Bahria University,

Karachi Campus



Course: CSL 220 Database management systems Lab

Term: Spring 2022, Class: BSE- 4(B)

Submitted By:

Abdul Quddos 69984

(Name) (Reg. No.)

Submitted To:

Mam Laraib Siddique/Engr. Ayesha Khan

Signed Remarks: Score:

INDEX

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SNO | DATE | LAB NO | LAB OBJECTIVE | SIGN |
| 1 | 8/3/22 | 1 | Intro To DBMS |  |
| 2 | 16/3/22 | 2 | GoupBy, OrderBy and aggregate functions |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Bahria University,

Karachi Campus



LAB EXPERIMENT NO.

1

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 1 | 1. Get the price of an order (by multiplying unit price by quantity). |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Submitted On:

(Date: 8/3/22)

**LAB # 01**

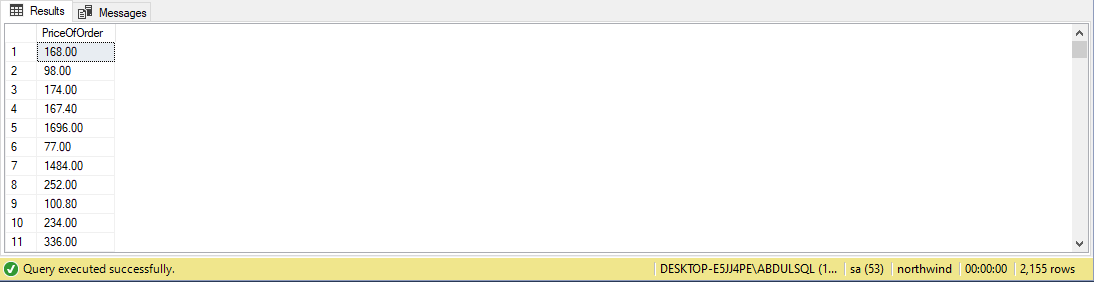
**Task # 01:**

1. **Get the price of an order (by multiplying unit price by quantity).**

**Querry:**

select UnitPrice\*Quantity as PriceOfOrder from [Order Details]

**Output:**



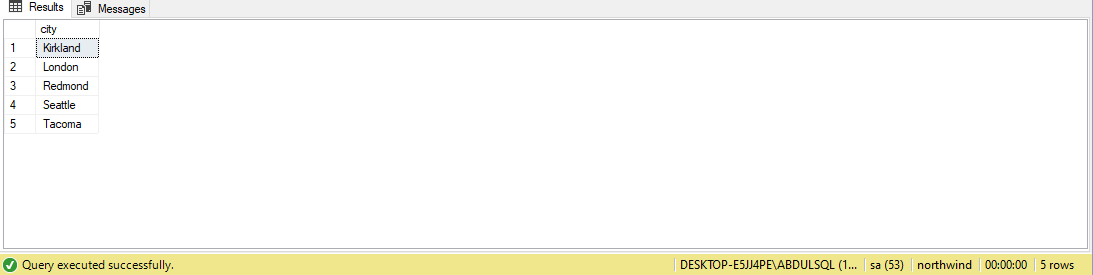
Task#2:

2. Display all cities that employees belong to but don’t allow repetition.

Query:

select distinct city from Employees

Output:



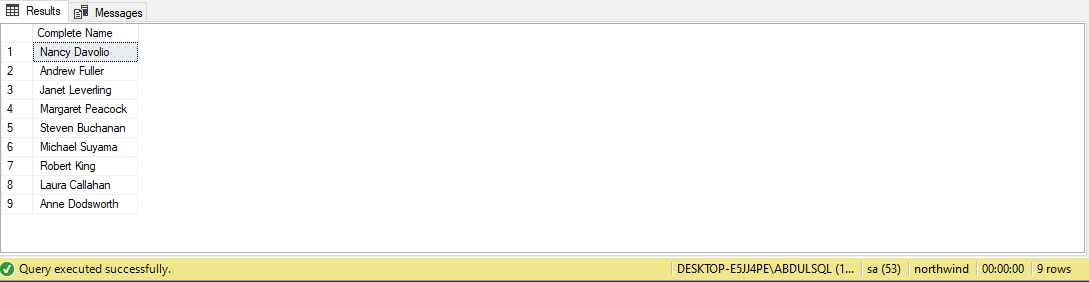
Task#3:

Find complete name of all employees.

Query:

select FirstName+' '+LastName as [Complete Name] from Employees

Output:



Task#4:

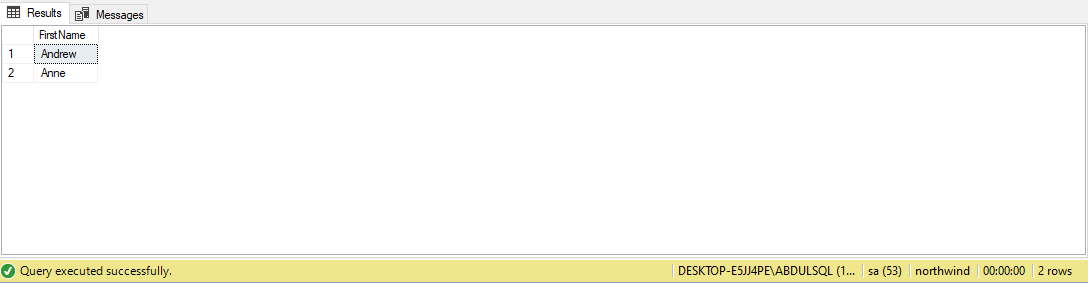
List the name of all employees whose first name starts with the letter ‘A’.

Query:

select FirstName from Employees

where FirstName like 'A%'

Output:



Task#5:

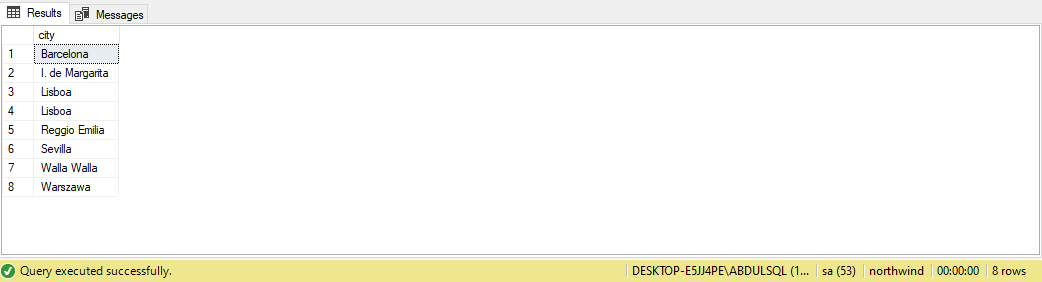
In Customer table, display all cities that ends with the letter ‘a’.

Query:

select city from customers

where City like '%A'

Output:



Task#6:

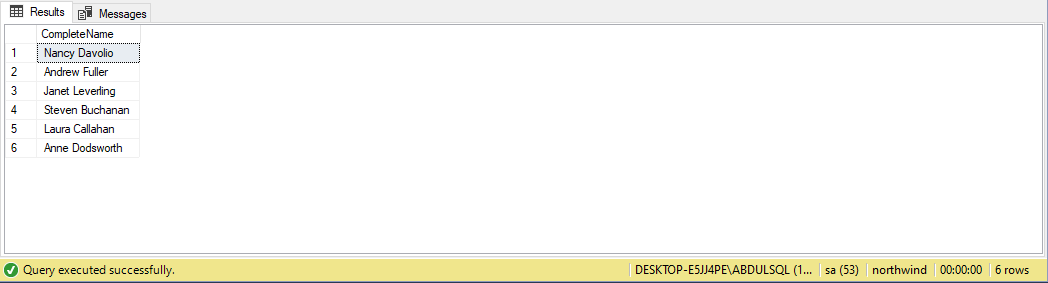
Display names of all employees whose name contain ‘an’.

Query:

select firstname+' '+lastname as CompleteName from Employees

where firstname+' '+lastname like '%an%'

Output:



Task#7:

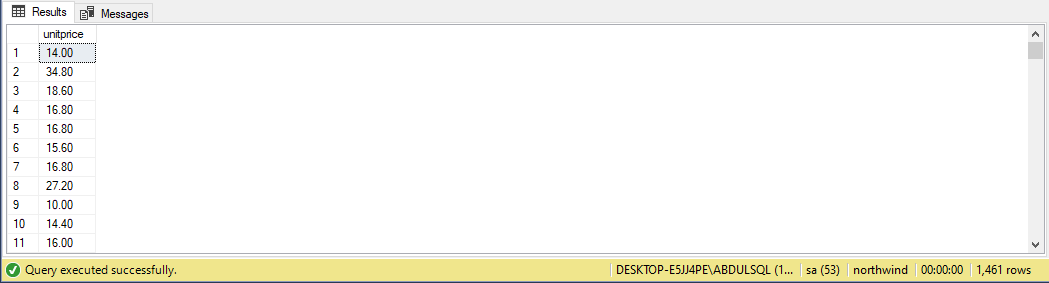
Display all the orders where unit price lies in the range of 10$ to 40$.

Query:

select unitprice from [Order Details]

where unitprice between 10 and 40

Output:



Task#8:

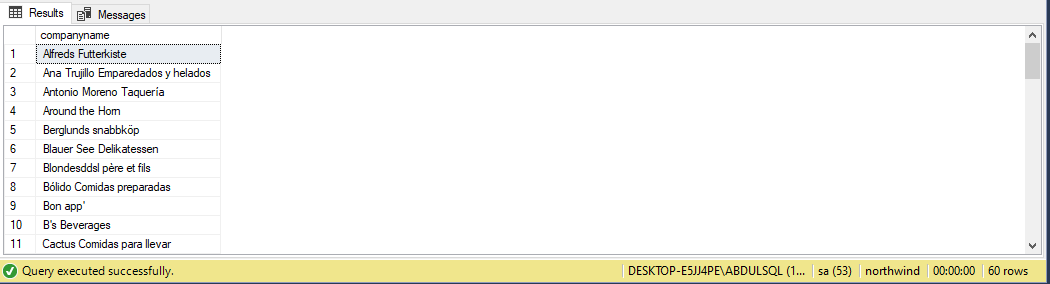
Display the company name where Region is NULL in Customer Table.

Query:

select companyname from customers

where region is null

Output:



Task#9:

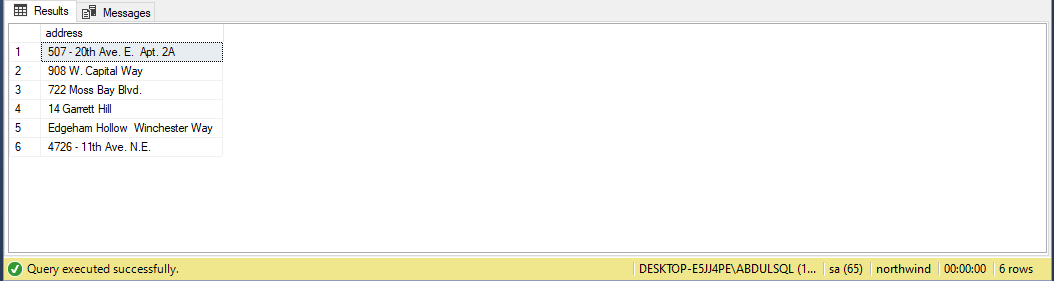
Write a query to list employees whose address does not contain Rd.

Query:

select address from employees

where Address not like '%Rd%'

Output:



Task#10:

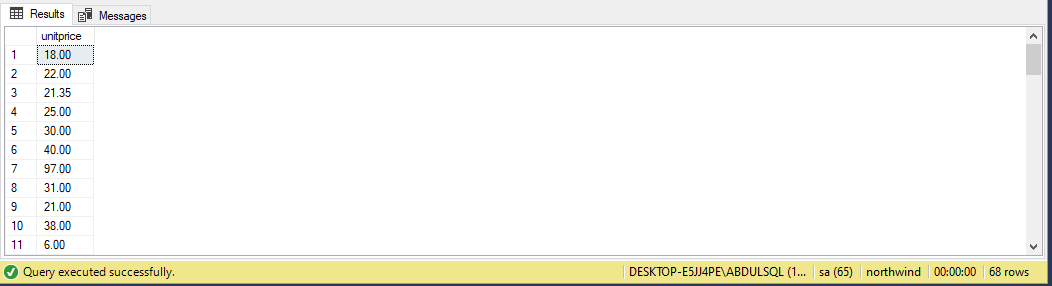
List all products where UnitPrice is not in 10,12,15,17 or 19

Query:

select unitprice from products

where unitprice not in (10,12,15,17,19)

Output:



Task#11:

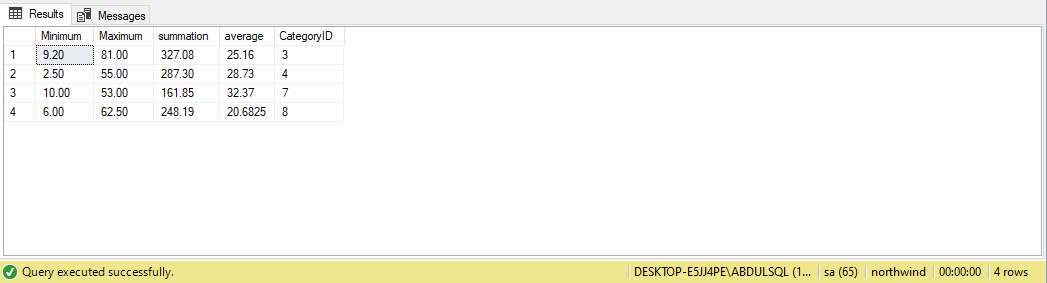
Display the highest, lowest, sum and average UnitPrice of each Category, where highest UnitPrice lies in the range of 50$ to 100$. Label column as CategoryId, Maximum, Minimum, Sum and Average, respectively. (Table: Products)

Query:

select MIN(unitprice) as Minimum,MAX(unitprice) as Maximum,SUM(unitprice) as summation,AVG(unitprice) as average,CategoryID from products group by CategoryID

having MAX(unitprice) between 50 and 100

Output:



Task#12:

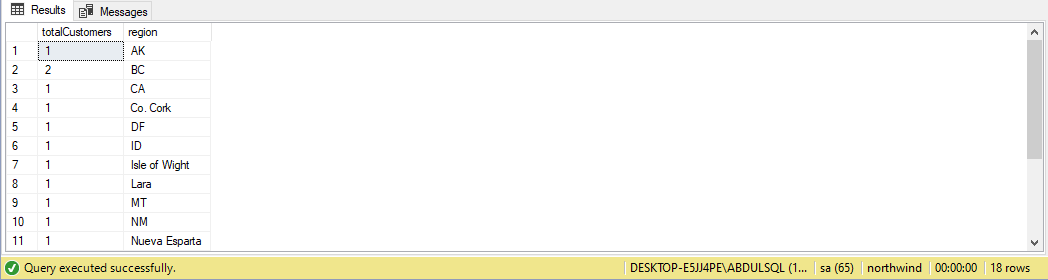
From customers table, Count all customers is each region where region is not null. (Table: Customers)

Query:

select count(customerid) as totalCustomers,region from customers group by region

having region is not null

Output:



Task#13:

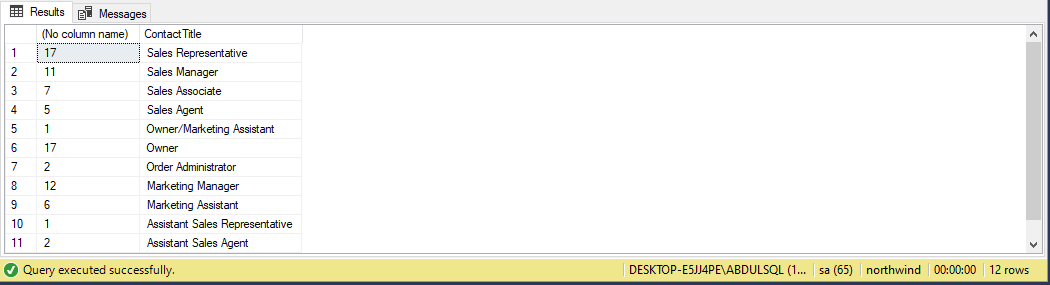
Write a query to display the number of ContactName with same ContactTitle. Sort contact title in descending order. (Table: Customers)

Query:

select count(contacttitle),ContactTitle from customers group by contacttitle

order by contacttitle desc

Output:



Task#14:

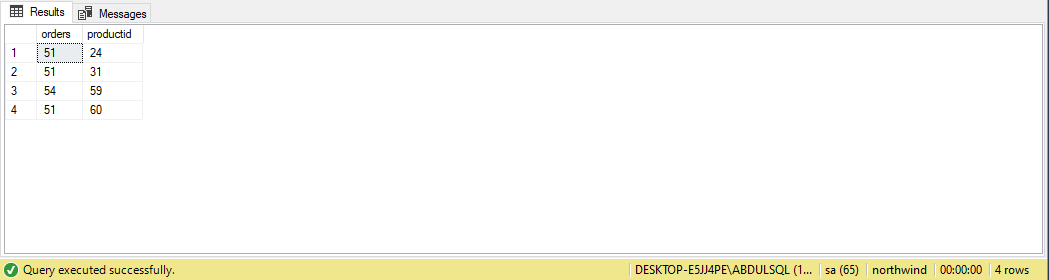
Write a query that count all orders against each product id. No of orders should be greater than 50. (Table: [Order Details])

Query:

select count(orderid) as orders,productid from [order details] group by productid

having count(orderid) > 50

Output:



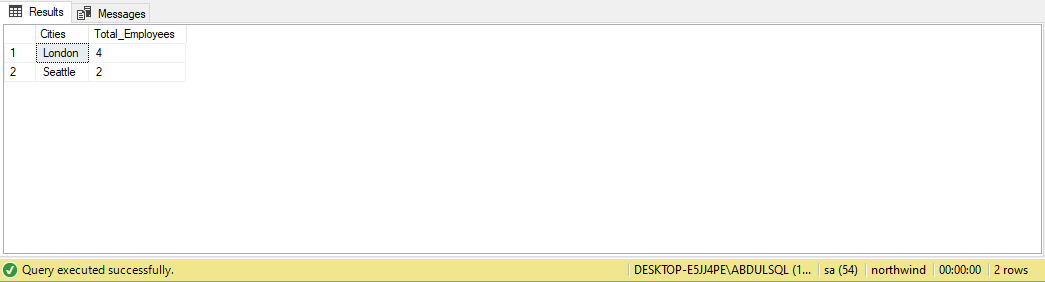
Task#15:

List only those cities in which more than or equals to 2 employees are living

Query:

select distinct City as Cities,count(EmployeeID) as Total\_Employees from Employees group by City having count(EmployeeID)>=2 and City is not null

Output:



Task#:

Query:

Output:

Task#:

Query:

Output: