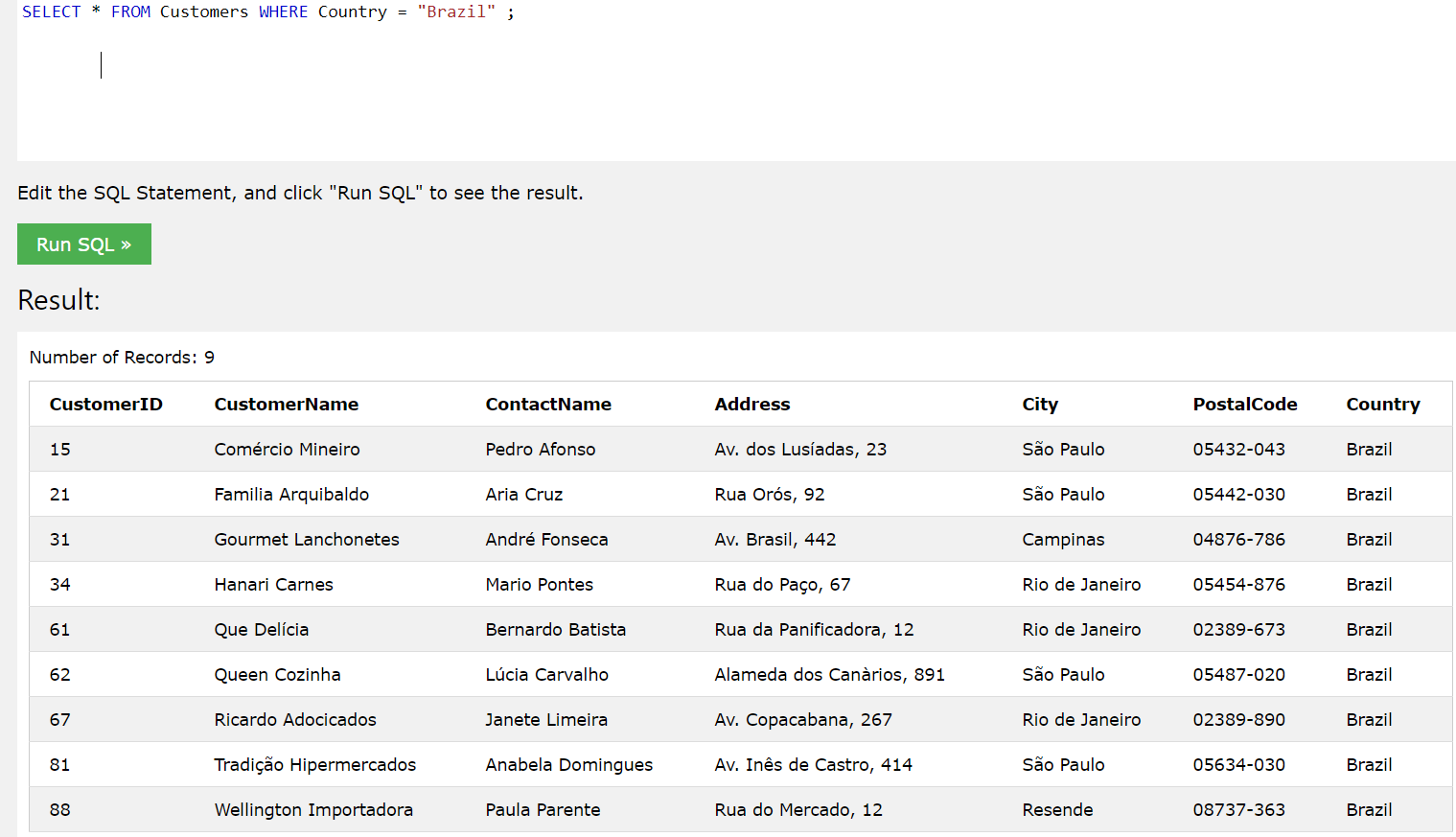
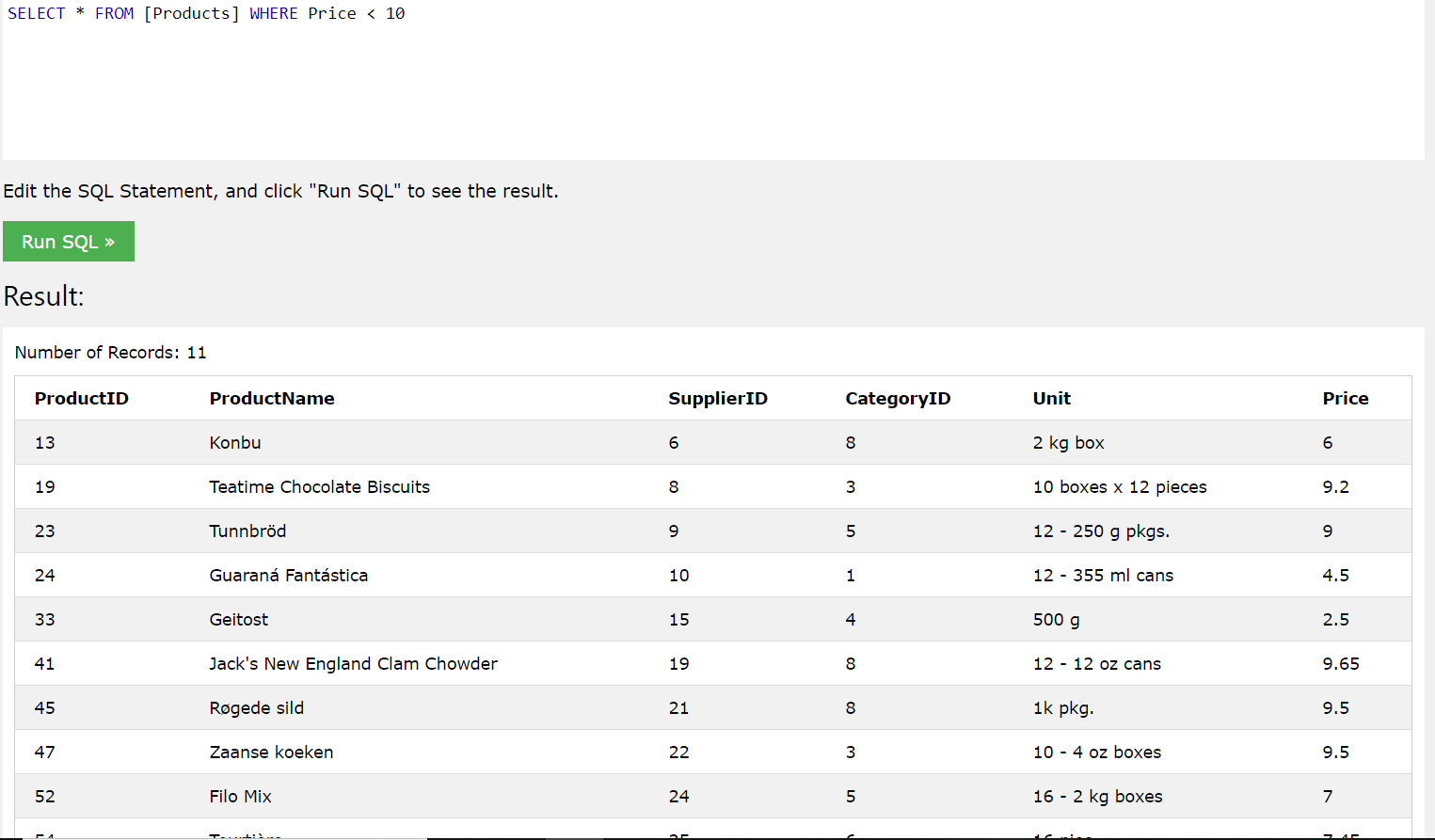
* Basic Queries

Note: Some of the queries that had 10 results and more did not fit in the screenshot. So, I took the most number of results (9 results) beside the command that was used.

1. Find the names of all the customers from Brazil



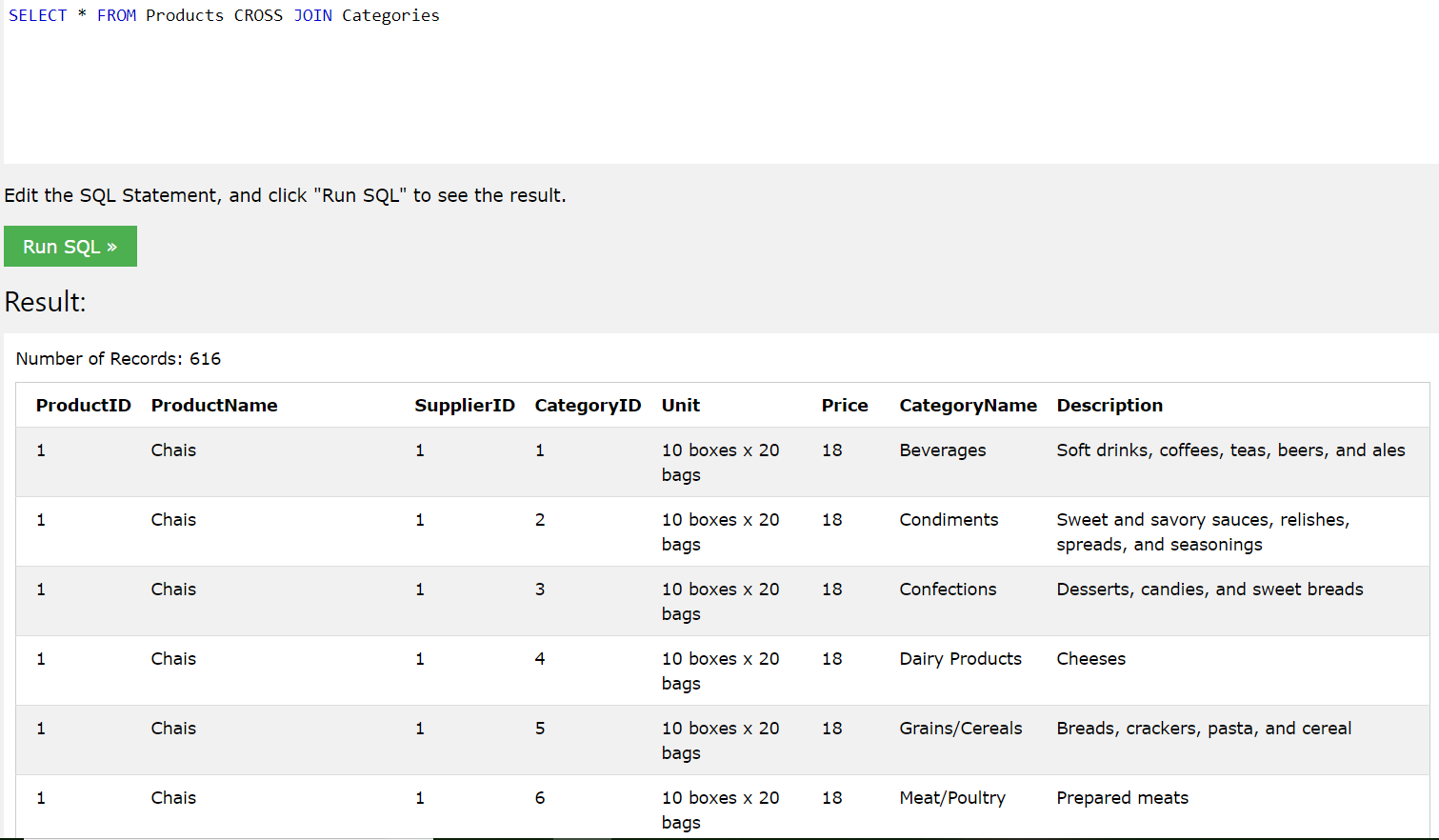
2. Find the names of products whose price is under 10



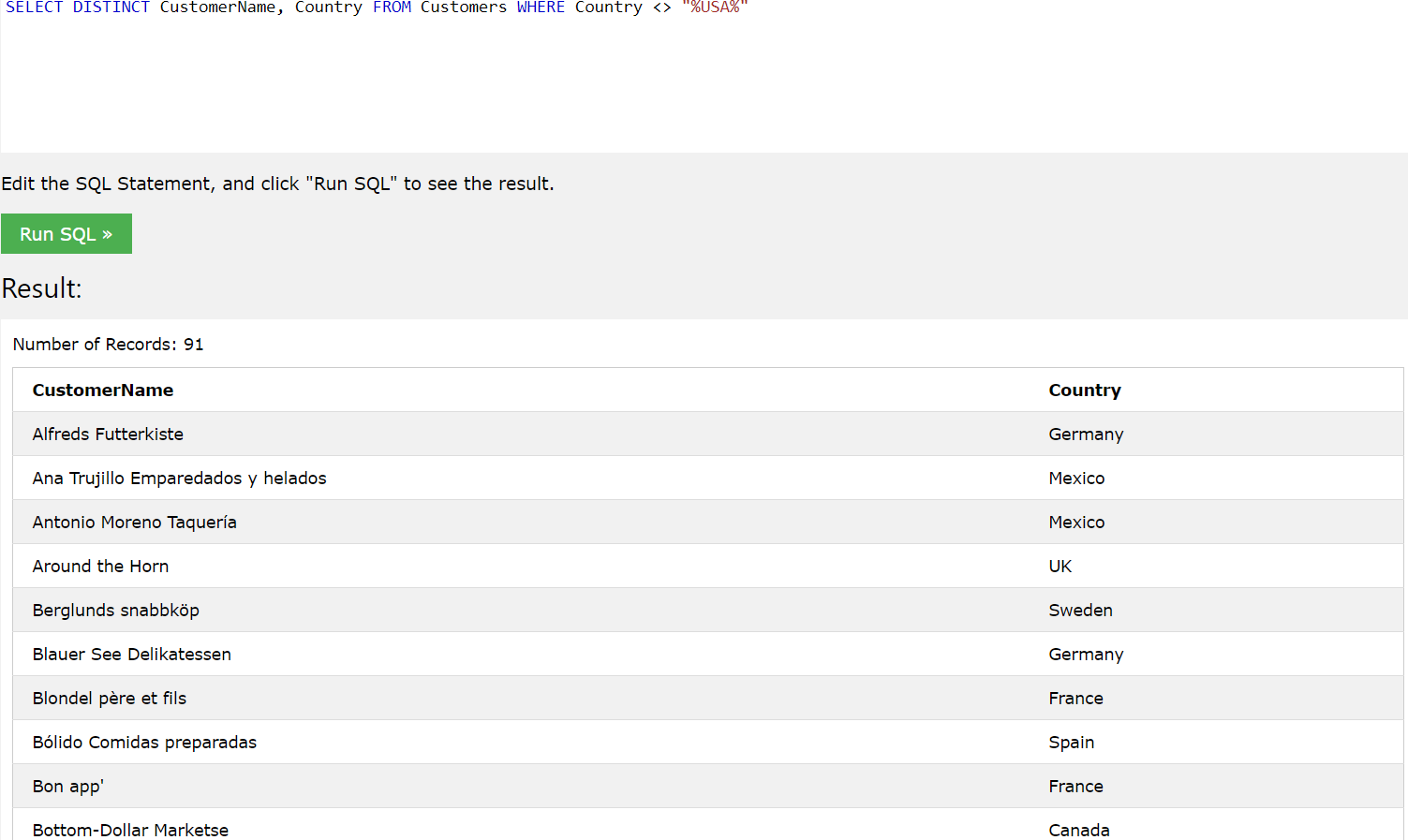
3. For the customer with ID 71, show all OrderIDs and order dates of all orders by the customer



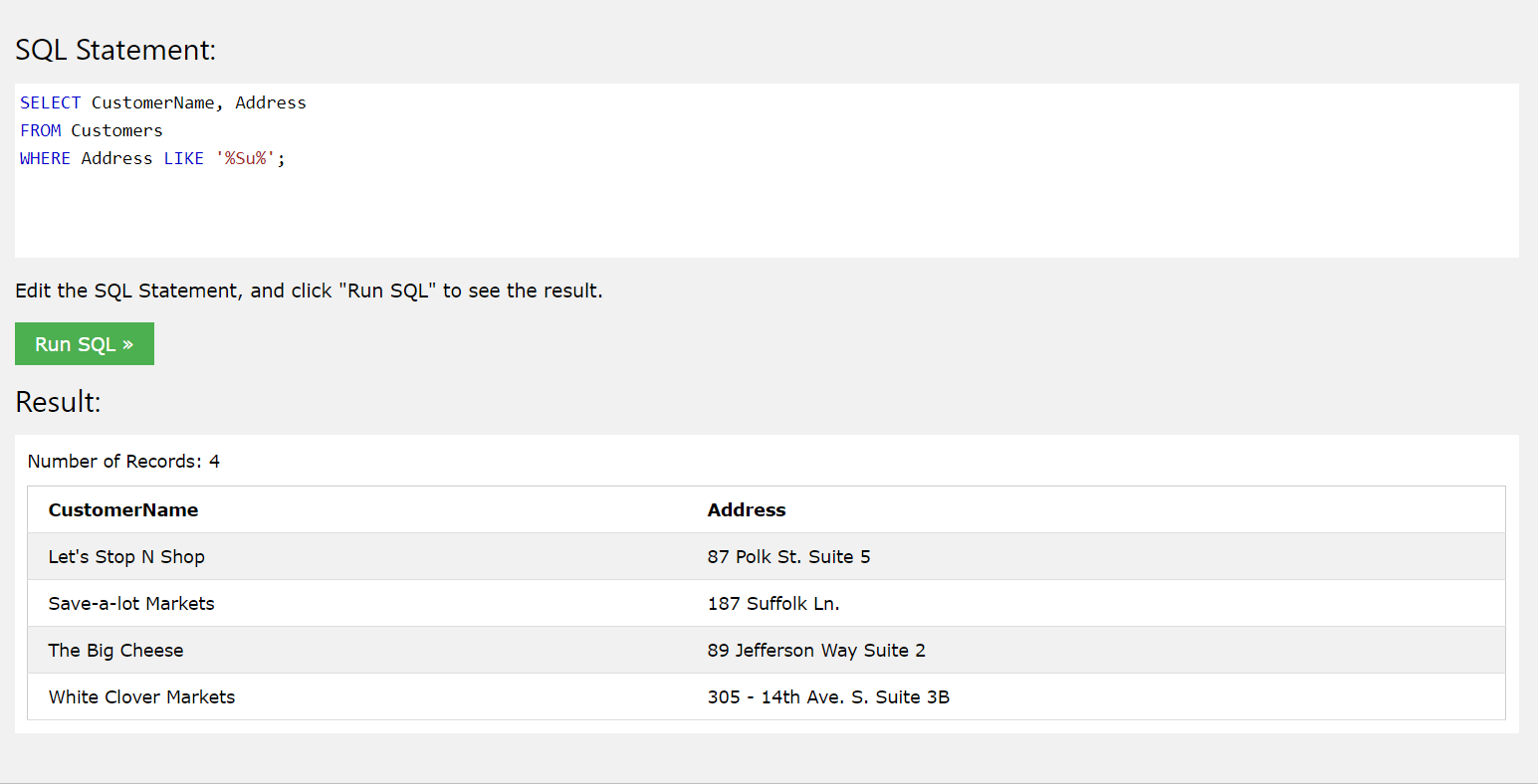
4. Show tables that cross join tables Products and Categories.



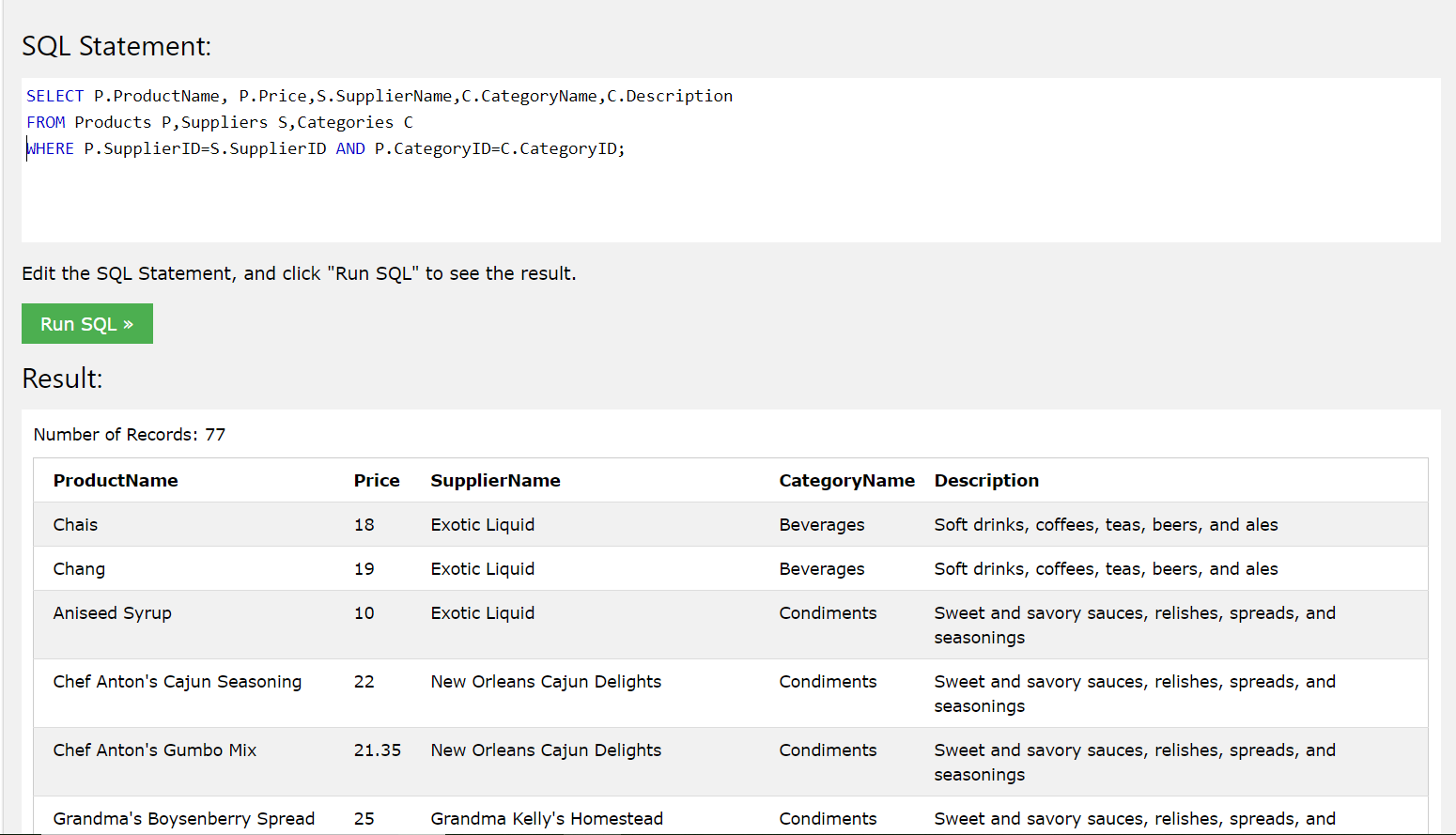
5. Find the names and countries of all customers who lives outside of the USA (there should be no duplicate names).



6. Find all the customers whose address contains “Su”:

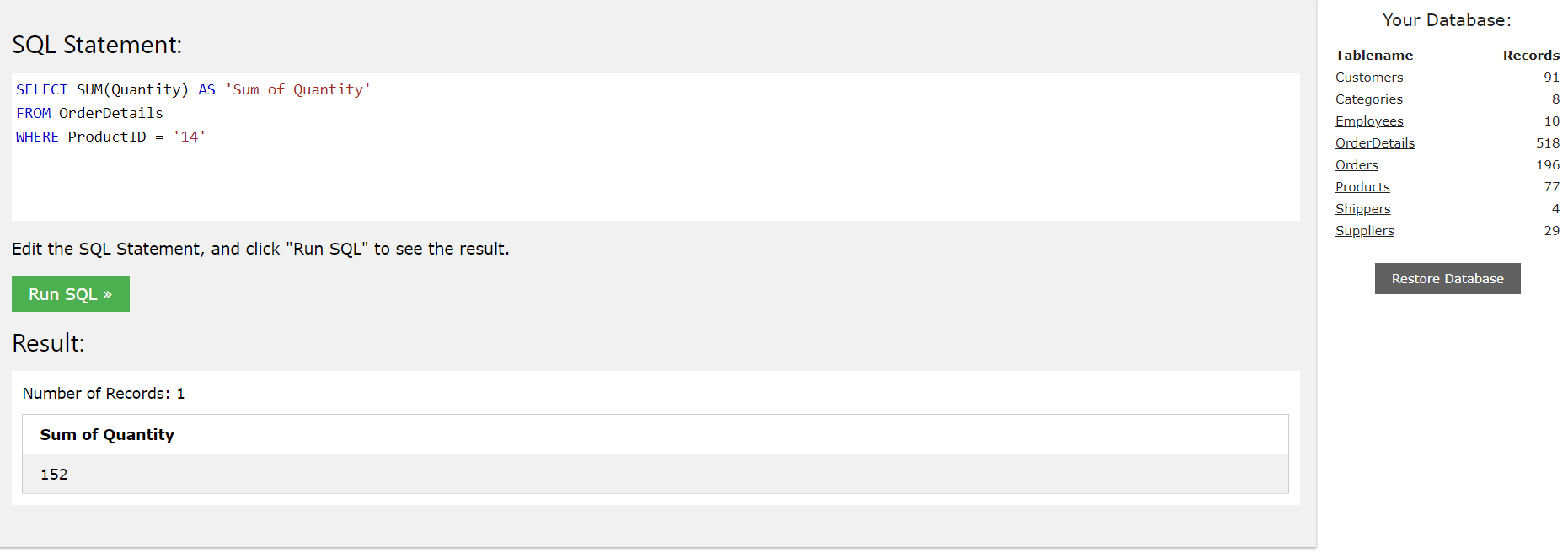


7. Make a single SQL statement that shows Product Name, Price, Supplier Name, Category Name and Description (hint need to join three tables)



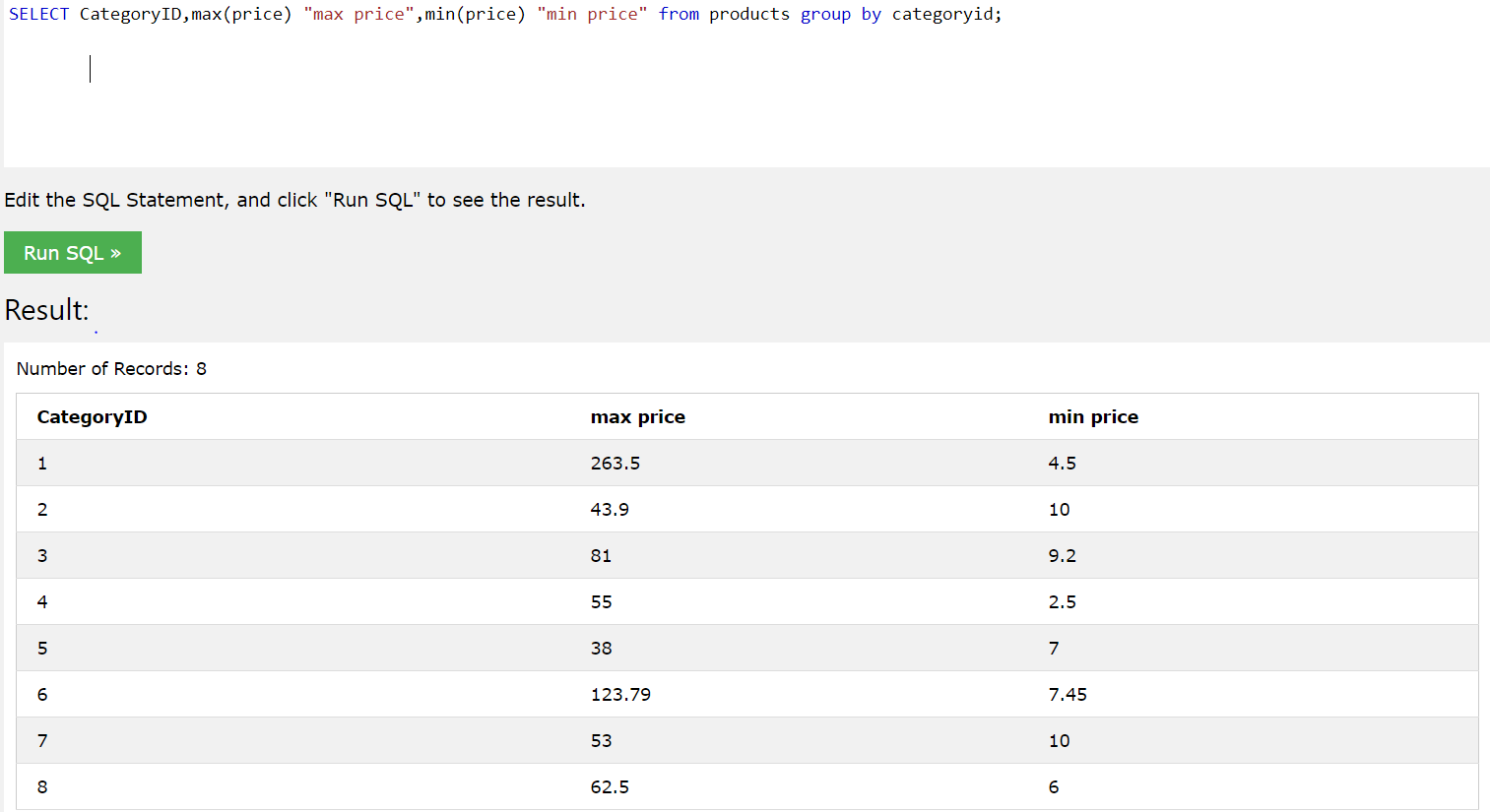
* aggregation and Union

1. For Product 14, find the total quantity ordered for such product. Display the title of this number as “Sum of Quantity”. You should use SQL aggregation on OrderDetails.

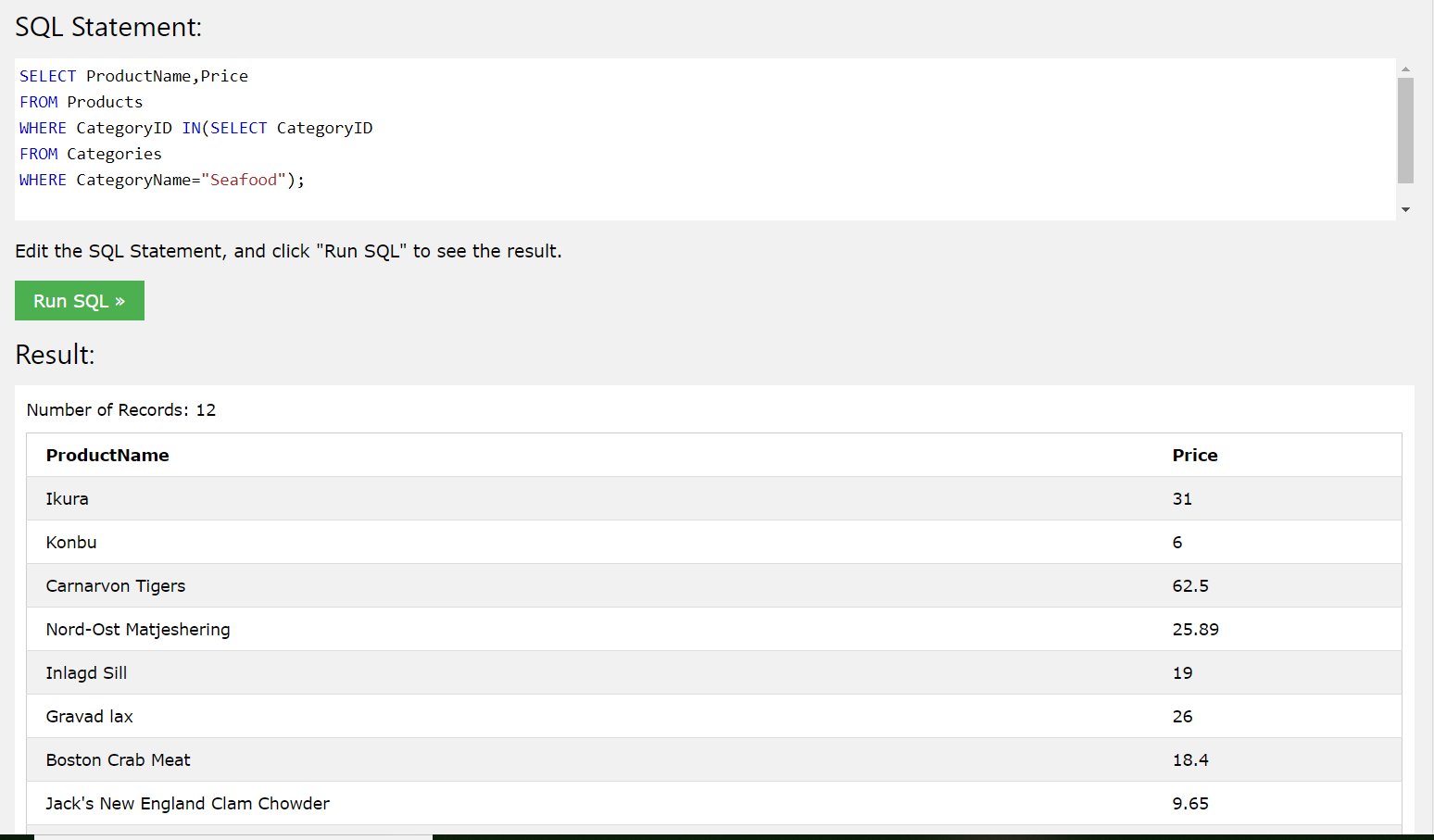


2. Find the maximum and minimum price across all Categories in Products shown as below (Note this is

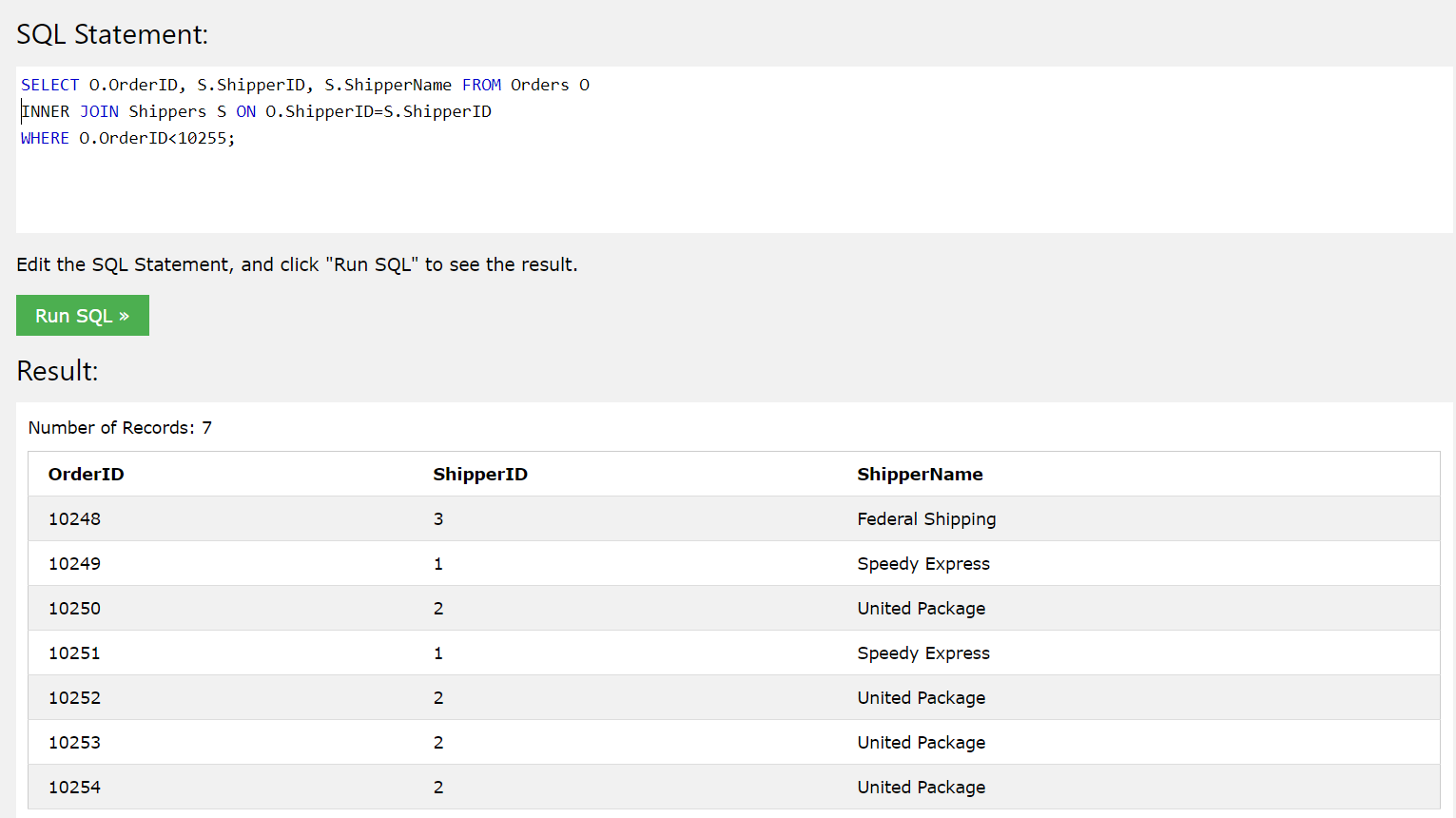
partial of the resulting table):



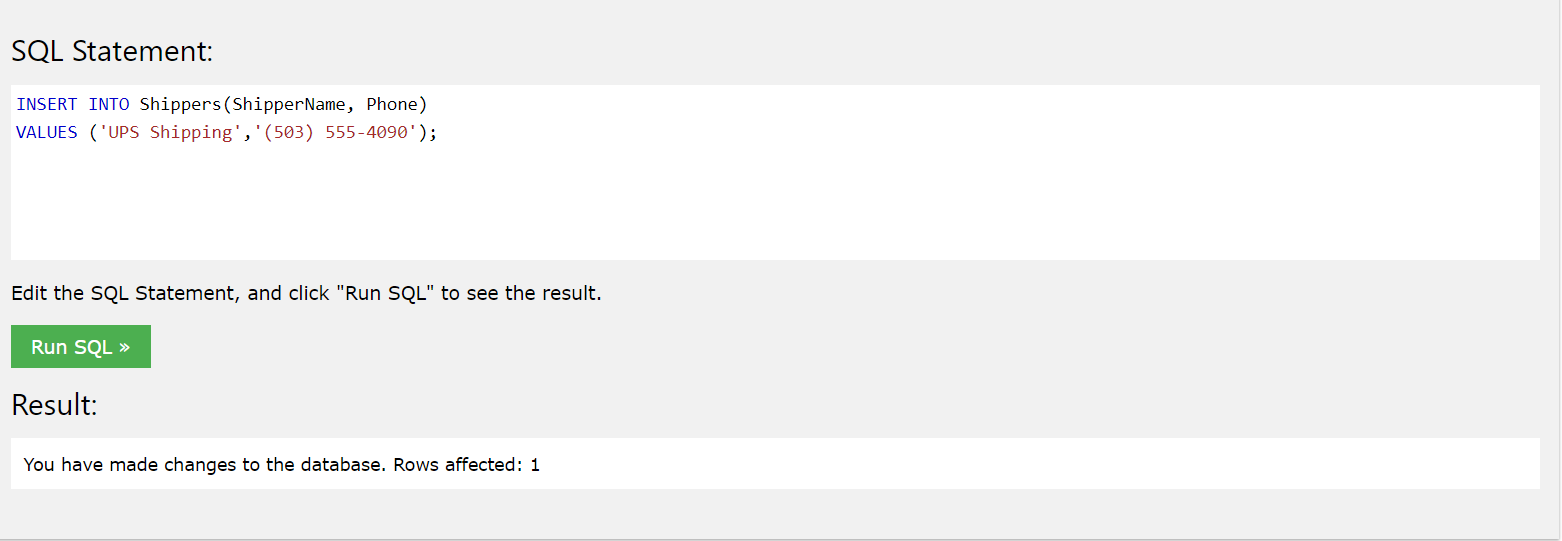
3. Find the ProductName and price whose category name is Seafood using subquery as shown below (Note this is partial of the resulting table):



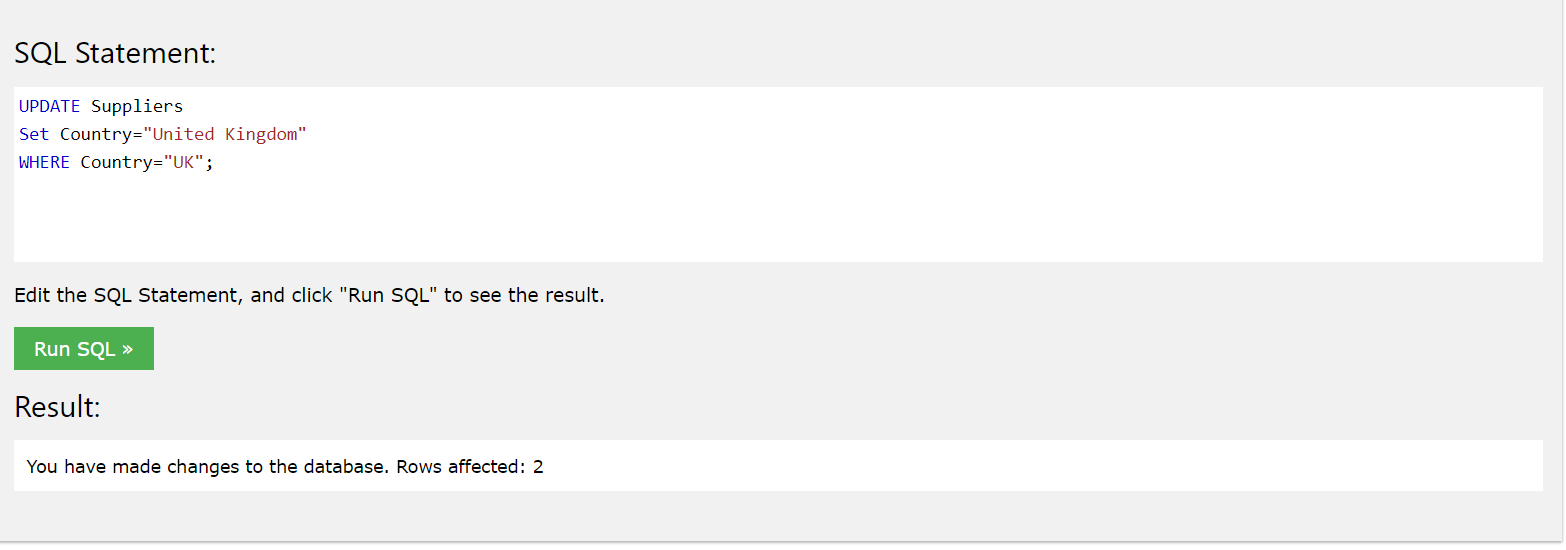
4. Use an inner join to find orderID. shipperID and Shipper name for orders whose ID is less than 10255. Order the result by orderID. Part of the result id shown below:



5. Add a new shipper and then show all the shippers with screen shot. Show the command used:



6. Change all Country UK to United Kingdom, take screen shot to show all the changes. Show the commands used.

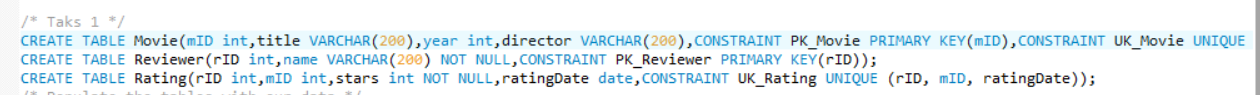


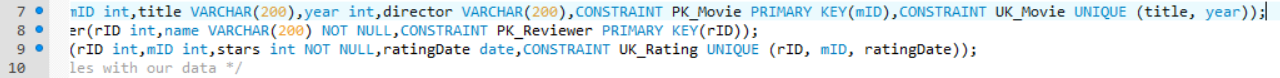
7. Find ProductID from Order Details when theses order is via ShipperID 3. Use subquery. ProductID should not have duplicates. Order your result in increasing order. The resulting table could be long. When copy to submission file, only copy the tuples where ProductID is less than 11.



1. Modify the three CREATE TABLE statements in the rating\_lab.sql to add the following constraints. Your three resulting CREATE TABLE statements should meet both the Key constraints and Not-Null:

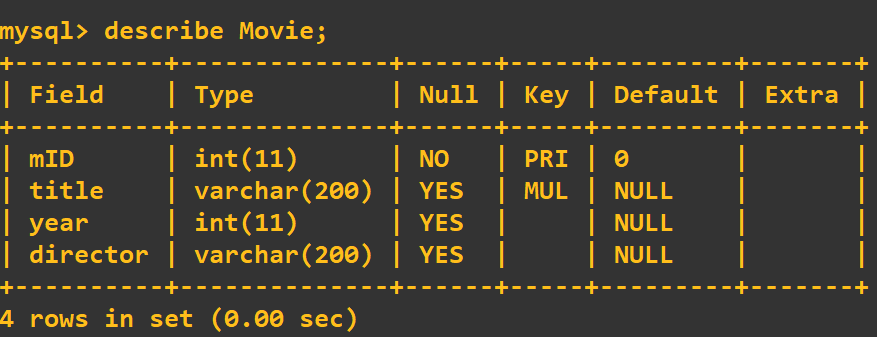
Note: The first Create table statement line is too long and cannot fit in one screenshot. So, it needed a second screenshot.



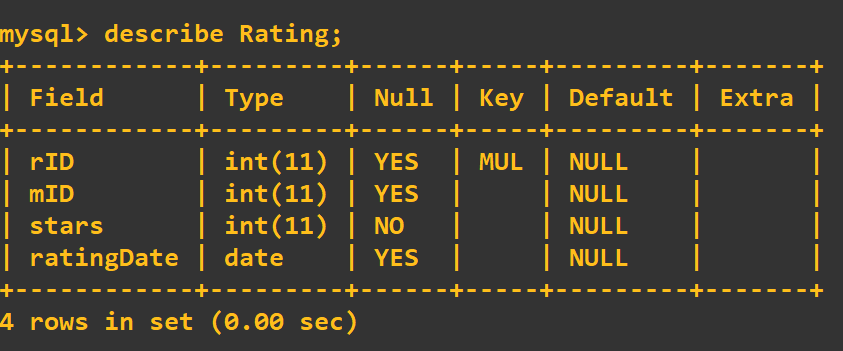


1. Load the Database tables

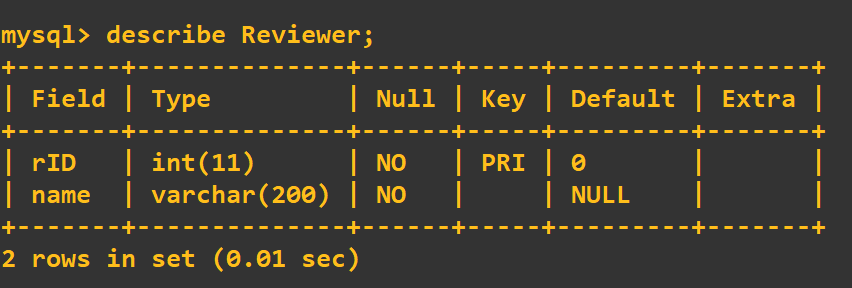
Movie Table:



Rating Table:



Reviewer Table:



FINAL