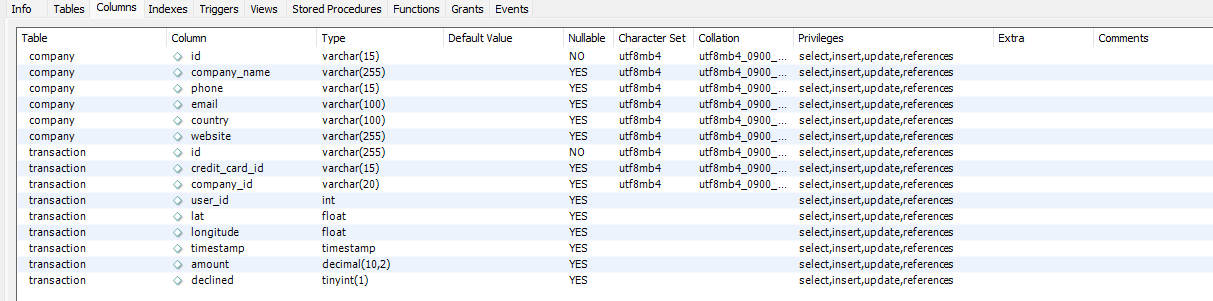
# NIVEL 1

## Ejercicio 1.1 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.

Para trabajar con estos sets de datos se han importado ambas tablas facilitadas “company” y “transactions” bajo el schema “Transactions” en MySQL Workbench,



La tabla “Company” esta compuesta por 6 columnas diversas. Tal y como se puede ver en la foto de arriba, estas columnas son del tipo VARCHAR() , que permite almacenar texto de longitud variable en bases de datos SQL. Por otro lado podemos observar que la columna id en esta tabla es NO “Nullable”, por lo que podemos asumir que es la PRIMARY KEY de la tabla, que se utiliza para mantener la integridad de los datos, garantizando que siempre tenga un valor.

La PRIMARY KEY se trata de una columna o un conjunto de columnas que identifican de forma única cada fila de una tabla, cuyo propósito es establecer relaciones entre tablas y mantener la integridad de los datos.

Por otro lado, la tabla “Transaction” esta compuesta por 9 columnas, pero de diversa tipología, entre las que destacan VARCHAR, INT / TINYINT, FLOAT, TIMESTAMP y DECIMAL

INT - Se utiliza para almacenar números enteros (sin decimales), utilizamos TINYINT para almacenar booleans de 0 o 1

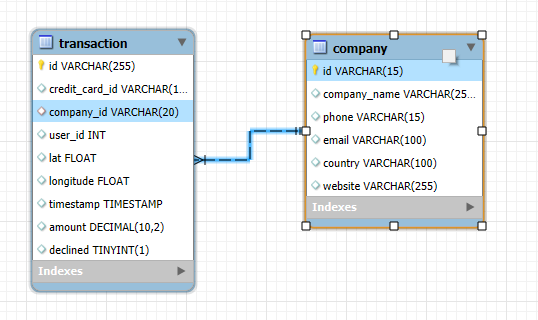
FLOAT se utiliza para almacenar números con decimales, en este caso, datos de geolocalización

TIMESTAMP: Se utiliza para almacenar fechas y horas con un formato estandarizado,

DECIMAL: Se utiliza para almacenar números con decimales con precisión exacta. Se especifica la precisión (el número total de dígitos) y la escala (el número de dígitos después del punto decimal). En este caso se utiliza para representar valores monetarios

El esta tabla, también podemos observar que la columna id también es No “Nullable”, por lo que podemos asumir que es la PRIMARY KEY de esta tabla

Pero, ¿como están relacionadas estas tablas? Además de poder confirmar que las columnas ID de ambas tablas son PRIMARY KEY, podemos observar en el siguiente flujo que la columna ID de la tabla “company” (PRIMARY Key) está directamente relacionada con la columna company\_id (FOREIGN KEY) de la tabla “transaction” .



La FOREIGN KEY es una columna o conjunto de columnas en una tabla que hace referencia a la PRIMARY KEY de otras tablas su propósito principal es crear una conexión lógica entre las dos tablas, permitiendo que los datos relacionados se vinculen entre sí y asegurar la integridad referencial, lo que significa que los valores en la clave foránea deben coincidir con los valores existentes en la clave primaria de la tabla referenciada, o ser nulos. (Cuando se inserta o actualiza un registro en la tabla relacionada, la base de datos verifica que el valor de FOREIGN KEY exista en la PRIMARY KEY de la tabla “padre”.)

En este caso, podremos vincular cada transacción a los datos de la compañía que la ha realizado.

## Ejercicio 1.2 Usando JOIN, realiza las siguientes queries

### Ejercicio 1.2.1 Lista de países desde donde se realizan las compras .

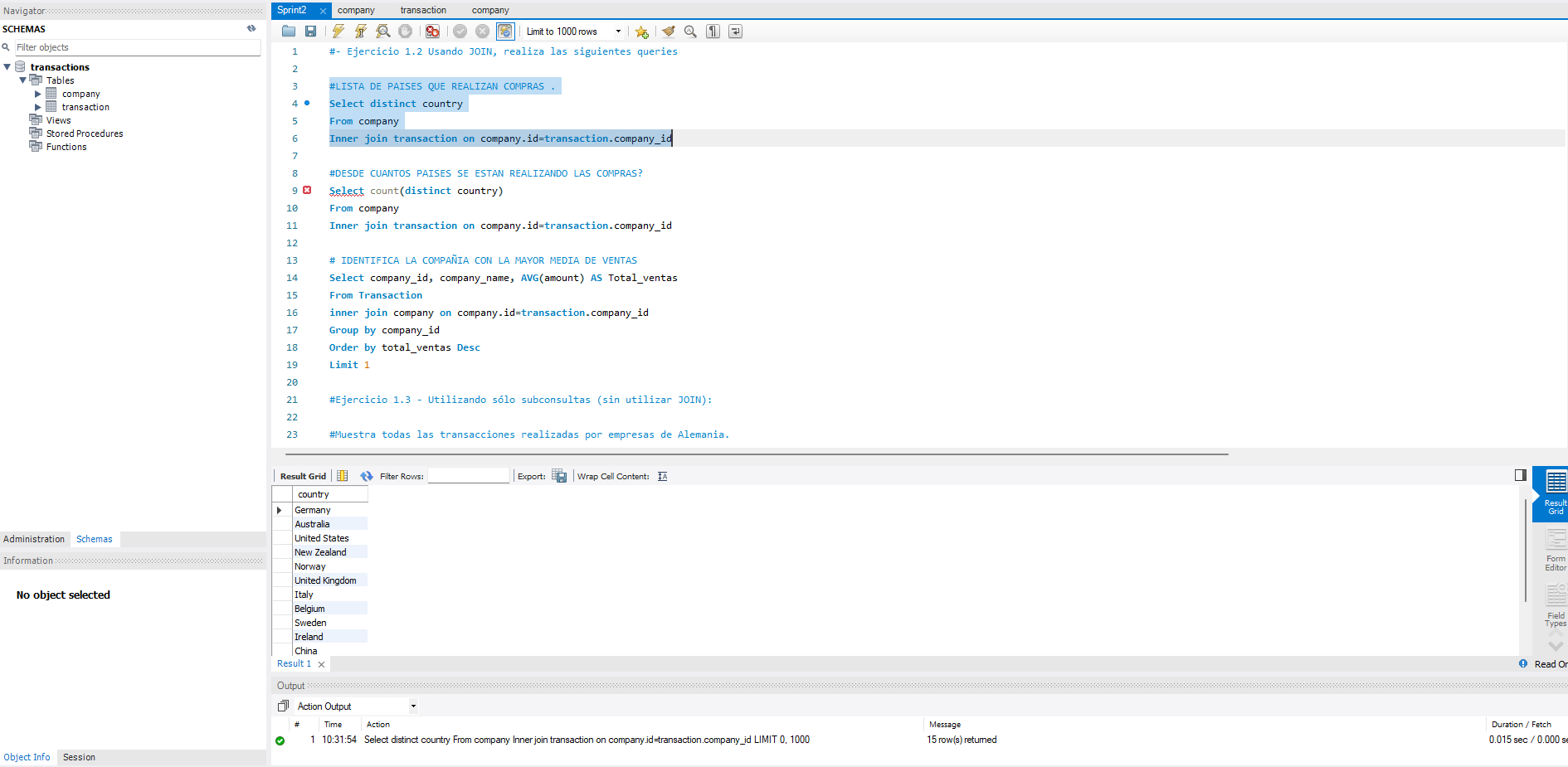
CONSULTA:

Select distinct country

From company

Inner join transaction on company.id=transaction.company\_id

CAPTURA



EXPLICACIÓN:

He extraído de la tabla “Company” los nombres únicos (DISTINCT) de países donde existen compañías que han realizado transacciones, combinando datos de las tablas "company" y "transaction" usando JOIN.

### Ejercicio 1.2.2 ¿DESDE CUANTOS PAISES SE ESTAN REALIZANDO LAS COMPRAS?

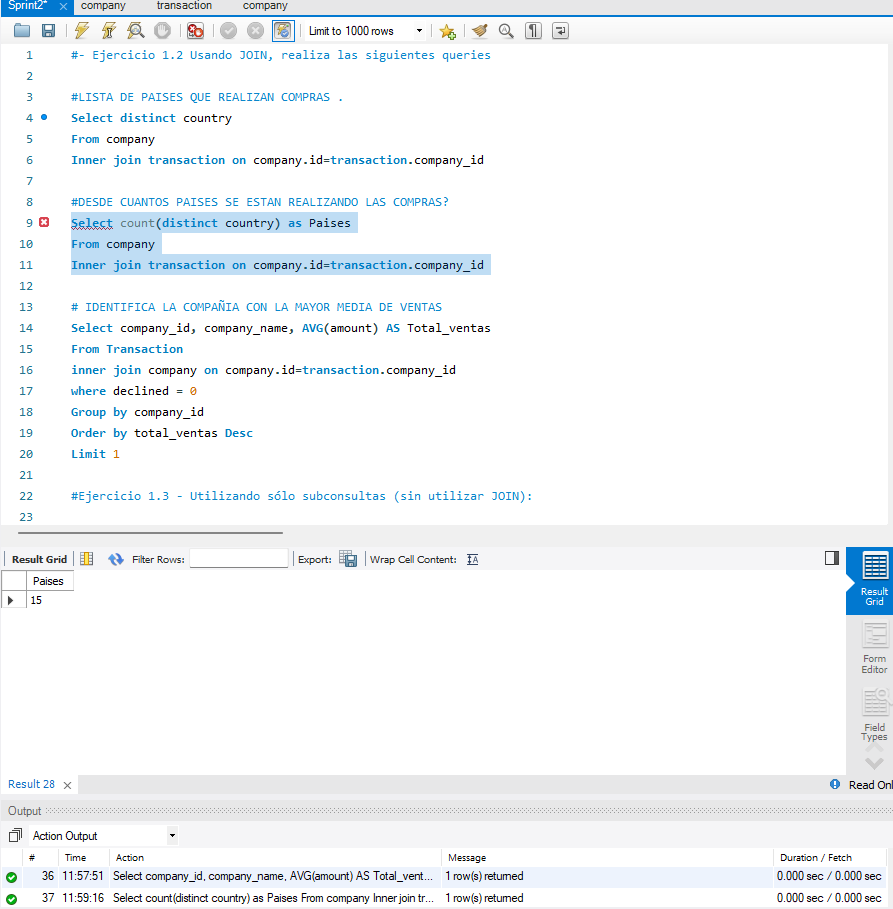
CONSULTA

Select count(distinct country) as Paises

From company

Inner join transaction on company.id=transaction.company\_id

CAPTURA



EXPLICACIÓN

La consulta combina las tablas "company" y "transaction" utilizando JOIN para vincular el id de las empresas con sus transacciones, seleccionando los nombres de países únicos donde estas empresas tienen actividad comercial registrada, contando el número total de esos países distintos, proporcionando así el recuento de países con transacciones -SELECT COUNT(Distinct Country)

### Ejercicio 1.2.3 IDENTIFICA LA COMPAÑIA CON LA MAYOR MEDIA DE VENTAS

CONSULTA:

Select company\_id, company\_name, AVG(amount) AS Total\_ventas

From Transaction

inner join company on company.id=transaction.company\_id

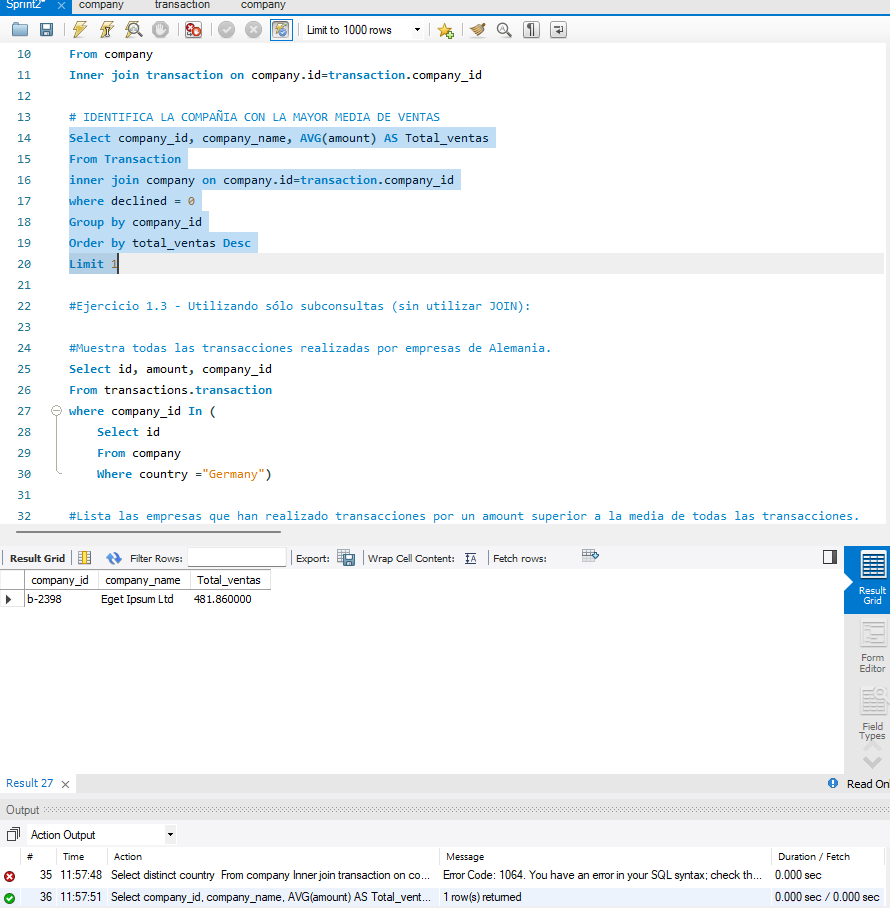
where declined = 0

Group by company\_id

Order by total\_ventas Desc

Limit 1

CAPTURA



EXPLICACIÓN

Esta consulta encuentra la compañía con el promedio de ventas más alto, mostrando el código de la compañía, su nombre, y la media (AVG) de ventas , combinando las tablas de “transaction” y “company” usando JOIN , calculando el promedio de ventas por compañía, ordenando los resultados de mayor a menor (ORDER BY \_ DESC ) y finalmente muestra solo la compañía con el promedio de ventas máximo (LIMIT 1) . Se han descartado las transacciones declinadas a la hora de calcular el promedio de ventas

## Ejercicio 1.3 - Utilizando sólo subconsultas (sin utilizar JOIN):

### Ejercicio 1.3.1 Muestra todas las transacciones realizadas por empresas de Alemania.

CONSULTA

Select id, amount, company\_id

From transactions.transaction

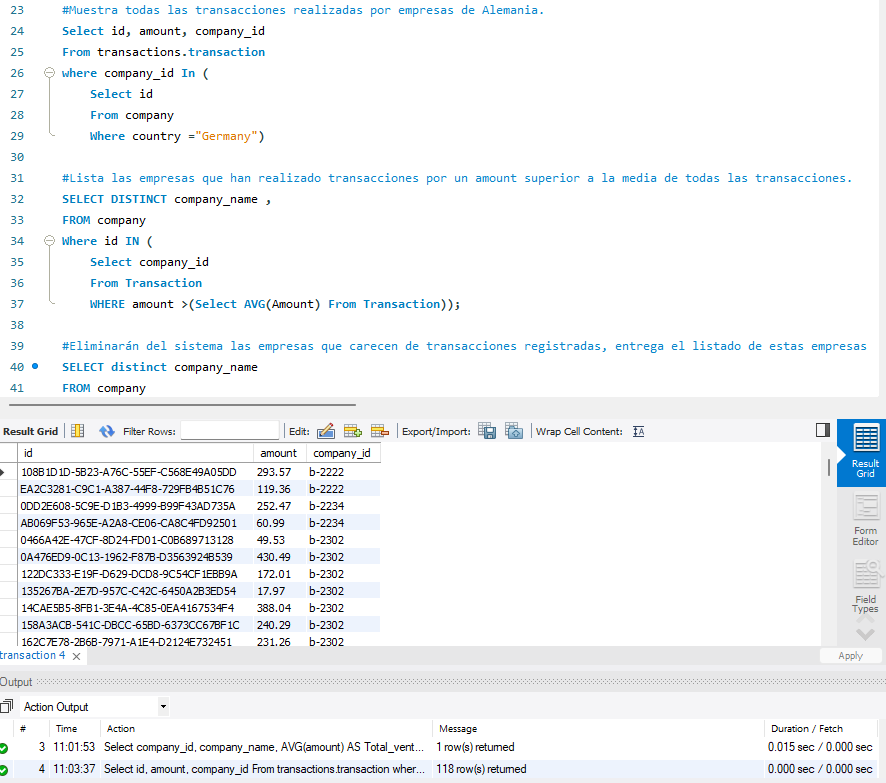
where company\_id In (

Select id

From company

Where country ="Germany")

CAPTURA



EXPLICACIÓN

Esta consulta muestra el ID de las transacciones, el montante, y el ID de la compañía de la tabla " transaction" para todas las transacciones realizadas por compañías ubicadas en "Germany". Para ello se ha verificado los IDs de las compañías con la tabla “company”y luego ha filtrado las transacciones para incluir solo aquellas cuyo país es “Germany”. En esta consulta se han tenido en cuenta todas las transacciones, incluidas las declinadas.

### Ejercicio 1.3.2 Lista las empresas que han realizado transacciones por un amount superior a la media de todas las transacciones.

CONSULTA

SELECT DISTINCT company\_name

FROM company

Where id IN (

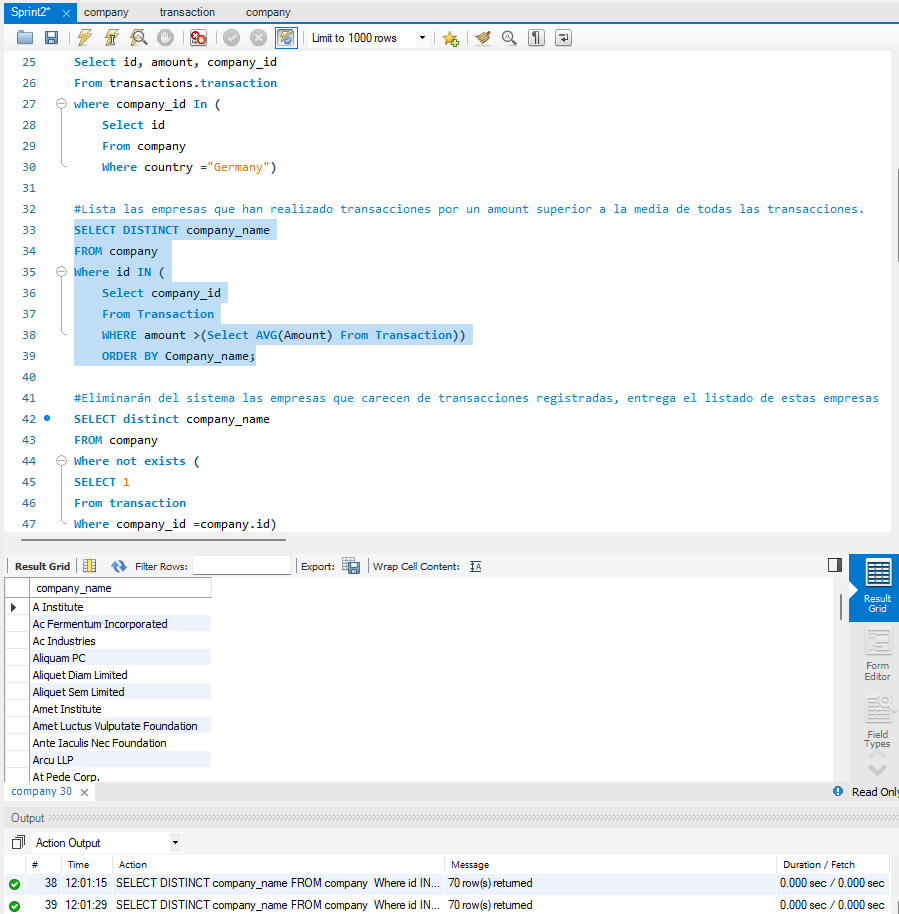
Select company\_id

From Transaction

WHERE amount >(Select AVG(Amount) From Transaction))

ORDER BY Company\_name ;

CAPTURA



EXPLICACIÓN

Esta consulta identifica y muestra los nombres únicos (DISTINCT) de las compañías (COMPANY\_NAME) cuyas transacciones superan el promedio general ( amount > AVG) de todas las transacciones. A través de una subconsulta, calcula el valor promedio de transacciones y luego selecciona las compañías que tienen al menos una transacción por encima de ese promedio. En esta consulta se han tenido en cuenta todas las transacciones, incluidas las declinadas.

### Ejercicio 1.3.3 Se eliminarán del sistema las empresas que carecen de transacciones registradas, entrega el listado de estas empresas

CONSULTA:

SELECT distinct company\_name

FROM company

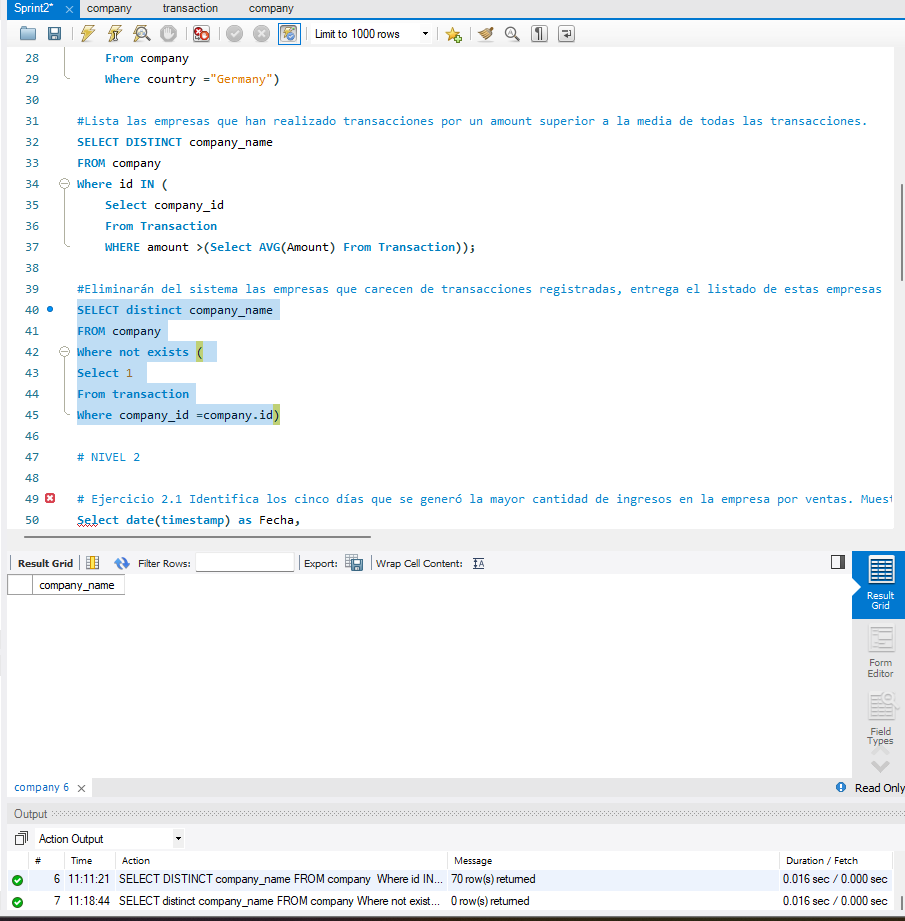
Where not exists (

Select 1

From transaction

Where company\_id =company.id)

CAPTURA



EXPLICACIÓN

Esta consulta encuentra y muestra los nombres únicos de las compañías que no tienen ninguna transacción registrada en la tabla "transaction", utilizando NOT EXISTS para filtrar las compañías que no tienen coincidencias en la tabla de transaction y company. En este caso, no hay ninguna compañía que no tenga ninguna transacción.

# NIVEL 2

## Ejercicio 2.1 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.

CONSULTA:

Select date(timestamp) as Fecha,

sum(amount) as Total\_diario

From Transaction

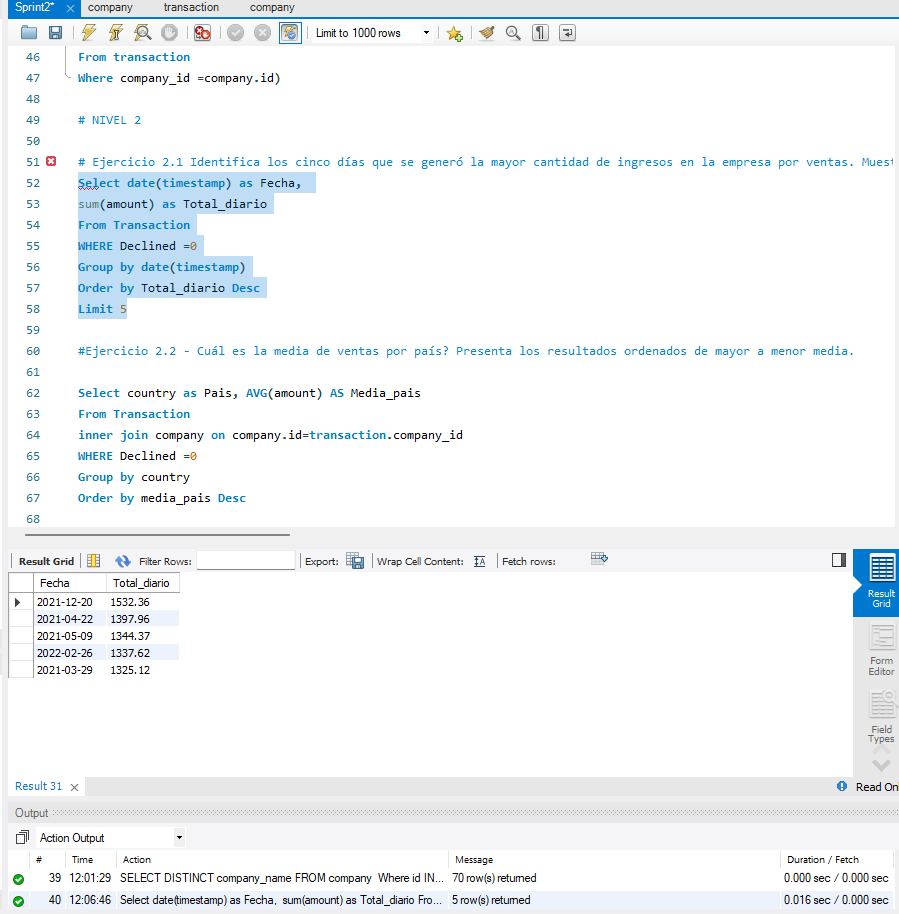
WHERE Declined =0

Group by date(timestamp)

Order by Total\_diario Desc

Limit 5

CAPTURA



EXPLICACION

Esta consulta devuelve el valor monetario total diario de transacciones ( sum amount) que no han sido rechazadas (where declined = 0) de 5 días con los mayores montos totales, ordenado de mayor a menor. Para identificar el dia, se ha extraído la fecha de la columna Timestamp (select date(timestamp) , ya que el timestamp incluye minutos y horas.

## Ejercicio 2.2 - Cuál es la media de ventas por país? Presenta los resultados ordenados de mayor a menor media.

CONSULTA:

Select country as Pais, AVG(amount) AS Media\_pais

From Transaction

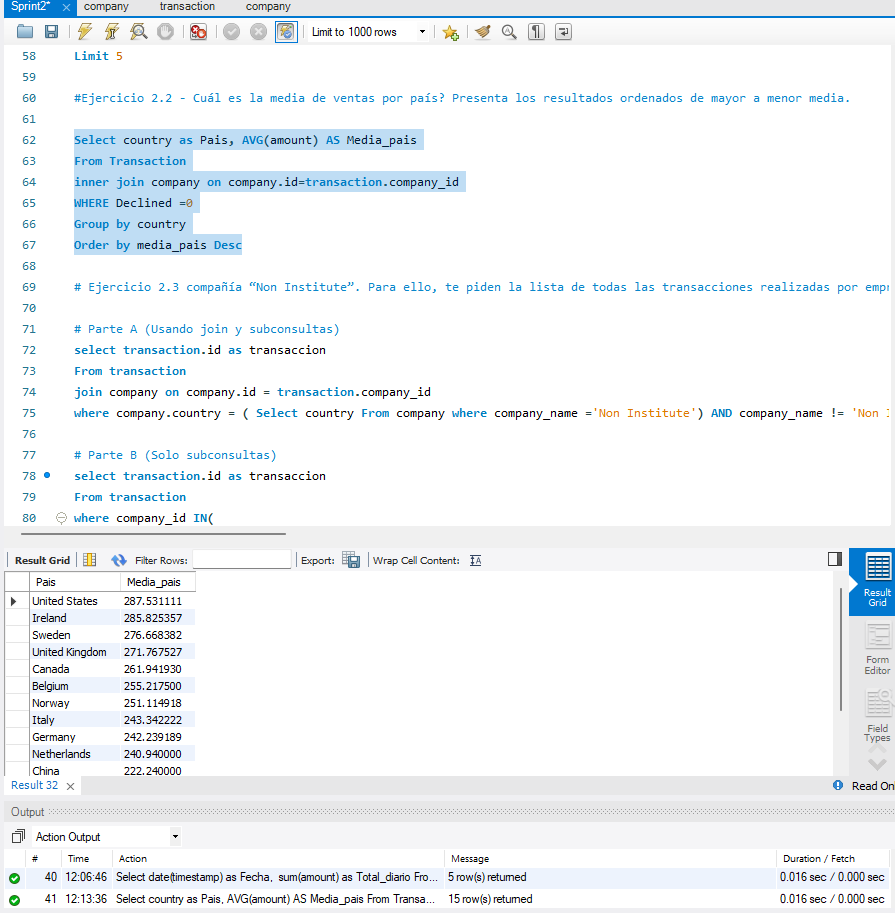
inner join company on company.id=transaction.company\_id

WHERE Declined =0

Group by country

Order by media\_pais Desc

CAPTURA



EXPLICACION

Esta consulta devuelve el valor promedio de las de transacciones exitosas (Declined = 0 ) para cada país, mostrando los resultados ordenados de mayor a menor promedio. Para ello se han combinado las tablas "Transaction" y "company", filtrado las transacciones rechazadas, agrupado por país y calcula el promedio del monto para cada grupo.

## Ejercicio 2.3 la lista de todas las transacciones realizadas por empresas que están ubicadas en el mismo país que la compañía “Non Institute”..

### # Ejercicio 2.3.1 (Usando join y subconsultas)

CONSULTA:

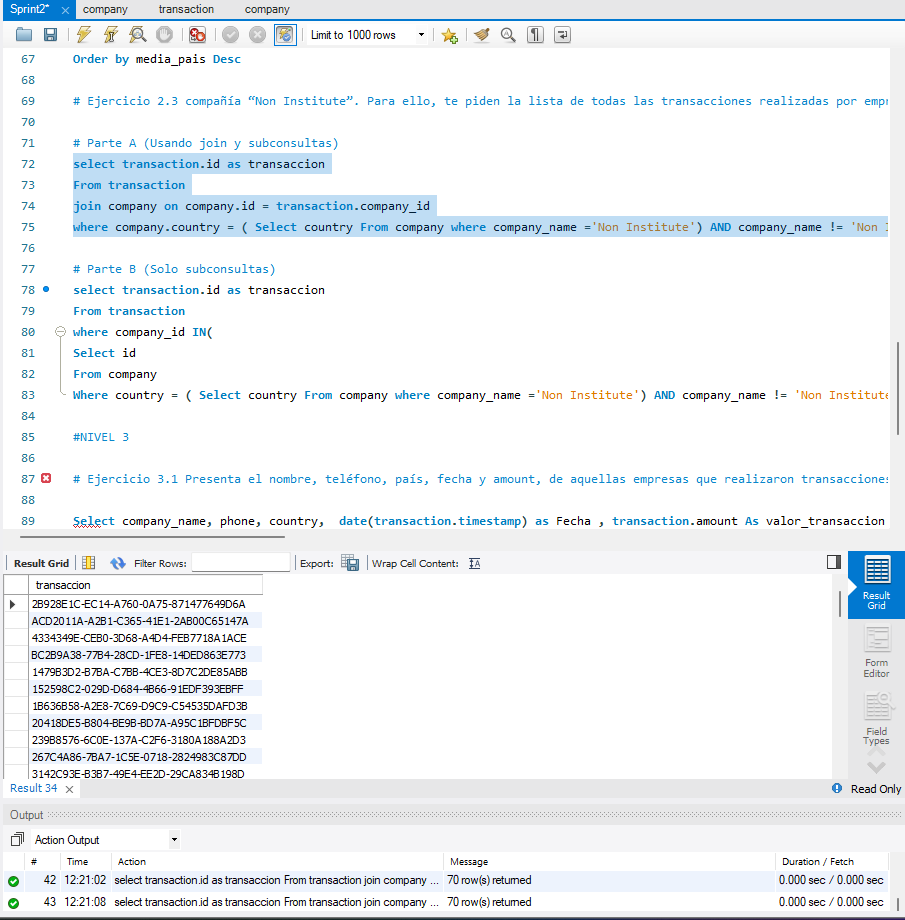
select transaction.id as transaccion

From transaction

join company on company.id = transaction.company\_id

where company.country = ( Select country From company where company\_name ='Non Institute') AND company\_name != 'Non Institute';

CAPTURA



EXPLICACIÓN:

Esta consulta devuelve una lista con los IDs de las transacciones realizadas, sin diferenciar si han sido declinadas o no, por compañías que están en el mismo país que "Non Institute", excluyendo las transacciones realizadas por la propia compañía "Non Institute". Para ello, primero, identifica el país de "Non Institute" y luego filtra las transacciones de otras compañías ubicadas en ese mismo país. Para esta consulta se ha utilizado un filtro AND \_\_\_ != , se podría haber utilizado también AND \_\_\_\_ NOT = . Se ha utlizado una combinación de JOIN y subconsultas

### Ejercicio 2.3.2 (Solo subconsultas)

CONSULTA:

select transaction.id as transaccion

From transaction

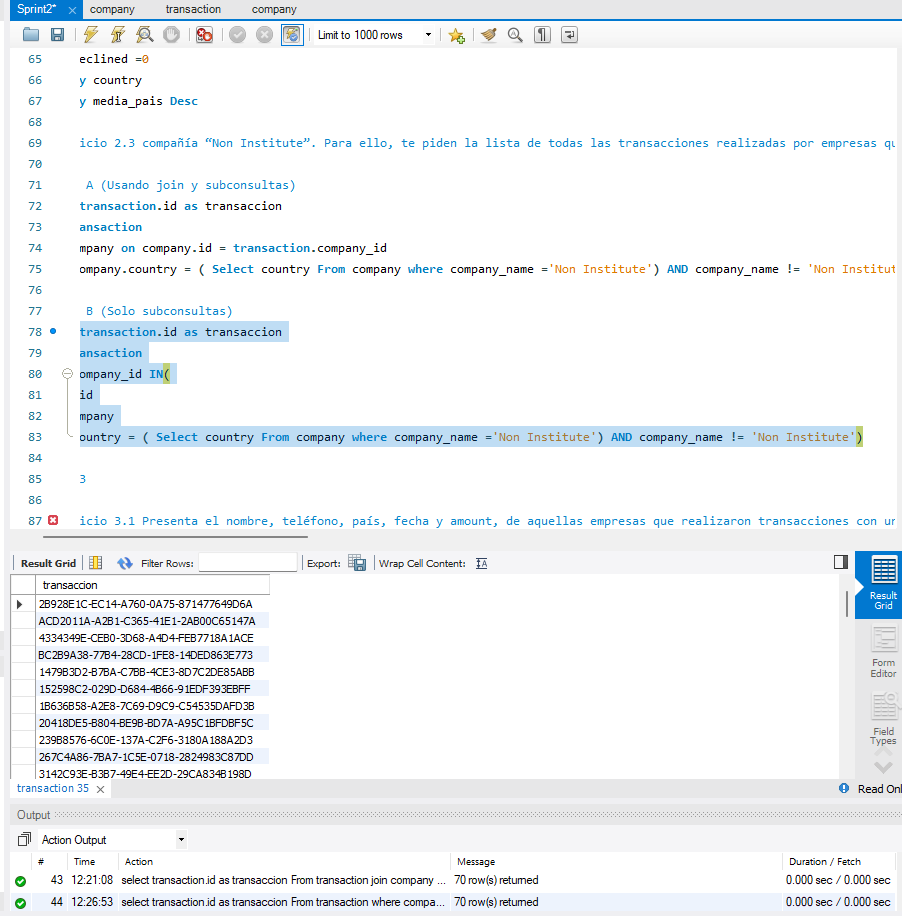
where company\_id IN(

Select id

From company

Where country = ( Select country From company where company\_name ='Non Institute') AND company\_name != 'Non Institute')

CAPTURA:



EXPLICACIÓN.

Al igual que el ejercicio anterior, esta consulta devuelve una lista con los IDs de las transacciones realizadas, sin diferenciar si han sido declinadas o no, por compañías que están en el mismo país que "Non Institute", excluyendo las transacciones realizadas por la propia compañía, sin embargo, aquí se emplea una subconsulta para obtener los IDs de las compañías que cumplen con el criterio del país y el nombre, y luego filtra las transacciones que tienen esos IDs de compañía.

# NIVEL 3

## Ejercicio 3.1 Presenta el nombre, teléfono, país, fecha y amount, de aquellas empresas que realizaron transacciones con un valor comprendido entre 100 y 200 euros y en alguna de estas fechas: 29 de abril de 2021, 20 de julio de 2021 y 13 de marzo de 2022. Ordena los resultados de mayor a menor cantidad.

CONSULTA:

Select company\_name, phone, country, date(transaction.timestamp) as Fecha , transaction.amount As valor\_transaccion

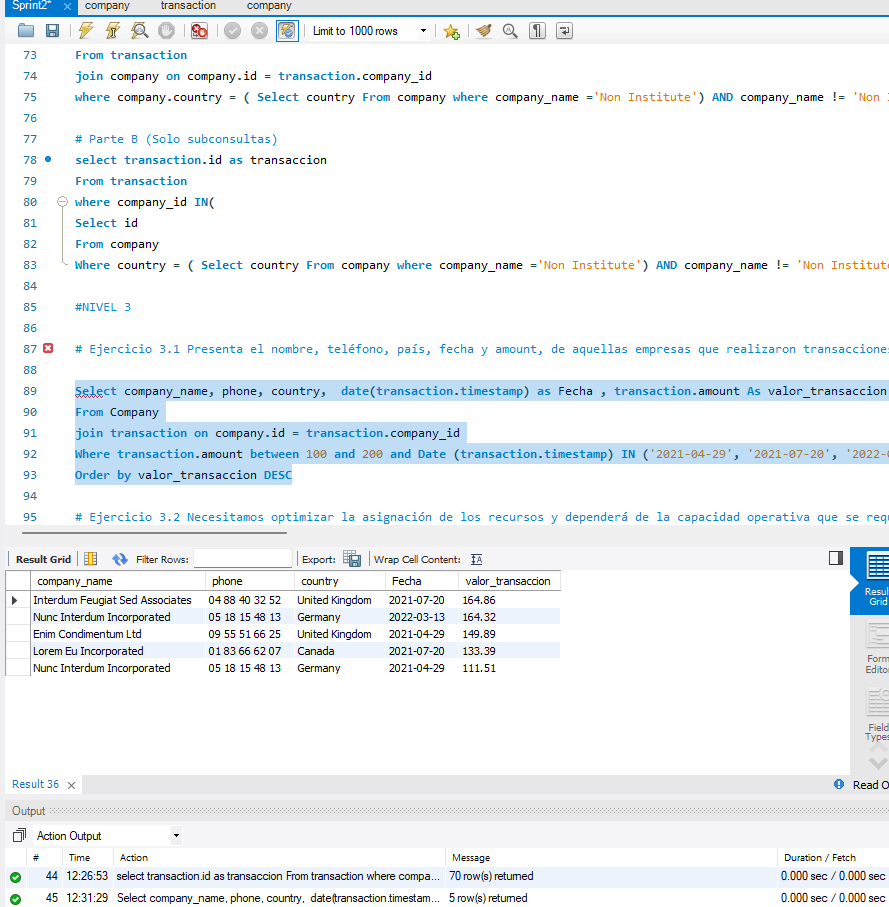
From Company

join transaction on company.id = transaction.company\_id

Where transaction.amount between 100 and 200 and Date (transaction.timestamp) IN ('2021-04-29', '2021-07-20', '2022-03-13')

Order by valor\_transaccion DESC

CAPTURA:



EXPLICACIÓN:

Esta consulta tiene como objetivo obtener datos específicos de transacciones y de las compañías asociadas, tal y como el nombre de la compañía, el teléfono, el país, con un valor y en unas fechas específicas (Para calcular las fechas se ha extraído la fecha de la tabla timestamp - , combinando las tablas "Company" y "transaction" para relacionar las transacciones con sus respectivas compañías (utilizando JOIN) .

Para obtener esta información se han aplicado filtro para seleccionar solo aquellas transacciones cuyo monto se encuentra entre 100 y 200 (BETWEEN) y cuya fecha coincide con una de las tres fechas especificadas: 29 de abril de 2021, 20 de julio de 2021 o 13 de marzo de 2022 (IN).

Finalmente, se ha ordenado los resultados de mayor a menor según la cuantía de cada transacción.

## # Ejercicio 3.2 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 4 transacciones o menos.

CONSULTA

Select company\_name as compañia,

CASE

WHEN COUNT(transaction.id) > 4

THEN 'Mas de 4'

ELSE '4 o menos'

END AS Cantidad\_transacciones

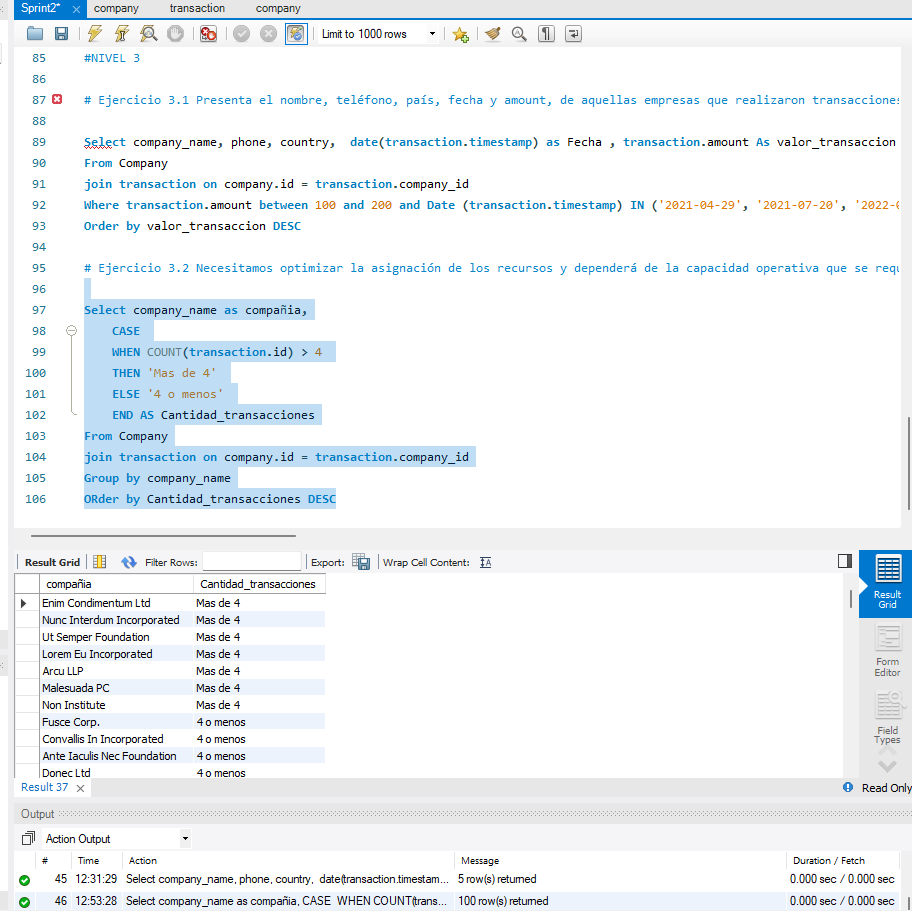
From Company

join transaction on company.id = transaction.company\_id

Group by company\_name

Order by Cantidad\_transacciones DESC

CAPTURA:



EXPLICACIÓN

Esta consulta clasifica las compañías según el número de transacciones que han realizado, combina las tablas "Company" y "transaction" con JOIN. Para contar y agrupar las transacciones por compañía, según se requiere por HR: se ha empleado una expresión CASE para crear una nueva categoría (Cantidad\_transacciones) : "Mas de 4" si tienen más de 4 transacciones, o "4 o menos" en caso contrario y GROUP BY. Finalmente se han ordenado los resultados según esta categoría en orden descendente. Para este caso, no se han diferenciado si las transacciones han sido declinadas o no, por lo que no se ha añadido WHERE declined = 0.