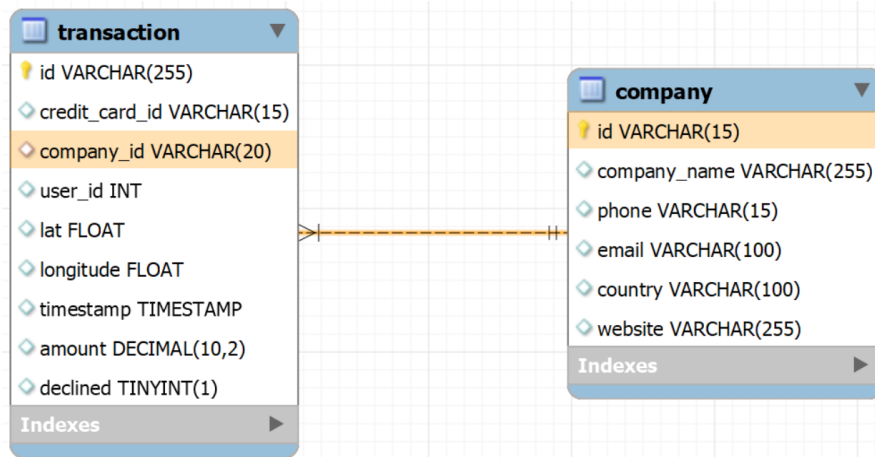


SPRINT 1: NOCIONES BÁSICAS Y COMPONENTES CLAVES EN EL DISEÑO DE BASES DE DATOS RELACIONALES

NIVEL 1

La base de datos *Transactions* importada se compone de dos entidades, Transaction y Company, que tienen una relación N a 1 (transaction N --> 1 company) entre sí, resaltada en el diagrama a continuación.



- Transaction: corresponde a la tabla de hechos de este modelo y contiene información sobre las diferentes transacciones que se realizaron. Tenemos las siguientes variables:
 - Id (primary key): identificador único (y no nulo) de cada transacción
 - Credit_card_id: identificador de la tarjeta de crédito utilizada para realizar la transacción
 - Company_id: identificador de la empresa/institución/organización involucrada en las transacciones registradas
 - Corresponde a la “foreign key” con la que se relaciona la tabla Company
 - User_id: identificador de los usuarios que realizaron las transacciones
 - Lat y longitude: coordenadas de latitud y longitud, respectivamente, desde donde se realizó la transacción
 - Timestamp: fecha (en formato año-mes-día) y hora (en formato hora-min-segundo) en que se realizó la transacción
 - Amount: cantidad de dinero gastado en cada transacción
 - Declined: variable tipo boolean (0: False; 1: True) que nos permite saber si la transacción se realizó correctamente (0) o, de lo contrario, si fue rechazada (1)
- Company: corresponde a la tabla dimensión y contiene información descriptiva sobre las compañías que participan en las transacciones. Tenemos las siguientes variables:
 - Id (primary key): identificador único (y no nulo) de cada empresa
 - Corresponde a la “primary key” que permite relacionarse con la tabla Transaction
 - Company_name: nombre de la empresa
 - Phone y email: teléfono y mail de contacto, respectivamente, de la empresa
 - Country: país donde está registrada la empresa
 - Website: página web de la empresa

Ejercicio 2. Obtén el nombre, email y país de cada compañía, ordena los datos en función del nombre de las compañías.

Realizamos la consulta de los campos que contienen la información deseada (nombre, mail y país de cada empresa) de la tabla Company y los ordenamos alfabéticamente por el nombre de la empresa.

```
SELECT company_name AS nombre_compañia,  
       email,  
       country AS pais  
FROM transactions.company  
ORDER BY nombre_compañia;
```

Con esta consulta obtenemos un total de 100 compañías, las 5 primeras entre las cuales son las siguientes, con sus respectivos datos:

nombre_compañia	email	pais
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

Ejercicio 3. Desde la sección de marketing te solicitan que les pases un listado de los países que están realizando compras.

Para recuperar los países que realizan las compras hacemos la consulta con SELECT DISTINCT de países de la tabla Company. Debemos aplicar DISTINCT para evitar países duplicados, ya que muchas compras pueden realizarse desde un mismo país.

Finalmente, aunque no lo pidan explícitamente, ordenamos los países alfabéticamente.

```
SELECT DISTINCT country AS pais  
FROM transactions.company  
ORDER BY pais;
```

Obtenemos una lista con 15 países en total, los primeros 5 de la cual son Australia, Bélgica, Canadá, China y Francia:

pais
Australia
Belgium
Canada
China
France

Ejercicio 4. Desde marketing también quieren saber desde cuántos países se realizan las compras.

Para saber la cantidad de países desde los que se realizan las compras, aplicamos un COUNT DISTINCT de los países de la tabla Company. Hay que aplicar el conteo distintivo por el mismo motivo explicado anteriormente, es decir, para evitar duplicados.

```
SELECT COUNT(DISTINCT country) AS cantidad_paises  
FROM transactions.company;
```

Obtenemos un total de 15 países desde los que se realizan las compras. Dicha cantidad coincide con la cantidad de países en la lista de la consulta previa, por lo que si combinamos las dos consultas sabemos tanto la cantidad como los nombres de países desde los que se realizaron las compras.

cantidad_paises
15

Ejercicio 5. Tu jefe identifica un error con la compañía que tiene identificador 'b-2354'. Por tanto, te solicita que le indiques el país y nombre de compañía de este identificador.

Para recuperar información de una compañía en concreto, debemos buscar los campos de interés (en este caso el país y el nombre de la compañía) de la tabla Company y aplicarle el filtro del identificador de esa compañía (b-2354).

```
SELECT country AS pais, company_name AS nombre_compañia
FROM transactions.company
WHERE id='b-2354';
```

Por tanto, la compañía donde se identificó el error es Ac Libero Inc., de Reino Unido.

pais	nombre_compañia
United Kingdom	Ac Libero Inc.

Ejercicio 6. Además, tu jefe te solicita que indiques ¿cuál es la compañía con mayor gasto promedio?

Para identificar a la compañía con mayor gasto promedio, debemos calcular el promedio de los gastos de cada compañía. Para ello, aplicamos el AVG de los gastos de la tabla Transaction, agrupándolos por la compañía (GROUP BY company_id); hay que tener en cuenta que para este cálculo únicamente se deben incluir las transacciones que no fueron rechazadas (filtro declined=0). Aunque no lo pide expresamente, redondeamos el promedio a 2 decimales, para coincidir con el formato de los datos en la tabla. Con esto ya sabemos los gastos promedio de cada compañía.

Para conocer el nombre de cada una de ellas debemos recurrir a la tabla Company, mediante una JOIN entre las dos tablas con el identificador de la compañía como elemento de relación.

Dado que solicitan la compañía con mayor gasto, ordenamos los resultados de la consulta por el gasto promedio decreciente y aplicamos Limit 1 para tener información de una única compañía.

```
SELECT t.company_id AS identificador,
       c.company_name AS nombre_compañia,
       round(AVG(t.amount),2) AS gasto_promedio
FROM transactions.transaction AS t
JOIN transactions.company AS c
  ON t.company_id=c.id
WHERE declined=0
GROUP BY identificador
ORDER BY gasto_promedio DESC
LIMIT 1;
```

En definitiva, la empresa con el mayor gasto promedio en las transacciones aceptadas (481.86 euros) es Eget Ipsum Ltd, cuyo identificador es b-2398 (dato incluido en caso de que sea de interés para futuros análisis).

identificador	nombre_compañía	gasto_promedio
b-2398	Eget Ipsum Ltd	481.86

NIVEL 2

Ejercicio 1. Tu jefe está redactando un informe de cierre del año y te solicita que le envíes información relevante para el documento. Para ello te solicita verificar si en la base de datos existen compañías con identificadores (id) duplicados.

La base de datos se compone de dos tablas, Company y Transaction. La tabla de company tiene los identificadores únicos de las compañías, tal y como se ha explicado en la introducción. En caso de duda, la presencia de duplicados se puede identificar haciendo un conteo de los id de compañías y un conteo distintivo (id únicos) de esos mismos id en las tablas que componen la base de datos.

```
SELECT COUNT(id) - COUNT(DISTINCT id) AS 'cantidad compañías duplicadas'
FROM transactions.company;
```

```
SELECT COUNT(company_id) 'cantidad total de compañías',
       COUNT(DISTINCT company_id) AS 'cantidad de compañías unicas'
FROM transactions.transaction;
```

En la tabla Company obtenemos el mismo número de id, por lo que sabemos que esa tabla contiene registros únicos para cada compañía (primary key de dicha tabla, tal y como se ha explicado en la introducción).

cantidad compañías duplicadas
0

En cambio, en la tabla Transaction obtenemos un total de 587 registros de compañías, 100 de los cuales corresponden a compañías con identificador único. Por tanto, en la tabla Transaction sí que hay varios registros de transacciones para la misma compañía.

cantidad total de compañías	cantidad de compañías unicas
587	100

Ejercicio 2. ¿En qué día se realizaron las cinco ventas más costosas? Muestra la fecha de la transacción y la sumatoria de la cantidad de dinero.

Para saber los días en que se realizaron las 5 mayores ventas, debemos hacer un sumatorio de la cantidad de gastos (amount) de la tabla Transaction, teniendo en cuenta únicamente las operaciones aceptadas (declined=0). El sumatorio se hace en función de la fecha, por lo que debemos aplicar un GROUP BY a la fecha. Como en la base de datos se incluye en un mismo campo tanto la fecha como la hora en que se realizó la transacción, aplicamos la función DATE para que nos detecte únicamente la fecha.

Finalmente, ordenamos los resultados en orden decreciente del sumatorio y aplicamos un LIMIT para tener los resultados de los 5 días con mayores ventas.

```
SELECT DATE(timestamp) Fecha_transacción,  
       SUM(amount) cantidad_dinero  
FROM transactions.transaction  
WHERE declined=0  
GROUP BY Fecha_transacción  
ORDER BY cantidad_dinero DESC  
LIMIT 5;
```

Una vez realizada la consulta, obtenemos que los 5 días con mayores ventas totales son: 20 de diciembre de 2021 (con 1532.36 euros), 22 de abril de 2021 (con 1397.96 euros), 9 de mayo de 2021 (con 1344.37 euros), 26 de febrero de 2022 (con 1337.62 euros) y 29 de marzo de 2021 (con 1325.12 euros).

Fecha transacción	cantidad de dinero
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

Ejercicio 3. ¿En qué día se realizaron las cinco ventas de menor valor? Muestra la fecha de la transacción y la sumatoria de la cantidad de dinero.

Para saber los días en que se realizaron las 5 ventas con menor valor se realiza el mismo procedimiento que en el caso anterior, con la diferencia de que, en este caso, hay que ordenar el sumatorio de gastos por orden ascendiente.

```
SELECT DATE(timestamp) AS Fecha_transacción,  
       SUM(amount) AS cantidad_dinero  
FROM transactions.transaction  
WHERE declined=0  
GROUP BY Fecha_transacción  
ORDER BY cantidad_dinero ASC  
LIMIT 5;
```

Una vez realizada la consulta, obtenemos que los 5 días con menores ventas totales son: 4 de enero de 2022 (con 15.05 euros), 27 de abril de 2021 (con 18.08 euros), 24 de enero de 2022 (con 23.86 euros), 27 de febrero de 2022 (con 30.76 euros) y 14 de enero de 2022 (con 37.55 euros).

Fecha transacción	cantidad de dinero
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

Ejercicio 4. ¿Cuál es el promedio de gasto por país? Presenta los resultados ordenados de mayor a menor promedio.

Para saber el promedio de gasto por país, debemos relacionar las dos tablas de la base de datos, ya que cada una contiene parte de la información que buscamos (Company - país, Transaction – gastos).

Aplicamos una (INNER) JOIN de las dos tablas con el id de la compañía como elemento de relación. Como nos piden el promedio de gasto por país, calculamos el AVG de los gastos aceptados (declined=0) en función del país (GROUP BY país) y redondeamos el resultado a 2 decimales para cumplir con el formato.

Finalmente, ordenamos los resultados por el AVG calculado, de mayor a menor (en orden decreciente).

```
SELECT country AS pais,
       round(AVG(amount),2) AS gasto_promedio
FROM transactions.company AS c
     JOIN transactions.transaction AS t
     ON c.id=t.company_id
WHERE declined=0
GROUP BY pais
ORDER BY gasto_promedio DESC;
```

Por tanto, con esta consulta obtenemos el gasto promedio de los 15 países presentes en la base de datos, los 5 primeros de los cuales son los representados en la tabla a continuación.

pais	gasto_promedio
United States	287.53
Ireland	285.83
Sweden	276.67
United Kingdom	271.77
Canada	261.94

En definitiva, tenemos a Estados Unidos con mayor gasto (287.53 euros), seguido de Irlanda (285.83 euros), de Suecia (276.67 euros), etc., y terminamos con España (26.22 euros).

NIVEL 3

Ejercicio 1. Presenta el nombre, teléfono y país de las compañías, junto con la cantidad total gastada, de aquellas que realizaron transacciones con un gasto comprendido entre 100 y 200 euros. Ordena los resultados de mayor a menor cantidad gastada.

Para obtener estos resultados, de nuevo tenemos que relacionar las dos tablas mediante una JOIN con identificador de compañía como elemento de relación.

Como se pide la cantidad de gasto de las transacciones comprendidas entre los 100 y 200 euros, aplicamos el filtro de que los gastos se encuentren en ese rango.

Finalmente, ordenamos los resultados por el orden decreciente de los gastos.

```
SELECT company_name AS nombre_empresa,  
       phone AS telefono,  
       country AS pais,  
       amount AS gasto  
FROM transactions.company AS c  
JOIN transactions.transaction AS t  
  ON c.id=t.company_id  
WHERE declined=0 AND amount BETWEEN 100 AND 200  
ORDER BY gasto DESC;
```

Con esta consulta obtenemos 101 registros de transacciones con gastos de entre 100 y 200 euros, los cinco primeros de los cuales se muestran en la tabla a continuación:

nombre_empresa	telefono	pais	gasto
Lorem Eu Incorporated	01 83 66 62 07	Canada	199.83
Ut Semper Foundation	01 60 36 33 06	Sweden	199.81
Non Institute	06 77 15 31 14	United Kingdom	196.71
Enim Condimentum Ltd	09 55 51 66 25	United Kingdom	195.06
Ut Semper Foundation	01 60 36 33 06	Sweden	193.64

Ejercicio 2. Indica el nombre de las compañías que realizaron compras el 16 de marzo de 2022, 28 de febrero de 2022 y 13 de febrero de 2022.

Para obtener la información deseada, tenemos que relacionar las dos tablas mediante una JOIN con identificador de compañía como elemento de relación.

Una vez hecho esto, buscamos las transacciones realizadas los días especificados, aplicando un filtro con las 3 fechas (aplicando función DATE en el campo que contiene fecha y hora).

```
SELECT c.id AS identificador,  
       company_name AS nombre_compañia,  
       amount AS gasto,  
       timestamp AS Fecha_hora_transaccion  
FROM transactions.company AS c  
JOIN transactions.transaction AS t  
  ON c.id=t.company_id  
WHERE DATE(timestamp) IN ('2022-03-16', '2022-02-28', '2022-02-13')  
ORDER BY Fecha_hora_transaccion;
```

En total, obtenemos 9 transacciones realizadas, ordenadas por la fecha y hora en que se realizaron, que se pueden observar en la tabla a continuación.

Podemos observar que hay ciertas compañías, como Malesuada PC, que realizaron transacciones en más de una fecha concretada; mientras que hay otras, como Arcu LLP, que realizaron compras en una única fecha de las tres especificadas.

identificador	nombre_compañia	gasto	Fecha_hora_transaccion
b-2302	Nunc Interdum Incorporated	87.44	2022-02-13 04:07:29
b-2278	Arcu LLP	432.55	2022-02-13 12:02:43
b-2494	Malesuada PC	408.58	2022-02-13 15:13:38
b-2274	Sed LLC	33.34	2022-02-13 16:33:50
b-2346	Ut Semper Foundation	186.94	2022-02-28 00:10:50
b-2302	Nunc Interdum Incorporated	67.65	2022-02-28 05:55:03
b-2494	Malesuada PC	410.49	2022-02-28 08:09:21
b-2494	Malesuada PC	200.72	2022-03-16 02:35:05
b-2362	Lorem Eu Incorporated	329.03	2022-03-16 14:01:36

Finalmente, como nos piden el nombre de las compañías sin más detalles, seleccionamos únicamente los campos del nombre de compañía y su identificador (en caso de necesidad para futuros análisis) y agrupamos los resultados por el identificador (eliminamos compañías duplicadas, no tenemos en cuenta la fecha en que realizaron las transacciones; también se podría aplicar SELECT DISTINCT del identificador, sin necesidad de agrupar).

```
SELECT c.id AS identificador,
       company_name AS nombre_compañia
FROM transactions.company AS c
     JOIN transactions.transaction AS t
     ON c.id=t.company_id
WHERE DATE(timestamp) IN ('2022-03-16', '2022-02-28', '2022-02-13')
GROUP BY identificador;
```

En definitiva, tenemos 6 compañías que realizaron las compras el 16 de marzo de 2022, 28 de febrero de 2022 y 13 de febrero de 2022: Sed LLC, Arcu LLP, Nunc Interdum Incorporated, Ut Semper Foundation, Lorem Eu Incorporated y Malesuada PC.

identificador	nombre_compañia
b-2274	Sed LLC
b-2278	Arcu LLP
b-2302	Nunc Interdum Incorporated
b-2346	Ut Semper Foundation
b-2362	Lorem Eu Incorporated
b-2494	Malesuada PC