

## Sprint 2

### Nivel 1

#### Ejercicio 1

A partir de los documentos adjuntos (estructura\_datos y datos\_introducir), importa las dos tablas. Muestra las principales características del esquema creado y explica las diferentes tablas y variables que existen. Asegúrate de incluir un diagrama que ilustre la relación entre las distintas tablas y variables.

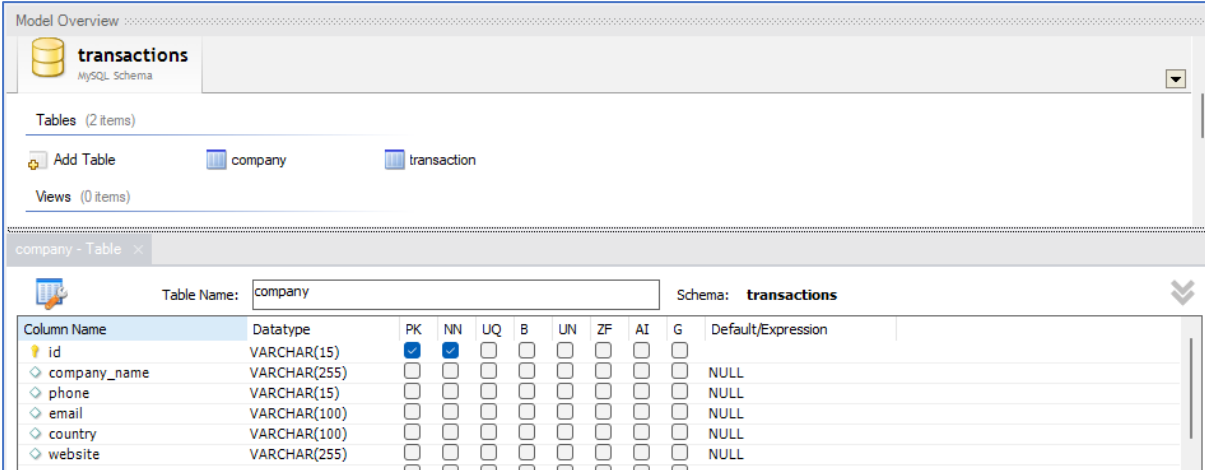
El esquema **'transactions'** tiene 2 tablas: **'company'** y **'transaction'**.

Tabla **'company'** – Datos de identificación de las empresas

Columnas: id (primary key), company\_name, phone, email, country, website

Tabla **'transactions'** – Datos de las transacciones de pago

Columnas: id(primary key), credit\_card\_id, company\_id (foreign key), user\_id, lat, longitude, timestamp, amount, declined



Model Overview

**transactions**  
MySQL Schema

Tables (2 items)

Add Table   company   transaction

Views (0 items)

company - Table

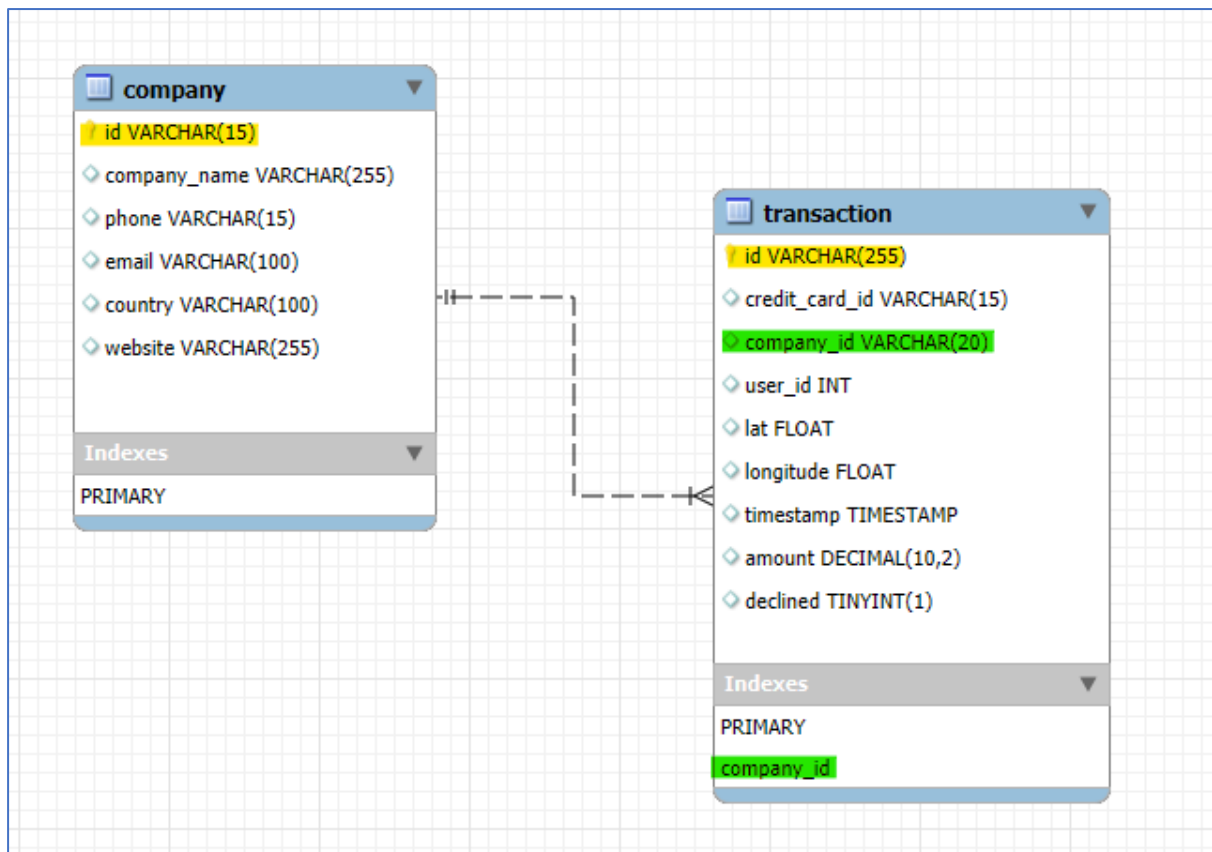
Table Name: company   Schema: transactions

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	VARCHAR(15)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
company_name	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
phone	VARCHAR(15)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
email	VARCHAR(100)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
country	VARCHAR(100)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
website	VARCHAR(255)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

# Data Analytics – SQL

transaction - Table										
Table Name: transaction		Schema: transactions								
Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	VARCHAR(255)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
credit_card_id	VARCHAR(15)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
company_id	VARCHAR(20)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
user_id	INT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
lat	FLOAT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
longitude	FLOAT	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
timestamp	TIMESTAMP	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
amount	DECIMAL(10,2)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
declined	TINYINT(1)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL

La tabla company se relaciona con la tabla transaction en una relación de uno (company) a N (transaction). La columna 'company\_id' es una foreign key y hace referencia a la primary key id de la tabla company.



## Nivel 1

### Ejercicio 2

Utilizando JOIN realizarás las siguientes consultas:

- Listado de los países que están generando ventas.

```
SELECT DISTINCT c.country
```

```
FROM transaction as t
```

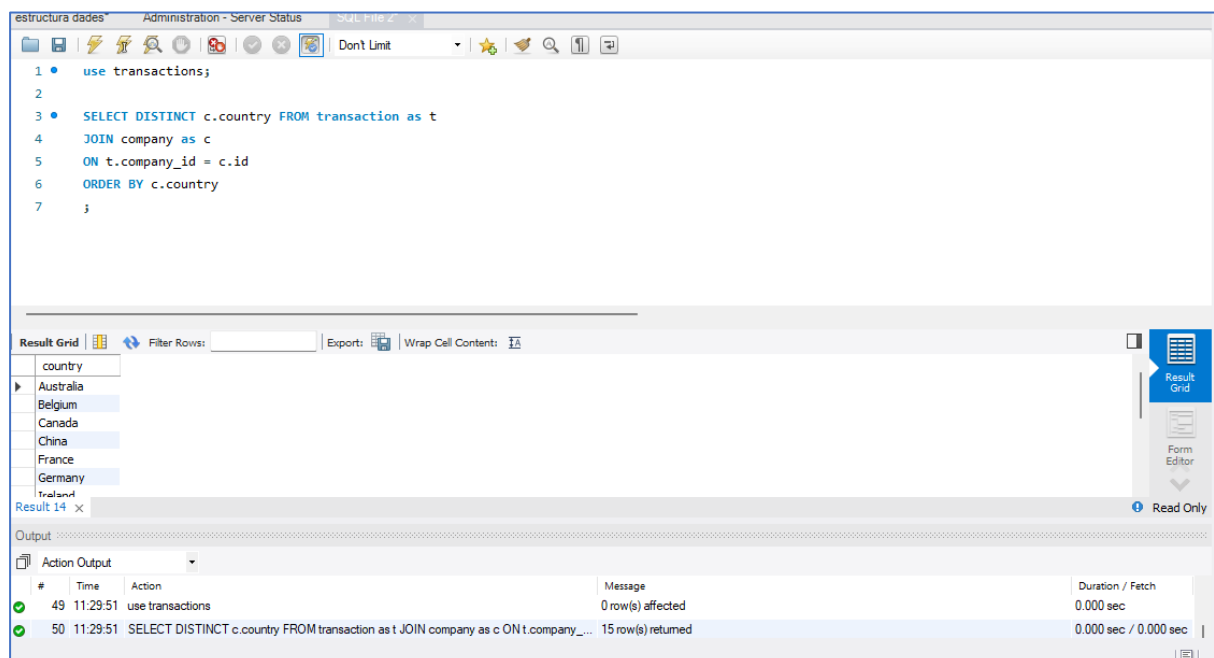
```
JOIN company as c
```

```
ON t.company_id = c.id
```

```
ORDER BY c.country
```

```
;
```

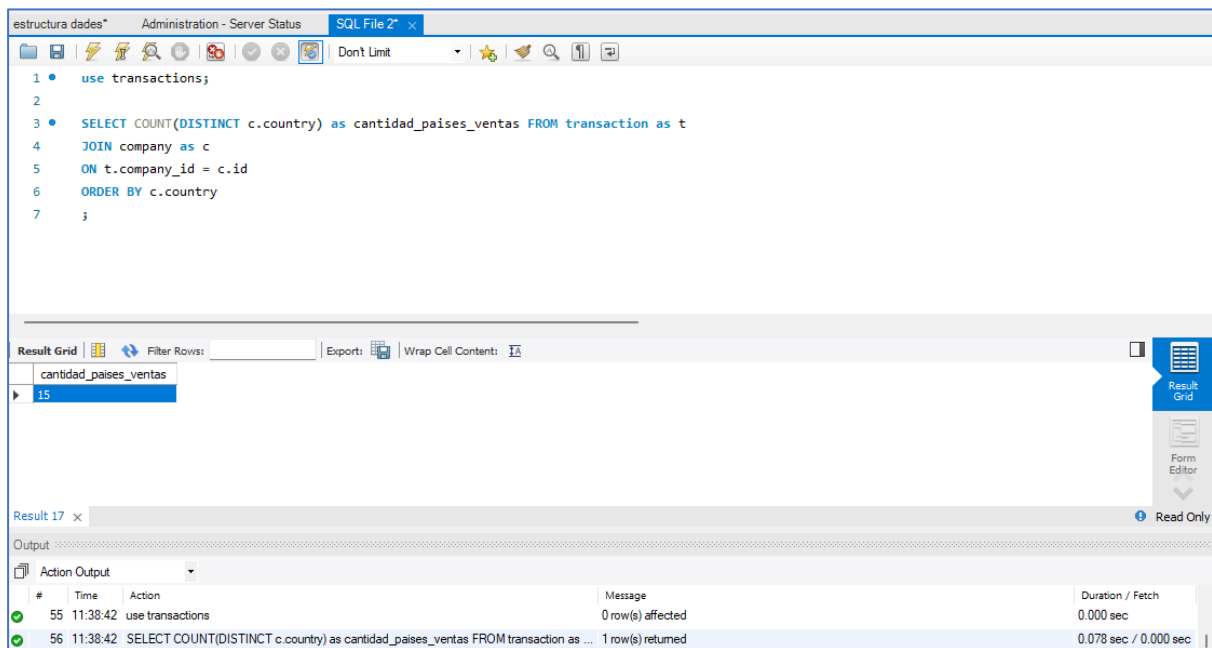
Australia	Netherlands	China	Spain	Ireland
Belgium	New Zealand	France	Sweden	Italy
Canada	Norway	Germany	United Kingdom	United States



# Data Analytics – SQL

- Desde cuántos países se generan las ventas.

```
SELECT COUNT(DISTINCT c.country) as cantidad_paises_ventas  
  
FROM transaction as t  
  
JOIN company as c  
  
ON t.companyid = c.id  
  
ORDER BY c.country  
  
;
```



- Identifica a la compañía con la mayor media de ventas.

284.867160      Ac Fermentum Incorporated

```
SELECT AVG(t.amount) as media_ventas, c.company_name  
  
FROM transaction as t  
  
JOIN company as c  
  
ON t.company_id = c.id  
  
GROUP BY c.company_name  
  
ORDER BY media_ventas DESC  
  
LIMIT 1  
  
;
```

# Data Analytics – SQL

The screenshot shows a SQL IDE interface with a query editor at the top and a results pane at the bottom. The query editor contains the following SQL code:

```
18
19 -- Identifica la compañía con la mayor media de ventas
20
21 • SELECT AVG(t.amount) as media_ventas, c.company_name FROM transaction as t
22 JOIN company as c
23 ON t.company_id = c.id
24 GROUP BY c.company_name
25 ORDER BY media_ventas DESC
26 LIMIT 1
27 ;
```

The results pane displays a single row of data:

media_ventas	company_name
284.867160	Ac Fermentum Incorporated

Below the results pane, the 'Output' section shows the execution log:

#	Time	Action	Message	Duration / Fetch
121	11:58:50	use transactions	0 row(s) affected	0.000 sec
122	11:58:50	SELECT AVG(t.amount) as media_ventas, c.company_name FROM transaction as t JOIN...	1 row(s) returned	0.344 sec / 0.000 sec

# Data Analytics – SQL

## Nivel 1

### Ejercicio 3

Utilizando sólo subconsultas (sin utilizar JOIN):

- Muestra todas las transacciones realizadas por empresas de Alemania.

**SELECT \***

**FROM transaction as t**

**WHERE t.company\_id IN**

**( -- Subquery para seleccionar apenas los id de Alemania**

**SELECT c.id**

**FROM company as c**

**WHERE c.country = 'Germany')**

**;**

The screenshot shows a SQL IDE interface with a query editor and a result grid. The query is as follows:

```
28
29  -- transacciones realizadas por empresas de Alemania.
30
31  SELECT * FROM transaction as t
32  WHERE t.company_id IN
33  (SELECT c.id FROM company as c
34   WHERE c.country = 'Germany')
35
36
37
```

The result grid displays the following data:

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
00138D3B-206D-4C03-94B7-63A2676EB9B4	CcS-4899	b-2222	318	41.3781	12.447	2020-03-25 10:43:43	426.36	0
0013C1B6-3884-4D6C-8154-E2B3FBCA8E9	CcS-5070	b-2222	489	41.3814	2.18176	2020-12-17 18:15:37	316.90	0
00201A11-2E62-44C4-941D-198FC8DB77F0	CcU-3512	b-2222	193	55.5704	-3.65129	2021-01-22 23:44:27	453.04	0
00235618-0A5C-4D49-9DCB-83A9405D8923	CcS-8137	b-2222	3556	59.8421	18.729	2020-09-09 15:43:19	263.14	0
005A5A7B-1F1A-4B6C-9B15-1625A78C9C38	CcS-8998	b-2222	4417	41.1591	-8.63905	2024-05-15 09:10:11	442.01	0
00687139-48E2-4FFA-8E73-B20376F04AB4	CcS-4870	b-2222	289	51.1966	10.4669	2019-03-09 19:37:49	524.84	0
0074E4A9-33F1-4877-8768-55906314673A	CcS-8081	b-2222	3500	30.7016	-8.50375	2016-12-26 23:06:57	401.00	0

The output section shows the execution of the query:

#	Time	Action	Message	Duration / Fetch
131	12:42:05	SELECT * FROM transaction as t WHERE t.company_id IN (SELECT c.id FROM com...	13291 row(s) returned	0.000 sec / 0.094 sec
132	12:49:44	SELECT * FROM transaction as t WHERE t.company_id IN (SELECT c.id FROM com...	13291 row(s) returned	0.000 sec / 0.062 sec

# Data Analytics – SQL

- Lista las empresas que han realizado transacciones por un amount superior a la media de todas las transacciones.

**SELECT c.company\_name**

**FROM company AS c**

**WHERE c.id IN**

**(** -- Subquery para encontrar las empresas con media de ventas por encima de la media general

**SELECT t.company\_id**

**FROM transaction as t**

**GROUP BY t.company\_id**

**HAVING AVG(t.amount) >**

**(** -- Subquery para encontrar la media general

**SELECT AVG(t.amount) as media\_general FROM transaction**

**as t**

**)**

**)**

**;**

The screenshot shows a SQL IDE interface with a query editor and a results grid. The query is as follows:

```
-- Empresas que han realizado transacciones por un amount superior a la media de todas las transacciones
SELECT t.company_id, c.company_name, avg(t.amount) as media_ventas_company FROM transaction as t
JOIN company as c
ON c.id = t.company_id
GROUP BY t.company_id, c.company_name
HAVING media_ventas_company >
(
    SELECT AVG(t.amount) as media_general FROM transaction as t
)
;
```

The results grid displays the following data:

company_id	company_name	media_ventas_company
b-2546	Lorem Ipsum Dolor Corp.	259.066194
b-2302	Nunc Interdum Incorporated	259.319156
b-2446	Risus Associates	259.800737
b-2226	Magna A Neque Industries	260.038610
b-2586	Tempor Diam Institute	260.096898
b-2602	Placerat LLP	260.210135
b-2598	Aliquam Iaculis Lacus Corp.	260.262606

The output section at the bottom shows the execution details:

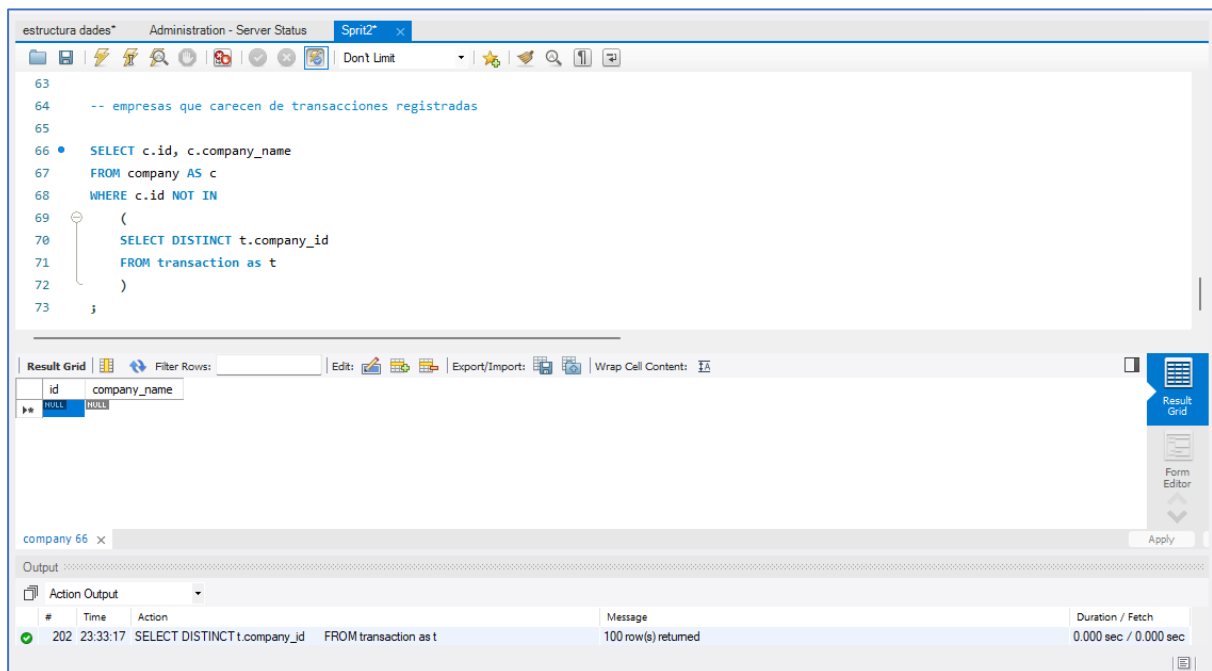
#	Time	Action	Message	Duration / Fetch
159	13:40:27	SELECT DISTINCT ventas_company_sup_media.company_id, ventas_company_sup_m...	100 row(s) returned	0.844 sec / 0.000 sec

- Eliminarán del sistema las empresas que carecen de transacciones registradas, entrega el listado de estas empresas.

Author: Ana Cláudia da Costa

# Data Analytics – SQL

```
SELECT c.id, c.company_name  
  
FROM company AS c  
  
WHERE c.id NOT IN  
  
    ( -- Subquery para identificar los company_id de las transacciones  
  
      SELECT DISTINCT t.company_id  
  
      FROM transaction as t  
  
    )  
  
; ;
```





# Data Analytics – SQL

## Nivel 2

### Ejercicio 1

Identifica los cinco días que se generó la mayor cantidad de ingresos en la empresa por ventas.  
Muestra la fecha de cada transacción junto con el total de las ventas.

```
SELECT DATE(t.timestamp), SUM(amount) AS total_ventas

FROM transaction AS t

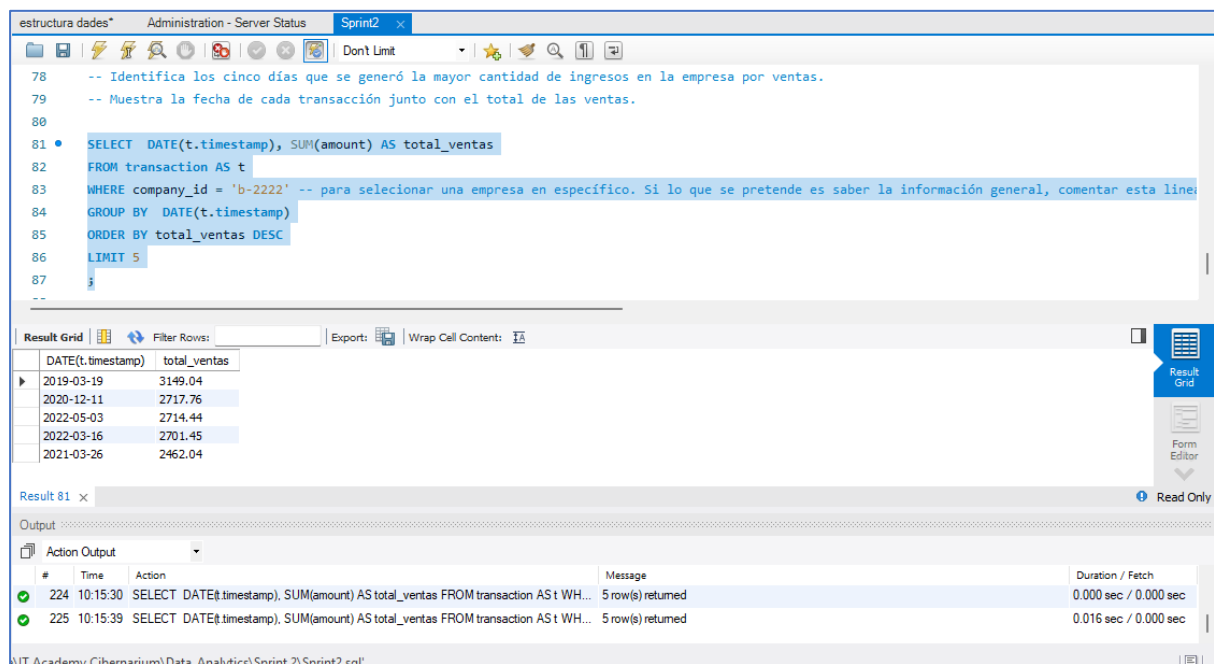
WHERE company_id = 'b-2222' -- para seleccionar una empresa en específico. Si lo que se pretende es saber la información general, comentar esta línea.

GROUP BY DATE(t.timestamp)

ORDER BY total_ventas DESC

LIMIT 5

;
```



The screenshot shows a SQL IDE interface with a query editor and a results grid. The query is as follows:

```
SELECT DATE(t.timestamp), SUM(amount) AS total_ventas
FROM transaction AS t
WHERE company_id = 'b-2222' -- para seleccionar una empresa en específico. Si lo que se pretende es saber la información general, comentar esta línea.
GROUP BY DATE(t.timestamp)
ORDER BY total_ventas DESC
LIMIT 5
;
```

The results grid displays the following data:

DATE(t.timestamp)	total_ventas
2019-03-19	3149.04
2020-12-11	2717.76
2022-05-03	2714.44
2022-03-16	2701.45
2021-03-26	2462.04

The output section shows the execution details:

#	Time	Action	Message	Duration / Fetch
224	10:15:30	SELECT DATE(t.timestamp), SUM(amount) AS total_ventas FROM transaction AS t WH...	5 row(s) returned	0.000 sec / 0.000 sec
225	10:15:39	SELECT DATE(t.timestamp), SUM(amount) AS total_ventas FROM transaction AS t WH...	5 row(s) returned	0.016 sec / 0.000 sec

## Nivel 2

### Ejercicio 2

¿Cuál es la media de ventas por país? Presenta los resultados ordenados de mayor a menor medio.

```
SELECT c.country, AVG(t.amount) AS media_ventas

FROM transaction AS t
```

Author: Ana Cláudia da Costa

# Data Analytics – SQL

**JOIN company as c**

**ON t.company\_id = c.id**

**GROUP BY c.country**

**ORDER BY media\_ventas DESC**

**;**

The screenshot shows a SQL IDE window with a query editor and a results grid. The query is as follows:

```
-- Media de ventas por país ordenado de maior a menor
SELECT c.country, AVG(t.amount) AS media_ventas
FROM transaction AS t
JOIN company as c
ON t.company_id = c.id
GROUP BY c.country
ORDER BY media_ventas DESC
```

The results grid displays the following data:

country	media_ventas
Australia	265.190742
United States	264.977877
Belgium	261.153042
Germany	260.841391
Ireland	260.644761
Spain	260.468125

The bottom of the screenshot shows the output log with two entries:

#	Time	Action	Message	Duration / Fetch
232	10:31:03	SELECT c.country, AVG(t.amount) AS media_ventas FROM transaction AS t JOIN comp...	15 row(s) returned	0.312 sec / 0.000 sec
233	10:32:48	SELECT c.country, AVG(t.amount) AS media_ventas FROM transaction AS t JOIN comp...	15 row(s) returned	0.329 sec / 0.000 sec

## Nivel 2

### Ejercicio 3

En tu empresa, se plantea un nuevo proyecto para lanzar algunas campañas publicitarias para hacer competencia a la compañía “Non Institute”. Para ello, te piden la lista de todas las transacciones realizadas por empresas que están ubicadas en el mismo país que esta compañía.

- Muestra el listado aplicando JOIN y subconsultas.

**SELECT \***

**FROM transaction as t**

**JOIN company as c**

**ON t.company\_id = c.id**

**WHERE c.country =**

**(-- select para identificar el país de la compañía 'Non Institute')**

# Data Analytics – SQL

```
SELECT c.country
FROM company as c
WHERE company_name ='Non institute'
)
AND c.id <>
(-- select para identificar el ID de la compañía 'Non Institute'
SELECT c.id
FROM company as c
WHERE company_name ='Non institute'
)
;
```

The screenshot shows a SQL IDE window titled 'estructura datos\*' with a tab 'Sprint2'. The query editor contains the following SQL code:

```
104 SELECT *
105 FROM transaction as t
106 JOIN company as c
107 ON t.company_id = c.id
108 WHERE c.country =
109 (
110     (-- select para identificar el pais de la compañía 'Non Institute'
111     SELECT c.country
112     FROM company as c
113     WHERE company_name ='Non institute'
114     )
115 )
116 AND c.id <>
```

Below the query editor, the 'Result Grid' is displayed, showing a table with 12 columns: id, credit\_card\_id, company\_id, user\_id, lat, longitude, timestamp, amount, declined, id, company\_name, and phone. The table contains 4 rows of data.

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined	id	company_name	phone
008629B4-C9A9-406C-A3D2-71FDA478C546	CcS-7063	b-2246	2482	45.7666	4.83048	2015-07-30 12:12:42	486.44	0	b-2246	Sed Nunc Ltd	02 62 64 73 48
00B72BA4-54A3-4B8E-B13F-2D57535AA17A	CcS-8475	b-2246	3894	55.6212	-3.7546	2017-10-26 22:08:26	414.06	0	b-2246	Sed Nunc Ltd	02 62 64 73 48
01F075B1-D7AE-4D02-AAD9-5FFD72A43F3C	CcS-8700	b-2246	4119	55.856	-3.15783	2018-01-27 13:44:36	103.73	0	b-2246	Sed Nunc Ltd	02 62 64 73 48
023FFCE8-E618-4938-BF56-C8DF80540ADD	CcS-7816	b-2246	3235	46.3568	1.82755	2016-12-19 11:53:45	219.28	0	b-2246	Sed Nunc Ltd	02 62 64 73 48

Below the result grid, the 'Output' window shows the execution log:

#	Time	Action	Message	Duration / Fetch
244	10:53:51	SELECT * FROM transaction as t JOIN company as c ON t.company_id = c.id WHERE c...	12233 row(s) returned	0.015 sec / 0.047 sec
245	10:54:44	SELECT * FROM transaction as t JOIN company as c ON t.company_id = c.id WHERE c...	12233 row(s) returned	0.000 sec / 0.062 sec

- Muestra el listado aplicando solo subconsultas.

```
SELECT *
FROM transaction
HAVING company_id IN
    ( -- select para identificar los IDs de las compañías del mismo pais que la compañía
    'Non Institute'
    SELECT c.id
```

# Data Analytics – SQL

**FROM company as c**

**WHERE c.country =**

**( -- select para identificar el pais da la compañía 'Non Institute'**

**SELECT c.country**

**FROM company as c**

**WHERE company\_name = 'Non Institute')**

**AND c.id <>**

**( -- select para identificar el id da la compañía 'Non Institute'**

**SELECT c.id**

**FROM company as c**

**WHERE company\_name = 'Non Institute')**

**)**

**;**

The screenshot shows a SQL IDE interface with a query editor and a results grid. The query is as follows:

```
123
124 • SELECT *
125 FROM transaction
126 HAVING company_id IN
127 ( -- select para identificar los IDs de las compañías del mismo pais que la compañía 'Non Institute'
128 SELECT c.id
129 FROM company as c
130 WHERE c.country =
131 ( -- select para identificar el pais da la compañía 'Non Institute'
132 SELECT c.country
133 FROM company as c
```

The results grid displays the following data:

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
00130BE3-3898-4DC3-B705-CE6723CC1F71	CcU-3736	b-2310	161	55.2114	-3.40245	2024-04-25 16:42:21	425.10	0
001A60EA-DC9C-4E5A-9460-6628B100E7E1	CcS-6225	b-2326	1644	51.7125	19.0674	2018-05-20 02:06:39	354.02	0
001ESD06-A391-4735-88D8-748F16C061A6	CcS-7425	b-2310	2844	59.9165	18.5518	2021-09-10 02:39:32	418.20	0
0022377F-4447-4328-B01A-CFE5416E336C	CcS-8582	b-2522	4001	50.7865	10.6173	2019-10-24 06:52:23	171.13	0
00285171-6887-4E96-9787-BD580BE4515D	CcS-8358	b-2310	3777	55.7751	-3.8232	2022-09-21 21:18:51	192.60	0

The output section shows the following messages:

#	Time	Action	Message	Duration / Fetch
251	11:21:36	SELECT * FROM transaction as t HAVING company_id IN ( -- select para identificar los l...	12233 row(s) returned	0.000 sec / 0.141 sec
252	11:22:52	SELECT * FROM transaction HAVING company_id IN ( -- select para identificar los IDs d...	12233 row(s) returned	0.000 sec / 0.125 sec

## Nivel 3

### Ejercicio 1

Presenta el nombre, teléfono, país, fecha y amount, de aquellas empresas que realizaron transacciones con un valor comprendido entre 350 y 400 euros y en alguna de estas fechas: 29 de abril de 2015, 20 de julio de 2018 y 13 de marzo de 2024. Ordena los resultados de mayor a menor cantidad.

```
SELECT * FROM  
  
(  
  
    SELECT c.company_name, c.phone, c.country, DATE(t.timestamp) as fecha,  
    t.amount  
  
    FROM transaction AS t  
  
    JOIN company AS c  
  
    ON t.company_id = c.id  
  
    HAVING t.amount BETWEEN 350.00 AND 400.00  
  
    ORDER BY amount DESC  
  
)consulta1  
  
HAVING consulta1.fecha = '2015-04-29' OR consulta1.fecha = '2018-07-20' OR  
consulta1.fecha = '2024-03-13'  
  
;
```

The screenshot shows a SQL IDE interface with a query editor and a results grid. The query is as follows:

```
SELECT * FROM  
(  
    SELECT c.company_name, c.phone, c.country, DATE(t.timestamp) as fecha, t.amount  
    FROM transaction AS t  
    JOIN company AS c  
    ON t.company_id = c.id  
    HAVING t.amount BETWEEN 350.00 AND 400.00  
    ORDER BY amount DESC  
)consulta1  
HAVING consulta1.fecha = '2015-04-29' OR consulta1.fecha = '2018-07-20' OR consulta1.fecha = '2024-03-13'
```

The results grid displays the following data:

company_name	phone	country	fecha	amount
Aliquam PC	01 45 73 52 16	Germany	2024-03-13	388.29
Ordi Adipiscing Limited	03 18 00 77 81	United Kingdom	2018-07-20	373.71
Fringilla LLC	08 29 15 93 57	New Zealand	2015-04-29	367.62
Pede Cum Ltd	07 62 26 48 38	Norway	2018-07-20	356.87
Auctor Mauris Vel LLP	08 09 28 74 14	United States	2024-03-13	353.75

The output section shows the execution of the query, indicating that 8 row(s) were returned and the duration was 0.328 sec / 0.000 sec.

## Nivel 3

### Ejercicio 2

Necesitamos optimizar la asignación de los recursos y dependerá de la capacidad operativa que se requiera, por lo que te piden la información sobre la cantidad de transacciones que realizan las empresas, pero el departamento de recursos humanos es exigente y quiere un listado de las empresas en las que especifiques si tienen más de 400 transacciones o menos.

**-- select para filtrar las empresas com más de 400 transacciones**

**SELECT c.company\_name, COUNT(t.id)**

**FROM company as c**

**JOIN transaction as t**

**ON c.id = t.company\_id**

**GROUP BY c.company\_name**

**HAVING COUNT(t.id) > 400;**

**-- select para filtrar las empresas com menos de 400 transacciones**

**SELECT c.company\_name, COUNT(t.id)**

**FROM company as c**

**JOIN transaction as t**

**ON c.id = t.company\_id**

**GROUP BY c.company\_name**

**HAVING COUNT(t.id) < 400;**

# Data Analytics – SQL

The screenshot shows a SQL IDE window with a query editor and a results grid. The query is as follows:

```
-- Ejercicio 2
-- listado de las empresas en las que especifiques si tienen más de 400 transacciones o menos
-- select para filtrar las empresas com más de 400 transacciones
SELECT c.company_name, COUNT(t.id)
FROM company as c
JOIN transaction as t
ON c.id = t.company_id
GROUP BY c.company_name
HAVING COUNT(t.id) > 400;
```

The results grid displays the following data:

company_name	COUNT(t.id)
Ac Fermentum Incorporated	2401
Magna A Neque Industries	410
Fusce Corp.	447
Convalis In Incorporated	1514
Ante Iaculis Nec Foundation	472

The output pane shows the execution of the query, indicating that 97 row(s) were returned.

The screenshot shows a SQL IDE window with a query editor and a results grid. The query is as follows:

```
GROUP BY c.company_name
HAVING COUNT(t.id) > 400;
-- select para filtrar las empresas com menos de 400 transacciones
SELECT c.company_name, COUNT(t.id)
FROM company as c
JOIN transaction as t
ON c.id = t.company_id
GROUP BY c.company_name
HAVING COUNT(t.id) < 400;
```

The results grid displays the following data:

company_name	COUNT(t.id)
Fringilla LLC	397
Lorem Eu Incorporated	380
Nec Luctus LLC	399

The output pane shows the execution of the query, indicating that 3 row(s) were returned.