

# Tarea S4.01. Creación de Base de Datos

## Nivel 1

Descarga los archivos CSV, estudiales y diseña una base de datos con un esquema de estrella que contenga, al menos 4 tablas de las que puedas realizar las siguientes consultas:

### Creando la Base de datos y las tablas de dimensiones.

**Tabla de usuarios.** En esta tabla se combinarán los dos archivos (American\_users y European\_users) para evitar problemas con las cléveas foráneas.

```
4 • USE VentasBD;
5   -- Creando una tabla con todos los usuarios para evitar problemas con la FK en la tabla Transactions
6 • CREATE TABLE D_Users (
7     id INT,
8     name VARCHAR(100),
9     surname VARCHAR(100),
10    phone VARCHAR(150),
11    email VARCHAR(150),
12    birth_date VARCHAR(100),
13    country VARCHAR(100),
14    city VARCHAR(150),
15    postal_code VARCHAR(100),
16    address VARCHAR(255),
17    PRIMARY KEY (id)
18 );
```

### Tabla Companies.

```
19
20 • USE VentasBD;
21   -- creando tabla Companies
22 • ○ CREATE TABLE Companies (
23     company_id VARCHAR(15),
24     company_name VARCHAR(255),
25     phone VARCHAR(15),
26     email VARCHAR(150),
27     country VARCHAR(100),
28     website VARCHAR(255),
29     PRIMARY KEY (company_id)
30   );
31
```

### Tabla Credit\_cards

```
32 • USE VentasBD;
33   -- creando tabla Credit_card
34 • ○ CREATE TABLE Credit_cards (
35     id VARCHAR(15),
36     user_id INT,
37     iban VARCHAR(50),
38     pan VARCHAR(19),
39     pin VARCHAR(4),
40     cvv VARCHAR(3),
41     track1 VARCHAR(100),
42     track2 VARCHAR(100),
43     expiring_date VARCHAR(10),
44     PRIMARY KEY (id)
45   );
```

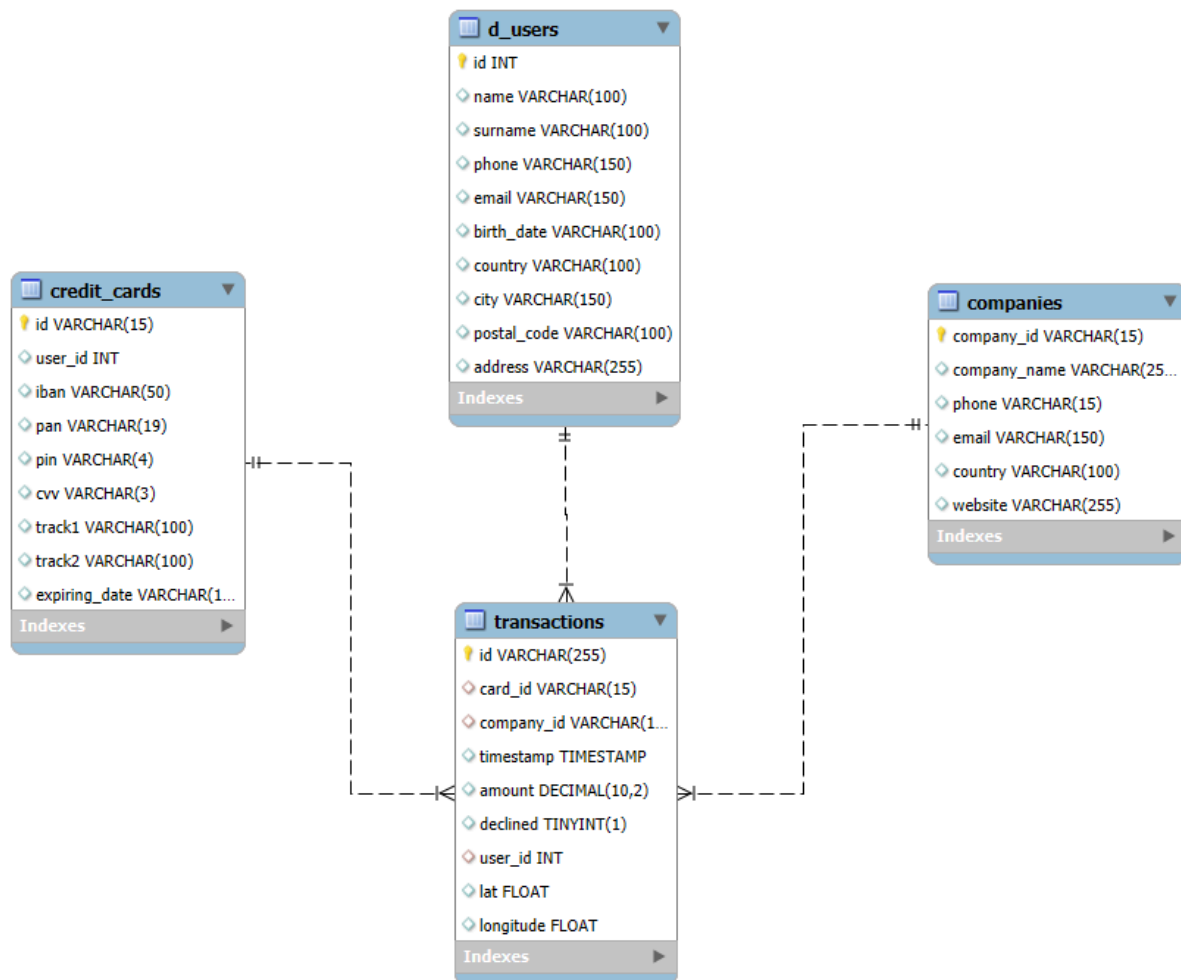
---

## Tabla de hechos Transactions.

```
47 • USE VentasBD;
48 • CREATE TABLE Transactions (
49     id VARCHAR(255) ,
50     card_id VARCHAR(15),
51     company_id VARCHAR(15),
52     timestamp timestamp,
53     amount DECIMAL(10,2),
54     declined BOOLEAN,
55     user_id INT,
56     lat FLOAT,
57     longitude FLOAT,
58     PRIMARY KEY (id),
59     CONSTRAINT fk_card_id FOREIGN KEY (card_id) REFERENCES Credit_cards (id),
60     CONSTRAINT fk_company_id FOREIGN KEY (company_id) REFERENCES Companies (company_id),
61     CONSTRAINT fk_D_users FOREIGN KEY (user_id ) REFERENCES D_users (id)
62 );
```

Nota. He omitido por el momento la tabla de productos y en la tabla de Transactions el campo de product\_ids para generar un modelo de estrella con relación 1:N.

## Modelo relacional de la base de datos VentasBD



Esta base de datos recoge información sobre las compras que realizan distintos clientes usando sus tarjetas de crédito. Cada vez que alguien hace una compra, se guarda quién la hizo, con qué tarjeta, en qué empresa, cuánto gastó, en qué lugar y si la operación fue aceptada o rechazada. Además, se almacenan los datos de los usuarios, sus tarjetas, y las empresas donde compran. Esta información está organizada en diferentes tablas que se conectan entre sí formando un modelo de estrella.

## Carga de Datos.

### Tabla companies

```
--
66 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\companies.csv'
67 INTO TABLE companies
68 FIELDS TERMINATED BY ',' -- Aquí se pone el separador del docu
69 ENCLOSED BY '"' -- Si usan comillas los campos
70 LINES TERMINATED BY '\\n' -- Salto de línea
71 IGNORE 1 LINES -- Para omitir la primera línea con títulos de la cabecera
72 (company_id,company_name,phone,email,country,website); -- Nombres de las columnas
73
```

### Tabla credit\_cards

```
75 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\credit_cards.csv'
76 INTO TABLE credit_cards
77 FIELDS TERMINATED BY ','
78 ENCLOSED BY '"'
79 LINES TERMINATED BY '\\n'
80 IGNORE 1 LINES
81 (id,user_id,iban,pan,pin, cvv, track1, track2, expiring_date);
82
```

**Tabla d\_users** (los dos archivos de usuarios en una misma tabla, por que al crear la tabla de transactions las Foreign Keys me percaté que tendría problemas con dos tablas de users y solo una FK)

```
83 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\european_users.csv'
84 INTO TABLE d_users
85 FIELDS TERMINATED BY ','
86 ENCLOSED BY '"'
87 LINES TERMINATED BY '\\n'
88 IGNORE 1 LINES
89 (id, name, surname, phone, email, birth_date, country, city, postal_code, address);
90
```

```

93 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\american_users.csv'
94 INTO TABLE d_users
95 FIELDS TERMINATED BY ','
96 ENCLOSED BY '"'
97 LINES TERMINATED BY '\\n'
98 IGNORE 1 LINES
99 (id, name, surname, phone, email, birth_date, country, city, postal_code, address);
100

```

## Tabla Transactions

He declarado un *dummy* (@dummy) para indicar que esos datos, que coinciden con products\_id, no se inserten en la tabla.

```

101 • LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\transactions.csv'
102 INTO TABLE Transactions
103 FIELDS TERMINATED BY ';'
104 LINES TERMINATED BY '\\n'
105 IGNORE 1 LINES
106 (id, card_id, company_id, timestamp, amount, declined, @dummy, user_id, lat, longitude);
107

```

## Comprobaciones

### Companies

The screenshot displays the MySQL Workbench interface. On the left, the 'Schemas' pane shows the 'ventasdb' database selected, with a tree view of its tables: companies, credit\_cards, d\_users, and transactions. The main area shows the 'companies' table structure and data. The table has 6 columns: company\_id, company\_name, phone, email, country, and website. The data is as follows:

company_id	company_name	phone	email	country	website
b-2222	Ac Fermentum Incorporated	06 85 56 52 33	donec.porttitor.tellus@yahoo.net	Germany	https://instagram.com/site
b-2226	Magna A Neque Industries	04 14 44 64 62	risus.donec.nibh@icloud.org	Australia	https://whatsapp.com/group/9
b-2230	Fusce Corp.	08 14 97 58 85	risus@protonmail.edu	United States	https://pinterest.com/sub/cars
b-2234	Convallis In Incorporated	06 66 57 29 50	mauris.ut@aol.couk	Germany	https://cnn.com/user/110
b-2238	Ante Iaculis Nec Foundation	08 23 04 99 53	sed.dictum.proin@outlook.ca	New Zealand	https://netflix.com/settings
b-2242	Donec Ltd	01 25 51 37 37	at.iaculis@hotmail.couk	Norway	https://nytimes.com/user/110
b-2246	Sed Nunc Ltd	02 62 64 73 48	nibh@yahoo.org	United Kingdom	https://cnn.com/one
b-2250	Amet Nulla Donec Corporation	07 15 25 14 74	mattis.integer.eu@protonmail.net	Italy	https://netflix.com/sub/cars
b-2254	Nascetur Ridiculus Mus Inc.	06 26 87 61 84	suspendisse.dui@icloud.net	United States	https://ebay.com/sub
b-2258	Vestibulum Lorem PC	02 02 87 33 40	aenean.massa.integer@aol.net	Belgium	https://pinterest.com/sub/cars
b-2262	Gravida Saeittis LLP	03 81 28 33 97	turois.vitae@ooodle.ca	Sweden	https://naver.com/site

The 'Output' pane at the bottom shows the execution of a 'SELECT \* FROM companies;' query. The first execution at 11:56:40 resulted in an error: 'Error Code: 1046. No database selected Select the default DB to be used'. The second execution at 11:56:55 was successful, returning 100 rows.

## Credit\_cards

Database Explorer: **ventasbd** (Schema: **ventasbd**)

- Tables
  - companies
  - credit\_cards
  - d\_users
  - transactions
- Views
- Stored Procedures
- Functions

SQL Editor:

```
108 SELECT *
109 FROM credit_cards;
110 SELECT *
111 FROM d_users;
112 SELECT *
```

Result Grid:

id	user_id	iban	pan	pin	cvv	track1	track2	expiring_date
CS-4857	276	XX4857591835292505850771	2314242385113924	1819	467	%62314242385113924*%LWCBUDLWCBUD*22...	%62314242385113924=24101015183631649	09/27/25
CS-4858	277	XX8581768137002436094025	6582720299715533	3964	817	%66582720299715533*%TIQNM%TIQNM%*2404...	%66582720299715533=2411101045462727	12/28/28
CS-4859	278	XX7826930491423553609370	8861684536289642	4983	277	%68861684536289642*%COPBDCOPBDC*280...	%68861684536289642=25021017616653717	11/26/26
CS-4860	279	XX5559590368835304645299	2481155515498459	6876	661	%62481155515498459*%TLJTUTLJTUT*31040...	%62481155515498459=26021015144143957	07/27/27
CS-4861	280	XX2035182877195191627307	1308930301149557	5710	398	%61308930301149557*%POBNZ*POBNZ*330...	%61308930301149557=28051017513050289	04/25/26
CS-4862	281	XX47747421462463645409758	6715617009807829	4042	174	%66715617009807829*%LDMW%LDMW%*33...	%66715617009807829=22101017023704289	11/27/26
CS-4863	282	XX1476829664245046207111	3140879819451394	5969	449	%63140879819451394*%OXJODOXJOD*230...	%63140879819451394=32101011586485999	12/27/29
CS-4864	283	XX8380298893385731196159	5793672133649114	8481	139	%65793672133649114*%NHWBYRNHWBYR*30...	%65793672133649114=23061018063671017	02/28/26
CS-4865	284	XX7085078596101025280599	5101552687251312	7847	903	%65101552687251312*%MJODH*MJODH*330...	%65101552687251312=30101016648627107	11/25/28
CS-4866	285	XX4792859188206596406839	8080768801072613	9271	961	%68080768801072613*%MBGOZ*MBGOZ*321...	%68080768801072613=28101017158856957	02/28/25
CS-4867	286	XX6038298816319374853717	7761849537661098	4820	862	%67761849537661098*%LDRSOLLDRSOL*2811...	%67761849537661098=22061017104113717	11/30/25

Output:

#	Time	Action	Message
1	11:56:40	SELECT * FROM companies	Error Code: 1046. No database selected Select the default DB to be used by double-clicking its name in the S...
2	11:56:55	SELECT * FROM companies	100 row(s) returned
3	11:58:09	SELECT * FROM credit_cards	5000 row(s) returned

## d\_users

Database Explorer: **ventasbd** (Schema: **ventasbd**)

- Tables
  - companies
  - credit\_cards
  - d\_users
  - transactions
- Views
- Stored Procedures
- Functions

SQL Editor:

```
110 SELECT *
111 FROM d_users;
112 SELECT *
113 FROM transactions;
114
115 #EJERCICIO 1
```

Result Grid:

id	name	surname	phone	email	birth_date	country	city	postal_code	address
1	Zeus	Gamble	1-282-581-0551	interdum.enim@protonmail.edu	Nov 17, 1985	United States	New York	10001	348-7818 Sagittis St.
2	Garrett	Mconnell	(718) 257-2412	integer.vitae.nibh@protonmail.org	Aug 23, 1992	United States	Philadelphia	19101	903 Sit Ave
3	Ciaran	Harrison	(522) 598-1365	interdum.feugiat@aol.org	Apr 29, 1998	United States	Houston	77001	736-2063 Tellus St.
4	Howard	Stafford	1-411-740-3269	omare.egestas@cloud.edu	Feb 18, 1989	United States	Phoenix	85001	Ap #545-2244 Erat. Rd.
5	Hayfa	Pierce	1-554-541-2077	et.malesuada.fames@hotmail.org	Sep 26, 1998	United States	Philadelphia	19101	341-2821 Ultrices Av.
6	Joel	Tyson	(718) 288-8020	gravida.nunc.sed@yahoo.ca	Oct 15, 1989	United States	San Jose	95101	888-2799 Amet Street
7	Rafael	Jimenez	(817) 689-0478	eget@outlook.ca	Dec 4, 1981	United States	Chicago	60601	8627 Malesuada Rd.
8	Nissim	Franks	(692) 157-3469	egestas.aliquam.fringilla@google.ca	Aug 1, 1993	United States	New York	10001	Ap #251-7144 Integer St.
9	Mannix	Mcclain	(590) 883-2184	aliquam.nisi@outlook.com	Jan 24, 1987	United States	San Antonio	78201	647-3080 Lacus. St.
10	Robert	Mccarthy	(324) 746-6771	fermentum@protonmail.com	Apr 30, 1984	United States	San Jose	95101	P.O. Box 773
11	Joan	Baird	(981) 429-8106	et@outlook.net	Feb 25, 1990	United States	Los Angeles	90001	P.O. Box 687

Output:

#	Time	Action	Message
1	11:56:40	SELECT * FROM companies	Error Code: 1046. No database selected Select the default DB to be used by d
2	11:56:55	SELECT * FROM companies	100 row(s) returned
3	11:58:09	SELECT * FROM credit_cards	5000 row(s) returned
4	11:59:01	SELECT * FROM d_users	5000 row(s) returned

## Transactions

Schema: ventasdb

```

111 FROM d_users)
112 SELECT *
113 FROM transactions;
114
115 #EJERCICIO 1

```

id	card_id	company_id	timestamp	amount	declined	user_id	lat	longitude
00043A49-2949-494B-A5DD-ASBAE38819DD	Cc5-9294	b-2458	2024-08-28 07:16:46	395.43	0	4713	46.1999	1.43554
000447FE-8650-4DCF-85DE-C7ED0EE1CAAD	Cc5-5019	b-2370	2016-12-21 20:07:18	155.63	0	438	41.5972	12.2218
00045D6B-ED2E-4F2F-8186-CEE074D875D0	Cc5-6699	b-2390	2020-07-14 15:37:45	326.01	0	2118	29.7573	-95.3796
000481C3-1C26-4FEF-83AD-4CD0EB0046BD	Cc5-6696	b-2230	2017-09-04 19:44:53	161.60	0	2115	53.5489	-113.503
00051AA4-9CBE-4268-B070-C38062A1B3E2	Cc5-7606	b-2266	2017-01-05 18:19:25	148.91	0	3025	52.2084	5.69081
0008A312-EDFE-4A4F-BC99-E9C92EC3CA4D	CcU-3358	b-2598	2023-09-23 04:51:43	294.59	0	215	53.5535	-113.499
0009A151-9BCF-4E31-9053-A468FF77FAAB	Cc5-7509	b-2546	2023-12-31 00:06:36	383.63	0	2928	51.9362	5.34265
0009D494-6245-4DF9-955D-2C084191CFFB	Cc5-8483	b-2526	2017-07-18 07:52:02	197.80	0	3902	45.492	-73.5706
000A1DEC-CDB6-4AB2-A619-71DAB80-4A262	Cc5-6467	b-2558	2018-09-08 05:29:58	339.94	0	1886	55.7425	-3.30009
000A1E64-1414-40B0-9092-5678A4D958E2	Cc5-5966	b-2550	2022-09-17 04:02:19	369.71	0	1385	52.0821	5.28424
000A5879-3472-41D9-AF60-42D3503B543C	CcU-4569	b-2590	2020-02-07 23:03:45	162.43	0	42	39.949	-75.1719

#	Time	Action	Message
2	11:56:55	SELECT * FROM companies	100 row(s) returned
3	11:58:09	SELECT * FROM credit_cards	5000 row(s) returned
4	11:59:01	SELECT * FROM d_users	5000 row(s) returned
5	12:00:00	SELECT * FROM transactions	100000 row(s) returned

!!NOTA. Los llamados de tabla.columna en la joins. Es lo que habíamos comentado de que así lo había aprendido y me quedé con la idea de que solo así se llamaban para las joins. Hice los cambios en el script sql

## Ejercicio 1

Realiza una subconsulta que muestre a todos los usuarios con más de 80 transacciones utilizando al menos 2 tablas.

```

111 #EJERCICIO 1
112 # Realiza una subconsulta que muestre a todos los usuarios con más de 80 transacciones utilizando al menos 2 tablas.
113
114 SELECT d_users.id, d_users.name as "Nombre", d_users.surname as "Apellido", COUNT(transactions.id) as CantidaddeTransacciones
115 FROM transactions
116 JOIN d_users
117 ON d_users.id = transactions.user_id
118 GROUP BY d_users.id, d_users.name, d_users.surname
119 HAVING CantidaddeTransacciones > 80;

```

id	Nombre	Apellido	CantidaddeTransacciones
185	Molly	Gilliam	110
289	Dxwgi	Hwicu	94
318	Bnyr	Astuw	91
454	Sfzoh	Xgvfridxs	81

#	Time	Action	Message
39	18:48:01	LOAD DATA INFILE 'C:\\ProgramData\\MySQL\\MySQL Server 8.0\\Uploads\\transactions.csv' INTO T...	100000 row(s) affected Records: 100000 Deleted: 0 Skipped: 0 Warnings: 0
40	18:49:19	SELECT * FROM transactions	100000 row(s) returned
41	21:18:44	SELECT d_users.id, d_users.name, d_users.surname, COUNT(transactions.id) as CantidaddeTransaccione...	4 row(s) returned
42	21:20:59	SELECT d_users.id, d_users.name as "Nombre", d_users.surname as "Apellido", COUNT(transactions.id) a...	4 row(s) returned



con el cambio en los llamamientos:

```
116 #EJERCICIO 1
117 # Realiza una subconsulta que muestre a todos los usuarios con más de 80 transacciones utilizando al menos 2 tablas.
118
119 • SELECT d_users.id, d_users.name as "Nombre", d_users.surname as "Apellido", COUNT(transactions.id) as CantidaddeTransacciones
120 FROM transactions
121 JOIN d_users
122 ON d_users.id = transactions.user_id
123 GROUP BY d_users.id, d_users.name, d_users.surname
124 HAVING CantidaddeTransacciones > 80;
125
```

## Ejercicio 2

Muestra la media de amount por IBAN de las tarjetas de crédito en la compañía Donec Ltd., utiliza por lo menos 2 tablas.

The screenshot shows a database management tool interface. On the left, a tree view displays the database structure for 'ventasdb', including tables like 'companies', 'credit\_cards', and 'transactions'. The main area shows a SQL query for Exercise 2:

```
125 #Ejercicio 2
126 #Muestra la media de amount por IBAN de las tarjetas de crédito en la compañía Donec Ltd., utiliza por lo menos 2 tablas.
127
128 • SELECT iban, ROUND(AVG(amount), 2) as MediaMonto
129 FROM transactions
130 JOIN credit_cards
131 ON credit_cards.id = transactions.card_id
132 JOIN companies
133 ON companies.company_id = transactions.company_id
134 WHERE company_name = "Donec Ltd"
135 GROUP BY iban
136 ORDER BY MediaMonto DESC;
```

Below the query, the 'Result Grid' shows the results of the query. The columns are 'iban' and 'MediaMonto'. The results are as follows:

iban	MediaMonto
XX383017813919620199366352	680.69
XX637706357397570394973913	680.01
XX971393971465292202312259	645.46
XX171847116928892375969307	628.89
XX225424638818542406223575	608.68
XX748890729057195711766071	607.29
TN9614563570667381893122	605.41
XX481908034037364242591185	605.36

Below the result grid, the 'Output' section shows the execution log with the following messages:

#	Time	Action	Message
6	12:04:36	SELECT * FROM credit_cards	5000 row(s) returned
7	12:12:42	SELECT iban, AVG(amount) as MediaMonto FROM transactions JOIN credit_cards ON credit_cards.id = tran...	371 row(s) returned
8	12:15:28	SELECT iban, ROUND(AVG(amount), 2) as MediaMonto FROM transactions JOIN credit_cards ON credit_...	371 row(s) returned
9	12:16:06	SELECT iban, ROUND(AVG(amount), 2) as MediaMonto FROM transactions JOIN credit_cards ON credit_...	371 row(s) returned

## Con el cambio en los llamados:

```
126 #Ejercicio 2
127 #Muestra la media de amount por IBAN de las tarjetas de crédito en la compañía Donec Ltd., utiliza por lo menos 2 tablas.
128
129 • SELECT credit_cards.iban, ROUND(AVG(transactions.amount), 2) as MediaMonto
130 FROM transactions
131 JOIN credit_cards
132 ON credit_cards.id = transactions.card_id
133 JOIN companies
134 ON companies.company_id = transactions.company_id
135 WHERE companies.company_name = "Donec Ltd"
136 GROUP BY credit_cards.iban
137 ORDER BY MediaMonto DESC;
138
```