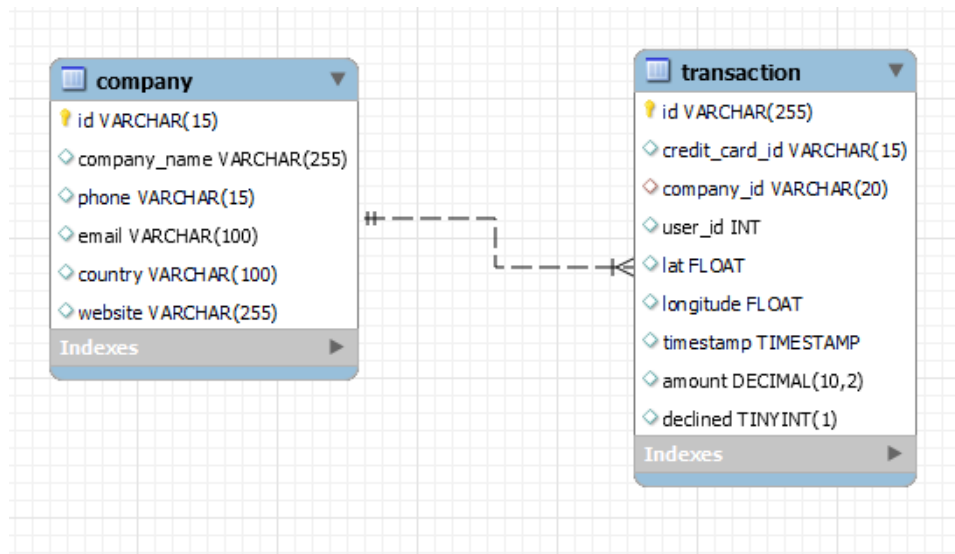


Nivel 1.

Ejercicio 1

DB TRANSACTIONS



La **base de datos “Transactions”** está compuesta por dos tablas: “Company” y “Transaction”.

La **tabla Company** está compuesta por **6 campos** que contienen información de las distintas compañías de la empresa entre ellos encontramos: ID que funciona como Primary Key, el nombre de la compañía (Company_name), el teléfono (phone), el email, el país (Country) y la website.

La **tabla Transaction** contiene información sobre las distintas transacciones realizadas por las compañías. La componen 9 campos: ID que es la PK, identificador de tarjeta de crédito (Credit_card_id), identificador de la compañía que actúa como Foreign Key (Company_id), identificador del usuario (User_id), latitude (lat), longitude, fecha y hora (timestamp), monto (cantidad) y rechazado (declined).

Transaction tiene una **relación de N a 1** con la tabla Company, por medio del campo Company_id=Id.

Ejercicio 2

El listado con la información de cada compañía está compuesto por 100 compañías, se puede ver el comienzo del mismo en la siguiente imagen.

The screenshot shows a SQL IDE window titled 'S1.T1*' with a file named 'SQL File 7*'. The query editor contains the following SQL code:

```

1 #Exercici 2: Has d'obtenir el nom, email i país de cada companyia, ordena les dades en funció del nom de
2 • SELECT company_name, email, country
3 FROM company
4 ORDER BY company_name;
5

```

Below the query editor is the 'Result Grid' showing the first 15 rows of the query results. The columns are 'company_name', 'email', and 'country'. The data is as follows:

company_name	email	country
A Institute	metus.aliquam@google.edu	Belgium
Ac Fermentum Incorporated	donec.porttitor.tellus@yahoo.net	Germany
Ac Industries	ipsum@yahoo.com	Germany
Ac Libero Inc.	mollis.lectus@protonmail.ca	United Kingdom
Aliquam Erat Volutpat LLP	pede.nunc@icloud.net	Italy
Aliquam Iaculis Lacus Corp.	dictum@aol.org	Belgium
Aliquam PC	scelerisque.mollis@icloud.org	Germany
Aliquet Diam Limited	eu.eros.nam@icloud.org	United States
Aliquet Sem Limited	sem.magna@yahoo.edu	Netherlands
Aliquet Vel Vulputate Incorporated	suspendisse.commodo@outlook.couk	Netherlands
Amet Faucibus Ut Foundation	malesuada@icloud.net	United Kingdom
Amet Institute	nullam.lobortis.quam@outlook.net	Australia
Amet Lorem LLP	ultrices.adipiscing@icloud.edu	Spain
Amet Luctus Vulputate Foundation	nulla@outlook.com	Canada
Amet Nulla Donec Corporation	mattis.integer.eu@protonmail.net	Italy

Below the result grid is the 'Output' section, which shows the execution of the query. It contains two entries:

#	Time	Action	Message
5	10:09:29	SELECT company_name, email, country FROM company ORDER BY company_name	100 row(s) returned
6	10:09:42	SELECT company_name, email, country FROM company ORDER BY company_name	100 row(s) returned



A partir de ahora tendremos en cuenta en las consultas al booleano "Declined" para que no incluya las compras rechazadas.

Ejercicio 3

Los países que están realizando compras son los siguientes quince: Alemania, Australia, EEUU, Nueva Zelanda, Noruega, Reino Unido, Italia, Bélgica, Suecia, Irlanda, China, Canadá, Francia, Países Bajos y España.

The screenshot shows the SQL Developer interface with a query window titled 'SQL File 7*'. The query is as follows:

```
7 • SELECT distinct country
8   FROM company
9   INNER JOIN transaction
10  ON company.id=transaction.company_id
11  WHERE declined = 0;
```

The 'Result Grid' displays the results of the query, showing a list of countries:

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

The 'Output' window shows the execution log with two entries:

#	Time	Action	Message
7	10:14:10	SELECT distinct country FROM company INNER JOIN transaction ON company.id=tr...	15 row(s) returned
8	10:14:25	SELECT distinct country FROM company INNER JOIN transaction ON company.id=tr...	15 row(s) returned

Ejercicio 4

Las compras se realizan desde 15 países diferentes.

The screenshot shows the SQL Developer interface with a query window titled 'SQL File 7*'. The query is as follows:

```
14 • SELECT COUNT(DISTINCT country) AS 'Cantidad_paises'
15   FROM company
16   INNER JOIN transaction
17   ON company.id=transaction.company_id
18   WHERE declined = 0;
```

The 'Result Grid' displays the results of the query, showing a single row with the count:

Cantidad_paises
15

Ejercicio 5

La compañía con ID = 'b-2354' se llama Ac Libero Inc. y es del Reino Unido.

The screenshot shows the SQL Developer interface with a query window titled 'estructura_dades SQL File 7*'. The query is as follows:

```
20
21 #Exercici 5: país i nom de companyia amb id = 'b-2354'.
22 • SELECT country, company_name
23   FROM company
24   WHERE id = 'b-2354';
25
```

The 'Result Grid' displays the results of the query, showing a single row with the company details:

country	company_name
United Kingdom	Ac Libero Inc.

Ejercicio 6

La compañía con mayor gasto medio es Eget Ipsum Ltd.

En la consulta incluimos el nombre de la empresa y también el valor del gasto medio, teniendo en cuenta las compras que han sido rechazadas.

The screenshot shows a SQL query in a text editor window titled 'estructura_dades' and 'SQL File 7*'. The query is as follows:

```

25 #Exercici 6: quina és la companyia amb major despesa mitjana?
26 • SELECT company_name, avg(amount) AS gasto_medio
27 FROM company
28 INNER JOIN transaction
29 ON company.id=transaction.company_id
30 WHERE declined = 0
31 GROUP BY company_name
32 ORDER BY gasto_medio DESC
33 LIMIT 1;

```

Below the query, the 'Result Grid' is displayed with the following data:

company_name	gasto_medio
Eget Ipsum Ltd	481.860000

Nivel 2

Ejercicio 1

No hay en la base de datos compañías con ID duplicados. Como ID es una PK, por definición, no puede tener valores duplicados. Por tanto, buscamos los duplicados en el campo "Company_name".

The screenshot shows a SQL query in a text editor window titled 'SQL File 7*'. The query is as follows:

```

36 # Exercici 1: verificar si en la base de dades existeixen companyies amb identificadors (id) duplicats.
37 • SELECT id, company_name
38 FROM company
39 WHERE company_name IN (
40     SELECT company_name
41     FROM company
42     GROUP BY company_name
43     HAVING count(*)>1
44 );

```

Below the query, the 'Result Grid' is displayed with the following data:

id	company_name
NULL	NULL

At the bottom of the screenshot, the 'Output' window shows the execution log:

#	Time	Action	Message
9	10:16:46	SELECT COUNT(DISTINCT country) AS 'Cantidad_paises' FROM company INNER J...	1 row(s) returned
10	10:30:04	SELECT company_name, avg(amount) AS gasto_medio FROM company INNER JOI...	1 row(s) returned
11	10:30:27	SELECT id, company_name FROM company WHERE company_name IN (SELECT ...	0 row(s) returned

Ejercicio 2

Los 5 días donde la suma de ventas ha sido mayor son los siguientes:

- 20/12/2021
- 22/04/2021
- 09/05/2021
- 26/02/2022
- 29/03/2021

The screenshot shows a SQL IDE window titled 'SQL File 7*'. The query editor contains the following SQL code:

```
45  
46 #N2.Exercici 2: En quin dia es van realitzar les cinc vendes més costoses? Mostra la data de la transacció i l  
47 • SELECT left(timestamp,10) as date, sum(amount) AS total_diario  
48 FROM transaction  
49 WHERE declined = 0  
50 GROUP BY date  
51 ORDER BY total_diario DESC  
52 LIMIT 5;  
53
```

Below the query editor, the 'Result Grid' shows the following data:

	date	total_diario
▶	2021-12-20	1532.36
	2021-04-22	1397.96
	2021-05-09	1344.37
	2022-02-26	1337.62
	2021-03-29	1325.12

The 'Output' pane at the bottom shows the execution log:

#	Time	Action	Message
71	11:36:12	SELECT left(timestamp,10) as date, sum(amount) AS total_diario FROM transaction W...	5 row(s) returned
72	11:37:26	SELECT left(timestamp,10) as date, sum(amount) AS total_diario FROM transaction W...	5 row(s) returned

Ejercicio 3

Los 5 días donde la suma de ventas ha sido menor son los siguientes:

- 04/01/2022
- 27/04/2021
- 24/01/2022
- 27/02/2022
- 14/01/2022

SQL File 7*

```

54 #N2. Exercici 3: En quin dia es van realitzar les cinc vendes de menor valor? Mostra la data de la transacció
55 • SELECT left(timestamp,10) as date, sum(amount) AS total_diario
56 FROM transaction
57 WHERE declined = 0
58 GROUP BY date
59 ORDER BY total_diario
60 LIMIT 5;

```

Result Grid

date	total_diario
2022-01-04	15.05
2021-04-27	18.08
2022-01-24	23.86
2022-02-27	30.76
2022-01-14	37.55

Result 45 x Read Only

Output

Action Output

#	Time	Action	Message
71	11:36:12	SELECT left(timestamp,10) as date, sum(amount) AS total_diario FROM transaction W...	5 row(s) returned
72	11:37:26	SELECT left(timestamp,10) as date, sum(amount) AS total_diario FROM transaction W...	5 row(s) returned
73	11:53:50	SELECT left(timestamp,10) as date, sum(amount) AS total_diario FROM transaction W...	5 row(s) returned

Ejercicio 4

La media de gasto por país ordenada de mayor a menor se puede observar en la siguiente imagen:

SQL File 7*

```

63 • SELECT country, AVG(amount)
64 FROM company
65 INNER JOIN transaction
66 ON company.id=transaction.company_id
67 WHERE declined = 0
68 GROUP BY country
69 ORDER BY AVG(amount) DESC;

```

Result Grid

country	AVG(amount)
United States	287.531111
Ireland	285.825357
Sweden	276.668382
United Kingdom	271.767527
Canada	261.941930
Belgium	255.217500
Norway	251.114918
Italy	243.342222
Germany	242.239189

Result 51 x Read Only

Output

Action Output

#	Time	Action	Message
80	12:09:18	SELECT country, amount FROM company INNER JOIN transaction ON company.id=...	500 row(s) returned
81	12:10:19	SELECT country, AVG(amount) FROM company INNER JOIN transaction ON compa...	15 row(s) returned
82	12:10:34	SELECT country, AVG(amount) FROM company INNER JOIN transaction ON compa...	15 row(s) returned

Nivel 3

Ejercicio 1

Las compañías que han gastado entre 100€ y 200€ en total, ordenadas de mayor a menor, pueden observarse en la siguiente imagen:

The screenshot shows a SQL query in SQL Studio. The query selects company details and total amount, filtered by a total amount between 100 and 200, ordered descending by total amount.

```

74 • SELECT company_name, phone, country, sum(amount) AS total_amount_company
75 FROM company
76 INNER JOIN transaction
77 ON company.id=transaction.company_id
78 WHERE declined = 0
79 GROUP BY company_name, phone, country
80 HAVING SUM(amount) >= 100 AND SUM(amount) <= 200
81 ORDER BY SUM(amount) DESC;

```

The Result Grid shows the following data:

company_name	phone	country	total_amount_company
Interdum Feugiat Sed Associates	04 88 40 32 52	United Kingdom	164.86
Mauris Incorporated	06 84 33 15 97	Norway	158.50
Mattis Foundation	06 74 03 95 74	Australia	155.57
Quisque Libero LLC	01 45 48 71 11	China	155.44
Non Justo Corp.	09 30 54 14 44	Sweden	151.32
Nec Luctus LLC	02 14 71 75 73	Norway	124.06
Donec Fringilla PC	01 51 58 14 44	France	119.68
Orci Adipiscing Limited	03 18 00 77 81	United Kingdom	111.64
Nulla Integer Vulputate Corp.	04 57 50 84 48	Sweden	105.42

The Output pane shows the execution log with messages indicating 16 rows returned for the main query and an error for a subsequent query.

Ejercicio 2

Las 8 compañías que han hecho compras el 16/03/2022, el 28/02/2022 y el 13/02/2022 pueden observarse en la siguiente imagen:

The screenshot shows a SQL query in SQL Studio. The query selects company names and dates for transactions on specific dates: 2022-03-16, 2022-02-28, and 2022-02-13.

```

81
82 #N3.Ejercicio 2: Indica el nom de les companyies que van fer compres el 16 de març del 2022, 28 de febrer del 2
83 • SELECT company_name, left(timestamp,10) as date
84 FROM company
85 INNER JOIN transaction
86 ON company.id=transaction.company_id
87 WHERE declined = 0
88 HAVING date IN ("2022-03-16", "2022-02-28", "2022-02-13");

```

The Result Grid shows the following data:

company_name	date
Arcu LLP	2022-02-13
Nunc Interdum Incorporated	2022-02-13
Nunc Interdum Incorporated	2022-02-28
Lorem Eu Incorporated	2022-03-16
Ut Semper Foundation	2022-02-28
Malesuada PC	2022-02-13
Malesuada PC	2022-02-28
Malesuada PC	2022-03-16

The Output pane shows the execution log with messages indicating 9 rows returned for the main query and 8 rows returned for a subsequent query.