

Spring2

- 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.

-Fact Table – Tabla transaction

Tabla que muestra todos los datos relacionados a una transacción efectuada o no. También datos de los clientes y empresas. Fk de esta tabla es company_id

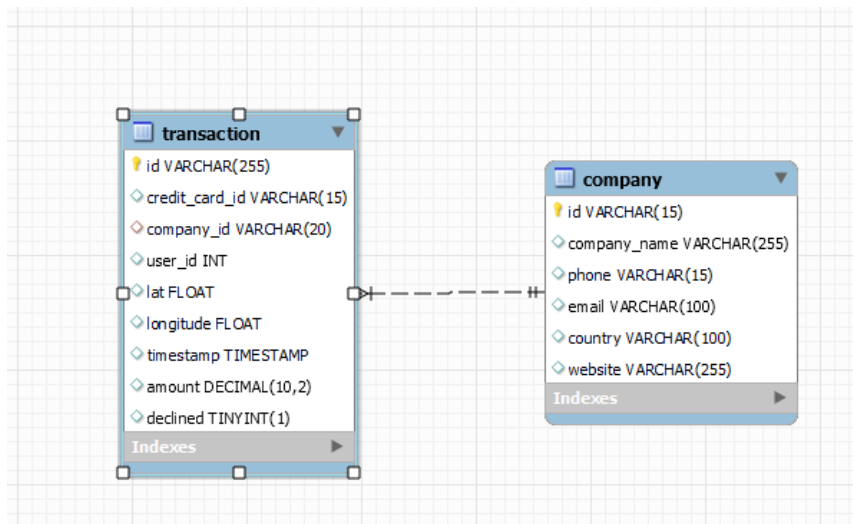
- id:Tipo Varchard(255) . Primary key .Identificador irrepetible alfanumérico.
- creditcard_id:Tipo Varchard(15) Dato flexible 15 dígitos max. Tarjeta de crédito.
- company_id:Tipo Varchard (20) Dato flexible 20 dígitos max. Identificador de compañía.
- user_id: Tipo Int. Dato de tipo entero. Identificador de usuario.
- lat:Tipo Float parecido a decimal, pero ocupa menos espacio. Latitud de un lugar.
- longitude:Tipo Float parecido a decimal ,pero ocupa menos espacio. Longitud de un lugar.
- timestamp: Tipo Timestamp es el tiempo y hora de un momento específico.
- amount:Tipo Decimal parecido a float , ocupa mas espacio. Numero específico del gasto de cada transacción.
- declined:Tipo Tinyint para almacenar tipos de datos enteros pequeños. 0 = si y 1= Se declino la transacción.

-Tabla de dimensión- Tabla company donde la Pk es id. Datos referentes a las empresas.

- id:Varchar(15). Dato flexible max 15 dígitos .Primary key e identificador único.
- company_name: Varchard(255). Dato flexible 255 digitos max. Nombre de las compañías.
- phone:Varchard (15) .Dato flexible 15 dígitos max. Teléfono empresas.
- email:Varchard(100).Dato flexible 100 dígitos max. Dirección de correo electrónico de las empresas.
- country:Varchard(100).Dato flexible 100 dígitos max. Pais de procedencia de las empresas.
- website:Varchard(255).Dato flexible 255 dígitos max. Pagina web de las empresas.

Granularidad company 1 →N transactions

Las tablas se unen por la Pk de company que es id y por la Fk de transaction que es company_id.



Select*
from transaction;

select*
from company;

- Ejercicio 2

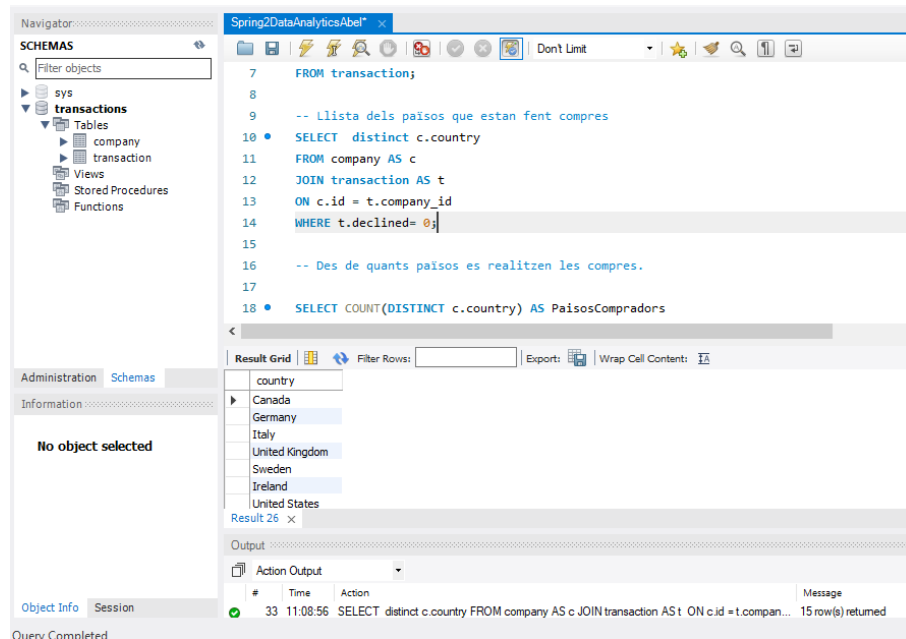
Utilizando JOIN realizarás las siguientes consultas:

Listado de los países que están realizando compras.

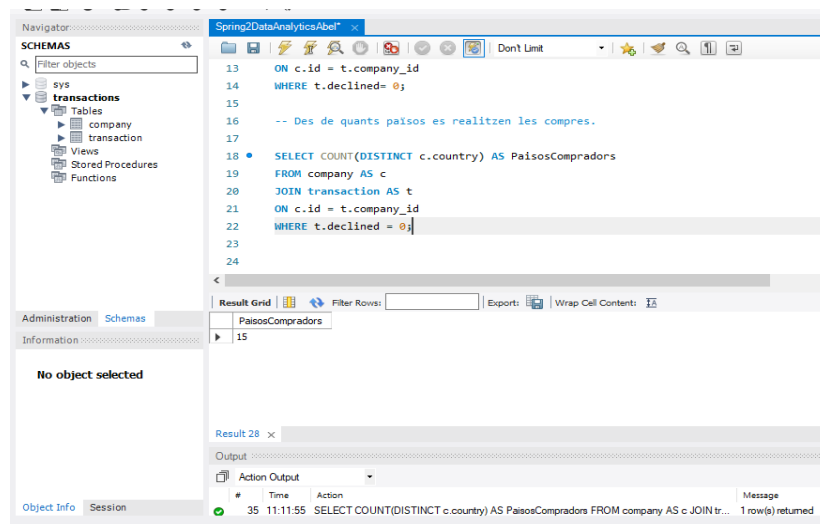
SELECT distinct c.country → Selecciono el campo de country de la tabla company y le añado distinct para no tener duplicados.

FROM company AS c → De la tabla company le añado un AS(Alias) = c para que sea mas rapido de escribir.

JOIN transaction AS t → Combino con la tabla transaction, que también le añado un alias t.
 ON c.id = t.company_id → Realizo la combinación a través de la pk c.id a t.company fk, para unir las dos tablas.
 WHERE t.declined = 0; → El filtro where con el campo declined, con el 0 para confirmar que se hizo la operación.

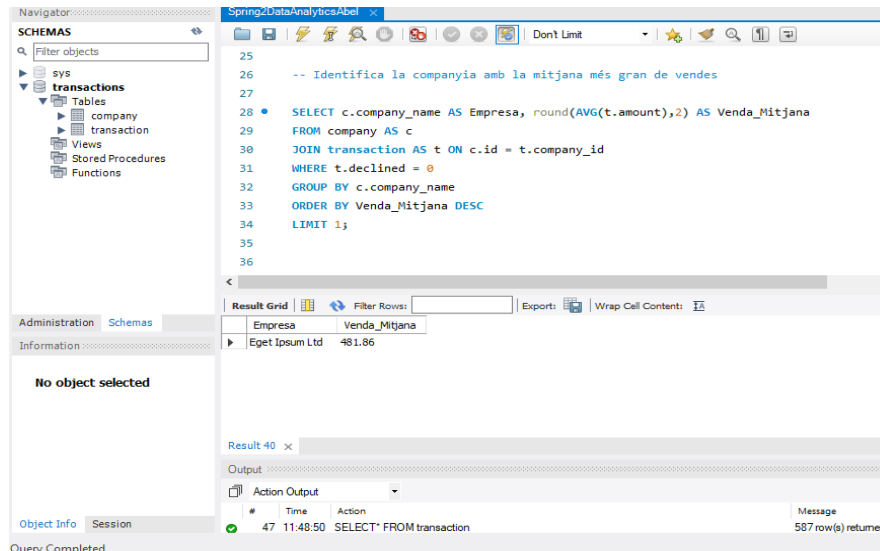


-Desde cuántos países se realizan las compras.
 SELECT COUNT(DISTINCT c.country) AS PaisosCompradors → Select con una función de agregación, con el campo con el distinct para no tener duplicidades. También un alias, para que sea más comprensible.
 FROM company AS c → De la tabla company le añado un AS (Alias) = c para que sea más rápido de escribir.
 JOIN transaction AS t → Combino con la tabla transaction, que también le añado un alias t.
 ON c.id = t.company_id → Realizo la combinación a través de la pk c.id a t.company fk, para unir las dos tablas.
 WHERE declined = 0; → El filtro where con el campo declined, con el 0 para confirmar que se hizo la operación.



-Identifica a la compañía con la mayor media de ventas.
 SELECT c.company_name AS Empresa, round(AVG(t.amount),2) AS Venda_Mitjana → Dos campos con alias, uno sin función de agregación y el otro sí, además con round (AVG tabla.campo), número de dígitos en el resultado final.

FROM company AS c → De la tabla company le añado un AS (Alias) = c para que sea mas rápido de escribir.
 JOIN transaction AS t → Combino con la tabla transaction, que también le añado un alias t.
 ON c.id = t.company_id → Realizo la combinación a través de la pk c.id a t.company fk, para unir las dos tablas
 WHERE t.declined = 0 → El filtro where con el campo declined, con el 0 para confirmar que se hizo la operación.
 GROUP BY c.company_name → Agrupo por este campo, porque al tener otro campo con una función de agregación. Si quisiera poner un filtro en el otro campo tendría que utilizar el HAVING.
 ORDER BY Venda_Mitjana DESC → Utilizo el campo t.amount (As-Venda_Mitjana) para ordenar y desc para que sea de mayor a menor.
 LIMIT 1; → Para que solo muestre un resultado. En este caso el primero con las condiciones del ORDER BY.



- Ejercicio 3

Utilizando sólo subconsultas (sin utilizar JOIN):

Muestra todas las transacciones realizadas por empresas de Alemania.

SELECT * → Todos los registros de la tabla.

FROM transaction as t → Tabla con alias de donde el select de la query exterior, reflejara los datos.

WHERE (SELECT c.country → Dentro el filtro Where empezamos la subconsulta, seleccionando el campo de donde obtendra la informacion la query exterior.

FROM company AS c → De esta tabla con un alias, para facilitar su escritura y lectura.

WHERE c.id = t.company_id = 'Germany' → Aplica una condición que involucra una subconsulta. La subconsulta se compara con el valor 'Germany'. Solo se incluirán en el resultado final las transacciones donde el país obtenido de la subconsulta sea igual a 'Germany'

;

```

40 -- Mostra totes les transaccions realitzades per empreses d'Alemanya.
41 SELECT *
42 FROM transaction AS t
43 WHERE (SELECT c.country
44        FROM company AS c
45        WHERE c.id = t.company_id) = 'Germany'
46
47
48 -- Llista les empreses que han realitzat transaccions per un amount superior a la mitjana de totes les transaccions.
49 SELECT c.company_name
50 FROM company AS c
51 WHERE EXISTS (

```

Result Grid

id	credit_card_id	company_id	user_id	lat	longitude	timestamp	amount	declined
0466A42E-47CF-4D24-FD01-C0B689713128	CCU-4219	b-2302	170	-43.9695	-117.525	2021-07-26 07:29:18	49.53	0
0A476ED9-0C13-1962-F87B-03563924B539	CCU-4359	b-2302	221	-56.4901	114.801	2022-02-26 20:33:54	430.49	0
00D2E608-5C9E-D1B3-4999-899F-43AD735A	CCU-2959	b-2234	275	9.68811	130.282	2021-04-17 05:30:17	252.47	1
10881D1D-5823-A76C-55EF-C568E49A05DD	CCU-2938	b-2222	275	83.7839	-178.86	2021-07-07 17:43:16	293.57	0
122DC333-E19F-D629-0CD8-9C54CF-IE8B9A	CCU-4366	b-2302	221	29.6372	-166.173	2021-06-09 06:04:14	172.01	0
1332678A-2E7D-957C-C48C-6459A3B3ED54	CCU-4520	b-2302	210	20.6724	14.9732	2021-12-29 20:36:23	17.97	0
14CAE5B5-9F81-3E4A-4C85-9EA4167534F4	CCU-4849	b-2302	189	-53.6202	93.0533	2021-12-31 00:29:42	388.04	0

Output

Action Output

#	Time	Action	Message
49	12:48:24	SELECT * FROM transaction AS t WHERE (SELECT c.country FROM company AS c WHERE c.id = t.company_id) = 'Germany'	118 row(s) returned

Query Completed

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

SELECT c.company_name → Recuperar el nombre de la empresa (company_name) de la tabla company (con alias c)
 FROM company AS c → Tabla de la que se obtendrán las empresas. Se utiliza un alias c para facilitar la escritura
 WHERE EXISTS (→ Intenta buscar transacciones en la tabla transaction (con alias t). Filtra los resultados para incluir solo transacciones donde el company_id coincida con el c.id de la empresa actual (en la tabla company). Básicamente, busca transacciones asociadas a la empresa actual. Aplica una condición que utiliza una subconsulta positiva (EXIST).

Una subconsulta positiva verifica si hay al menos un registro coincidente en otra consulta.

SELECT * → Subconsulta interna que filtra los resultados para incluir solo transacciones donde el company_id coincida con el c.id de la empresa actual (en la tabla company).

FROM transaction AS t → Busca transacciones en la tabla transaction (con alias t).

WHERE t.company_id = c.id → Subconsulta anidada calcula el precio pagado promedio de transacción en todas las transacciones de la tabla transaction. Utiliza la función AVG y asigna un alias avg_transaction al resultado.

AND t.amount > (→ verifica si el cuenta (amount) de alguna transacción (t) asociada a la empresa actual (c) es mayor que el promedio de transacciones (avg_transaction) calculado en la subconsulta anidada.

SELECT AVG(amount) AS avg_transaction → Verifica si el precio pagado (amount) de alguna transacción (t) asociada a la empresa actual (c) es mayor que el promedio de transacciones (avg_transaction) calculado en la subconsulta anidada.

FROM transaction → Desde la tabla transaction. Se le puede añadir el alias (AS t).

)

```

49 SELECT c.company_name
50 FROM company AS c
51 WHERE EXISTS (
52     SELECT *
53     FROM transaction AS t
54     WHERE t.company_id = c.id
55     AND t.amount > (
56         SELECT AVG(amount) AS avg_transaction
57         FROM transaction
58     )
59 )
60

```

Result Grid

company_name
Ac Fermentum Incorporated
Magna A Neque Industries
Fusce Corp.
Ante Jaculis Nec Foundation
Donec Ltd
Sed Nunc Ltd
Nascetur Ridiculus Mus Inc.

Output

Action Output

#	Time	Action	Message
51	12:49:39	SELECT c.company_name FROM company AS c WHERE EXISTS (SELECT * FROM transaction AS t WHERE t.company_id = c.id AND t.amount > (SELECT AVG(amount) AS avg_transaction FROM transaction))	70 row(s) returned

Query Completed

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

SELECT c.company_name → recuperar el nombre de la empresa (company_name) de la tabla company (con alias c).
 FROM company AS c → La tabla de la que se obtendrán las empresas. Se utiliza un alias c.
 WHERE NOT EXISTS (→ subconsulta negativa (NOT EXISTS) verifica si no hay registros coincidentes en otra consulta.

SELECT *→ Subconsulta interna intenta encontrar transacciones en la tabla transaction (con alias t). Filtra los resultados para incluir solo transacciones donde el company_id coincida con el c.id de la empresa actual (en la tabla company). Básicamente, busca transacciones asociadas a la empresa actual.

FROM transaction AS t→ Tabla subquery interior.

WHERE t.company_id = c.id→ Filtrado de las transacciones que son relevantes para la empresa actual que se está examinando en la consulta exterior

);

