SPRINT 2

NIVEL1

EJERCICIO 1

IMPORTAR ESTRUCTURA

Se crea la BBDD “transactions” e Importamos las tablas “company” y “transaction”

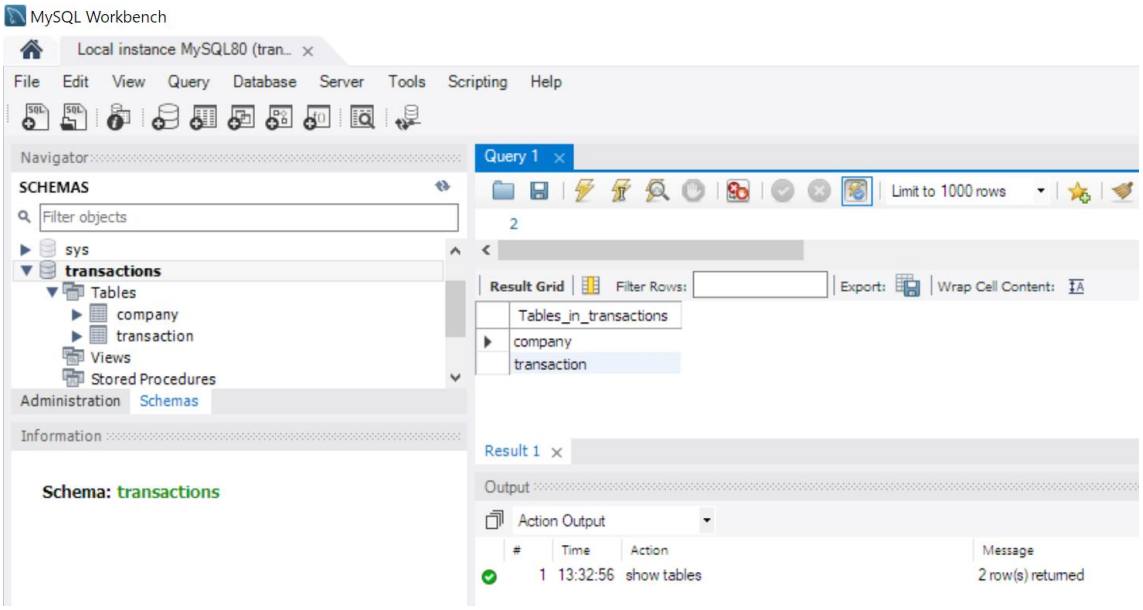
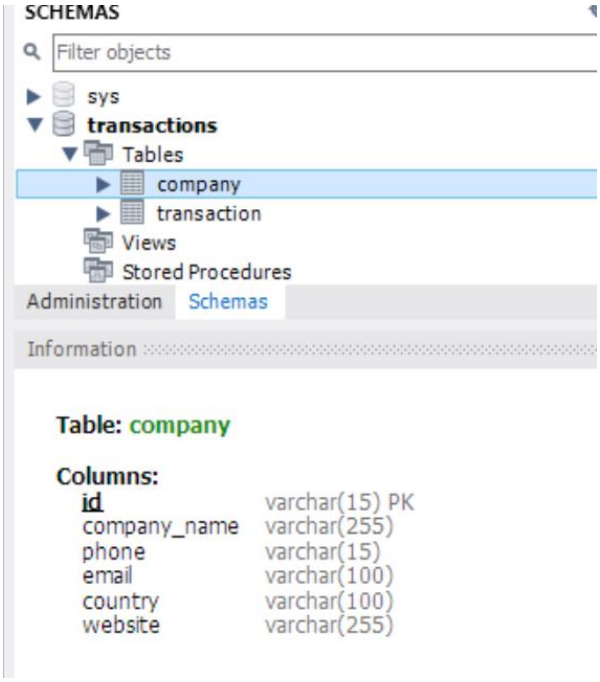


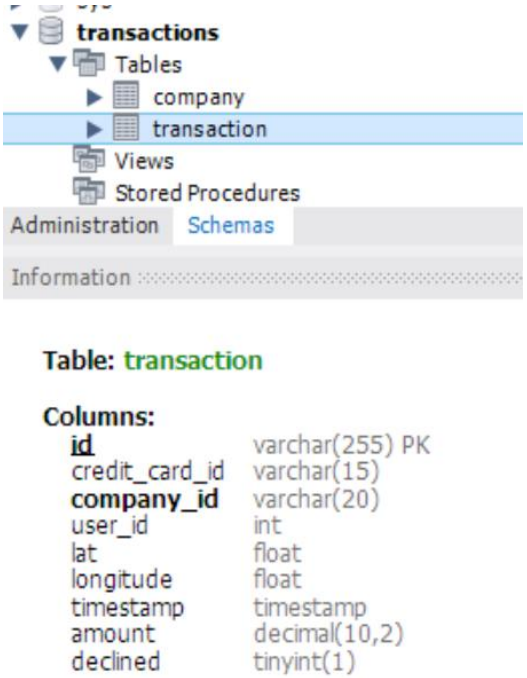
Tabla company:



La tabla company tiene los siguientes campos (columnas) de datos

- id : cadena de datos de longitud máxima 15 y valores únicos -> Primary key
- company_name: cadena de datos de longitud máxima de 255
- phone: cadena de datos de longitud máxima de 15
- email: cadena de datos de longitud máxima de 100
- country: cadena de datos de longitud máxima de 100
- website: cadena de datos de longitud máxima de 255

la tabla “transaction” :

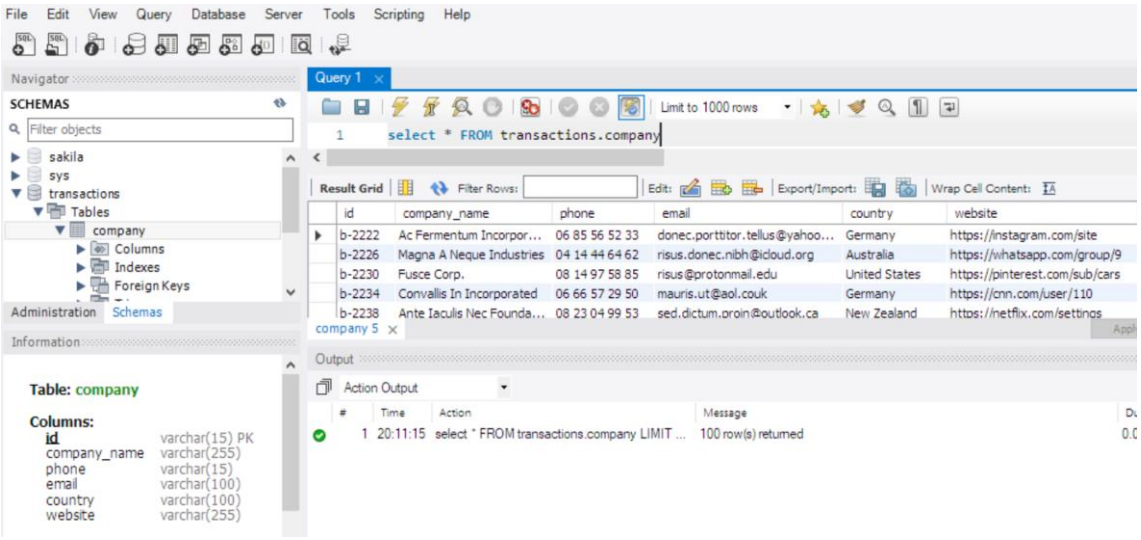


La tabla “transaction” tiene los siguientes campos (columnas) de datos:

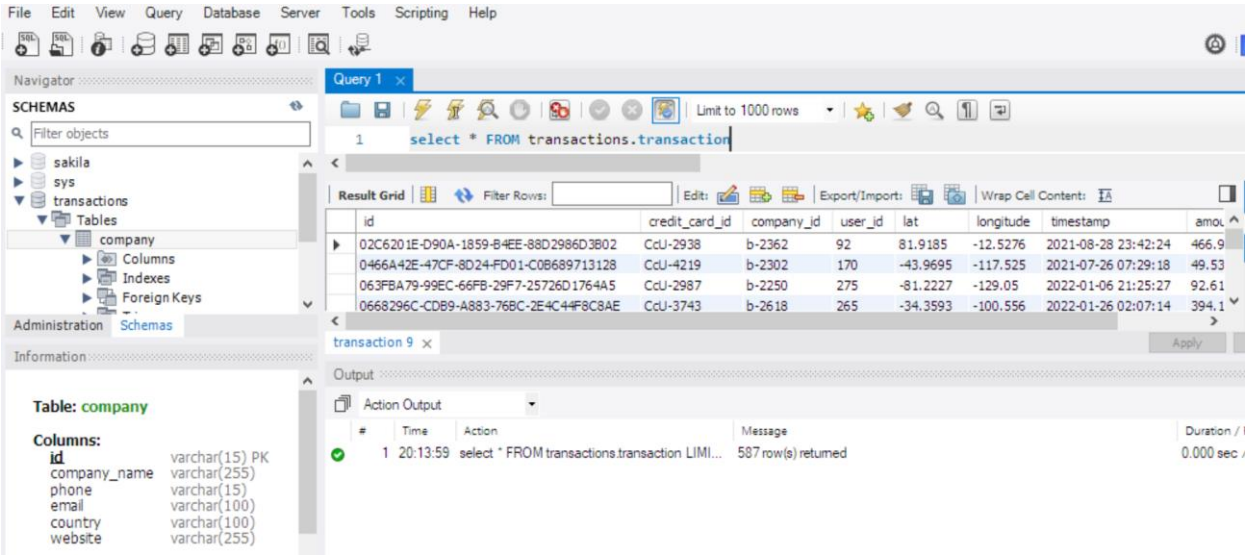
- id : cadena de datos de longitud máxima 255 y valores únicos -> Primary key
- credit_card_id: cadena de datos de longitud máxima de 15
- company_id varchar: cadena de datos de longitud máxima de 20
- user_id : Número entero
- lat : Número decimal
- longitude: Número decimal
- timestamp: Fecha
- amount: número decimal de 10 dígitos, dos decimales de precision
- declined: variable booleana

IMPORTAR DATOS

La tabla company tiene 100 filas

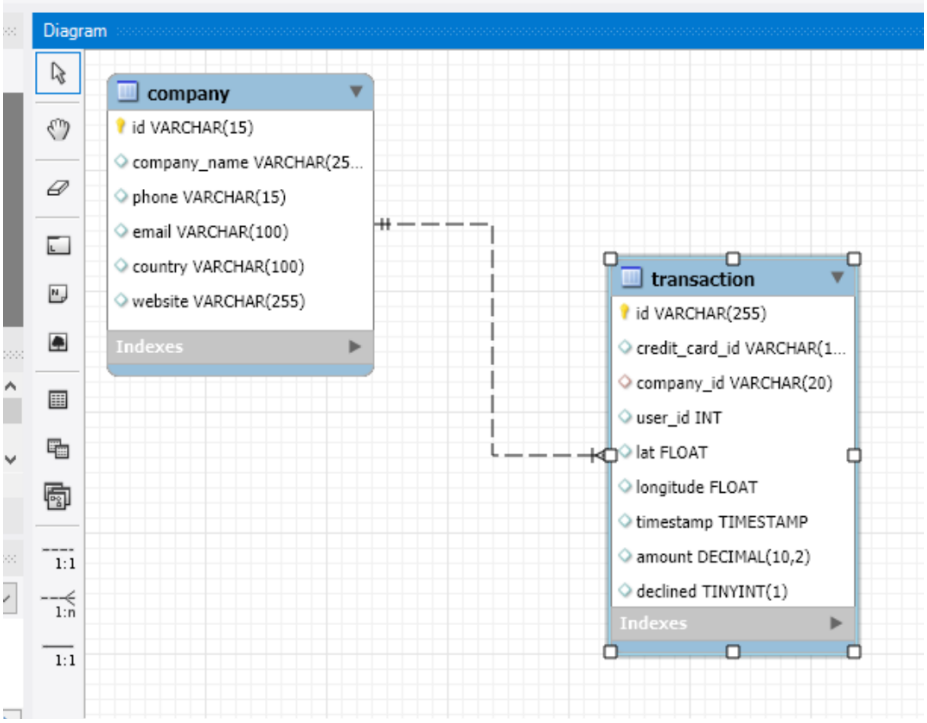


La tabla transaction tiene 587 filas



RELACIONES ENTRE TABLAS

El campo ‘company_name’ de la tabla ‘company’ se relaciona 1:N con el campo ‘id’ de la tabla ‘transaction’.



EJERCICIO 2

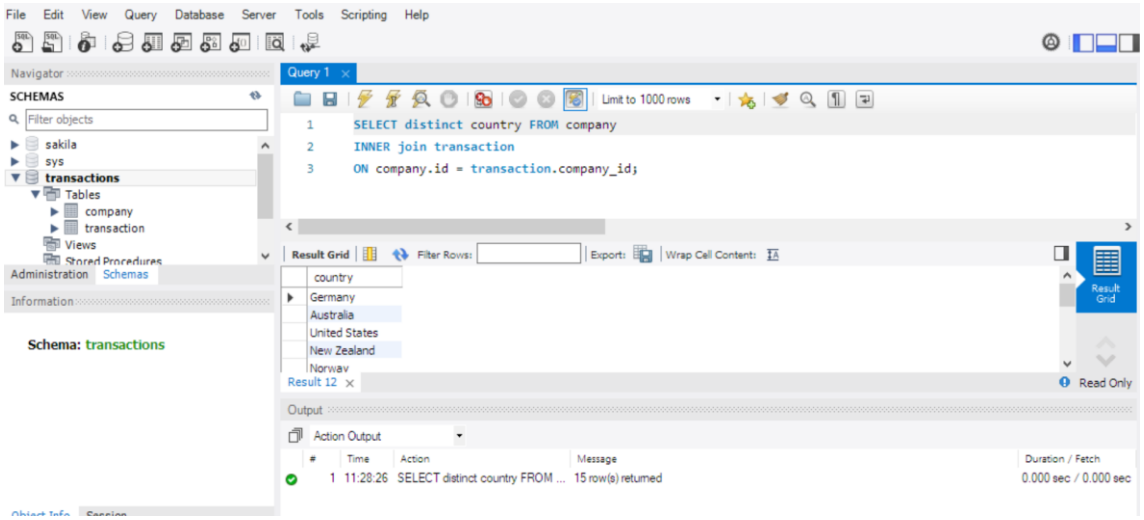
LISTADO DE PAISES QUE ESTÁN HACIENDO COMPRAS

Interpreto que nos piden todos los paises que han hecho transacciones (estén ‘declined’ o no)

La primary key es ‘id’ que identifica de modo único cada empresa con su país ‘country’ en la tabla ‘company’.

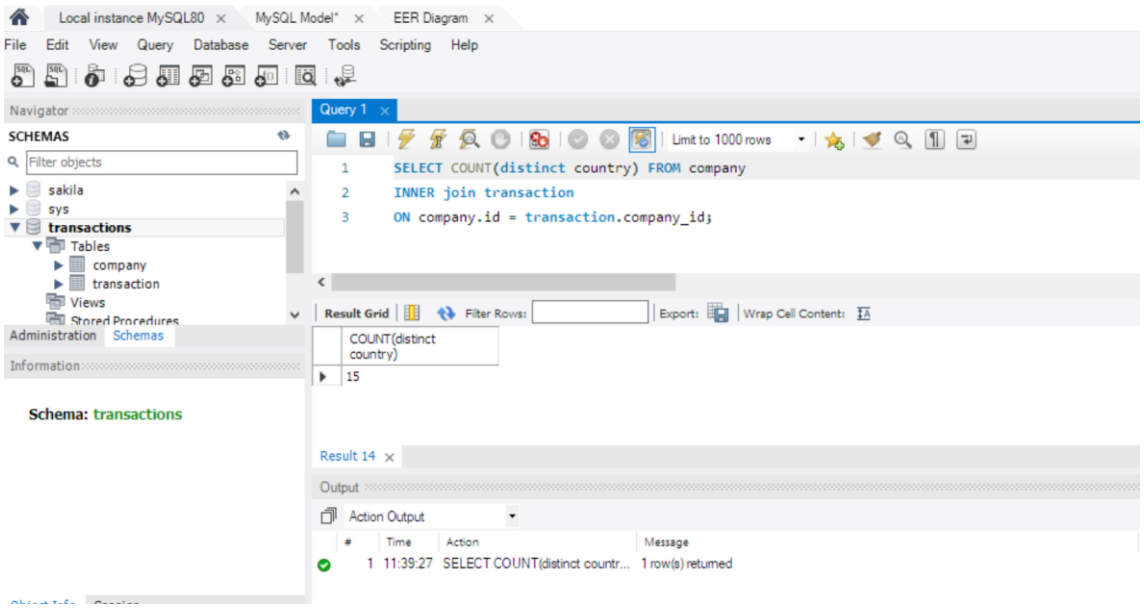
Como mantienen una relación 1:N tendremos que saber las compañias de la tabla ‘transaction’ que son comunes a la tabla ‘country’. Por tanto aplicamos un INNER JOIN siendo ‘transaction’ Table 1 y ‘company’ Table 2. En la selección de columnas a mostrar aplicamos un filtro ‘DISTINTC’ al campo ‘country’

Aun no sabiendo si todas las compañias de la tabla ‘transaction’ están incluidas en la tabla ‘company’ no serviría de nada utilizar un OUTER JOIN ya que solo sabriamos los paises de la tabla ‘company’ y ‘transaction’ comunes ya que campo ‘country’ solo aparece en ‘company’



DESDE CUANTOS PAISES SE REALIZAN LAS COMPRAS

Seria contabilizar el número de filas que tiene la query anterior.



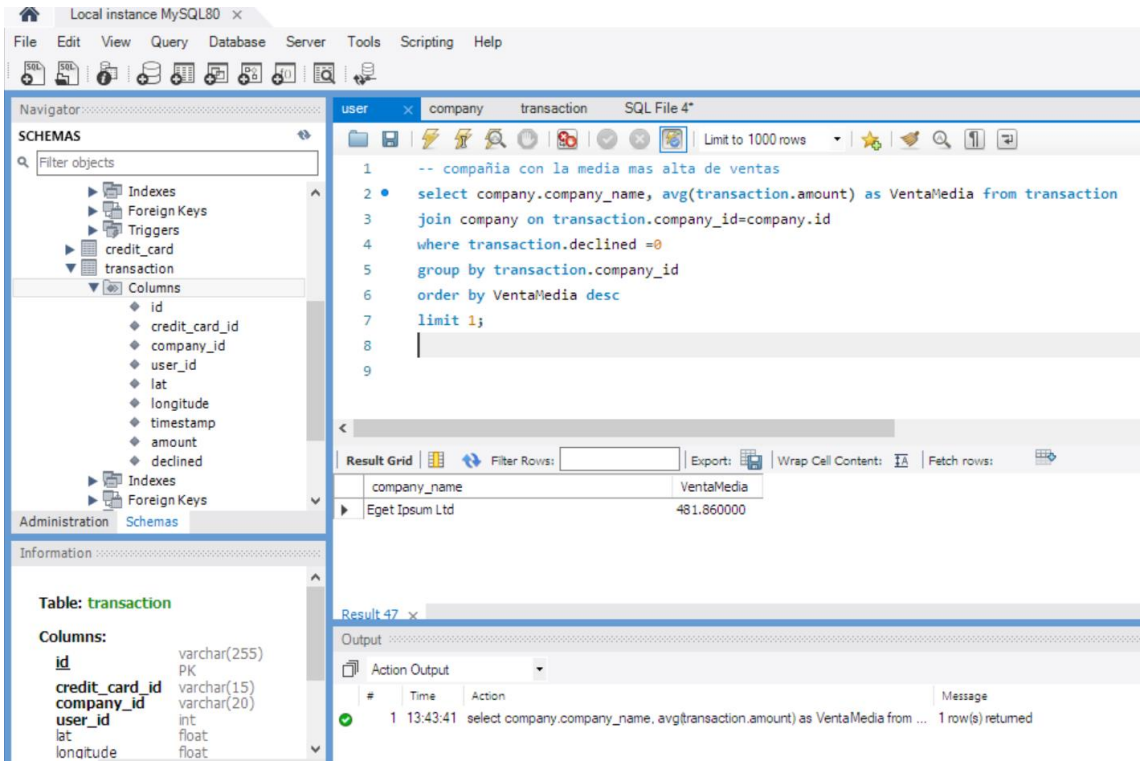
IDENTIFICAR LA COMPAÑÍA CON MEDIA MAS ALTA DE VENTAS

Considero que las ventas son las transacciones realizadas pero que no hayan sido rechazadas o ‘declined’.

Nos piden el nombre de la compañía que está en ‘company’ y la media de las ventas que es el campo de ‘amount’ pero filtrado con ‘declined’=0 y que están en ‘transacciones’

Por tanto, hacemos un join entre ambas transaction.company_id =company.id

Agrupamos por ‘company-id’ y ordenamos por la media de las ventas.

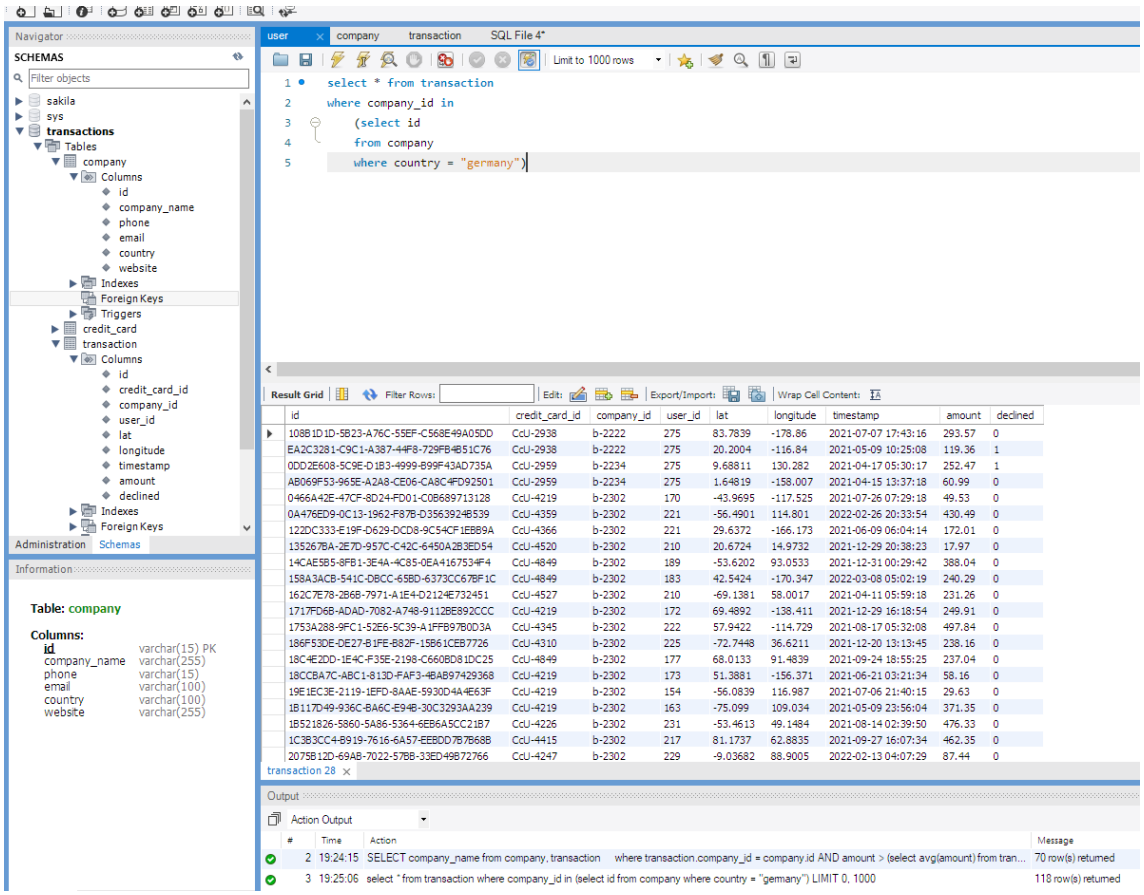


EJERCICIO 3

LISTADO DE TRANSACCIONES REALIZADAS POR ALEMANIA SIN UTILIZAR JOIN

Realizamos una ‘inner subquery’ para que nos identifique las Primary key (id) que mantienen la condición de que ‘country’ sea igual a “germany” .

Cómo ‘outer subquery’ mediante el operador “in” seleccionamos todos los campos en ‘transaction’ donde la Foreign key cumpla el condicionante anterior.



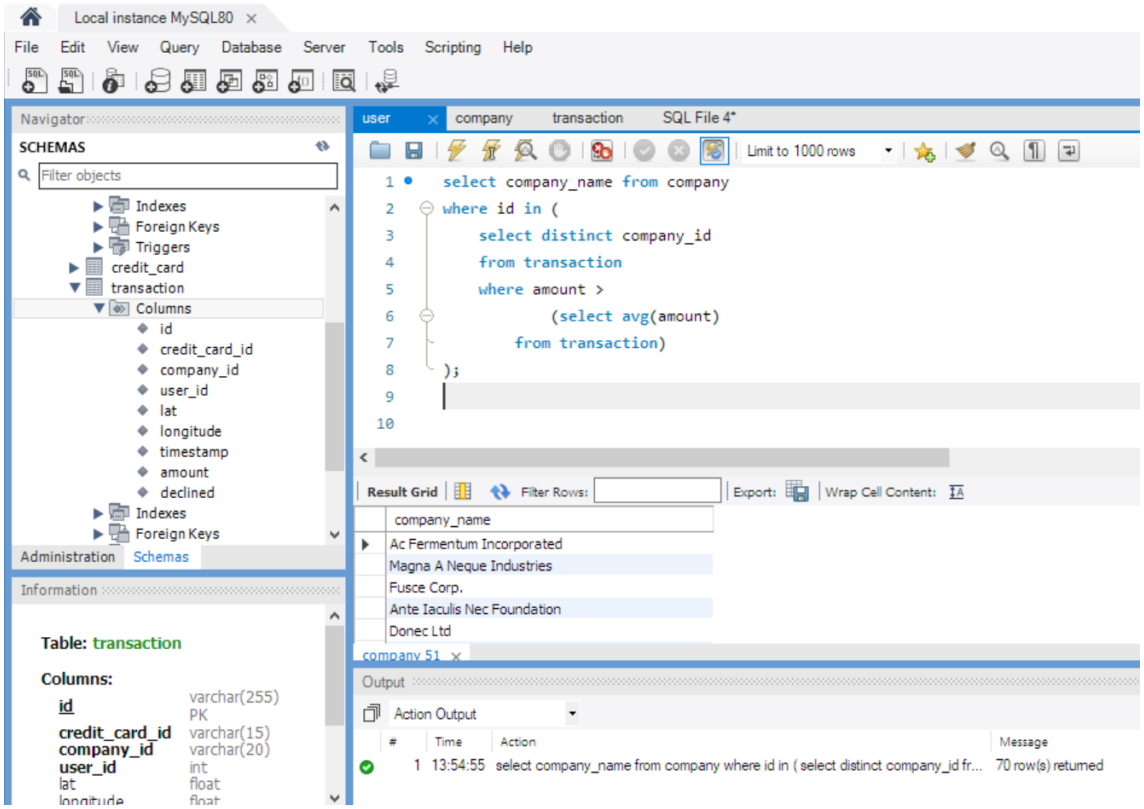
LISTA EMPRESAS CON TRANSACCIONES SUPERIORES A LA MEDIA DEL TOTAL SIN UTILIZAR JOIN

Creo la primera subquery para obtener la media de todas las transacciones

select avg(amount) from transaction)

La siguiente query sera seleccionar los distintos company_id donde el amount sea mayor que el valor obtenido antes

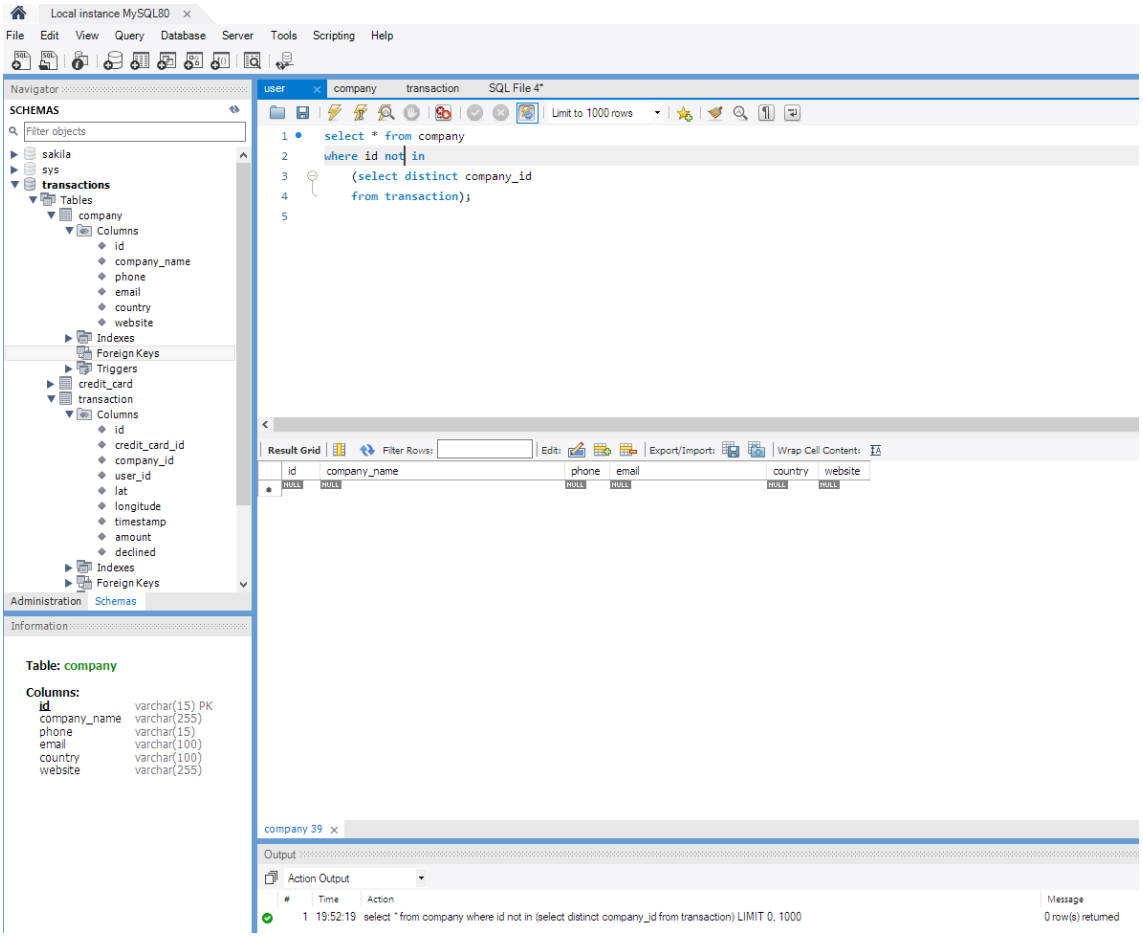
Estos company_id en transaction corresponden a los id en company mediante la condicion where ... in



EMPRESAS QUE NO TIENEN TRANSACCIONES SIN USAR JOIN

la primera subquery que realizo es identificar los valores únicos de ‘company_id’ en la tabla transaction.

La segunda subquery será ver en la tabla ‘company’ cuales de estos valores de ‘company_id’ no corresponden a los ‘id’ de tabla ‘company’



El resultado es negativo y no hay empresas que no tengan transacciones.

NIVEL 2

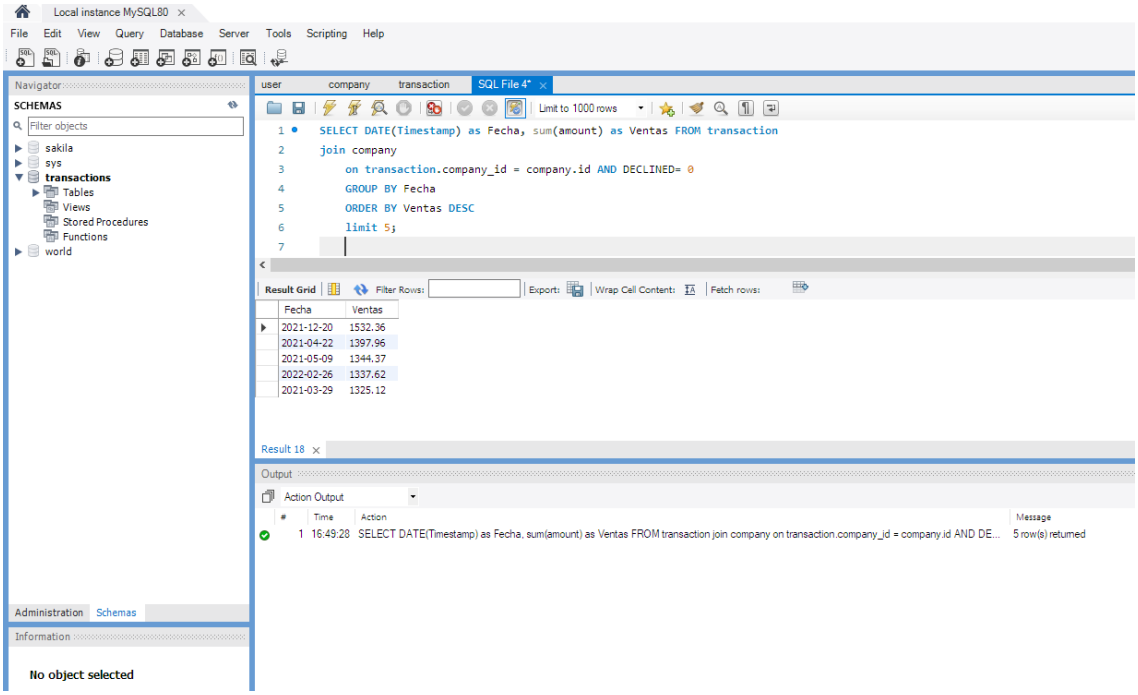
EJERCICIO 1

LOS CINCO DÍAS QUE MAYOR NÚMERO DE INGRESOS SE REGISTRARON

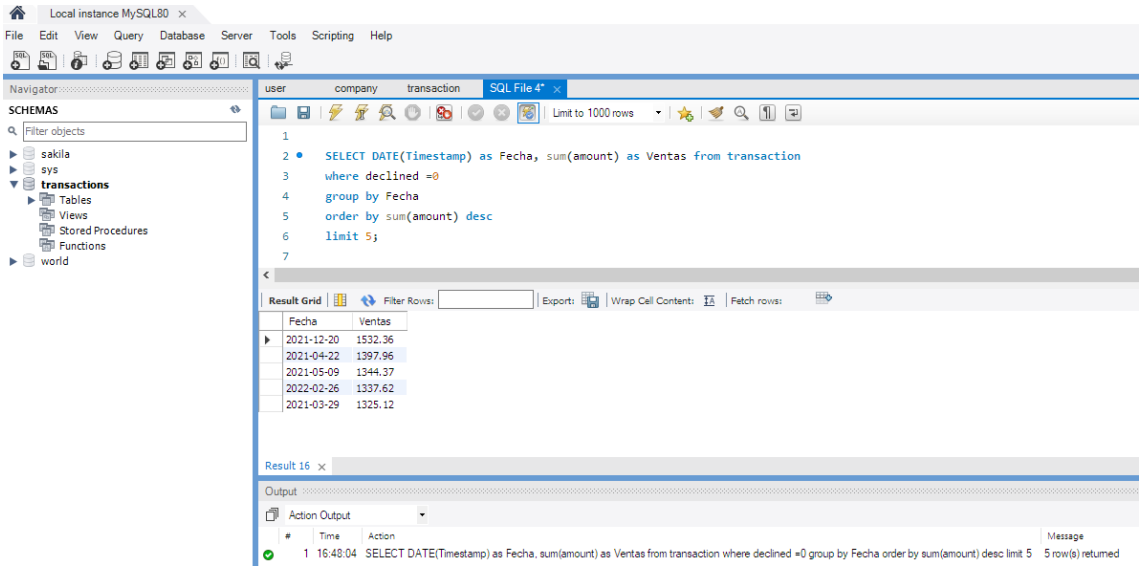
USANDO JOIN

Seleccionamos los campos ‘Timestamp’ le aplicamos el operador DATE y el operador suma con amount de la tabla ‘Transaction’

Aplicamos join con las key y añadimos la condicion DECLINED=0 para obtener las ventas y no incluir las transacciones que no se hayan realizado.Cojemos las 5 primeras filas.



SIN USAR JOIN

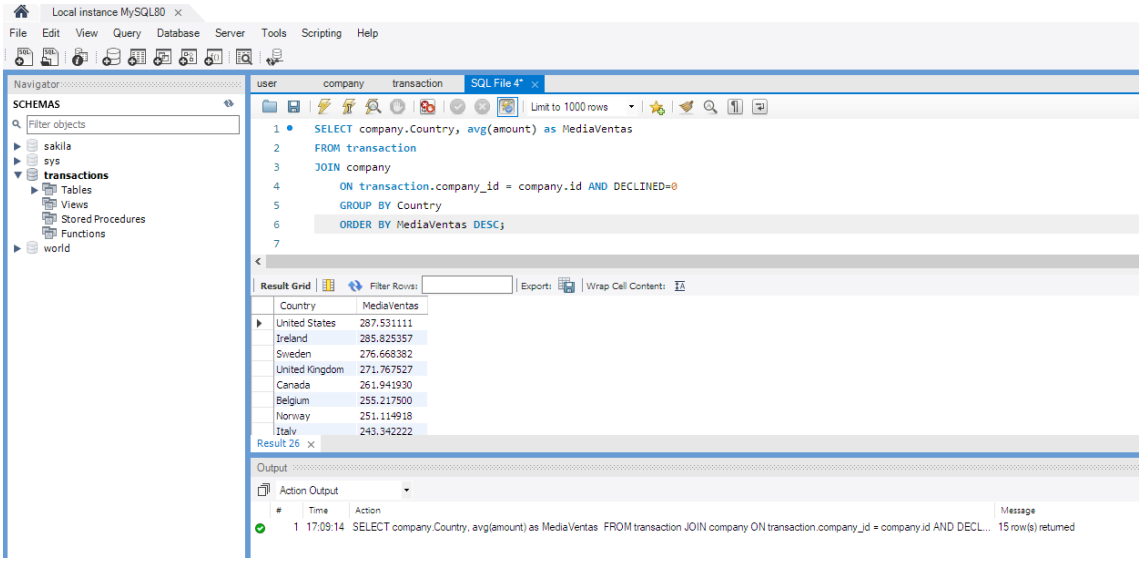


EJERCICIO 2

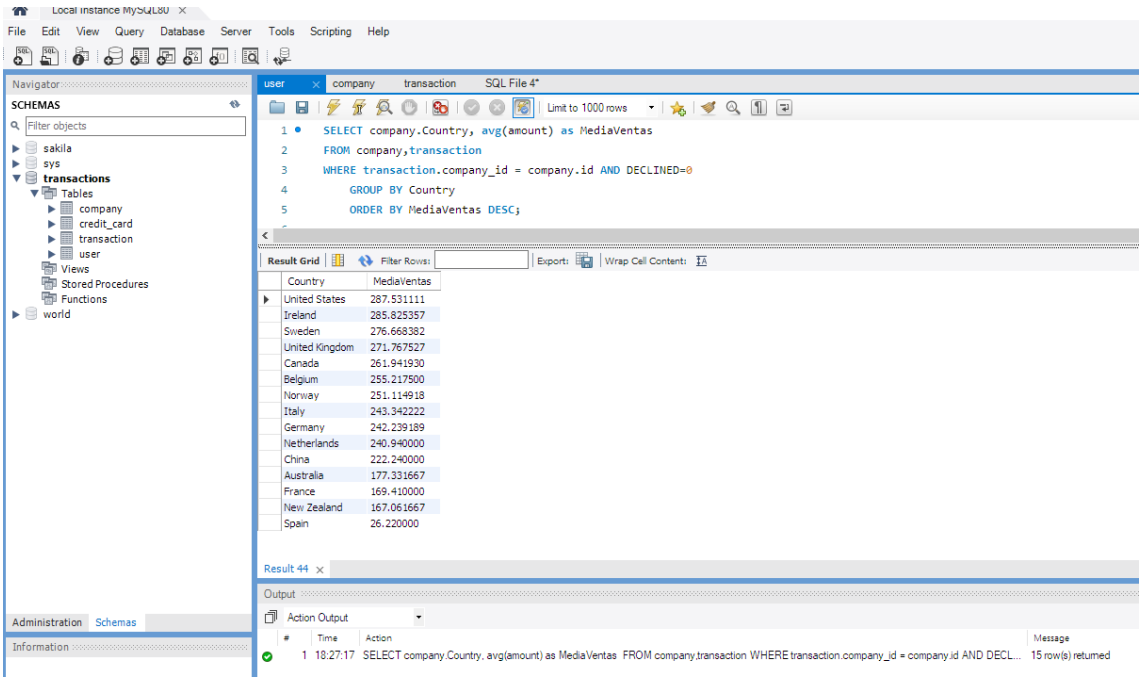
RANKING POR PAIS DEL PROMEDIO DE LAS VENTAS

USANDO JOIN

Aplicamos la condición utilizando las Primary Keys ya que tenemos que usar ambas tablas, agrupamos por pais y ordenamos por la media de las ventas



UTILIZANDO SUBQUERIES



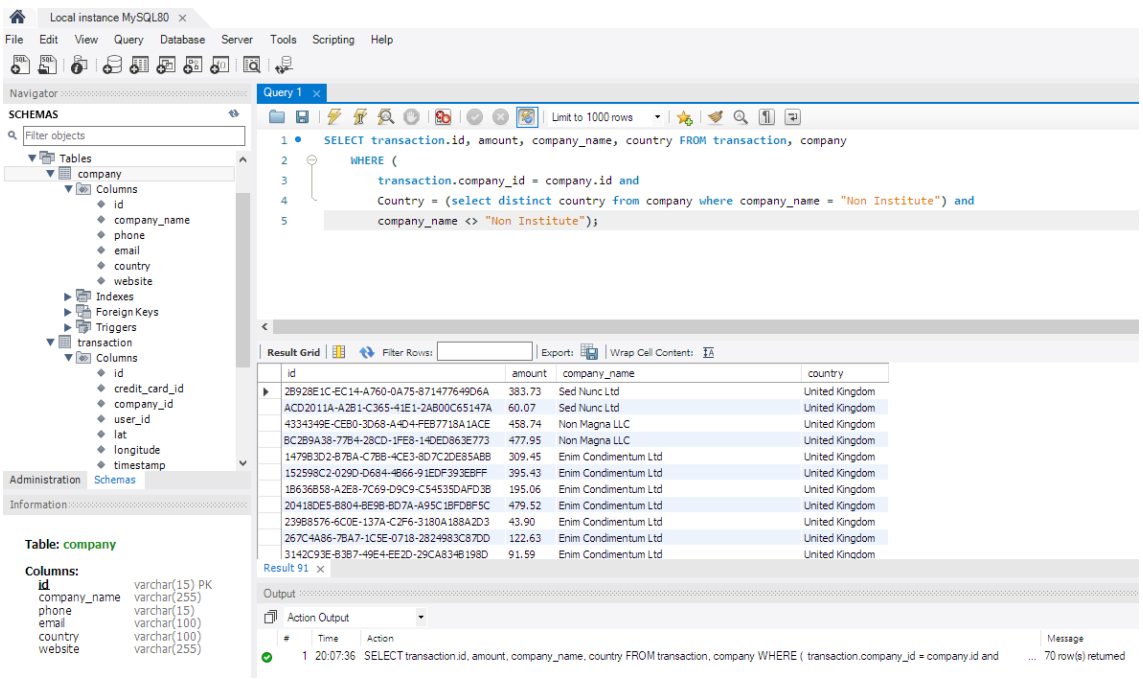
EJERCICIO 3

Listado de las transacciones realizadas en el mismo pais que 'Non Institute'

SIN JOINS

Mediante la condicion where añadimos las tres condiciones:

- 'company_id' = 'id'
- Country tiene que ser igual al 'country' cuando el nombre de la compañía es "Non Institute"
- El nombre de la compañía no es igual a "Non Institute"



UTILIZANDO QUERIES