Lab 04

# Objectives:

The purpose of this lab is to familiarize you with tables and subquery operations.

# LAB 04 – SUBMISSION

You will be submitting a Word document with the questions listed followed by the answers with the SQL queries and screenshots of the results. Use () brackets for subqueries and try to run the subquery first if it executes fine without any errors

## In the Word document header, you should have your Name, Student ID number, section.

**QUESTION 1**

Write a SQL query using subqueries to display payment date in the format “DD-MON-YYYY’ along with customer number of those customer number which is below the customer number of retail customer name ‘Lisboa Souveniers, Inc’ and the payment date are above the order date of retail order number 10361.

**SELECT**

**TO\_CHAR(p.paymentdate, 'DD-MON-YYYY') AS formatted\_payment\_date,**

**p.customernumber**

**FROM**

**retailpayments p**

**JOIN**

**retailcustomers c ON p.customernumber = c.customernumber**

**JOIN**

**retailorders o ON o.ordernumber = 10361**

**WHERE**

**p.paymentdate > o.orderdate**

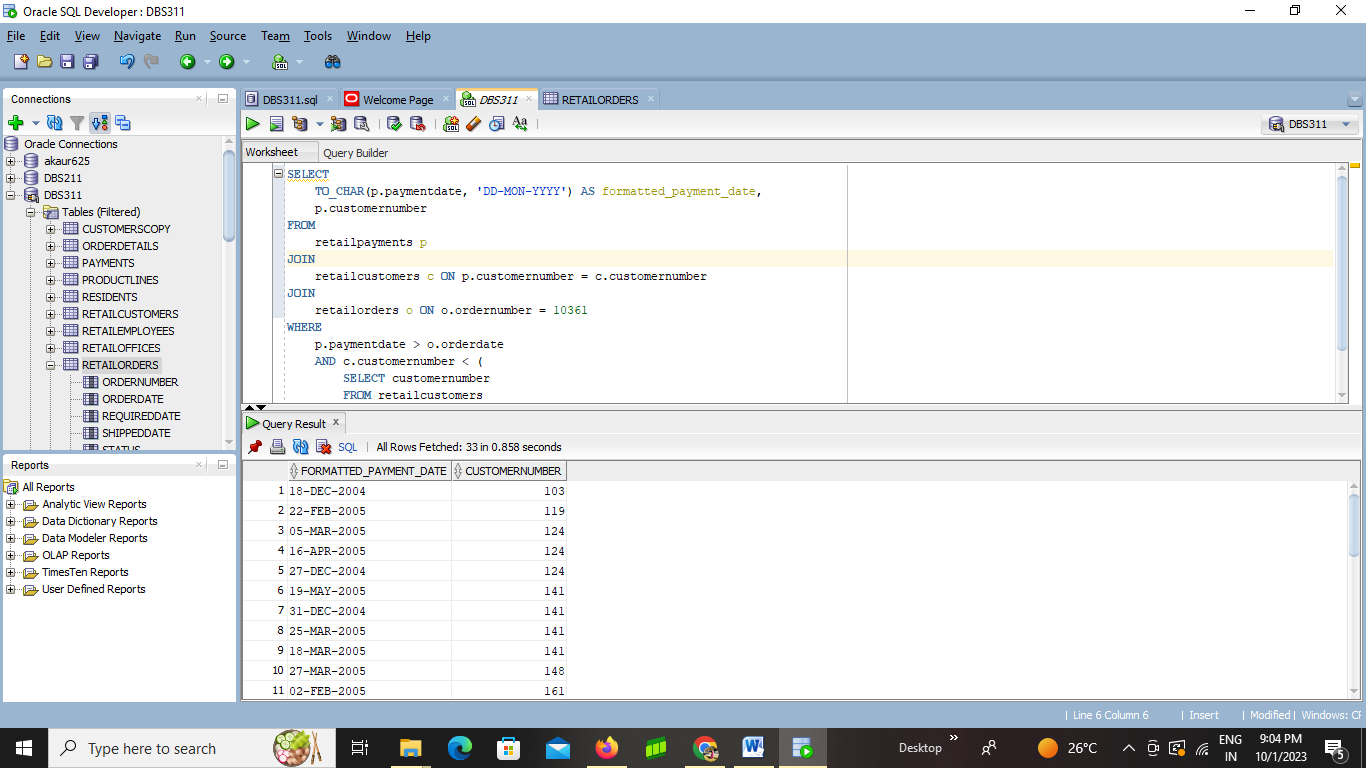
**AND c.customernumber < (**

**SELECT customernumber**

**FROM retailcustomers**

**WHERE customername = 'Lisboa Souveniers, Inc'**

**);**



**QUESTION 2**

Write a SQL query using subqueries, single row functions to display the retail product’s product name and quantity in stock where the product names contain the first character of retail employee’s first name of employee number 1286 and the quantity in stock is above the minimum payment amount of customer number 398.

**SELECT productname, quantityinstock**

**FROM RETAILPRODUCTS**

**WHERE LOWER(productname) LIKE CONCAT (CONCAT('%',(SELECT LOWER(SUBSTR(firstname,1,1))**

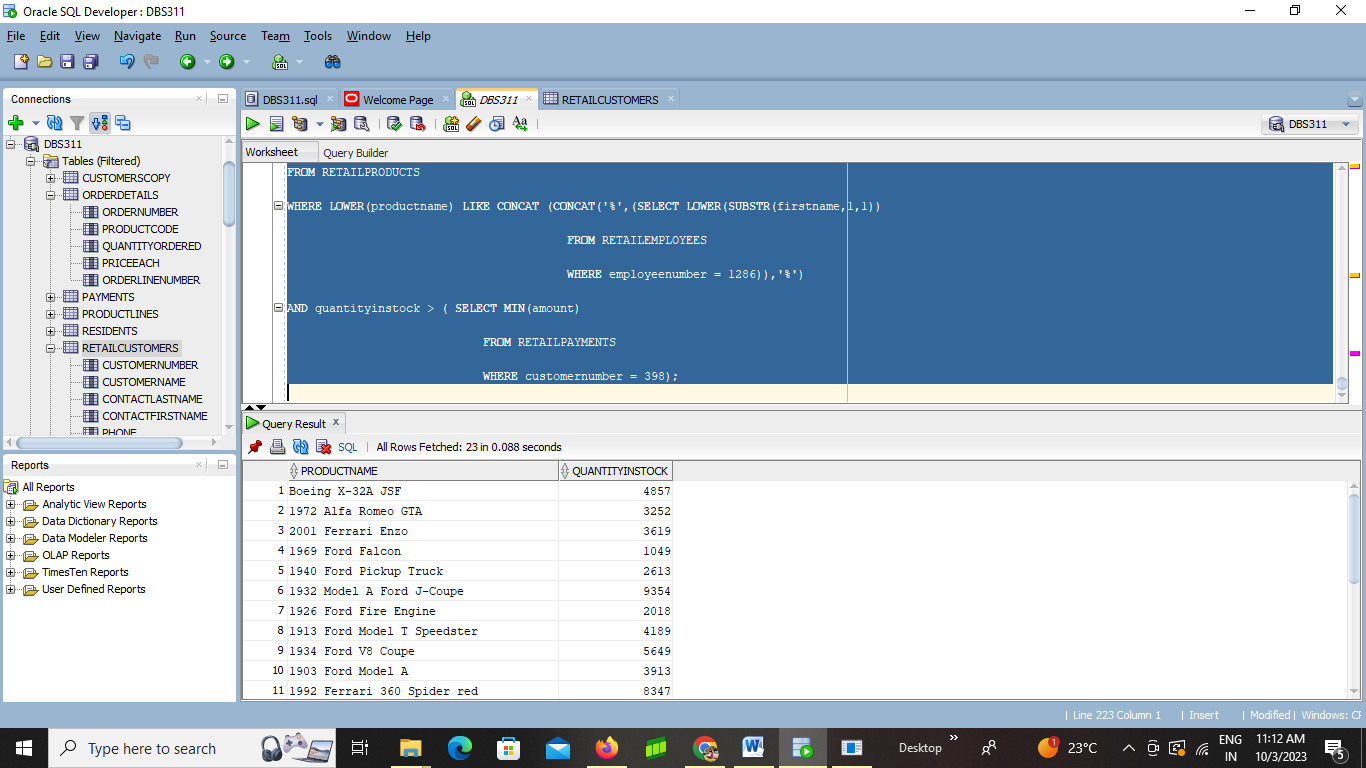
**FROM RETAILEMPLOYEES**

**WHERE employeenumber = 1286)),'%')**

**AND quantityinstock > ( SELECT MIN(amount)**

**FROM RETAILPAYMENTS**

**WHERE customernumber = 398);**



**QUESTION 3**

Write a SQL query using subqueries to display the retail payment’s customer number along with their minimum payment amount grouped by their customer number which are above the minimum payment amount of customer number 462.

**SELECT customernumber, MIN(amount)**

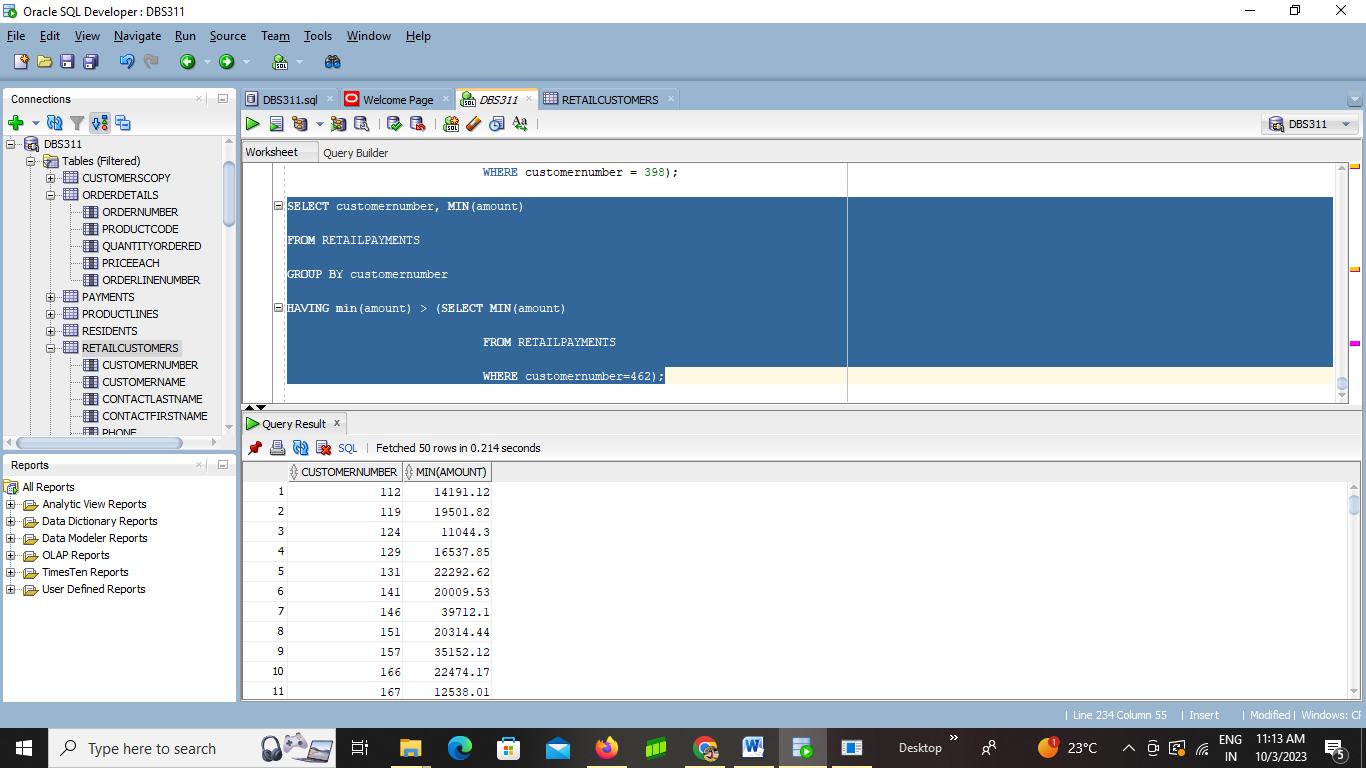
**FROM RETAILPAYMENTS**

**GROUP BY customernumber**

**HAVING min(amount) > (SELECT MIN(amount)**

**FROM RETAILPAYMENTS**

**WHERE customernumber=462);**



**QUESTION 4**

Write a SQL query using subqueries to display the order details’ order number, and their average price each of those below the average price each of order number 10129 and those product code that are in the product code list of retail products’ product line ‘Vintage Cars’

**SELECT**

**od1.ordernumber,**

**AVG(od1.priceeach) AS averagePrice**

**FROM**

**orderdetails od1**

**WHERE**

**od1.ordernumber IN (**

**SELECT**

**ordernumber**

**FROM**

**orderdetails**

**WHERE**

**ordernumber = 10129**

**)**

**AND od1.productcode IN (**

**SELECT**

**productcode**

**FROM**

**retailproducts**

**WHERE**

**productline = 'Vintage Cars'**

**)**

**GROUP BY**

**od1.ordernumber**

**HAVING**

**AVG(od1.priceeach) < (**

**SELECT**

**AVG(od2.priceeach)**

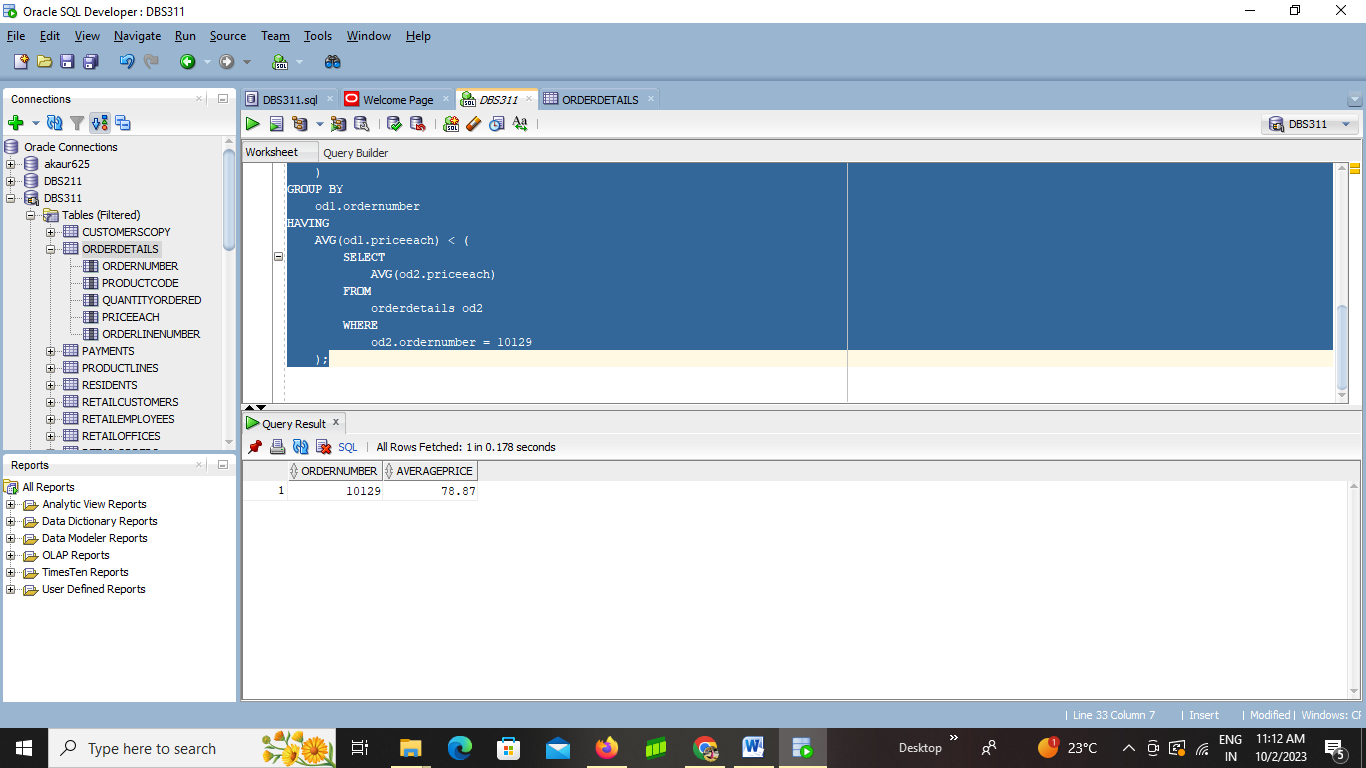
**FROM**

**orderdetails od2**

**WHERE**

**od2.ordernumber = 10129**

**);**



**QUESTION 5**

Write a SQL query using subqueries to find all retail customers that have customer number greater than 'Mini Wheels Co.' and whose credit limit are greater than customer number 121 (Hint 2 subqueries are needed and display customer number, credit limit, customer name in the main query)

**SELECT**

**customernumber,**

**creditlimit,**

**customername**

**FROM**

**retailcustomers**

**WHERE**

**customernumber > (**

**SELECT**

**customernumber**

**FROM**

**retailcustomers**

**WHERE**

**customername = 'Mini Wheels Co.'**

**)**

**AND creditLimit > (**

**SELECT**

**creditLimit**

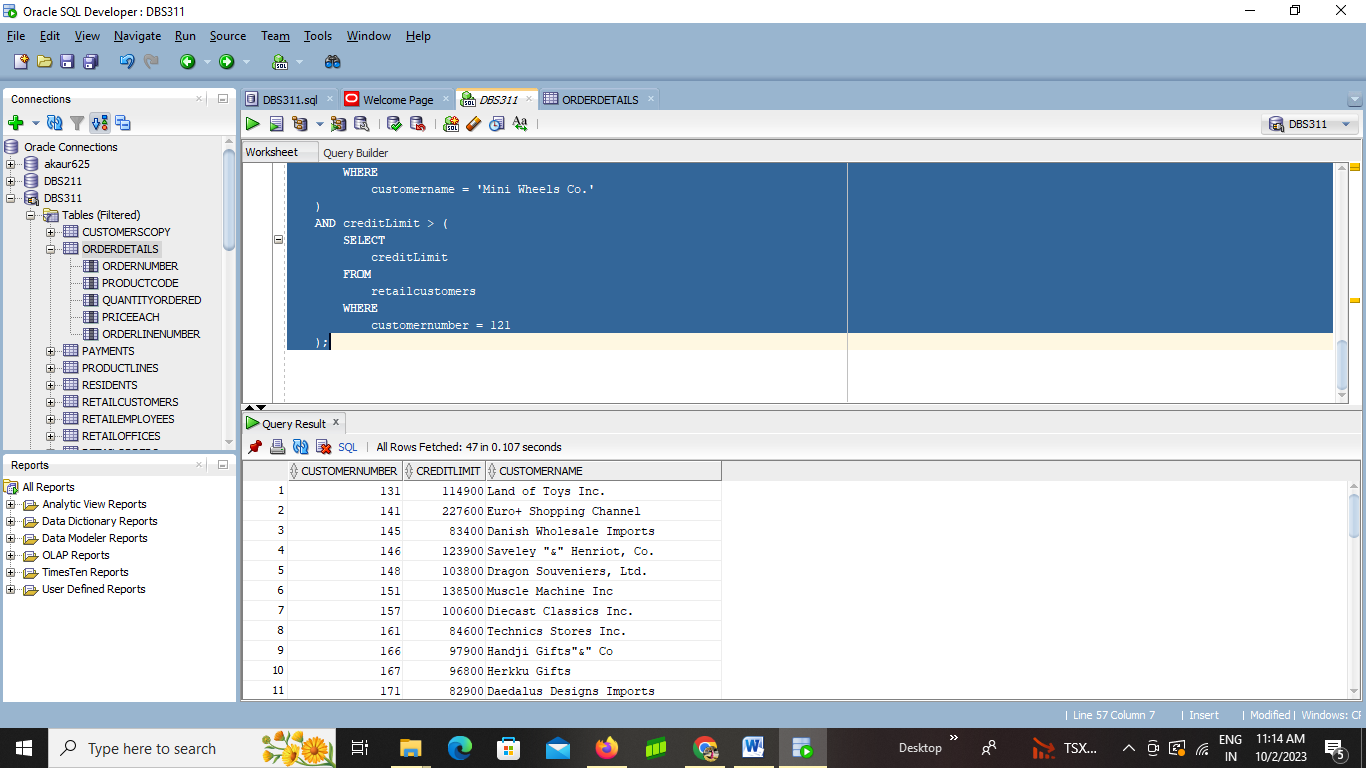
**FROM**

**retailcustomers**

**WHERE**

**customernumber = 121**

**);**



## QUESTION 6

Write a SQL query using subqueries to find all retail employees that have employee number greater than 'Andy Fixter' and whose office code are less than “Barry Jones” (Hint 2 subqueries are needed and display employee number, employee name, office code in the main query)

**SELECT employeenumber, firstname || ' ' || lastname AS employeeName, officecode**

**FROM retailemployees**

**WHERE employeenumber > (**

**SELECT employeenumber**

**FROM retailemployees**

**WHERE firstname || ' ' || lastname = 'Andy Fixter'**

**)**

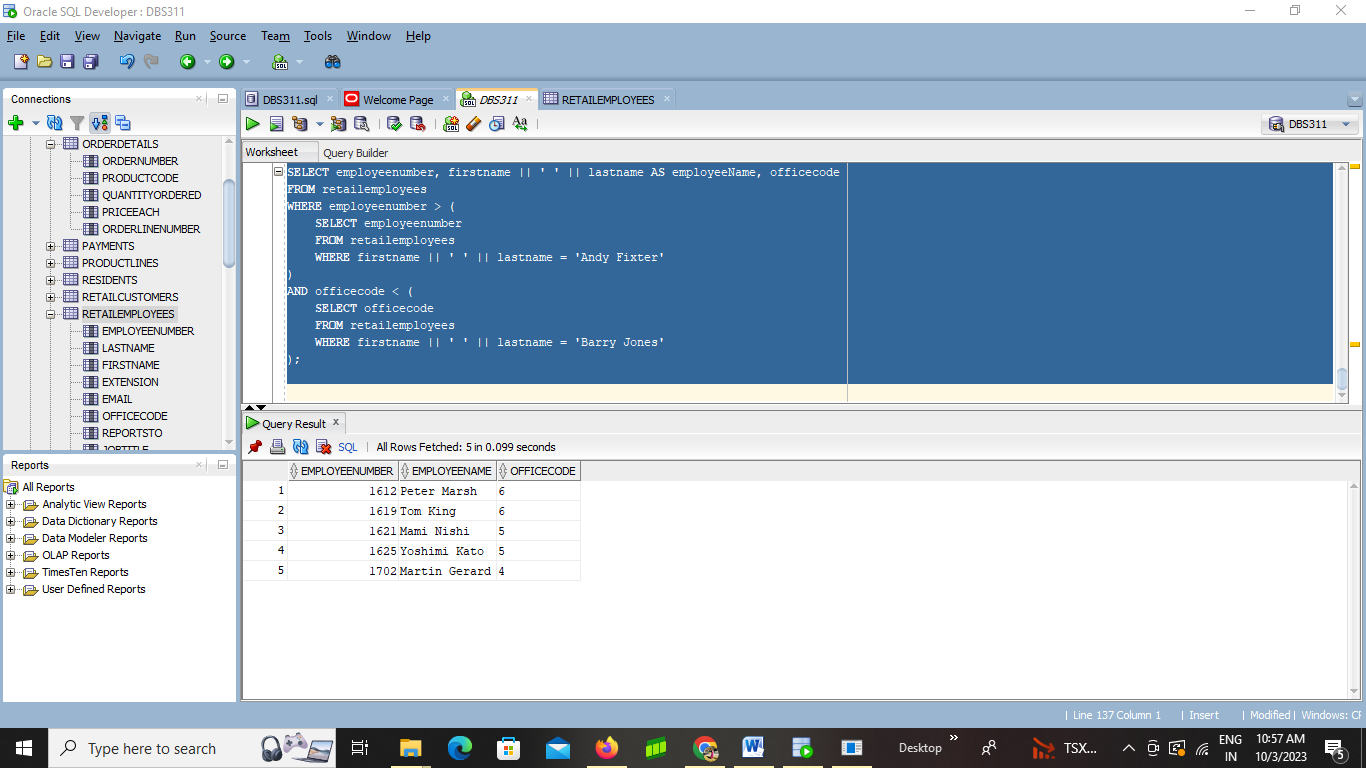
**AND officecode < (**

**SELECT officecode**

**FROM retailemployees**

**WHERE firstname || ' ' || lastname = 'Barry Jones'**

**);**



## QUESTION 7

Write a SQL query using subqueries to find all retail customers who have the same credit limit as the maximum credit limit of all retail customers(Hint 1 subquery and group function are needed and display contact full name, customer number and creditlimit in the main query)

**SELECT**

**contactfirstname || ' ' || contactlastname AS contactFullName,**

**customernumber,**

**creditlimit**

**FROM**

**retailcustomers**

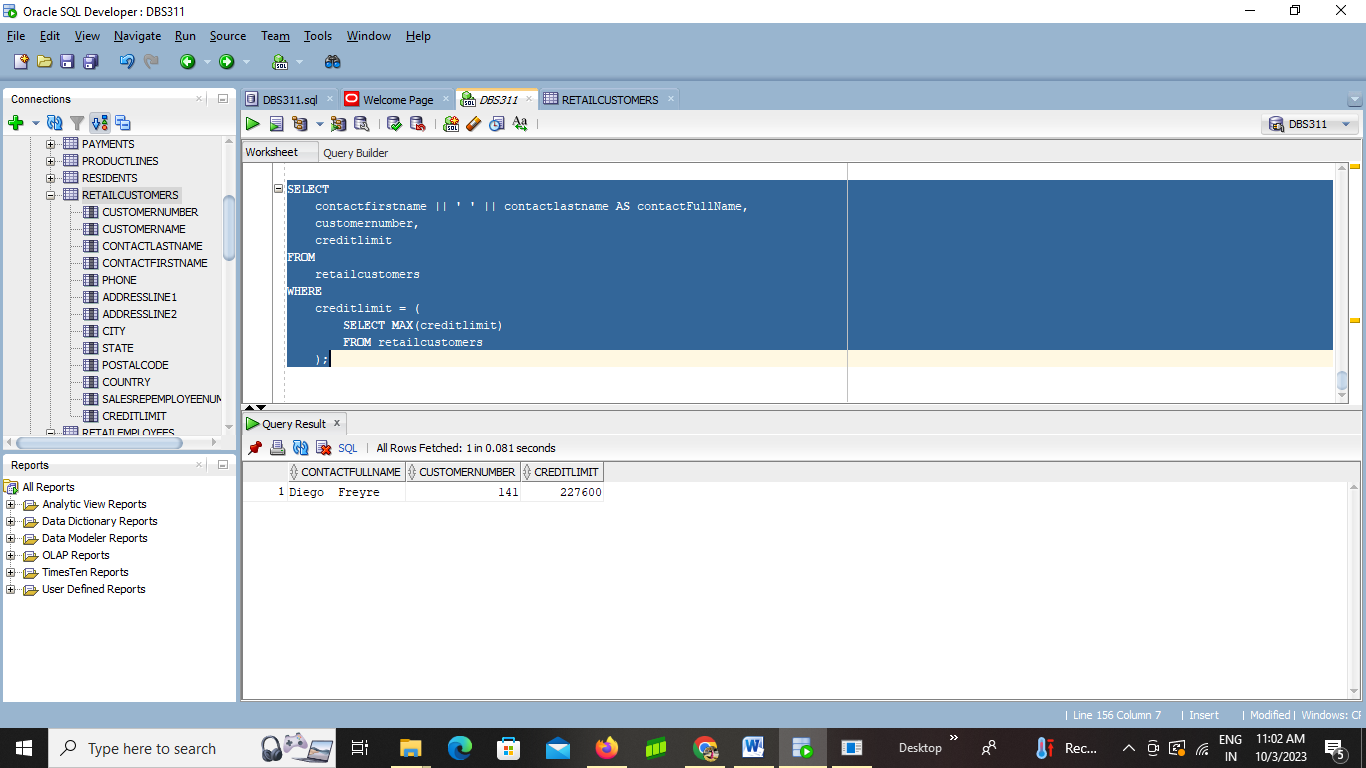
**WHERE**

**creditlimit = (**

**SELECT MAX(creditlimit)**

**FROM retailcustomers**

**);**



## QUESTION 8

Write a SQL query using subqueries to find all retail orders who have the order date above the minimum order date and who have ordered before order number 10107(Hint 2 subquery and group function are needed and display order number, customer number and order date in the main query)

**SELECT ordernumber, customernumber, orderdate**

**FROM retailorders**

**WHERE orderdate > (**

**SELECT MIN(orderdate)**

**FROM retailorders**

**)**

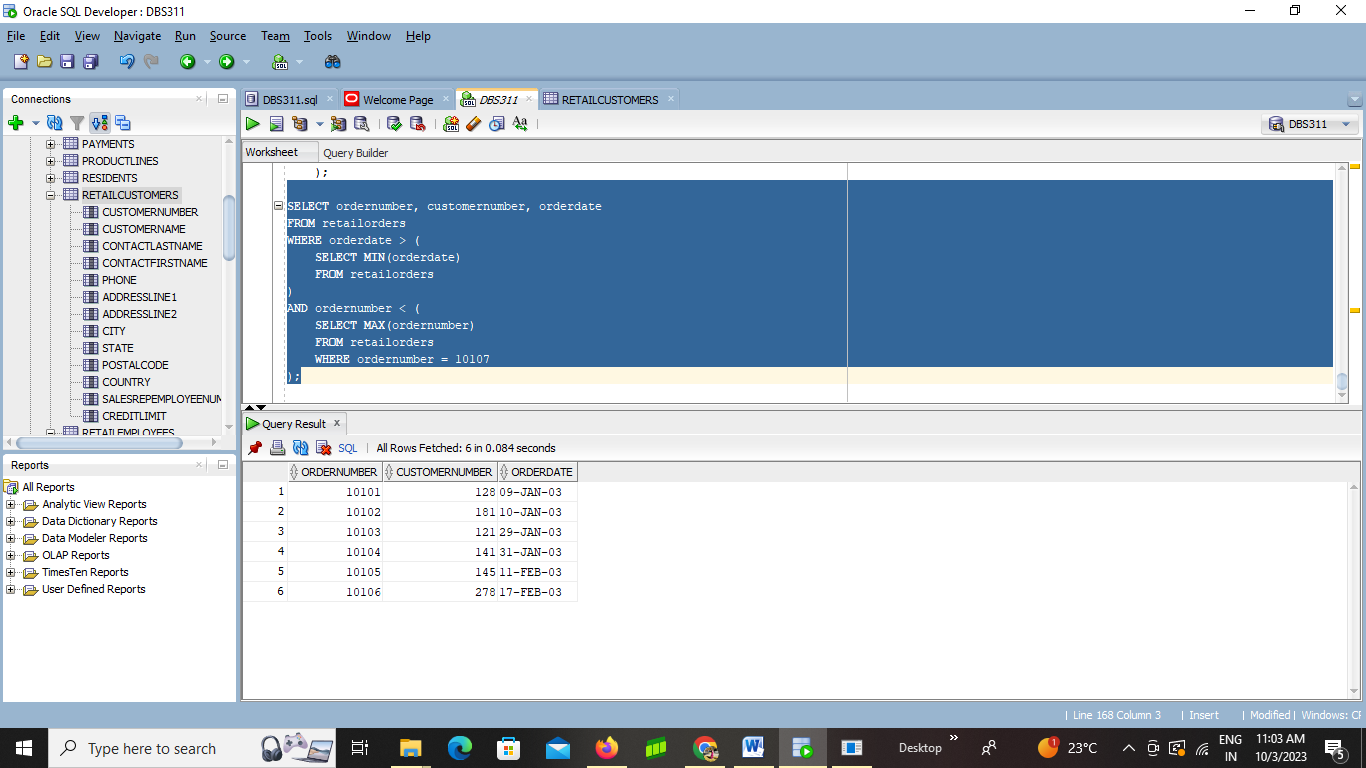
**AND ordernumber < (**

**SELECT MAX(ordernumber)**

**FROM retailorders**

**WHERE ordernumber = 10107**

**);**



## QUESTION 9

Write a SQL query using subqueries to display all orders with minimum order date grouped by the customer number and less than customer number 458’s order date (Hint you will have group by

clause, group function and 1 subquery to display the customer number and the minimum of order date in retail orders table)

**SELECT customernumber, MIN(orderdate) AS minOrderDate**

**FROM retailorders**

**GROUP BY customernumber**

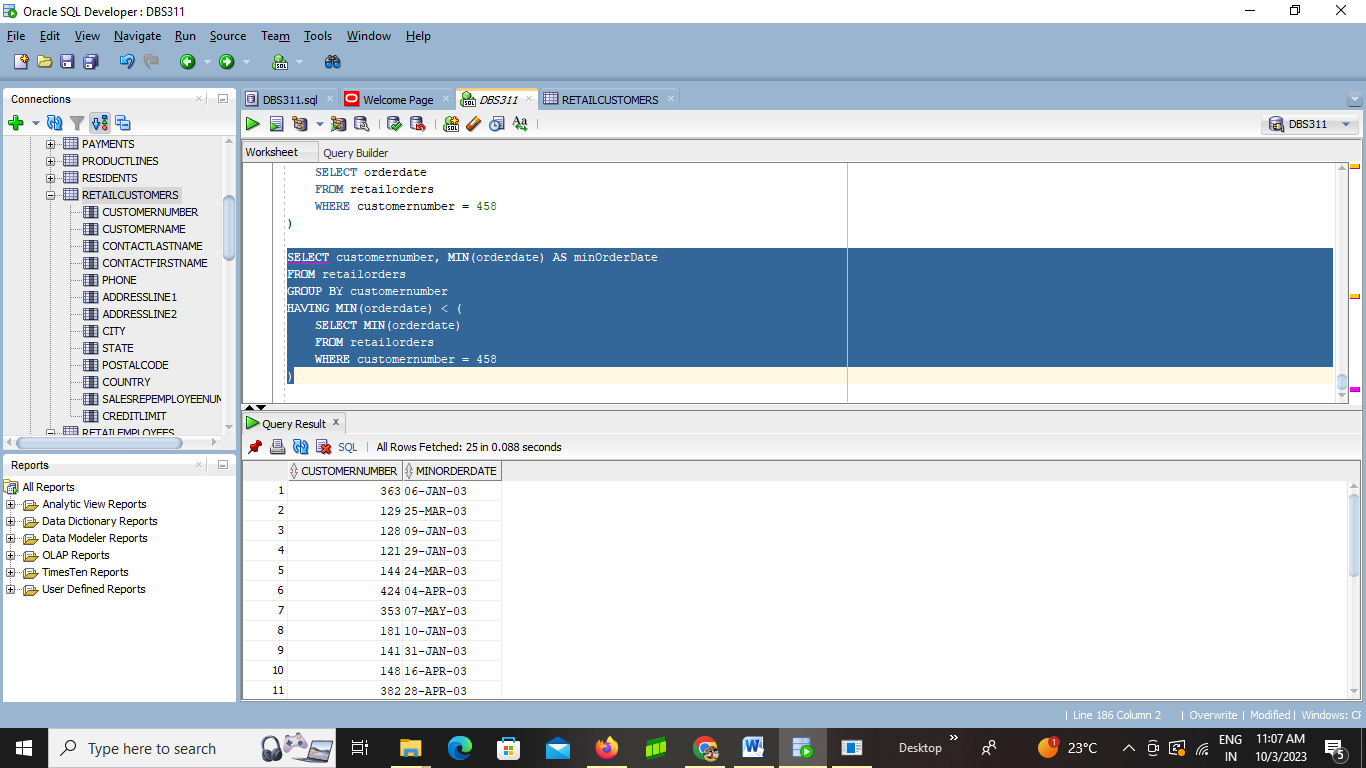
**HAVING MIN(orderdate) < (**

**SELECT MIN(orderdate)**

**FROM retailorders**

**WHERE customernumber = 458**

**)**



## QUESTION 10

Write a SQL query using subqueries to find the order details whose price of each item are above the average of the lowest price item (Hint use over() function in main query, you will have to group by clause, group function in 1 subquery to display the quantity, price, item count of the order number in orders details table)

**SELECT**

**ordernumber,**

**productcode,**

**quantityordered,**

**priceeach**

**FROM (**

**SELECT**

**ordernumber,**

**productcode,**

**quantityordered,**

**priceeach,**

**AVG(priceeach) OVER() AS avg\_lowest\_price**

**FROM orderdetails**

**WHERE priceeach > (**

**SELECT MIN(priceeach)**

**FROM orderdetails**

**)**

**) subquery**

**WHERE priceeach > avg\_lowest\_price;**

