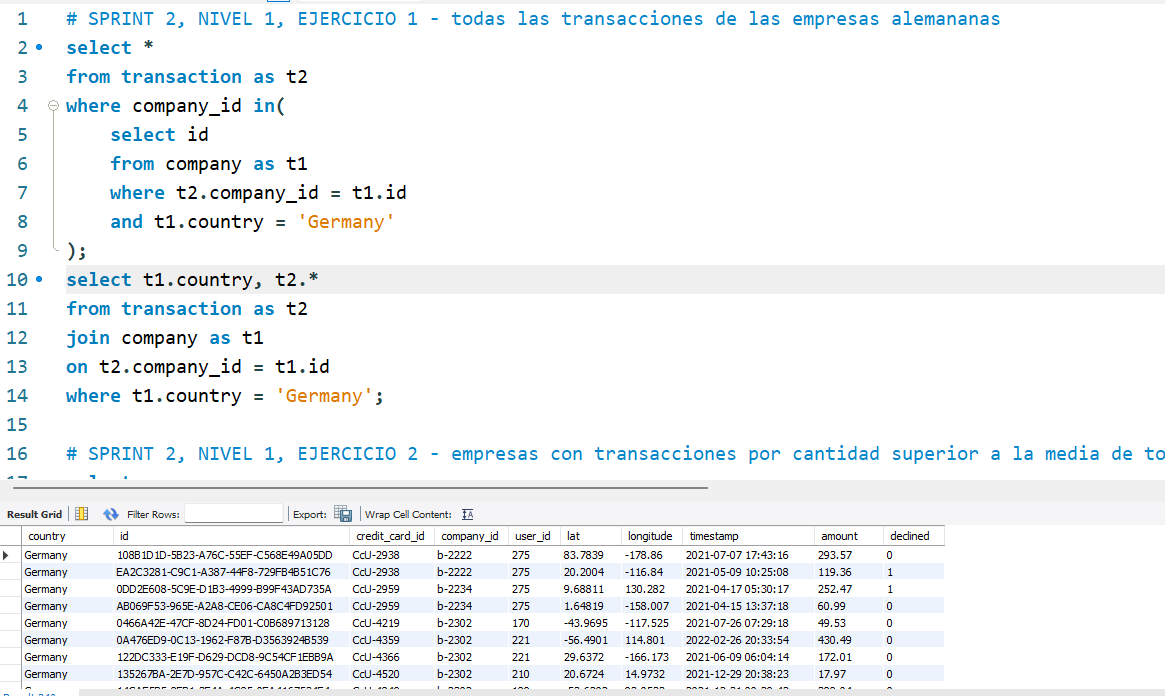
**SPRINT 2 – NIVEL 1**

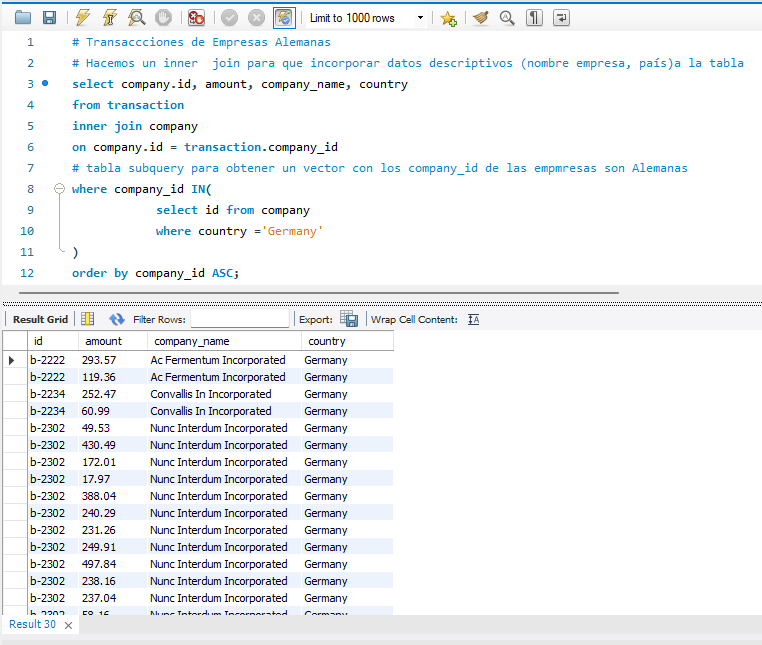
**Sprint 2, Nivel 1, Ejercicio 1**

**Empresas Alemanas que han hecho compras. Creamos un subquery para obtener el Vector con los ID de las empresas alemanas y que usaremos en la tabla de Transaciones para listarlas.**



**Veamos a continuación un método alternativo para resolver el problema (menos eficiente)**

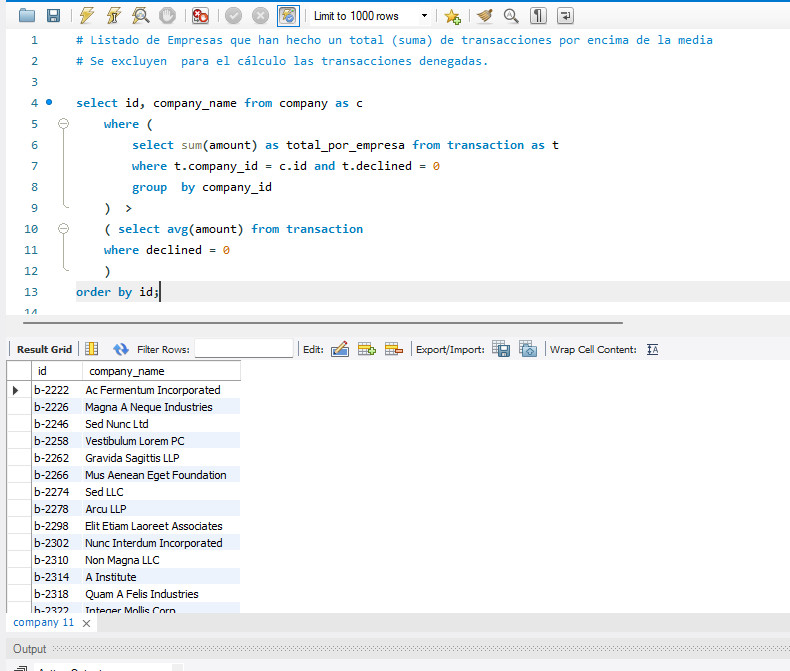
Vemos a continuación una alternativa con INNER JOIN para que la lectura de la tabla será más Amigable. Este código es más ineficiente



**Sprint 2, Nivel 1, Ejercicio 2**

**Listado con las Empresas que han hecho compras por encima de la media de todas las transacciones**

Creamos un subquery en el que se pregunta si la cantidad está por encima de la media, pero en el mismo acto, se vincula a la tabla company para extraer el nombre.



**Veamos a continuación un método alternativo para resolver el problema (menos eficiente)**

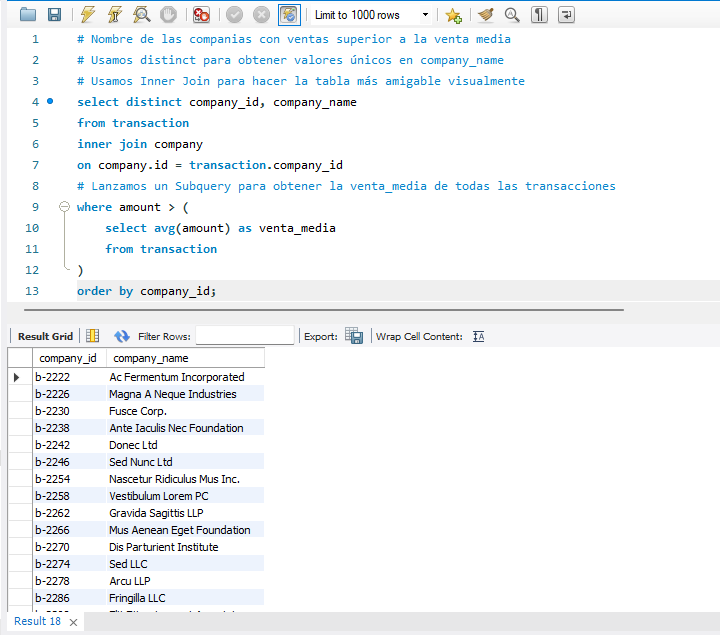
Creamos un Subquery para obtener la venta media de todas las transacciones.

Comparamos cada una de las transacciones con la cifra obtenida para ver si la operación es superior al valor medio calculado.

Con un Inner Join, llamamos al nombre de la empresa y confeccionamos el listado sin repetirlo.

Usamos Distinct para no repetir nombres de compañía.

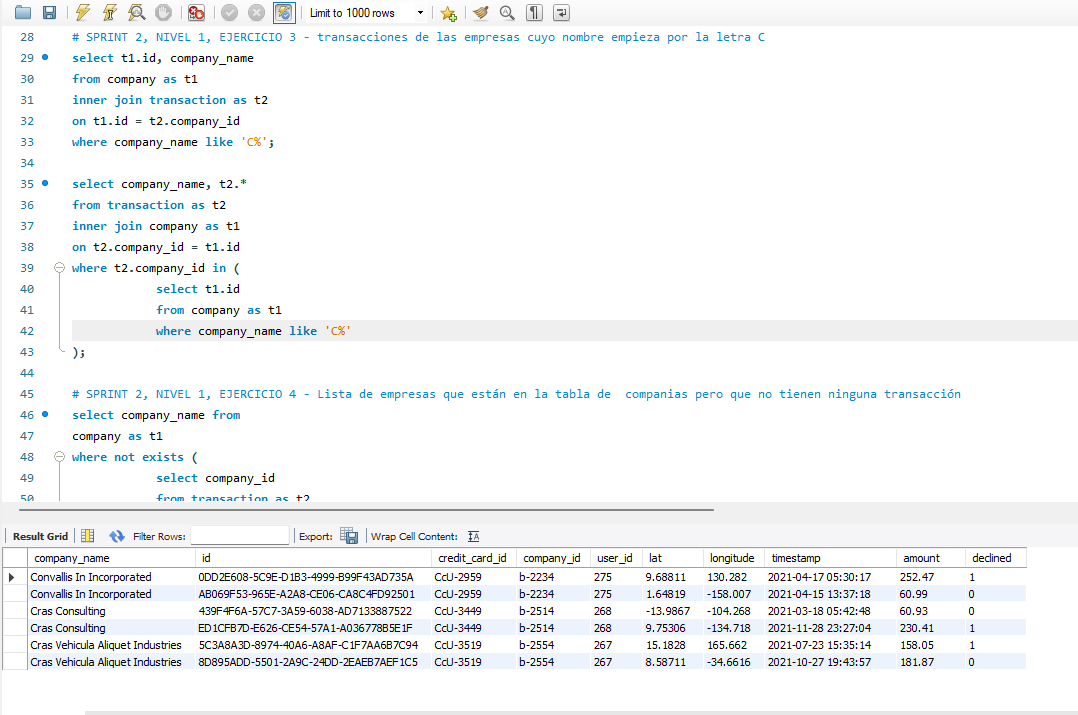
Este método es menos eficiente, a efectos prácticos, se ha comparado con la media todas las transacciones sin agregar por empresa, por lo tanto, queda anulado por no cumplir con el enunciado.



**Sprint 2, Nivel 1, Ejercicio 3**

**Recuperar transacciones de las empresas cuyo nombre empieza por c**

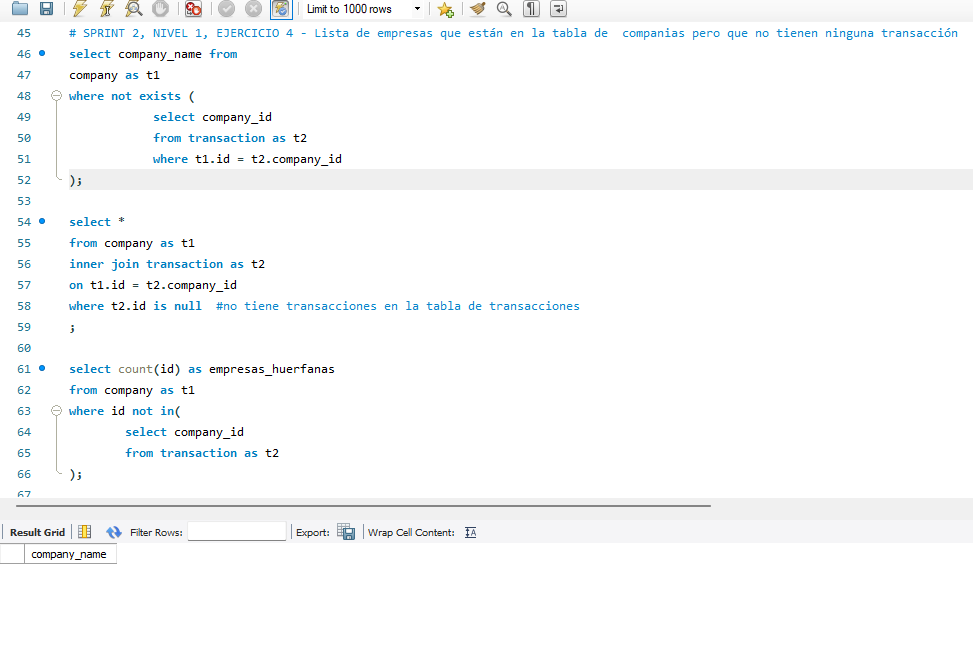
Aplicamos EXIST a la subtabla con dos condiciones, la primera que los id de las clave primaria y externa coincidan, y luego un LIKE ‘C%’ para empresas que empiecen por C.



**Sprint 2, Nivel 1, Ejercicio 4**

**Listado de empresas que aparecen en la tabla company pero no en la de transaction, es decir, empresas registradas que no han comprado.**

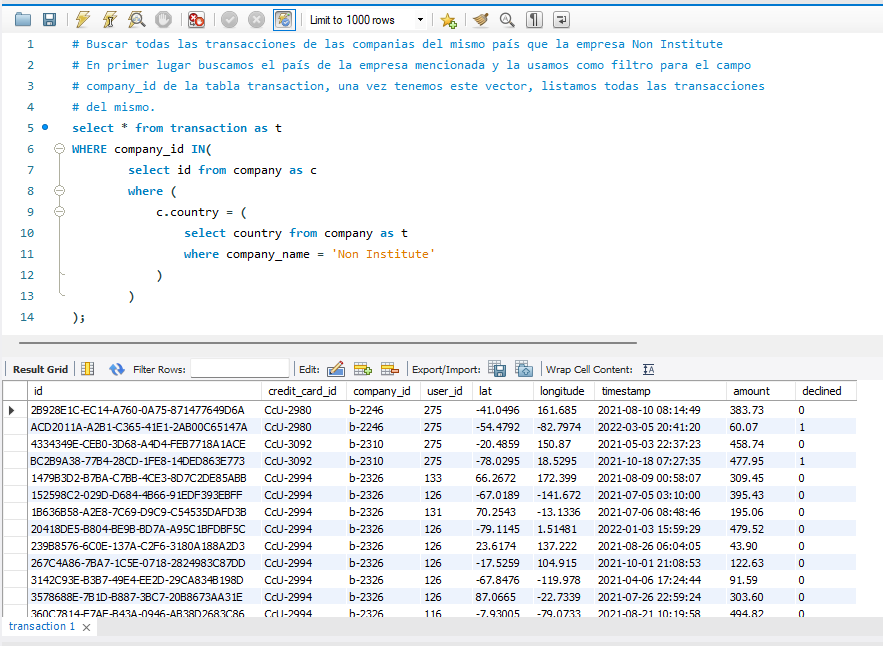
Aplicamos NOT EXIST en la tabla company para ver empresas que según su id no tiene correspondencia en la tabla transaction según su company\_id



**Sprint 2, Nivel 2, Ejercicio 1**

**Buscamos las transacciones de aquellas empresas que pertenecen al mismo país que la empresa Non Institute**

Tras obtener el country de esta empresa, el método es crear un select para obtener un vector con los id de las empresas que pertenecen al mismo país, para finamente iterar por el mismo con el método IN y poder así listar las transacciones.

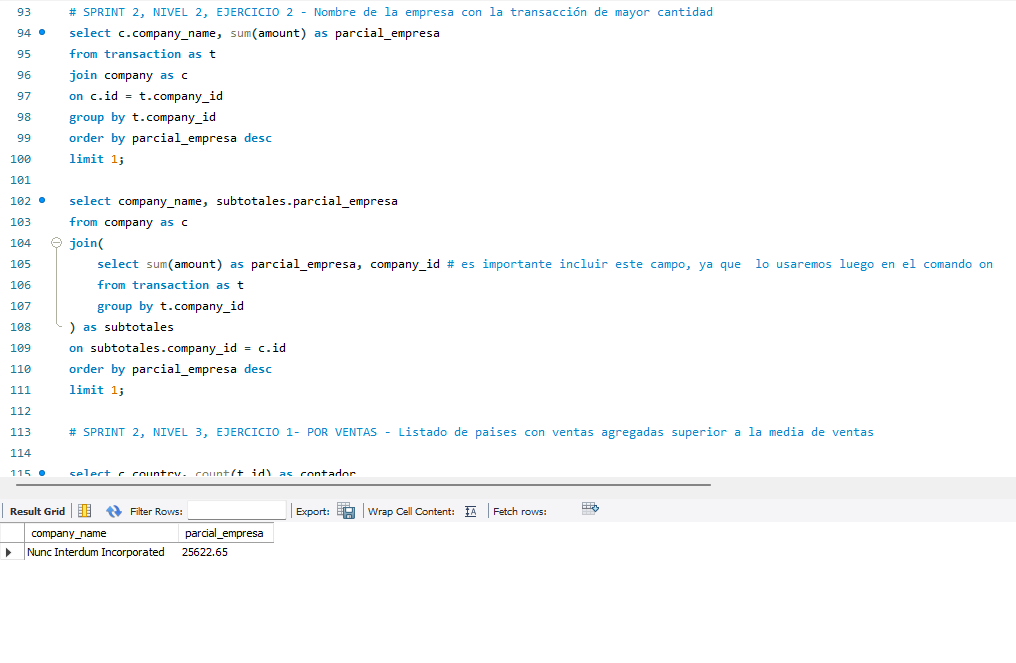


**Sprint 2, Nivel 2, Ejercicio 2**

**Buscamos la empresa que ha realizado la mayor transacción**

ordenamos por orden descendiente la tabla transación según las transacciones en el campo amount para tomar luego la primera fil.

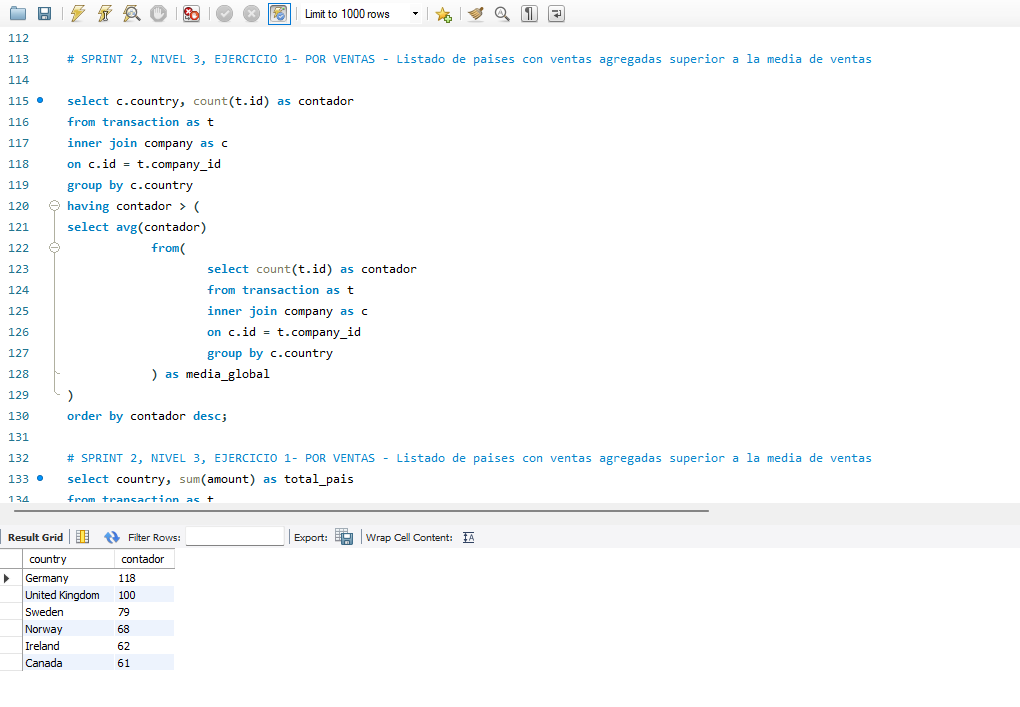
Finalmente vemos a que compañia corresponde el id de esta transacción.



**Sprint 2, Nivel 3, Ejercicio 1**

**Paises cuya ventas en cantidad de operacionessea superior a la venta media de todos los países**

Primero usamos el comando SUM para agrupar la cantidad amount por país para vectorizarlos en total\_pais . Luego usamos este vector para obtener la media, luego usamos el comando cross join para comparar las dos tablas, y finamente añadimos un Where para comprar total\_pais con media\_total\_paises.



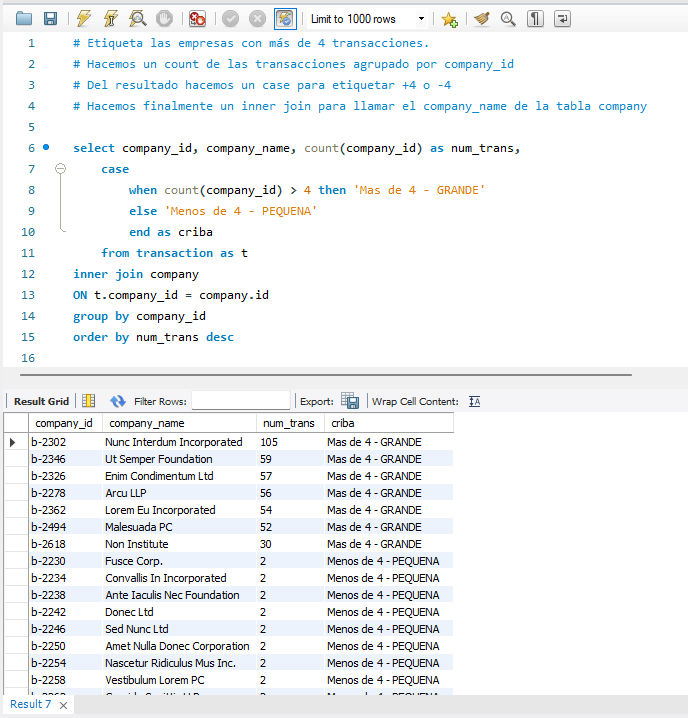
**Sprint 2, Nivel 3, Ejercicio 2**

**Necesitamos etiquetar las empresas que tienen más o menos de 4 transacciones**

Hacemos un COUNT del fichero transacciones agrupado por company\_id

Usamos un case para etiquetar +4 o -4 el resultado.

Finalmente hacemos un inner join para llamar al compo company\_name e la tabla company



**Sprint 2, Nivel 3, Ejercicio 2 – Alternativa**

Mismo ejercicio anterior pero sin etiquetar, el código devuelve el nombre de las empresas que han hecho más de 4 transacciones.

