	255.14

**Action Output:**

Time	Action	Response	Duration / Fetch Time
32 11:36:49	SELECT * FROM transactions.vistamarketing	101 row(s) returned	0.303 sec / 0.000021...
33 11:37:22	SELECT * FROM transactions.vistamarketing WHERE country = 'Germany'	8 row(s) returned	0.053 sec / 0.000011...

**SPRINT 3 - NIVEL 3**

## Exercici 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:

En aquesta activitat, és necessari que descrigus el "pas a pas" de les tasques realitzades. És important realitzar descripcions senzilles, simples i fàcils de comprendre. Per a realitzar aquesta activitat hauràs de treballar amb els arxius denominats "estructura\_dades\_user" i "dades\_introducir\_user". Recorda continuar treballant sobre el model i les taules amb les quals ja has treballat fins ara.

Para ayudar a mi compañero a recordar cómo se han creado las tablas, he utilizado la cláusula **SHOW CREATE**, que nos devuelve la query utilizada en la creación de cada tabla y también el tipo de dato de cada campo.

En primer lugar, creamos la estructura de la tabla **transaction** y agregamos todas las transacciones mediante la cláusula **INSERT INTO**. A continuación, hicimos lo mismo con la tabla **company**.

Posteriormente, añadimos la tabla **credit\_card**, cuya query de creación también se puede consultar mediante **SHOW CREATE TABLE credit\_card**, y añadimos su **FOREIGN KEY** utilizando la cláusula **ALTER TABLE** para relacionarla con la tabla **transaction**, que es nuestra **tabla principal**.

Realizamos algunos cambios necesarios, como añadir transacciones, eliminar la columna *pan* y modificar el tipo de dato del campo *expiring\_date* a **DATE**.

Después, utilizamos la cláusula **CREATE VIEW** para crear una vista, que funciona como una query almacenada (*stored query*), con el objetivo de presentar al equipo de marketing únicamente los campos necesarios.

Creamos la estructura de la tabla **user** e insertamos sus 5.000 registros. Sin embargo, en la estructura original de la tabla **user** no se había definido la **FOREIGN KEY** hacia la tabla principal **transaction**. Por ello, utilizamos **ALTER TABLE MODIFY COLUMN** para ajustar el tipo de dato de la **PRIMARY KEY** de **user** y hacerlo coincidir con el tipo de dato de **user\_id** en la tabla **transaction**. Posteriormente, usamos **ALTER TABLE ADD FOREIGN KEY** para relacionar el campo **id** (PRIMARY KEY de **user**) con el campo **user\_id** (FOREIGN KEY en la tabla principal).

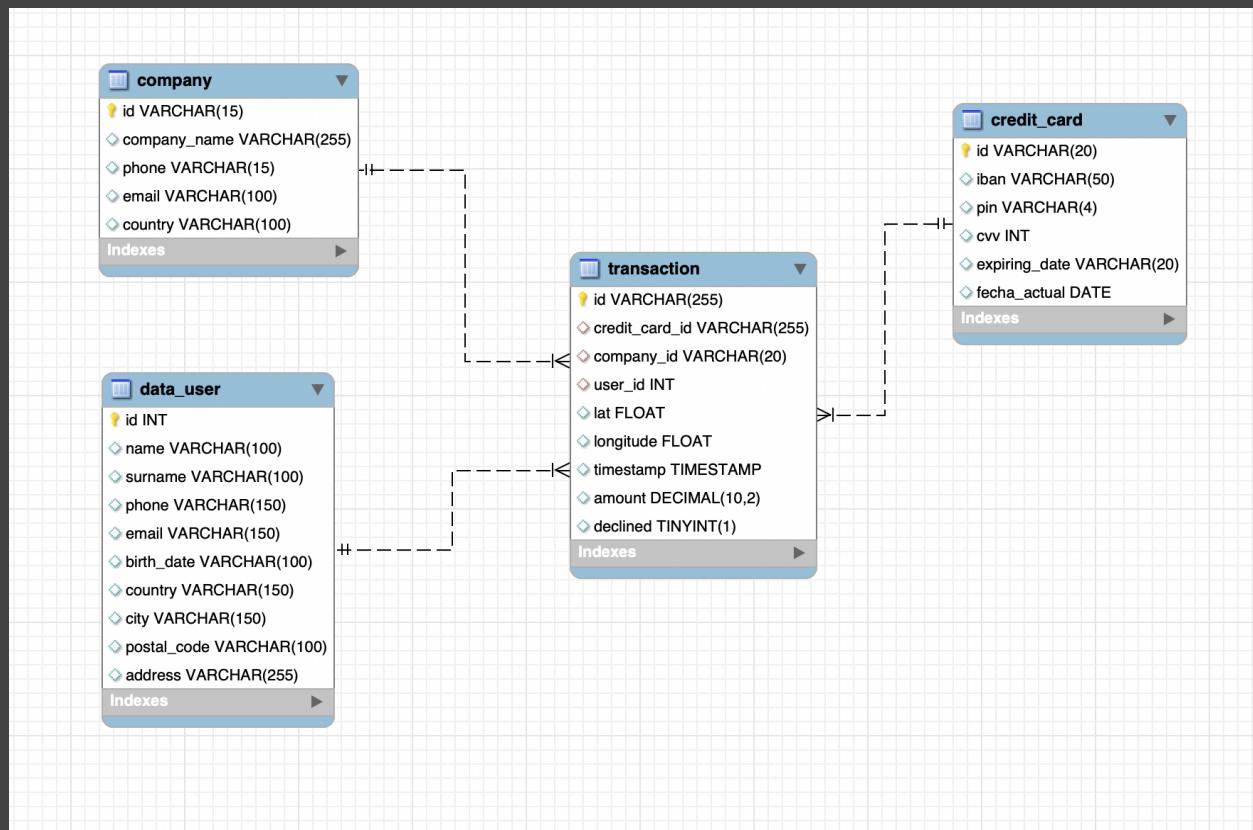
Cumpliendo con lo solicitado, cambiamos el nombre de la tabla **user** a **data\_user** mediante la cláusula **RENAME TABLE**.

En la tabla **company** eliminamos el campo **website**.

Además, modificamos algunos tipos de datos en varias tablas: en la tabla **transaction**, cambiamos el tipo de dato de los campos *id* y *credit\_card\_id* a **VARCHAR(255)**. En la tabla **credit\_card**, cambiamos el tipo de dato de *id* a **VARCHAR(20)**, *iban* a **VARCHAR(50)**, *pin* a **VARCHAR(4)** y *cvv* a **INT**. También modificamos el campo *expiring\_date* para que fuera de tipo **VARCHAR(20)**.

Finalmente, añadimos una nueva columna en la tabla **credit\_card** llamada *fecha\_actual* y utilizamos la función **CURDATE()** para asignar automáticamente la fecha actual como valor por defecto.

Todos los cambios se encuentran documentados con comentarios en el archivo **sprint\_3.sql**.



```
sprint_3* | SQL File 5 | 180
181
182 -- CHECK QUERIES USED TO CREATE TABLES
183
184 • SHOW CREATE TABLE transaction;
185 • SHOW CREATE TABLE company;
186 • SHOW CREATE TABLE credit_card;
187 • SELECT * FROM user;
188
189 -- CHANGE DATATYPE OF FOREIGN KEY, COLUMN USER.ID
190
191 • ALTER TABLE user
192   MODIFY COLUMN id INT;
193
194 -- CREATE USERS TABLE FOREIGN KEY
195
196 • ALTER TABLE transaction
197   ADD FOREIGN KEY(user_id) REFERENCES user(id);
198
199 -- CHANGE TABLE NAME
200
201 • RENAME TABLE user TO data_user;
202
203 -- DELETE WEBSITE COLUMN FROM COMPANY TABLE
100% 17:187 |
```

Result Grid    Filter Rows:  Search    Export:

Table	Create Table
transaction	CREATE TABLE `transaction` ( `id` varchar(255) NOT NULL, `credit_card_id` varchar(255) DEFAULT NULL, `company_id` varchar(20) DEF...

Result Grid    Form Editor

Vertical Output    Result 24    Read Only

Action Output

Time	Action	Response	Duration / Fetch Time
33 11:37:22	SELECT * FROM transactions.vistamarketing WHERE country = 'Germany'	8 row(s) returned	0.053 sec / 0.000011...
34 11:38:17	SHOW CREATE TABLE transaction	1 row(s) returned	0.0064 sec / 0.0000...

```
200
201 •   ⚡ ⚡ 🔎 🛡 🔍 🗃 🗁 🗂 Don't Limit
202
203 -- DELETE WEBSITE COLUMN FROM COMPANY TABLE
204
205 •   SELECT * FROM company;
206 •   ALTER TABLE company DROP website;
207 •   SHOW COLUMNS FROM company;
208
209 -- CHANGE DATATYPE OF TRANSACTION.ID TO VARCHAR(255)
210
211 •   ALTER TABLE transaction
212     MODIFY COLUMN id VARCHAR(255);
213
214 -- CHANGE DATATYPE OF TRANSACTION.CREDIT_CARD_ID TO VARCHAR(255)
215
216 •   ALTER TABLE transaction
217     MODIFY COLUMN credit_card_id VARCHAR(255);
218
219 -- CHANGE DATATYPE OF CREDIT_CARD.ID TO VARCHAR(20)
220
221 •   ALTER TABLE credit_card
222     MODIFY COLUMN id VARCHAR(20);
223
```

Result Grid  Filter Rows:  Search

Export: 

**Result**  
Grid

Table	Create Table
transaction	CREATE TABLE `transaction` ( `id` varchar(255) NOT NULL, `credit_card_id` varchar(255) DEFAULT NULL, `company_id` varchar(20) DEF... ...

Vertical Output

Result 24

Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
✓ 33	11:37:22	SELECT * FROM transactions.vistamarketing WHERE country = 'Germany'	8 row(s) returned	0.053 sec / 0.00011...
✓ 34	11:38:17	SHOW CREATE TABLE transaction	1 row(s) returned	0.0064 sec / 0.0000...

```
220
221 • MODIFY COLUMN id VARCHAR(20);
222
223
224 -- CHANGE DATATYPE OF CREDIT_CARD.IBAN TO VARCHAR(50)
225
226 • ALTER TABLE credit_card
227   MODIFY COLUMN iban VARCHAR(50);
228
229 -- CHANGE DATATYPE OF CREDIT_CARD.PIN TO VARCHAR(04)
230
231 • ALTER TABLE credit_card
232   MODIFY COLUMN pin VARCHAR(04);
233
234 -- CHANGE DATATYPE OF CREDIT_CARD.CVV TO INT
235
236 • ALTER TABLE credit_card
237   MODIFY COLUMN cvv INT;
238
239 -- CHANGE DATATYPE OF EXPIRING_DATE.CREDIT_CARD TO VARCHAR(20)
240
241 • ALTER TABLE credit_card
242   MODIFY COLUMN expiring_date VARCHAR(20);
243
```

100% 17:187

Result Grid   Filter Rows:  Search   Export:

Table Create Table

transaction	CREATE TABLE `transaction` ( `id` varchar(255) NOT NULL, `credit_card_id` varchar(255) DEFAULT NULL, `company_id` varchar(20) DEF...
-------------	--

Result Grid   Form Editor

Vertical Output   Result 24   Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
33	11:37:22	SELECT * FROM transactions.vistamarketing WHERE country = 'Germany'	8 row(s) returned	0.053 sec / 0.000011...
34	11:38:17	SHOW CREATE TABLE transaction	1 row(s) returned	0.0064 sec / 0.0000...

```
243
244 -- ADD NEW COLUMN FECHA_ACTUAL DATE TO CREDIT_CARD TABLE
245
246 • ALTER TABLE credit_card
247   ADD COLUMN fecha_actual DATE;
248
249 • UPDATE credit_card
250   SET fecha_actual = CURDATE();
251
252 • SELECT *
253   FROM credit_card
254   LIMIT 3;
255
256 • DESCRIBE data_user;
257 • DESCRIBE transactions;
258 • DESCRIBE company;
259 • DESCRIBE credit_card;
```

```
260
261
262
```

```
100% | 16:259 |
```

Result Grid Filter Rows:  Search Export:

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	

Vertical Output Result 25 Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
34	11:38:17	SHOW CREATE TABLE transaction	1 row(s) returned	0.00064 sec / 0.0000...
35	11:39:30	DESCRIBE credit_card	6 row(s) returned	0.0012 sec / 0.00001...

**Exercici 2**

L'empresa també us demana 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.
- Assegureu-vos d'incloure informació rellevant de les taules que coneixereu i utilitzeu àlies per canviar de nom columnes segons calgui.

Mostra els resultats de la vista, ordena els resultats de forma descendent en funció de la variable ID de transacció.

```

270
271     Make sure to include relevant information from the tables you know and use aliases to rename columns as needed.
272     Display the results of the view, ordering them in descending order based on the Transaction ID.
273
274     */
275
276     -- VIEW REQUIRED
277
278 • CREATE VIEW InformeTecnico AS
279     SELECT t.id, du.name AS first_name, du.surname AS last_name, cc.iban, c.company_name
280     FROM transaction t
281     JOIN company c ON t.company_id = c.id
282     JOIN data_user du ON t.user_id = du.id
283     JOIN credit_card cc ON t.credit_card_id = cc.id
284     ORDER BY t.id DESC;
285
286 • SELECT *
287     FROM transactions.InformeTecnico;
288
289

```

100% | 10:287

Result Grid    Filter Rows:    Search    Export:    Fetch rows:

id	first_name	last_name	iban	company_name
FFF9E8D-27C7-4ADE-98F2-7533EF4DF126	Securp	Faofovfy	XX162677143304223631437567	Nunc Interdum Incorporated
FFF8270D-F53A-4D5D-9666-E5307C53CC84	Ggzjpa	Uirzjulh	XX395114267082019952567052	Viverra Donec Foundation
FFF9E3CE-234E-408C-A8EF-F9CAD577224A	Yshimq	Zpsjsleel	XX8845462156537570367941	Convallis In Incorporated
FFF9E178-6CD2-4DF9-99B0-49AE068809B1	Jevepx	Xwczwmm	XX321405515711654384711481	Mus Aenean Eget Foundation
FFF867C9-17B5-4B1F-AFD9-F8023AAA449E	Fqlngd	Lvhfgyxi	XX278446342932680979729426	Cras Vehicula Aliquet Industries
FFF7042D-18C6-4DDD-B23C-4D90A4AC8F26	Njoraa	Egsqcuil	XX405009272572550082027209	Placerat LLP
FFF660D4-4244-47F6-9210-E5D1DCB99DB0	Lopzaj	Itgryfay	XX63376859736627454015125	Pede Cum Ltd

Result Grid    Form Editor

Vertical Output    InformeTecnico 26    Read Only

Action Output

	Time	Action	Response	Duration / Fetch Time
✓ 35	11:39:30	DESCRIBE credit_card	6 row(s) returned	0.0012 sec / 0.00001...
✓ 36	11:41:18	SELECT * FROM transactions.InformeTecnico	99999 row(s) returned	0.671 sec / 0.034 sec