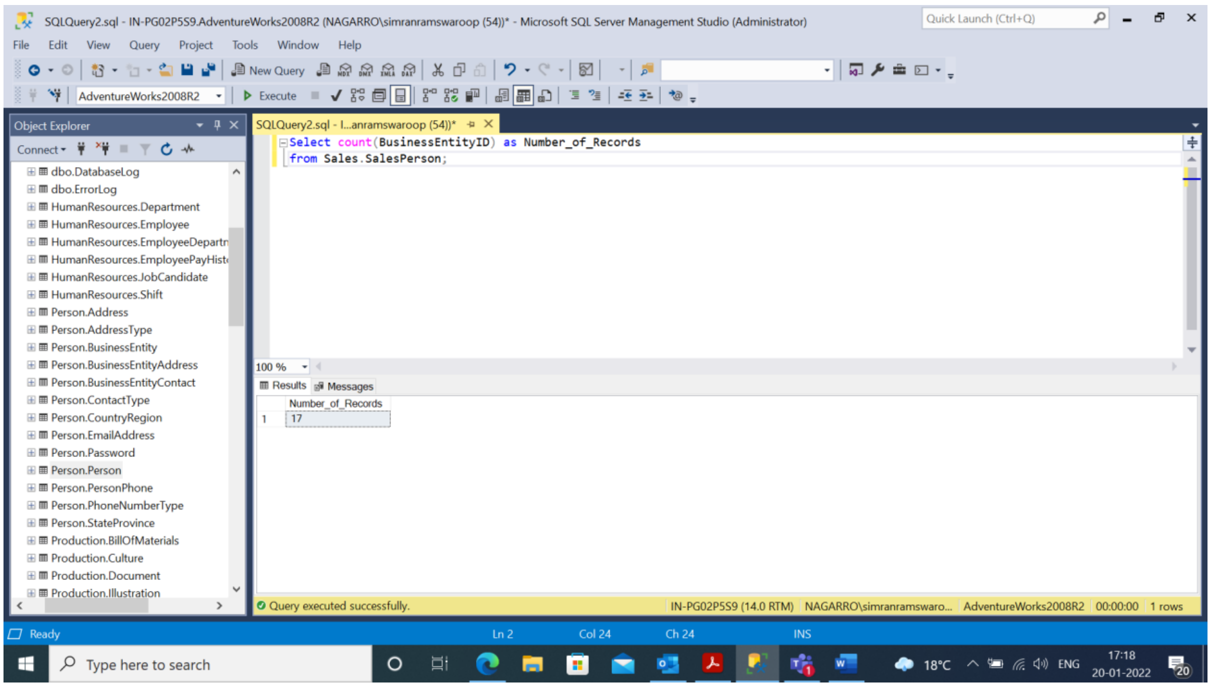
Assignment-2

Exercise-1

1. Display the number of records in the [SalesPerson] table. (*Schema(s) involved: Sales*)

Query - **Select count(BusinessEntityID) as Number_of_Records**
from Sales.SalesPerson



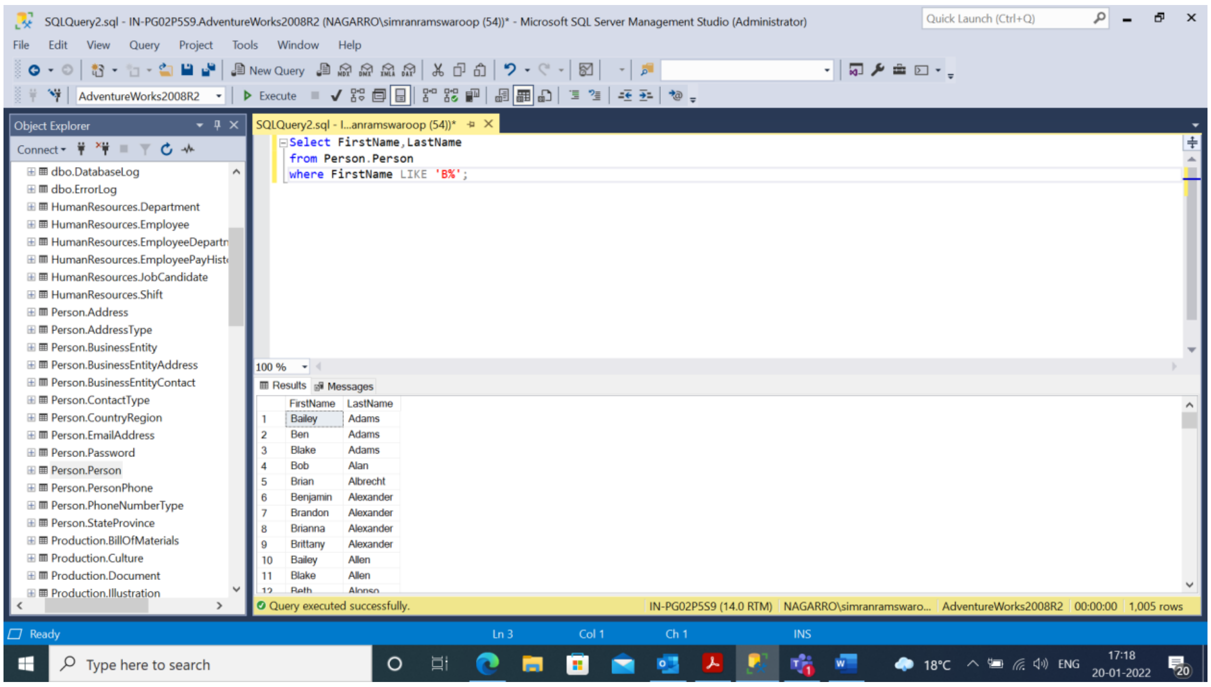
```
Select count(BusinessEntityID) as Number_of_Records
from Sales.SalesPerson;
```

```
   Number_of_Records
1  17
```

Query executed successfully.   IN-PG02P5S9 (14.0 RTM)   NAGARRO\simranramswaro...   AdventureWorks2008R2   00:00:00   1 rows

2.Select both the FirstName and Last Name of records from the Person table where the FirstName begins with the letter ‘B’.

Query - **Select FirstName,LastName**
from Person.Person
where FirstName LIKE 'B%';



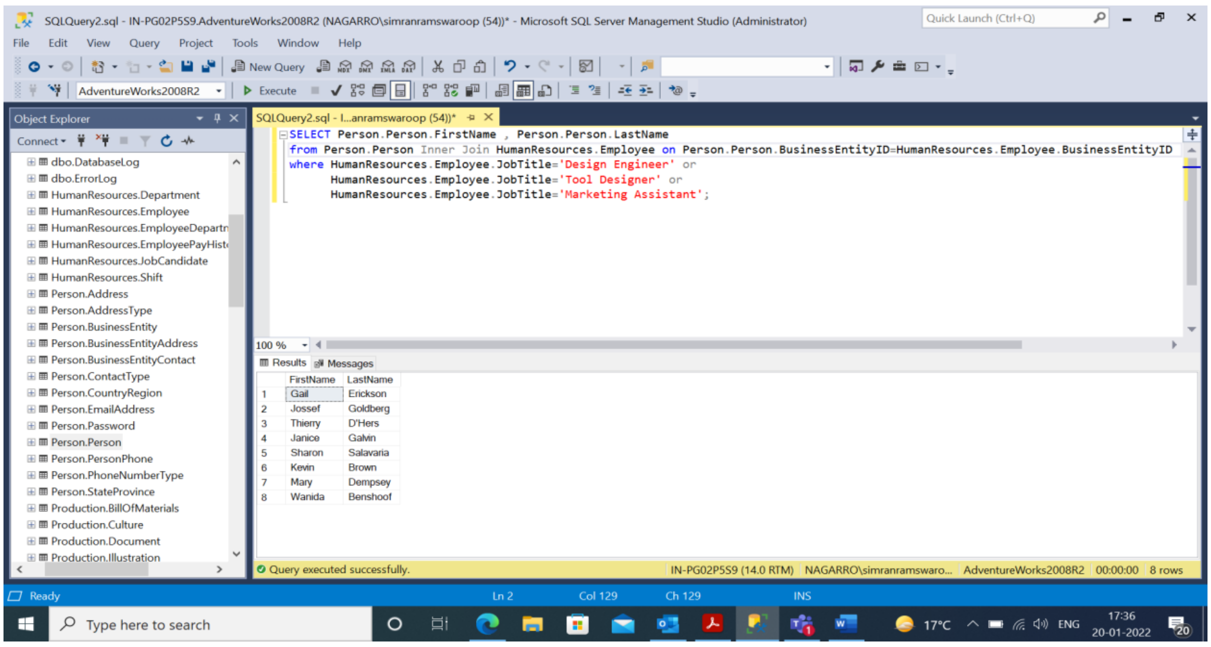
```
Select FirstName,LastName
from Person.Person
where FirstName LIKE 'B%';
```

```
   FirstName  LastName
1  Bailey     Adams
2  Ben        Adams
3  Blake      Adams
4  Bob        Alan
5  Brian      Albrecht
6  Benjamin   Alexander
7  Brandon    Alexander
8  Brianna    Alexander
9  Brittany   Alexander
10 Bailey     Allen
11 Blake      Allen
12 Beth       Alonso
```

```
Query executed successfully.   IN-PG02P5S9 (14.0 RTM)   NAGARRO\simranramswaro...   AdventureWorks2008R2   00:00:00   1,005 rows
```

3.Select a list of FirstName and LastName for employees where Title is one of Design Engineer, Tool Designer or Marketing Assistant.

Query – **SELECT Person.Person.FirstName , Person.Person.LastName**
from Person.Person Inner Join **HumanResources.Employee**
on Person.Person.BusinessEntityID=HumanResources.Employee.BusinessEntityID
where HumanResources.Employee.JobTitle='Design Engineer' or
HumanResources.Employee.JobTitle='Tool Designer' or
HumanResources.Employee.JobTitle='Marketing Assistant';



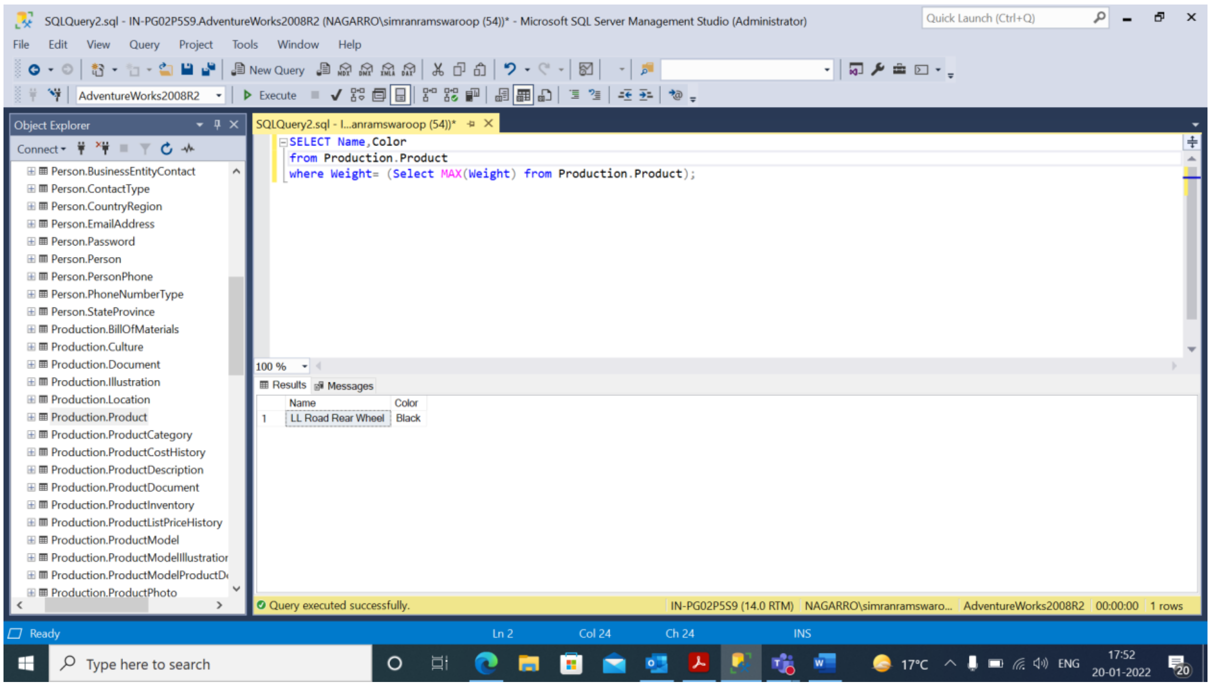
```
SELECT Person.Person.FirstName , Person.Person.LastName
from Person.Person Inner Join HumanResources.Employee on Person.Person.BusinessEntityID=HumanResources.Employee.BusinessEntityID
where HumanResources.Employee.JobTitle='Design Engineer' or
      HumanResources.Employee.JobTitle='Tool Designer' or
      HumanResources.Employee.JobTitle='Marketing Assistant';
```

```
   FirstName  LastName
1  Gail       Erickson
2  Jossef     Goldberg
3  Thierry    D'Hers
4  Janice     Galvin
5  Sharon     Salavaria
6  Kevin      Brown
7  Mary       Dempsey
8  Wanida     Benshoof
```

```
Query executed successfully.   IN-PG02P5S9 (14.0 RTM)   NAGARRO\simranramswaro...   AdventureWorks2008R2   00:00:00   8 rows
```

4.Display the Name and Color of the Product with the maximum weight.

Query – **SELECT Name,Color**
from Production.Product
where Weight= (Select MAX(Weight) from Production.Product);



SQLQuery2.sql - IN-PG02P5S9.AdventureWorks2008R2 (NAGARRO\simranramswaroop (54))* - Microsoft SQL Server Management Studio (Administrator)
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
AdventureWorks2008R2
Execute
Object Explorer
Connect
Person.BusinessEntityContact
Person.ContactType
Person.CountryRegion
Person.EmailAddress
Person.Password
Person.Person
Person.PersonPhone
Person.PhoneNumberType
Person.StateProvince
Production.BillOfMaterials
Production.Culture
Production.Document
Production.Illustration
Production.Location
Production.Product
Production.ProductCategory
Production.ProductCostHistory
Production.ProductDescription
Production.ProductDocument
Production.ProductInventory
Production.ProductListPriceHistory
Production.ProductModel
Production.ProductModelIllustratior
Production.ProductModelProductD
Production.ProductPhoto
SQLQuery2.sql - I...anramswaroop (54))*
100 %
Results
Messages
Ready
Ln 2
Col 24
Ch 24
INS
Type here to search
17°C
ENG
17:52
20-01-2022

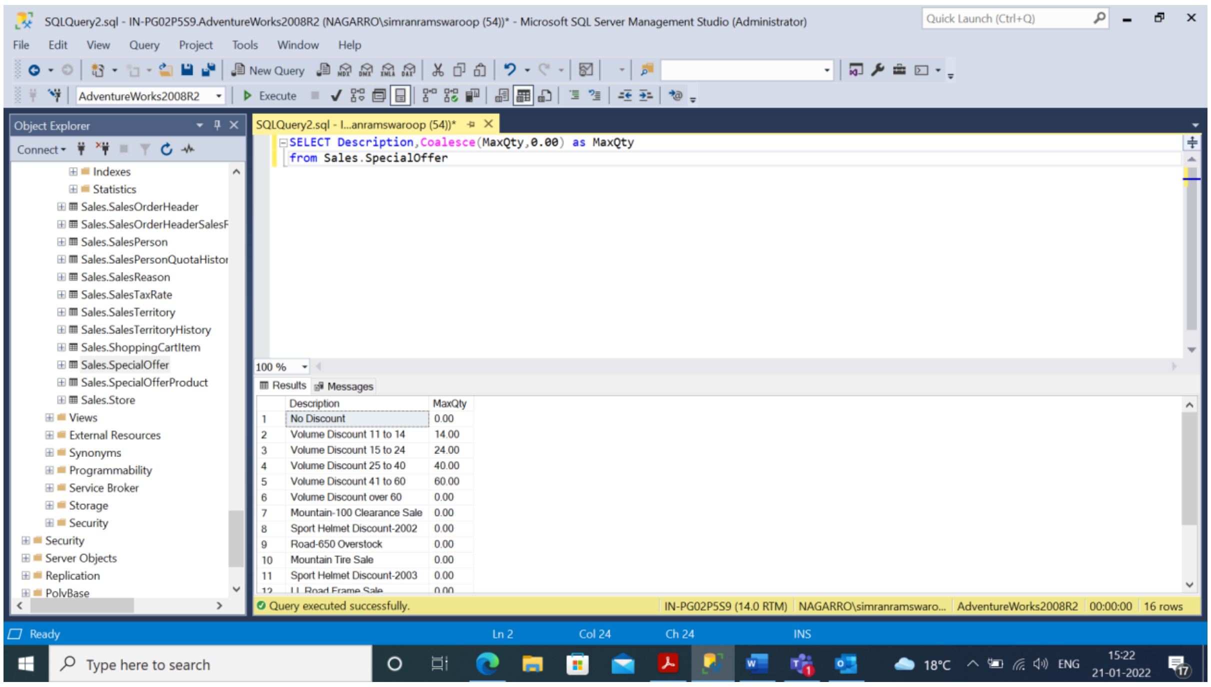
```
SELECT Name,Color
from Production.Product
where Weight= (Select MAX(Weight) from Production.Product);
```

	Name	Color
1	LL Road Rear Wheel	Black

Query executed successfully. IN-PG02P5S9 (14.0 RTM) NAGARRO\simranramswaro... AdventureWorks2008R2 00:00:00 1 rows

5.Display Description and MaxQty fields from the SpecialOffer table. Some of the MaxQty values are NULL, in this case display the value 0.00 instead.

Query – **SELECT Description,Coalesce(MaxQty,0.00) as MaxQty**
from Sales.SpecialOffer



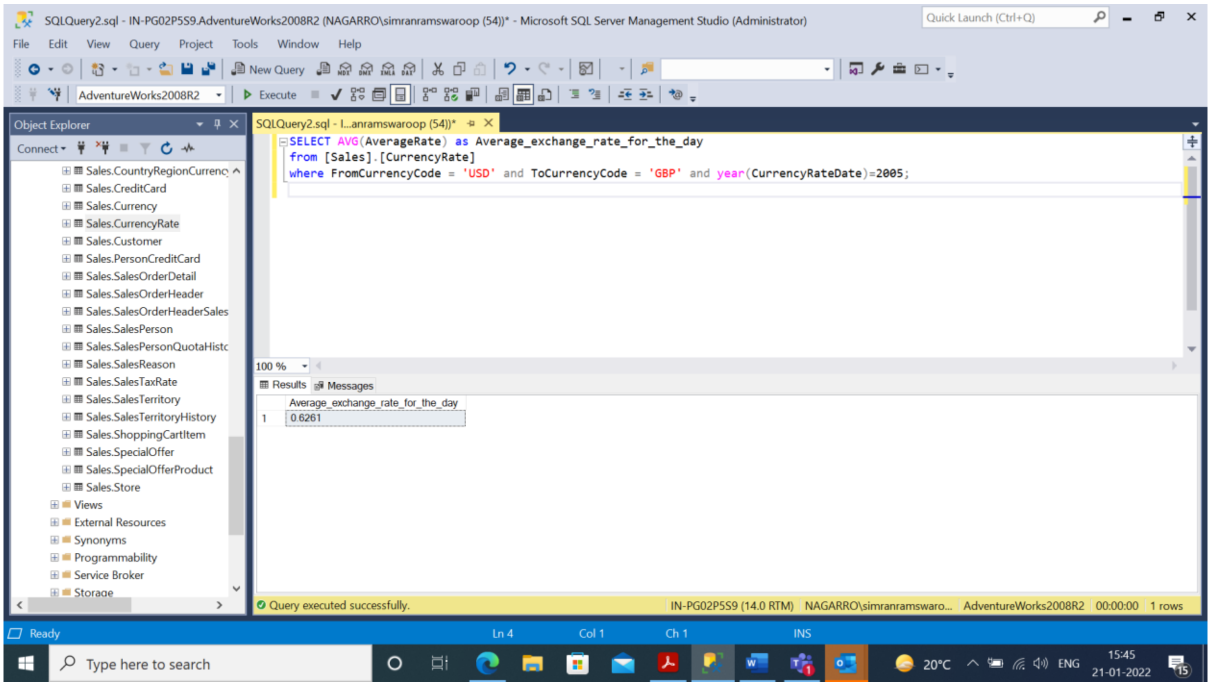
```
SELECT Description,Coalesce(MaxQty,0.00) as MaxQty
from Sales.SpecialOffer
```

```
   Description                  MaxQty
1  No Discount                  0.00
2  Volume Discount 11 to 14     14.00
3  Volume Discount 15 to 24     24.00
4  Volume Discount 25 to 40     40.00
5  Volume Discount 41 to 60     60.00
6  Volume Discount over 60      0.00
7  Mountain-100 Clearance Sale  0.00
8  Sport Helmet Discount-2002   0.00
9  Road-650 Overstock           0.00
10 Mountain Tire Sale           0.00
11 Sport Helmet Discount-2003   0.00
12 LL Road Frame Sale           0.00
```

```
Query executed successfully.   IN-PG02P5S9 (14.0 RTM)   NAGARRO\simranramswaro...   AdventureWorks2008R2   00:00:00   16 rows
```

6.Display the overall Average of the [CurrencyRate].[AverageRate] values for the exchange rate ‘USD’ to ‘GBP’ for the year 2005 i.e. FromCurrencyCode = ‘USD’ and ToCurrencyCode = ‘GBP’. **Note**: The field [CurrencyRate].[AverageRate] is defined as 'Average exchange rate for the day.'

Query – **SELECT AVG(AverageRate) as Average_exchange_rate_for_the_day**
from [Sales].[CurrencyRate]
where FromCurrencyCode = 'USD' and ToCurrencyCode = 'GBP' and year(CurrencyRateDate)=2005;



```
SELECT AVG(AverageRate) as Average_exchange_rate_for_the_day
from [Sales].[CurrencyRate]
where FromCurrencyCode = 'USD' and ToCurrencyCode = 'GBP' and year(CurrencyRateDate)=2005;
```

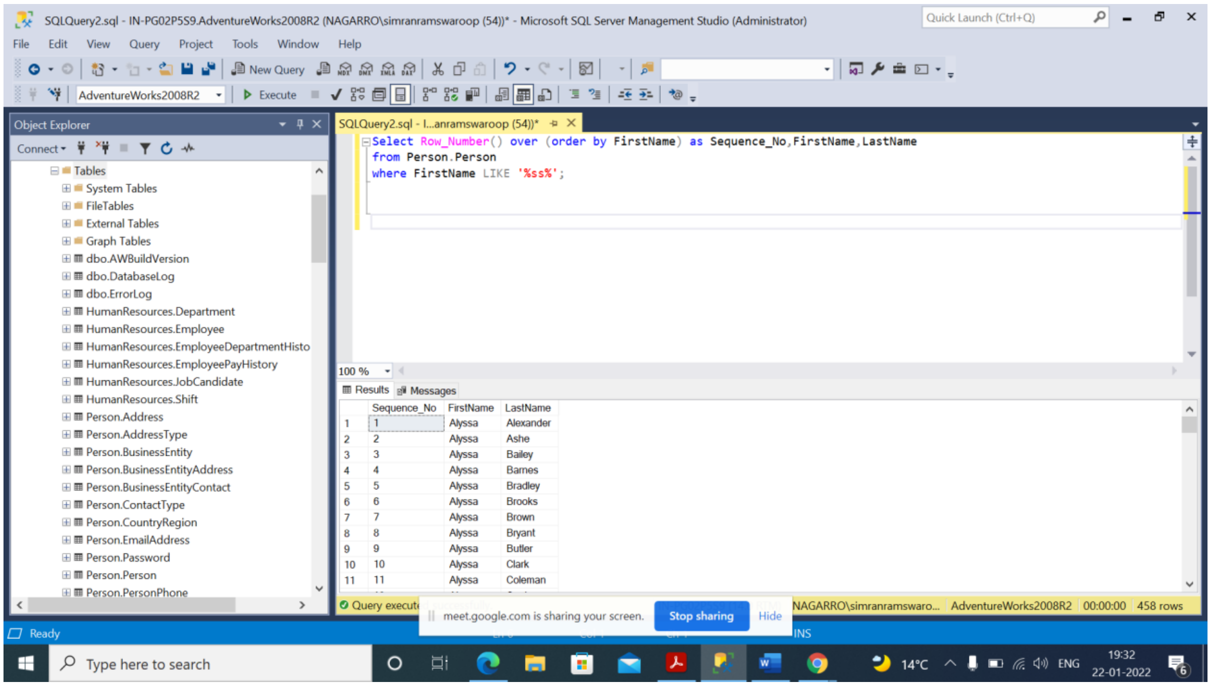
```
   Average_exchange_rate_for_the_day
1  0.6261
```

```
Query executed successfully.   IN-PG02P5S9 (14.0 RTM)   NAGARRO\simranramswaro...   AdventureWorks2008R2   00:00:00   1 rows
```

7.Display the FirstName and LastName of records from the Person table where FirstName contains the letters ‘ss’. Display an additional column with sequential numbers for each row returned beginning at integer 1.

Query –

```
Select Row_Number() over (order by FirstName) as Sequence_No,FirstName,LastName
from Person.Person
where FirstName LIKE '%ss%';
```



```
Select Row_Number() over (order by FirstName) as Sequence_No,FirstName,LastName
from Person.Person
where FirstName LIKE '%ss%';
```

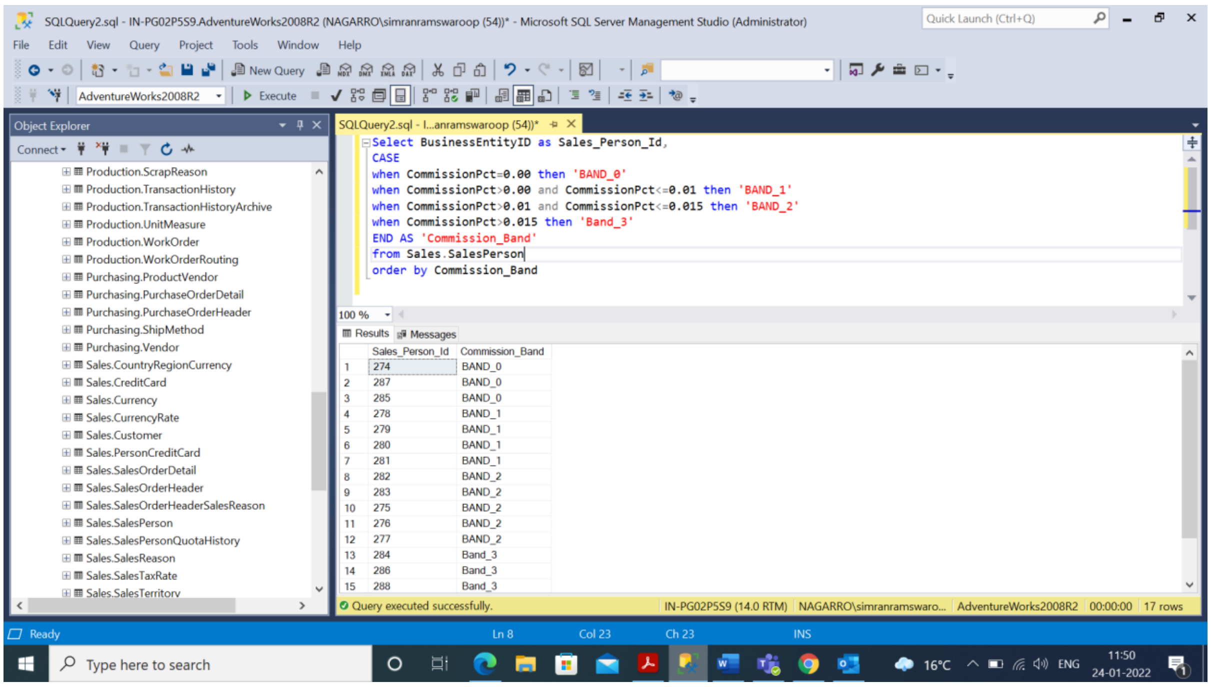
```
    Sequence_No  FirstName  LastName
1   1            Alyssa     Alexander
2   2            Alyssa     Ashe
3   3            Alyssa     Bailey
4   4            Alyssa     Barnes
5   5            Alyssa     Bradley
6   6            Alyssa     Brooks
7   7            Alyssa     Brown
8   8            Alyssa     Bryant
9   9            Alyssa     Butler
10  10           Alyssa     Clark
11  11           Alyssa     Coleman
```

```
Query executed ... NAGARRO\simranramswaro...  AdventureWorks2008R2  00:00:00  458 rows
```

8.Sales people receive various commission rates that belong to 1 of 4 bands .
Display the [SalesPersonID] with an additional column entitled ‘Commission Band’ indicating the appropriate band as above.

Query –

```
Select BusinessEntityID as Sales_Person_Id,
CASE
when CommissionPct=0.00 then 'BAND_0'
when CommissionPct>0.00 and CommissionPct<=0.01 then 'BAND_1'
when CommissionPct>0.01 and CommissionPct<=0.015 then 'BAND_2'
when CommissionPct>0.015 then 'Band_3'
END AS 'Commission_Band'
from Sales.SalesPerson
order by Commission_Band
```



```sql
Select BusinessEntityID as Sales_Person_Id,
CASE
when CommissionPct=0.00 then 'BAND_0'
when CommissionPct>0.00 and CommissionPct<=0.01 then 'BAND_1'
when CommissionPct>0.01 and CommissionPct<=0.015 then 'BAND_2'
when CommissionPct>0.015 then 'Band_3'
END AS 'Commission_Band'
from Sales.SalesPerson
order by Commission_Band
```

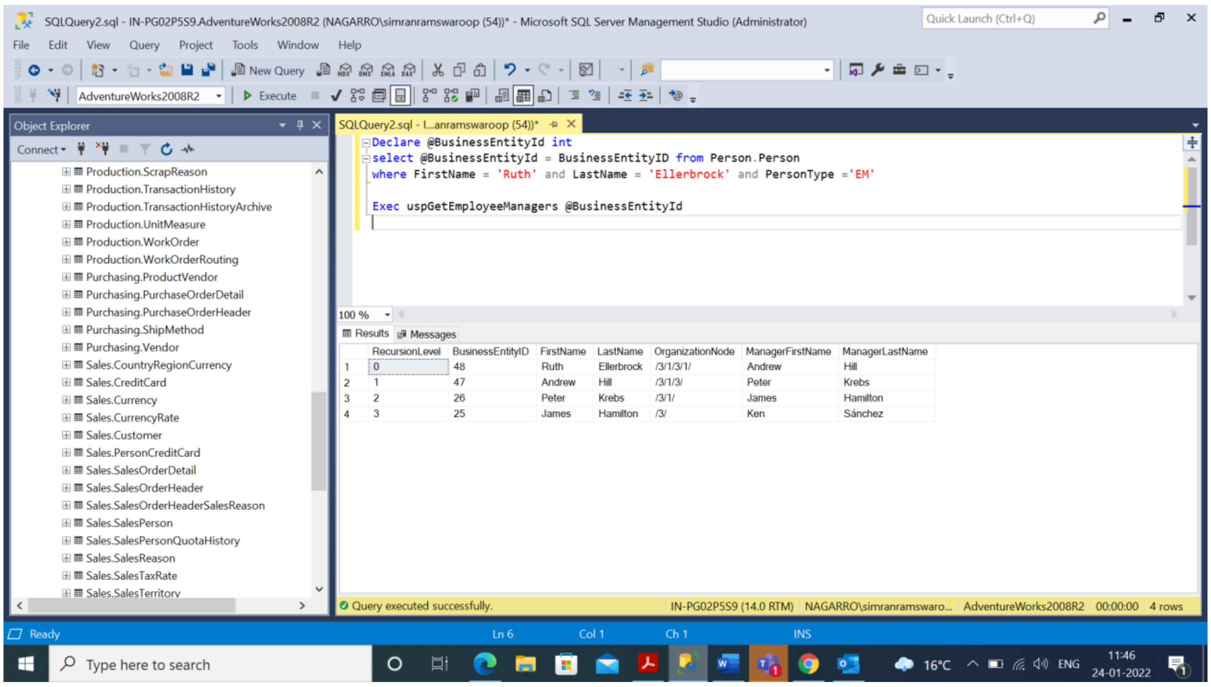
	Sales_Person_Id	Commission_Band
1	274	BAND_0
2	287	BAND_0
3	285	BAND_0
4	278	BAND_1
5	279	BAND_1
6	280	BAND_1
7	281	BAND_1
8	282	BAND_2
9	283	BAND_2
10	275	BAND_2
11	276	BAND_2
12	277	BAND_2
13	284	Band_3
14	286	Band_3
15	288	Band_3

Query executed successfully. IN-PG02P5S9 (14.0 RTM) NAGARRO\simranramswaro... AdventureWorks2008R2 00:00:00 17 rows

9. Display the managerial hierarchy from Ruth Ellerbrock (person type – EM) up to CEO Ken Sanchez.

Query –

```sql
Declare @BusinessEntityId int
select @BusinessEntityId = BusinessEntityID from Person.Person
where FirstName = 'Ruth' and LastName = 'Ellerbrock' and PersonType ='EM'

Exec uspGetEmployeeManagers @BusinessEntityId
```



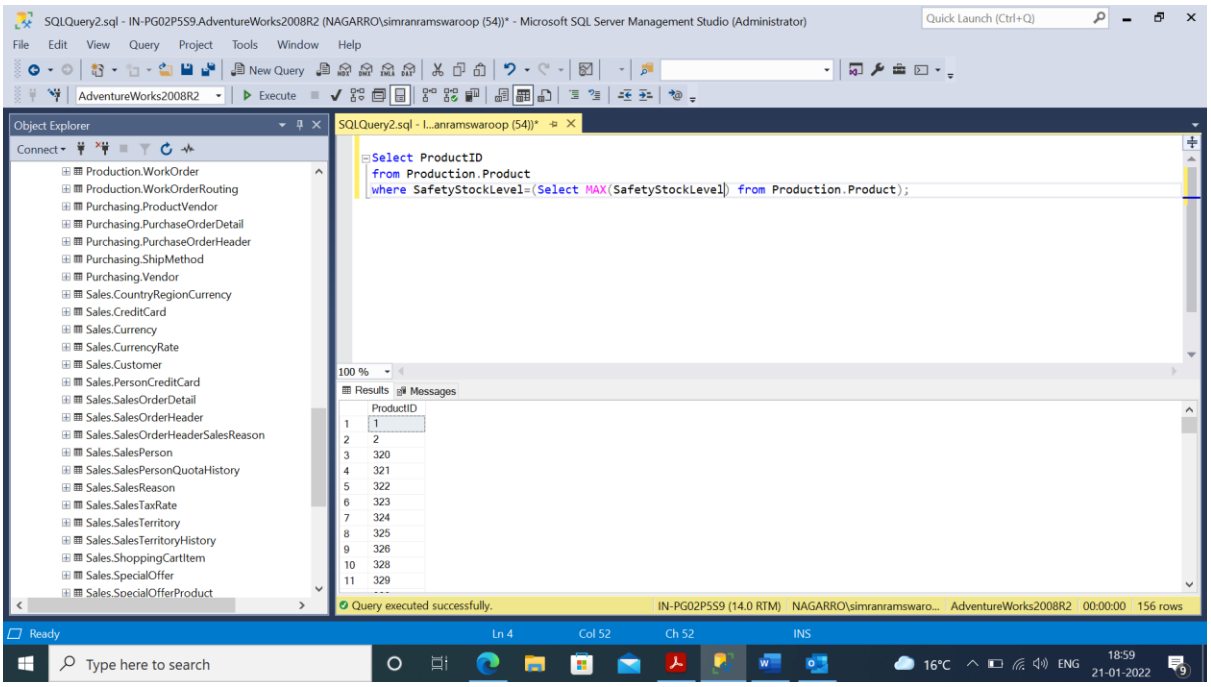
```
Declare @BusinessEntityId int
select @BusinessEntityId = BusinessEntityID from Person.Person
where FirstName = 'Ruth' and LastName = 'Ellerbrock' and PersonType ='EM'

Exec uspGetEmployeeManagers @BusinessEntityId
```

```
   RecursionLevel  BusinessEntityID  FirstName  LastName   OrganizationNode  ManagerFirstName  ManagerLastName
1  0               48                Ruth       Ellerbrock /3/1/3/1/         Andrew            Hill
2  1               47                Andrew     Hill       /3/1/3/           Peter             Krebs
3  2               26                Peter      Krebs      /3/1/             James             Hamilton
4  3               25                James      Hamilton   /3/               Ken               Sánchez
```

Query executed successfully. IN-PG02P5S9 (14.0 RTM) NAGARRO\simranramswaro... AdventureWorks2008R2 00:00:00 4 rows

10.Display the ProductId of the product with the largest stock level.

Query –
Select ProductID
from Production.Product
where SafetyStockLevel=(Select MAX(SafetyStockLevel) from Production.Product);



```
Select ProductID
from Production.Product
where SafetyStockLevel=(Select MAX(SafetyStockLevel) from Production.Product);
```

```
    ProductID
1   1
2   2
3   320
4   321
5   322
6   323
7   324
8   325
9   326
10  328
11  329
```

Query executed successfully. IN-PG02P5S9 (14.0 RTM) NAGARRO\simranramswaro... AdventureWorks2008R2 00:00:00 156 rows

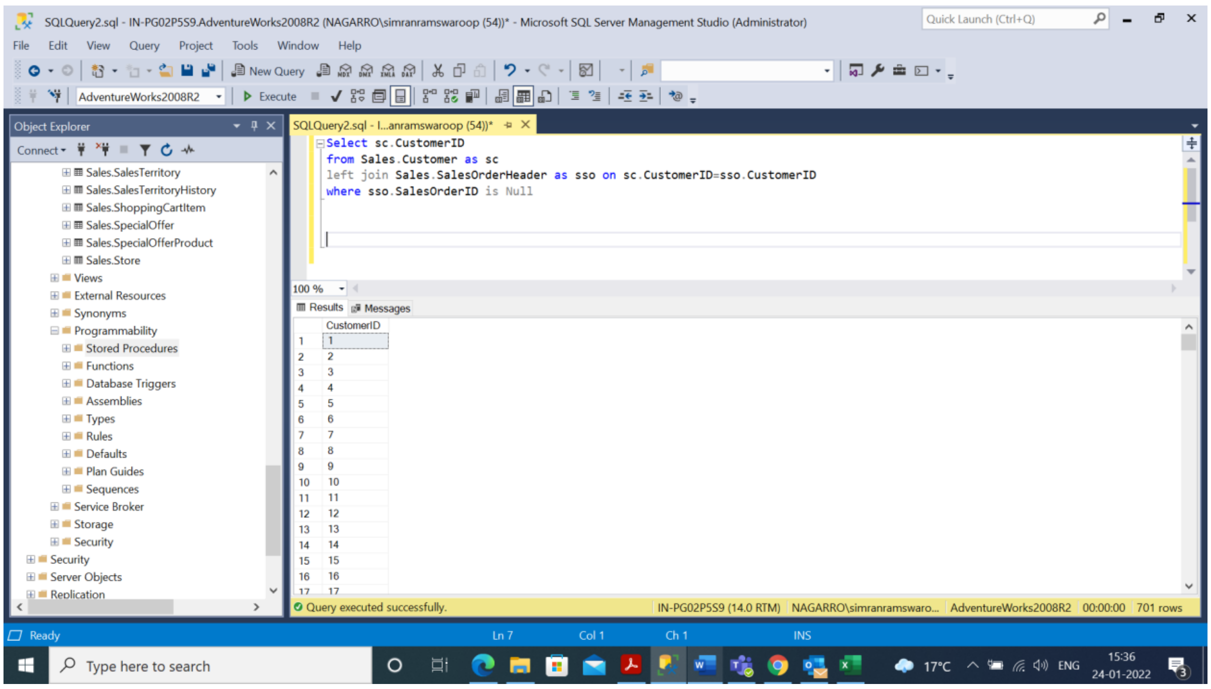
Exercise-2

Write separate queries using a join, a subquery, a CTE, and then an EXISTS to list all AdventureWorks customers who have not placed an order.

Query –

Using Join

```
Select sc.CustomerID
from Sales.Customer as sc
left join Sales.SalesOrderHeader as sso on sc.CustomerID=sso.CustomerID
where sso.SalesOrderID is Null
```



SQLQuery2.sql - IN-PG02P5S9.AdventureWorks2008R2 (NAGARRO\simranramswaroop (54))* - Microsoft SQL Server Management Studio (Administrator)

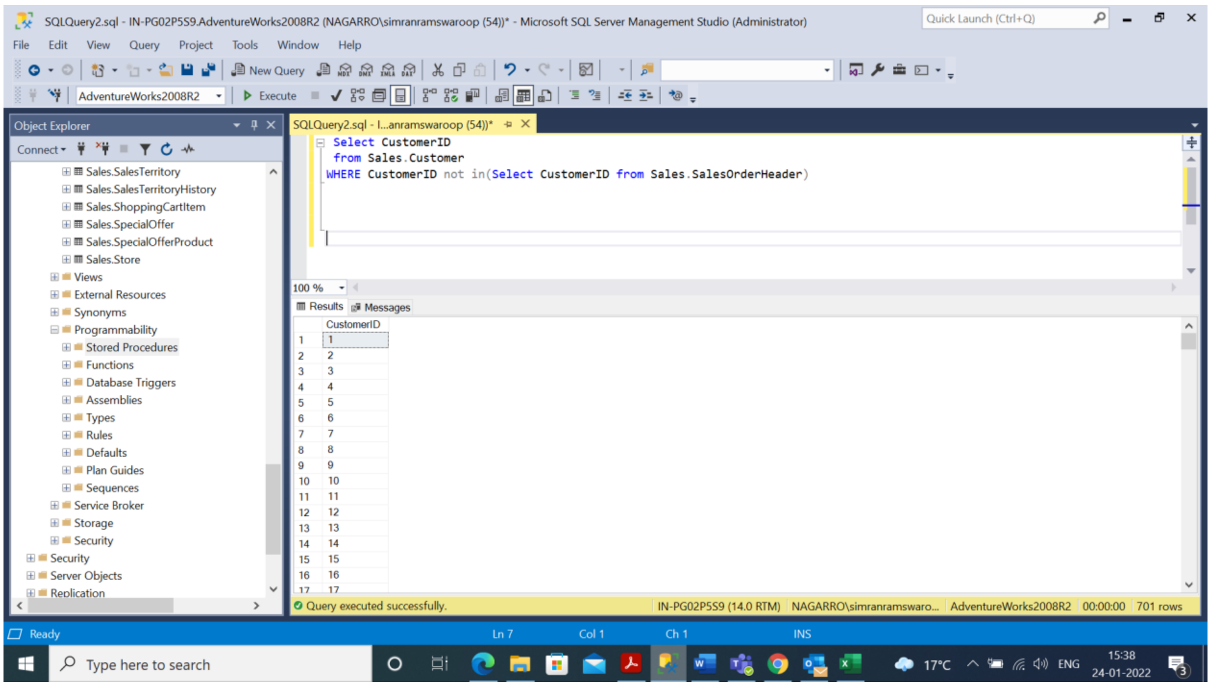
```
Select sc.CustomerID
from Sales.Customer as sc
left join Sales.SalesOrderHeader as sso on sc.CustomerID=sso.CustomerID
where sso.SalesOrderID is Null
```

```
   CustomerID
1  1
2  2
3  3
4  4
5  5
6  6
7  7
8  8
9  9
10 10
11 11
12 12
13 13
14 14
15 15
16 16
17 17
```

Query executed successfully. | IN-PG02P5S9 (14.0 RTM) | NAGARRO\simranramswaro... | AdventureWorks2008R2 | 00:00:00 | 701 rows

Using Subquery-

```
Select CustomerID
from Sales.Customer
WHERE CustomerID not in(Select CustomerID from Sales.SalesOrderHeader)
```



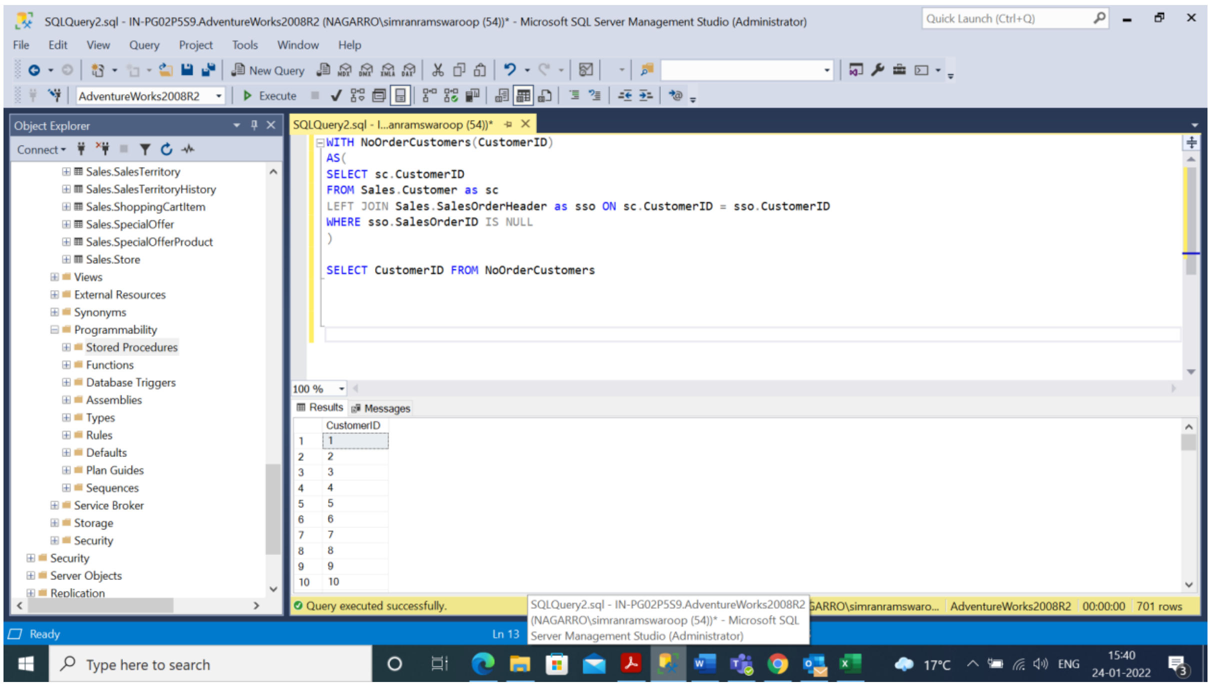
```
Select CustomerID
from Sales.Customer
WHERE CustomerID not in(Select CustomerID from Sales.SalesOrderHeader)
```

```
    CustomerID
1   1
2   2
3   3
4   4
5   5
6   6
7   7
8   8
9   9
10  10
11  11
12  12
13  13
14  14
15  15
16  16
17  17
```

Query executed successfully. IN-PG02P5S9 (14.0 RTM) NAGARRO\simranramswaro... AdventureWorks2008R2 00:00:00 701 rows

<u>**Using CTE**</u>

```
WITH NoOrderCustomers(CustomerID)
AS(
SELECT sc.CustomerID
FROM Sales.Customer as sc
LEFT JOIN Sales.SalesOrderHeader as sso ON sc.CustomerID = sso.CustomerID
WHERE sso.SalesOrderID IS NULL
)

SELECT CustomerID FROM NoOrderCustomers
```



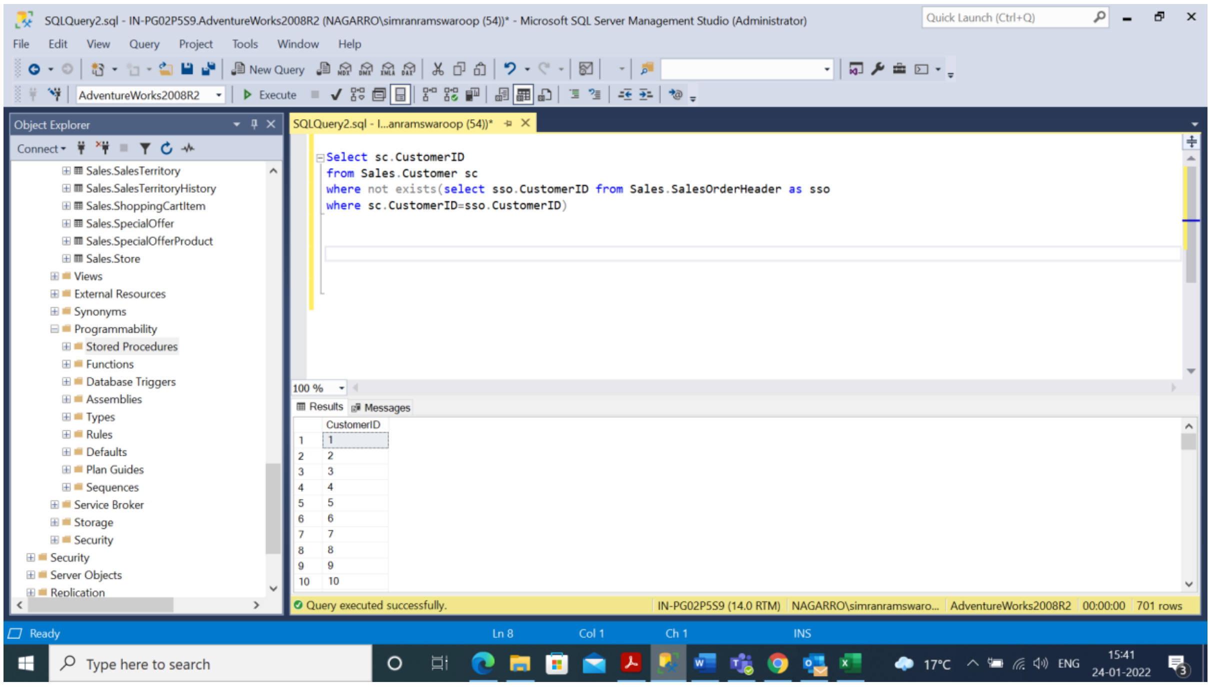
```
WITH NoOrderCustomers(CustomerID)
AS(
SELECT sc.CustomerID
FROM Sales.Customer as sc
LEFT JOIN Sales.SalesOrderHeader as sso ON sc.CustomerID = sso.CustomerID
WHERE sso.SalesOrderID IS NULL
)

SELECT CustomerID FROM NoOrderCustomers
```

```
   CustomerID
1  1
2  2
3  3
4  4
5  5
6  6
7  7
8  8
9  9
10 10
```

```
Query executed successfully.   AdventureWorks2008R2   00:00:00   701 rows
```

Using EXISTS

```
Select sc.CustomerID
from Sales.Customer sc
where not exists(select sso.CustomerID from Sales.SalesOrderHeader as sso
where sc.CustomerID=sso.CustomerID)
```



```
Select sc.CustomerID
from Sales.Customer sc
where not exists(select sso.CustomerID from Sales.SalesOrderHeader as sso
where sc.CustomerID=sso.CustomerID)
```

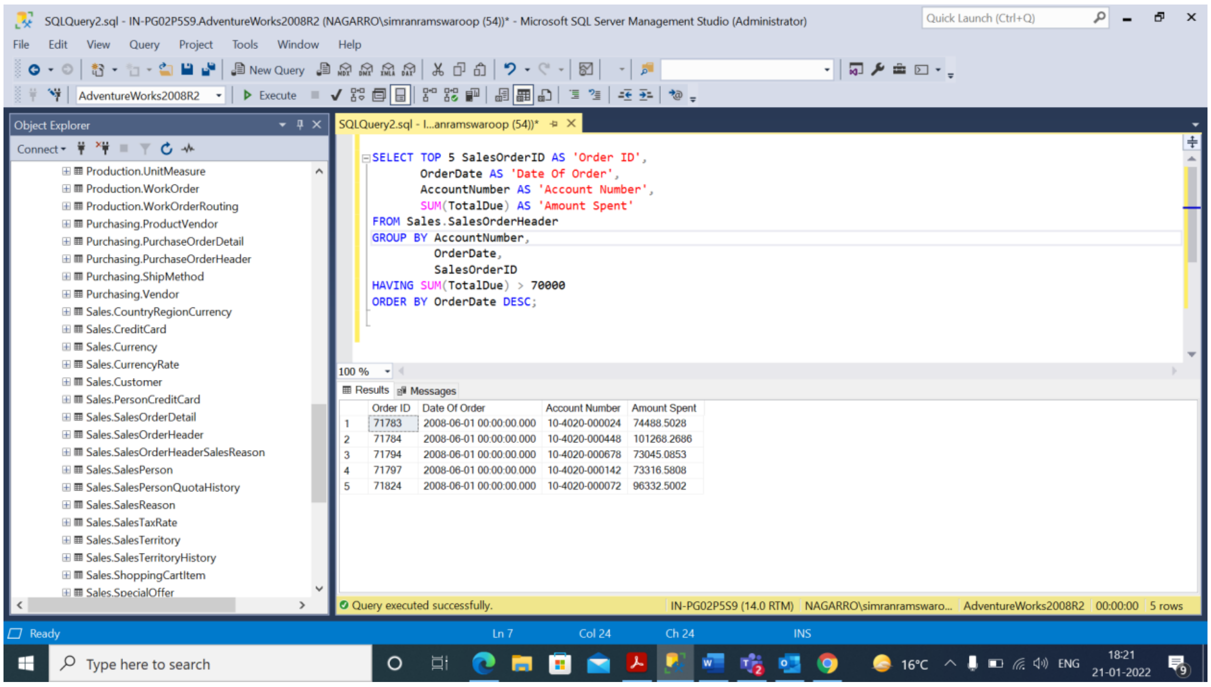
```
   CustomerID
1  1
2  2
3  3
4  4
5  5
6  6
7  7
8  8
9  9
10 10
```

```
Query executed successfully.   IN-PG02P5S9 (14.0 RTM)   NAGARRO\simranramswaro...   AdventureWorks2008R2   00:00:00   701 rows
```

Exercise-3

Show the most recent five orders that were purchased from account numbers that have spent more than $70,000 with AdventureWorks

Query –
```
SELECT TOP 5 SalesOrderID AS 'Order ID',
OrderDate AS 'Date Of Order',
AccountNumber AS 'Account Number',
SUM(TotalDue) AS 'Amount Spent'
FROM Sales.SalesOrderHeader
GROUP BY AccountNumber,
         OrderDate,
         SalesOrderID
HAVING SUM(TotalDue) > 70000
ORDER BY OrderDate DESC;
```



```
SELECT TOP 5 SalesOrderID AS 'Order ID',
       OrderDate AS 'Date Of Order',
       AccountNumber AS 'Account Number',
       SUM(TotalDue) AS 'Amount Spent'
FROM Sales.SalesOrderHeader
GROUP BY AccountNumber,
         OrderDate,
         SalesOrderID
HAVING SUM(TotalDue) > 70000
ORDER BY OrderDate DESC;
```

	Order ID	Date Of Order	Account Number	Amount Spent
1	71783	2008-06-01 00:00:00.000	10-4020-000024	74488.5028
2	71784	2008-06-01 00:00:00.000	10-4020-000448	101268.2686
3	71794	2008-06-01 00:00:00.000	10-4020-000678	73045.0853
4	71797	2008-06-01 00:00:00.000	10-4020-000142	73316.5808
5	71824	2008-06-01 00:00:00.000	10-4020-000072	96332.5002

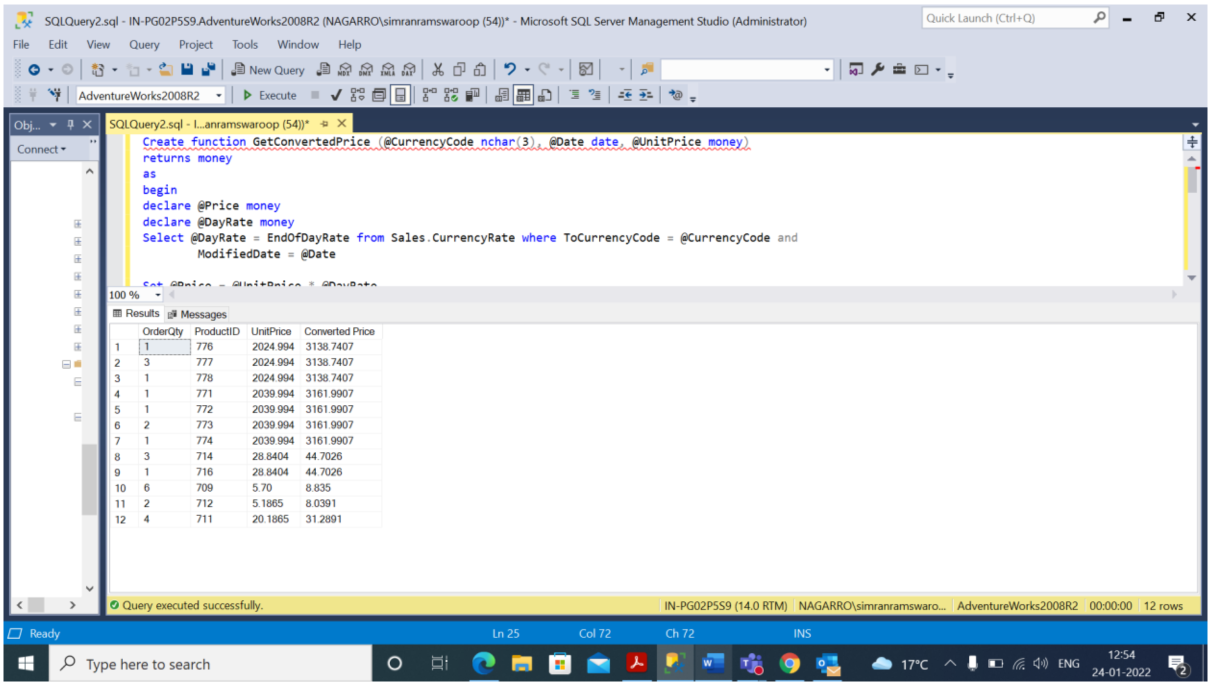
Query executed successfully. IN-PG02P5S9 (14.0 RTM) NAGARRO\simranramswaro... AdventureWorks2008R2 00:00:00 5 rows

Exercise-4

Create a function that takes as inputs a SalesOrderID, a Currency Code, and a date, and returns a table of all the SalesOrderDetail rows for that Sales Order including Quantity, ProductID, UnitPrice, and the unit price converted to the target currency based on the end of day rate for the date provided. Exchange rates can be found in the Sales.CurrencyRate table.

Query –

```sql
Create function GetConvertedPrice (@CurrencyCode nchar(3), @Date date, @UnitPrice
money)
returns money
as
begin
declare @Price money
declare         @DayRate money
Select @DayRate = EndOfDayRate from Sales.CurrencyRate where ToCurrencyCode =
@CurrencyCode and ModifiedDate = @Date
Set @Price = @UnitPrice * @DayRate
return @Price
end

--Main Function
Create Function GetSalesOrderDetail (@SalesOrderID int , @CurrencyCode nchar(3),
@Date datetime)
Returns Table
as
Return (Select OrderQty, ProductID, UnitPrice, dbo.GetConvertedPrice(@CurrencyCode,
@Date, UnitPrice)
as 'Converted Price'from Sales.SalesOrderDetail
where SalesOrderID=@SalesOrderID and ModifiedDate=@Date)

--Calling Function
Select * from dbo.GetSalesOrderDetail(43659, 'AUD', '2005-07-01 00:00:00')
```


SQLQuery2.sql - IN-PG02P5S9.AdventureWorks2008R2 (NAGARRO\simranramswaroop (54))* - Microsoft SQL Server Management Studio (Administrator)

```sql
Create function GetConvertedPrice (@CurrencyCode nchar(3), @Date date, @UnitPrice money)
returns money
as
begin
declare @Price money
declare @DayRate money
Select @DayRate = EndOfDayRate from Sales.CurrencyRate where ToCurrencyCode = @CurrencyCode and
        ModifiedDate = @Date
```

	OrderQty	ProductID	UnitPrice	Converted Price
1	1	776	2024.994	3138.7407
2	3	777	2024.994	3138.7407
3	1	778	2024.994	3138.7407
4	1	771	2039.994	3161.9907
5	1	772	2039.994	3161.9907
6	2	773	2039.994	3161.9907
7	1	774	2039.994	3161.9907
8	3	714	28.8404	44.7026
9	1	716	28.8404	44.7026
10	6	709	5.70	8.835
11	2	712	5.1865	8.0391
12	4	711	20.1865	31.2891

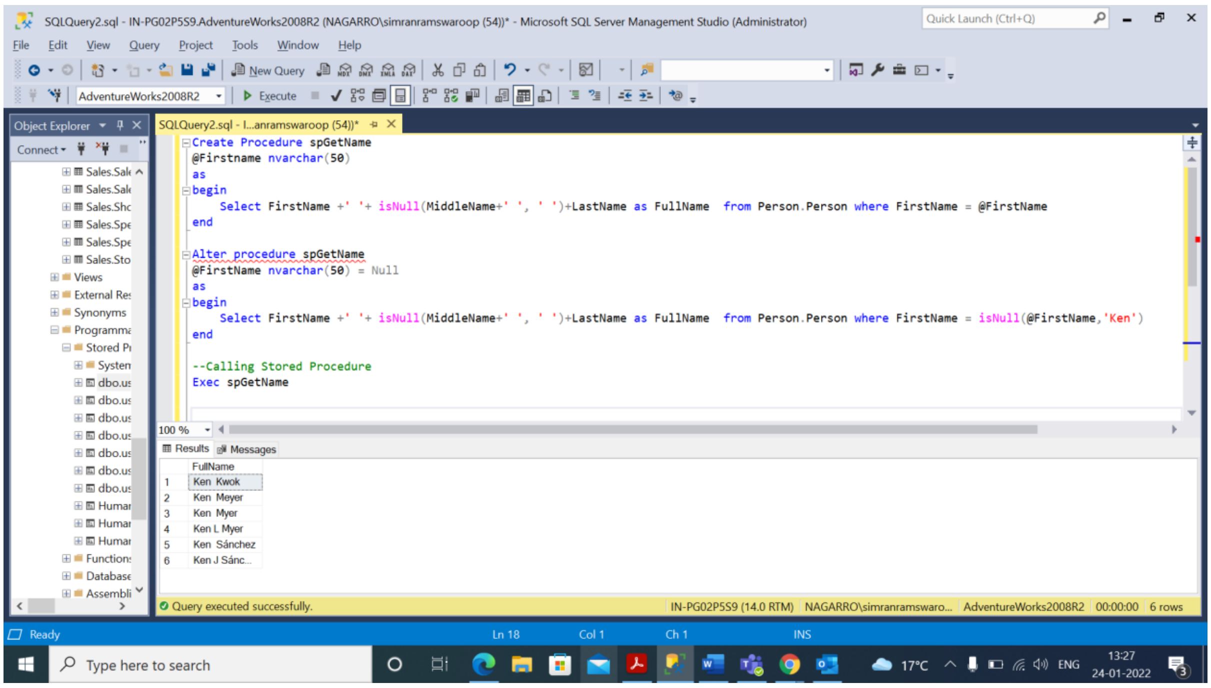
Query executed successfully. IN-PG02P5S9 (14.0 RTM) NAGARRO\simranramswaro... AdventureWorks2008R2 00:00:00 12 rows

<u>Exercise-5</u>

Write a Procedure supplying name information from the **Person.Person** table and accepting a filter for the first name. Alter the above Store Procedure to supply Default Values if user does not enter any value.

Query-

```sql
Create Procedure spGetName
@Firstname nvarchar(50)
as
begin
Select FirstName +' '+ isNull(MiddleName+' ', ' ')+LastName as FullName  from
Person.Person where FirstName = @FirstName
end

Alter procedure spGetName
@FirstName nvarchar(50) = Null
as
begin
Select FirstName +' '+ isNull(MiddleName+' ', ' ')+LastName as FullName  from
Person.Person where FirstName = isNull(@FirstName,'Ken')
end

--Calling Stored Procedure
Exec spGetName
```



SQLQuery2.sql - IN-PG02P5S9.AdventureWorks2008R2 (NAGARRO\simranramswaroop (54))* - Microsoft SQL Server Management Studio (Administrator)

```sql
Create Procedure spGetName
@Firstname nvarchar(50)
as
begin
    Select FirstName +' '+ isNull(MiddleName+' ', ' ')+LastName as FullName  from Person.Person where FirstName = @FirstName
end

Alter procedure spGetName
@FirstName nvarchar(50) = Null
as
begin
    Select FirstName +' '+ isNull(MiddleName+' ', ' ')+LastName as FullName  from Person.Person where FirstName = isNull(@FirstName,'Ken')
end

--Calling Stored Procedure
Exec spGetName
```

	FullName
1	Ken Kwok
2	Ken Meyer
3	Ken Myer
4	Ken L Myer
5	Ken Sánchez
6	Ken J Sánc...

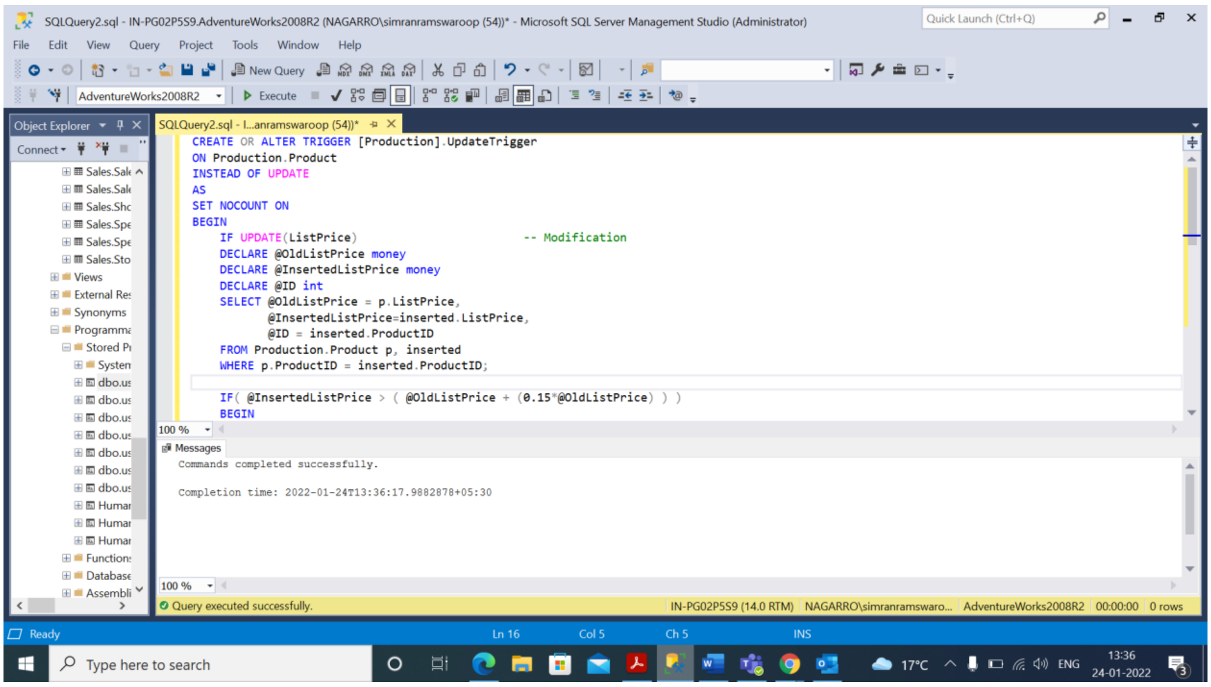
Query executed successfully. | IN-PG02P5S9 (14.0 RTM) | NAGARRO\simranramswaro... | AdventureWorks2008R2 | 00:00:00 | 6 rows

<u>Exercise-6</u>

Write a trigger for the Product table to ensure the list price can never be raised more than 15 Percent in a single change. Modify the above trigger to execute its check code only if the ListPrice column is updated

Query-

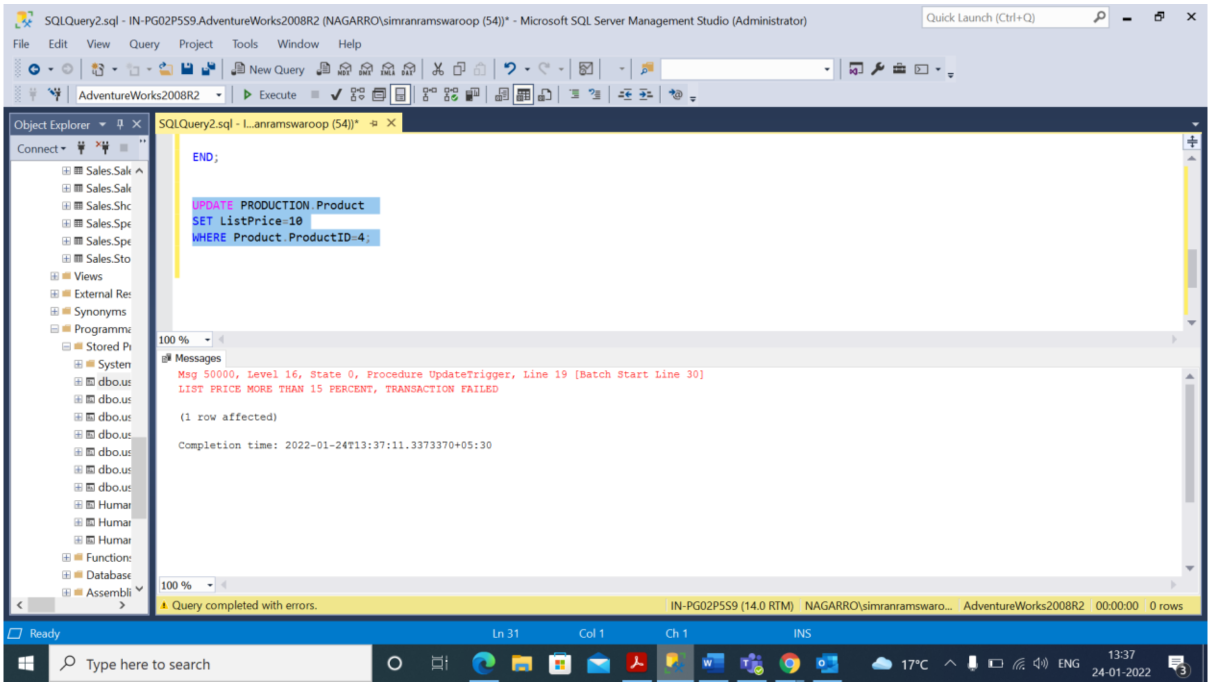
```
CREATE OR ALTER TRIGGER [Production].UpdateTrigger
ON Production.Product
INSTEAD OF UPDATE
AS
SET NOCOUNT ON
BEGIN
        IF UPDATE(ListPrice)                                    -- Modification
        DECLARE @OldListPrice money
        DECLARE @InsertedListPrice money
        DECLARE @ID int
        SELECT @OldListPrice = p.ListPrice,
                  @InsertedListPrice=inserted.ListPrice,
                  @ID = inserted.ProductID
        FROM Production.Product p, inserted
        WHERE p.ProductID = inserted.ProductID;

        IF( @InsertedListPrice > ( @OldListPrice + (0.15*@OldListPrice) ) )
        BEGIN
                RAISERROR('LIST PRICE MORE THAN 15 PERCENT, TRANSACTION FAILED',16,0)
                return
        END
        ELSE
        BEGIN
                Update Production.Product SET ListPrice=@InsertedListPrice
                WHERE Production.Product.ProductID = @ID;
        END

END;


UPDATE PRODUCTION.Product
SET ListPrice=10
WHERE Product.ProductID=4;
```



```
CREATE OR ALTER TRIGGER [Production].UpdateTrigger
ON Production.Product
INSTEAD OF UPDATE
AS
SET NOCOUNT ON
BEGIN
    IF UPDATE(ListPrice)                          -- Modification
    DECLARE @OldListPrice money
    DECLARE @InsertedListPrice money
    DECLARE @ID int
    SELECT @OldListPrice = p.ListPrice,
           @InsertedListPrice=inserted.ListPrice,
           @ID = inserted.ProductID
    FROM Production.Product p, inserted
    WHERE p.ProductID = inserted.ProductID;

    IF( @InsertedListPrice > ( @OldListPrice + (0.15*@OldListPrice) ) )
    BEGIN
```

```
Commands completed successfully.

Completion time: 2022-01-24T13:36:17.9882878+05:30
```

```
Query executed successfully.    IN-PG02P5S9 (14.0 RTM)    NAGARRO\simranramswaro...    AdventureWorks2008R2    00:00:00    0 rows
```



```
END;


UPDATE PRODUCTION.Product
SET ListPrice=10
WHERE Product.ProductID=4;
```

```
Msg 50000, Level 16, State 0, Procedure UpdateTrigger, Line 19 [Batch Start Line 30]
LIST PRICE MORE THAN 15 PERCENT, TRANSACTION FAILED

(1 row affected)

Completion time: 2022-01-24T13:37:11.3373370+05:30
```

```
Query completed with errors.    IN-PG02P5S9 (14.0 RTM)    NAGARRO\simranramswaro...    AdventureWorks2008R2    00:00:00    0 rows
```