Project - Final Exam MCIT

21.12.2023

—

Florent VAGBA
xxxx, Place de la Malicorne
Anjou,Qc,CA, xxxxx
514-970-8323

Introduction

ABC is a fictional Garments company based in Canada, specializing in the design and production of high-quality apparel.

With a focus on contemporary fashion trends, ABC caters to a diverse demographic, offering a wide range of clothing for men, women, and children. The company prides itself on delivering stylish and comfortable garments that align with the Canadian lifestyle. Known for its commitment to sustainable practices, ABC employs eco-friendly manufacturing processes and sources materials responsibly. With a presence in major Canadian cities, ABC aims to provide fashion-forward clothing options while contributing positively to the local and global community.

Please prepare the following Objects in the MCIT database and insert the required number of rows in each table..

Credentials for my Virtual lab

ComputerName:

Username:

Password:

Queries file path :

E:\Florent VAGBA\SQL\Final Exam Project\Project_FinalExam_SQL_MCIT_Florent VAGBA_Final

1. Create a Customers table

Create Customers table and Insert 10 records into Customers table

```
Use MCIT

DROP TABLE IF EXISTS Customers

CREATE TABLE Customers (
        CustomerID INT PRIMARY KEY
        ,FirstName VARCHAR(50)
        ,LastName VARCHAR(50)
        ,Email VARCHAR(100)
        );

INSERT INTO Customers
VALUES
(1, 'John', 'Doe', 'john.doe@gmail.com'),
(2, 'Jane', 'Smith', 'jane.smith@yahoo.com'),
(3, 'Nadreyh', 'NABEHY', 'nadreyh.nabehy@trevor.com'),
(4, 'Amenon', 'DACEEH', 'amenon.daceeh@gnimale.com'),
(5, 'Akaba', 'DJIMON', 'akaba.djimon@hotmail.com'),
(6, 'Okyo', 'ZADY', 'okyo.zady@bobo.com'),
(7, 'Kalie', 'ZOUKOUBY', 'kalie.zoukouby@pinky.com'),
(8, 'Digbehy', 'CAPORAL', 'digbehy.caporal@aol.com'),
(9, 'Gnakale', 'Za', 'gnakale.za@sql.com'),
```

(10, 'Gadega', 'NAWIHO', 'gadega.nawiho@azouakou.com')

Results Messages

	CustomerID	FirstName	LastName	Email
1	1	John	Doe	john.doe@gmail.com
2	2	Jane	Smith	jane.smith@yahoo.com
3	3	Nadreyh	NABEHY	nadreyh.nabehy@trevor.com
4	4	Amenon	DACEEH	amenon.daceeh@gnimale.com
5	5	Akaba	DJIMON	akaba.djimon@hotmail.com
6	6	Okyo	ZADY	okyo.zady@bobo.com
7	7	Kalie	ZOUKOUBY	kalie.zoukouby@pinky.com
8	8	Digbehy	CAPORAL	digbehy.caporal@aol.com
9	9	Gnakale	Za	gnakale.za@sql.com
10	10	Gadega	NAWIHO	gadega.nawiho@azouakou.com

Add 20 more records with similar structure with different domains like @gmail, @outlook

Final query to add 20 more records

The CustomerID will start from 11 and end at 30. We will have a total of 30 records.

```
INSERT INTO Customers (CustomerID, FirstName, LastName, Email)

VALUES

(11, 'Lola', 'BROWN', 'lola.brown@gmail.com'),

(12, 'Max', 'GREEN', 'max.green@outlook.com'),

(13, 'Zara', 'WHITE', 'zara.white@yahoo.com'),

(14, 'Leo', 'BLACK', 'leo.black@hotmail.com'),
```

(15, 'Eva', 'PINK', 'eva.pink@icloud.com'),

(16, 'Noah', 'GRAY', 'noah.gray@aol.com'),

(17, 'Mia', 'RED', 'mia.red@msn.com'),

(18, 'Lucas', 'BLUE', 'lucas.blue@protonmail.com'),

(19, 'Emma', 'PURPLE', 'emma.purple@zoho.com'),

(20, 'Liam', 'ORANGE', 'liam.orange@yandex.com'),

(21, 'Olivia', 'YELLOW', 'olivia.yellow@mail.com'),

(22, 'Ethan', 'GOLD', 'ethan.gold@edu.com'),

(23, 'Sophia', 'SILVER', 'sophia.silver@biz.com'),

(24, 'Mason', 'BRONZE', 'mason.bronze@org.com'),

(25, 'Isabella', 'PEARL', 'isabella.pearl@net.com'),

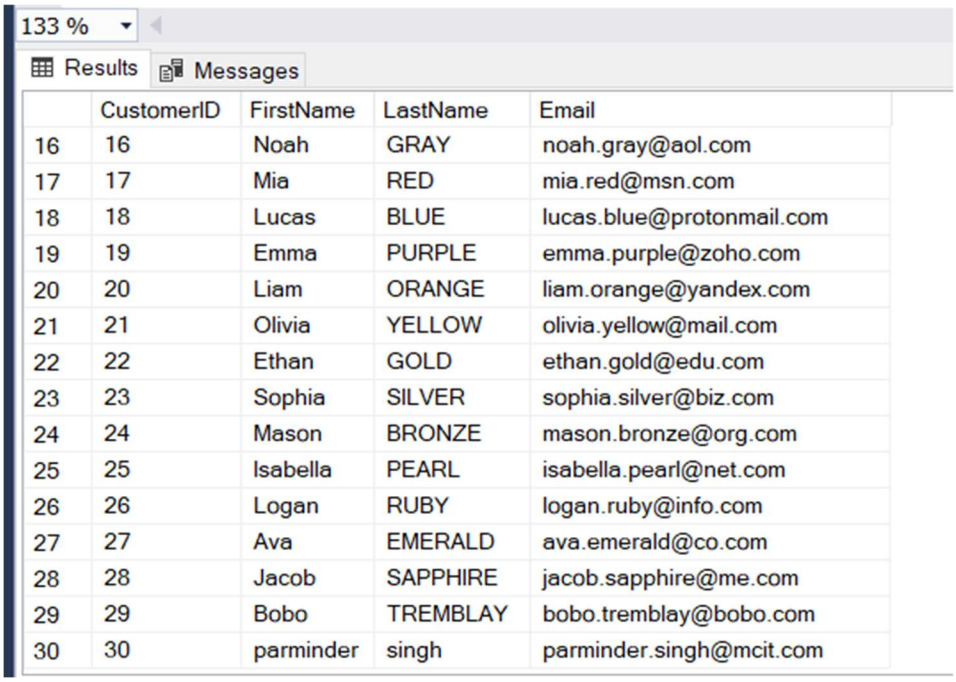
(26, 'Logan', 'RUBY', 'logan.ruby@info.com'),

(27, 'Ava', 'EMERALD', 'ava.emerald@co.com'),

(28, 'Jacob', 'SAPPHIRE', 'jacob.sapphire@me.com'),

(29, 'Bobo', 'TREMBLAY', 'bobo.tremblay@bobo.com'),

(30, 'parminder', 'singh', 'parminder.singh@mcit.com');



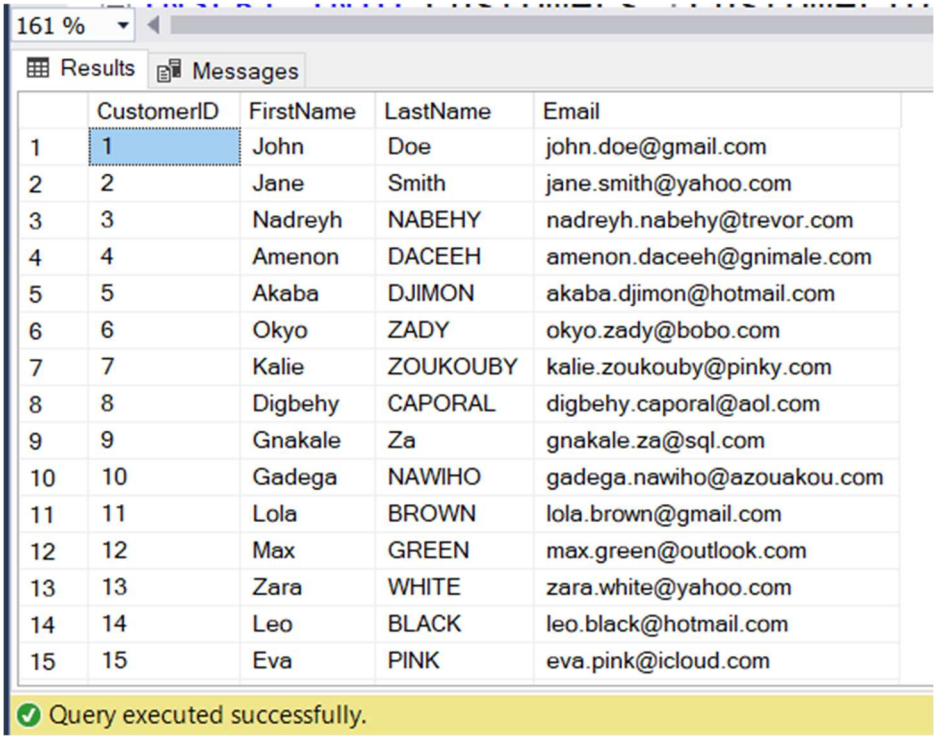
133 %

Results | Messages

	CustomerID	FirstName	LastName	Email
16	16	Noah	GRAY	noah.gray@aol.com
17	17	Mia	RED	mia.red@msn.com
18	18	Lucas	BLUE	lucas.blue@protonmail.com
19	19	Emma	PURPLE	emma.purple@zoho.com
20	20	Liam	ORANGE	liam.orange@yandex.com
21	21	Olivia	YELLOW	olivia.yellow@mail.com
22	22	Ethan	GOLD	ethan.gold@edu.com
23	23	Sophia	SILVER	sophia.silver@biz.com
24	24	Mason	BRONZE	mason.bronze@org.com
25	25	Isabella	PEARL	isabella.pearl@net.com
26	26	Logan	RUBY	logan.ruby@info.com
27	27	Ava	EMERALD	ava.emerald@co.com
28	28	Jacob	SAPPHIRE	jacob.sapphire@me.com
29	29	Bobo	TREMBLAY	bobo.tremblay@bobo.com
30	30	parminder	singh	parminder.singh@mcit.com

(20 rows affected)

Completion time: 2023-12-20T14:52:07.4590730-05:00



161 %

Results | Messages

	CustomerID	FirstName	LastName	Email
1	1	John	Doe	john.doe@gmail.com
2	2	Jane	Smith	jane.smith@yahoo.com
3	3	Nadreyh	NABEHY	nadreyh.nabehy@trevor.com
4	4	Amenon	DACEEH	amenon.daceeh@gnimale.com
5	5	Akaba	DJIMON	akaba.djimon@hotmail.com
6	6	Okyo	ZADY	okyo.zady@bobo.com
7	7	Kalie	ZOUKOUBY	kalie.zoukouby@pinky.com
8	8	Digbehy	CAPORAL	digbehy.caporal@aol.com
9	9	Gnakale	Za	gnakale.za@sql.com
10	10	Gadega	NAWIHO	gadega.nawiho@azouakou.com
11	11	Lola	BROWN	lola.brown@gmail.com
12	12	Max	GREEN	max.green@outlook.com
13	13	Zara	WHITE	zara.white@yahoo.com
14	14	Leo	BLACK	leo.black@hotmail.com
15	15	Eva	PINK	eva.pink@icloud.com

Query executed successfully.

2. Create Suppliers Table

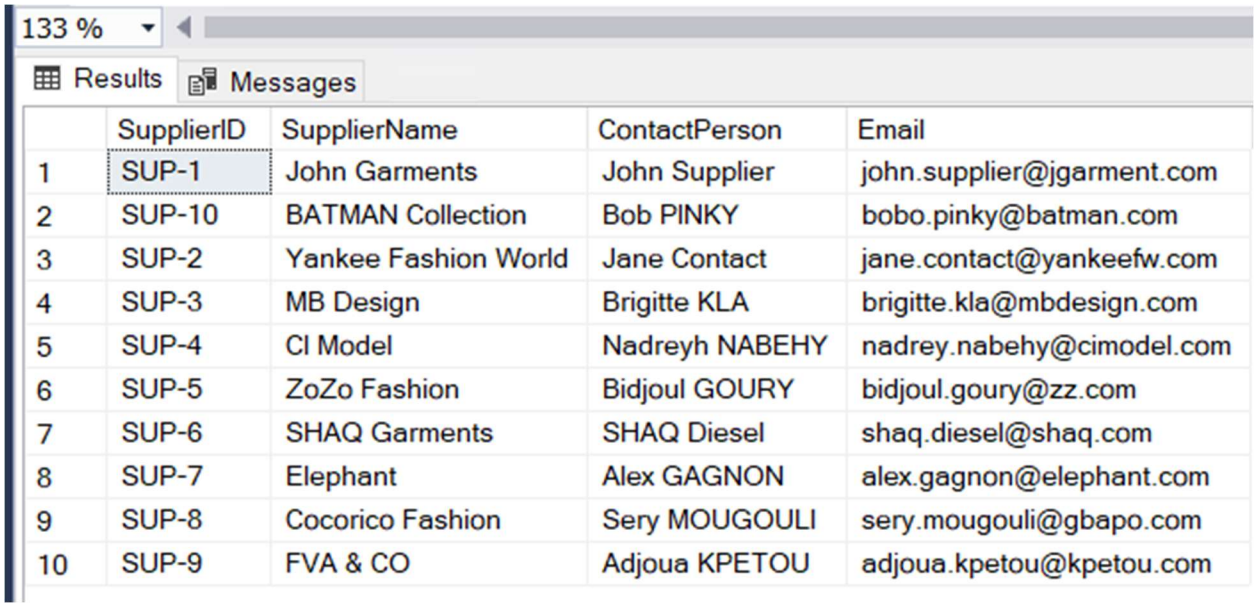
Create the suppliers table and insert 10 records

```
Use MCIT

DROP TABLE IF EXISTS Suppliers

CREATE TABLE Suppliers ( SupplierID VARCHAR(10) PRIMARY KEY

                        ,SupplierName VARCHAR(100)

                        ,ContactPerson VARCHAR(50)

                        ,Email VARCHAR(100)

                    )

INSERT INTO Suppliers

VALUES

('SUP-1', 'John Garments', 'John Supplier', 'john.supplier@jgarment.com'),

('SUP-2', 'Yankee Fashion World', 'Jane Contact', 'jane.contact@yankeefw.com'),

('SUP-3', 'MB Design', 'Brigitte KLA', 'brigitte.kla@mbdesign.com'),

('SUP-4', 'CI Model', 'Nadreyh NABEHY', 'nadrey.nabehy@cimodel.com'),

('SUP-5', 'ZoZo Fashion', 'Bidjoul GOURY', 'bidjoul.goury@zz.com'),

('SUP-6', 'SHAQ Garments', 'SHAQ Diesel', 'shaq.diesel@shaq.com'),

('SUP-7', 'Elephant', 'Alex GAGNON', 'alex.gagnon@elephant.com'),
```

('SUP-8', 'Cocorico Fashion', 'Sery MOUGOULI', 'sery.mougouli@gbapo.com'),

('SUP-9', 'FVA & CO', 'Adjoua KPETOU ', 'adjoua.kpetou@kpetou.com'),

('SUP-10', 'BATMAN Collection', 'Bob PINKY', 'bobo.pinky@batman.com');



133 %

Results | Messages

	SupplierID	SupplierName	ContactPerson	Email
1	SUP-1	John Garments	John Supplier	john.supplier@jgarment.com
2	SUP-10	BATMAN Collection	Bob PINKY	bobo.pinky@batman.com
3	SUP-2	Yankee Fashion World	Jane Contact	jane.contact@yankeefw.com
4	SUP-3	MB Design	Brigitte KLA	brigitte.kla@mbdesign.com
5	SUP-4	CI Model	Nadreyh NABEHY	nadrey.nabehy@cimodel.com
6	SUP-5	ZoZo Fashion	Bidjoul GOURY	bidjoul.goury@zz.com
7	SUP-6	SHAQ Garments	SHAQ Diesel	shaq.diesel@shaq.com
8	SUP-7	Elephant	Alex GAGNON	alex.gagnon@elephant.com
9	SUP-8	Cocorico Fashion	Sery MOUGOULI	sery.mougouli@gbapo.com
10	SUP-9	FVA & CO	Adjoua KPETOU	adjoua.kpetou@kpetou.com

(10 rows affected)

(10 rows affected)

Completion time: 2023-12-20T15:02:40.9420151-05:00

3. Create Product table

```
CREATE TABLE Product (
        PID VARCHAR(10) PRIMARY KEY
        ,PName VARCHAR(100)
        ,Size VARCHAR(10) -- Sizes are S,M,L,X-L,XXX-L,X4-L,X5-L
        ,Price DECIMAL(8, 2) -- 1.50 , 2.00 etc
        );
```

-- Insert 20 records into Product Tables and the Products must belong to a classification (Category)

- Tops: Shirt, blouse, t-shirt, sweater, jacket, tank top, cardigan, blazer, bustier, camisole, cape, corset, etc.
- Bottoms: Pant, short, skirt, trouser, jeans, Jumpsuit, legging, etc.
- Dresses: It is a one-piece garment that cover both the upper and lower body

etc

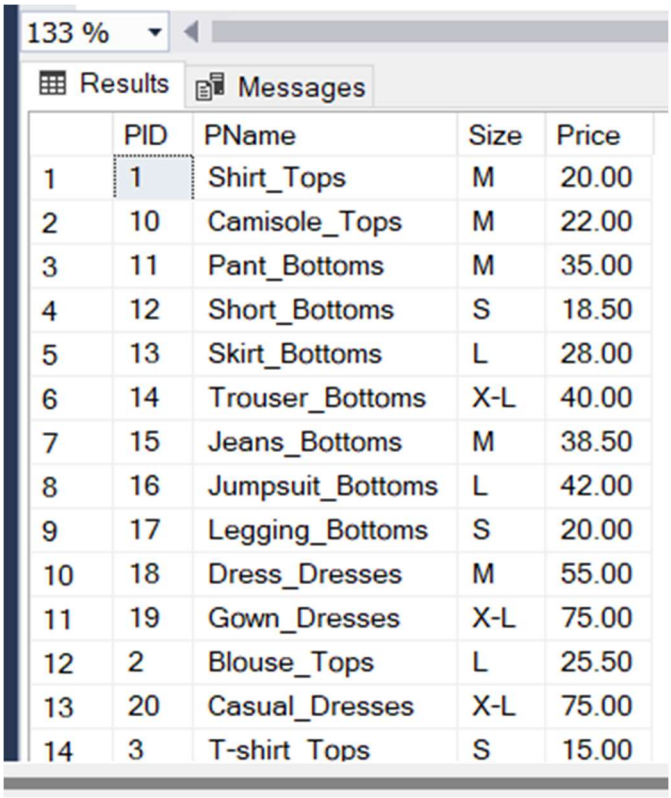
--ProductName must contain ProductName_Category For example Shirt_Tops , Pan_Bottoms

INSERT INTO Products (PName, Size, Price)

Final query

```
        Use MCIT

        DROP TABLE IF EXISTS Product

        CREATE TABLE Product (
                PID VARCHAR(10) PRIMARY KEY
```

```
    ,PName VARCHAR(100)
    ,Size VARCHAR(10) -- Sizes are S,M,L,X-L,XXX-L,4X-L,5X-L
    Price DECIMAL(8, 2) -- 1.50 , 2.00 etc
    );

INSERT INTO Product (PName, Size, Price)
VALUES
    (1,'Shirt_Tops', 'M', 20.00),
    (2,'Blouse_Tops', 'L', 25.50),
    (3,'T-shirt_Tops', 'S', 15.00),
    (4,'Sweater_Tops', 'X-L', 30.00),
    (5,'Jacket_Tops', 'M', 45.00),
    (6,'Tank Top_Tops', 'S', 12.50),
    (7,'Cardigan_Tops', 'L', 35.00),
    (8,'Blazer_Tops', 'X-L', 50.00),
    (9,'Bustier_Tops', 'S', 18.00),
    (10,'Camisole_Tops', 'M', 22.00),
    (11,'Pant_Bottoms', 'M', 35.00),
    (12,'Short_Bottoms', 'S', 18.50),
    (13,'Skirt_Bottoms', 'L', 28.00),
    (14,'Trouser_Bottoms', 'X-L', 40.00),
    (15,'Jeans_Bottoms', 'M', 38.50),
    (16,'Jumpsuit_Bottoms', 'L', 42.00),
    (17,'Legging_Bottoms', 'S', 20.00),
    (18,'Dress_Dresses', 'M', 55.00),
    (19,'Gown_Dresses', 'X-L', 75.00);
    (20,'Casual_Dresses', 'X-L', 75.00);
```



	PID	PName	Size	Price
1	1	Shirt_Tops	M	20.00
2	10	Camisole_Tops	M	22.00
3	11	Pant_Bottoms	M	35.00
4	12	Short_Bottoms	S	18.50
5	13	Skirt_Bottoms	L	28.00
6	14	Trouser_Bottoms	X-L	40.00
7	15	Jeans_Bottoms	M	38.50
8	16	Jumpsuit_Bottoms	L	42.00
9	17	Legging_Bottoms	S	20.00
10	18	Dress_Dresses	M	55.00
11	19	Gown_Dresses	X-L	75.00
12	2	Blouse_Tops	L	25.50
13	20	Casual_Dresses	X-L	75.00
14	3	T-shirt_Tops	S	15.00

(20 rows affected)

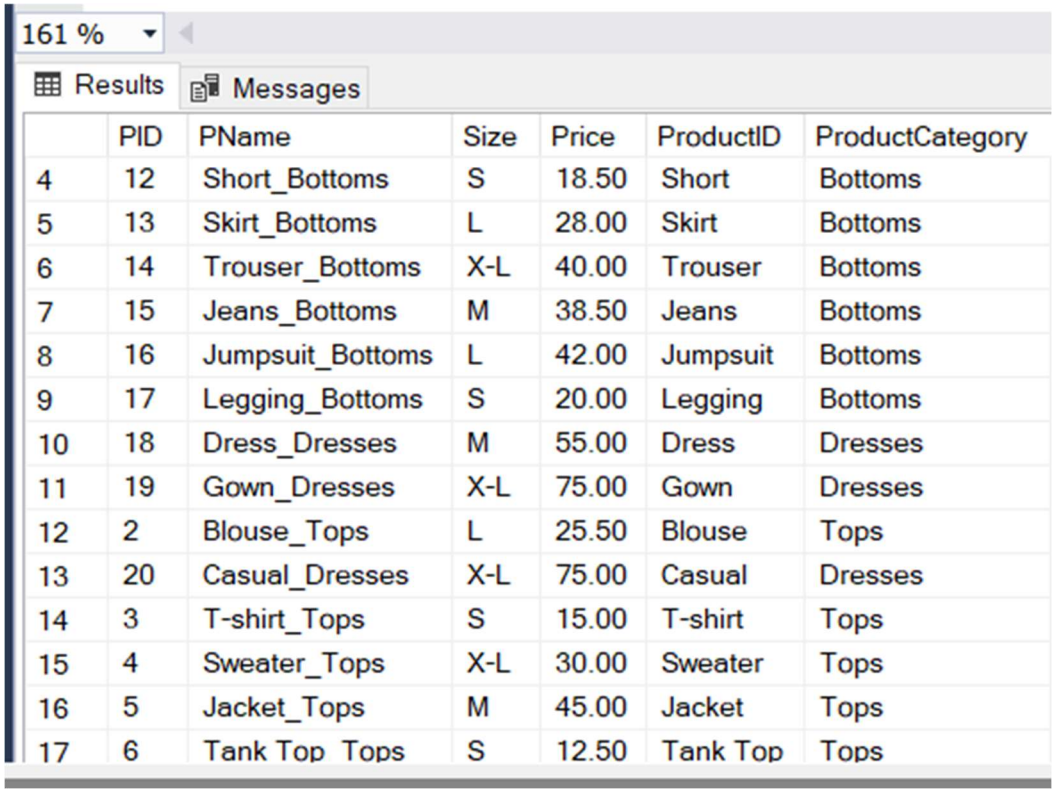
Completion time: 2023-12-20T15:08:51.2539199-05:00

Complete Product table as Products

/*** Creating a complete Product table with a split Pname in ProductCategory and ProductID (Product name). I added the Quantity here as It was requested in the final table***/

```
SELECT TOP (1000) [PID]
```

```
,[PName]
,[Size]
,[Price]
,Left([PName],CHARINDEX('_',[PName]) - 1) As ProductID
,RIGHT([PName], LEN([PName]) - CHARINDEX('_',[PName]) ) As ProductCategory
INTO Products
From [dbo].[Product]
```



161 %

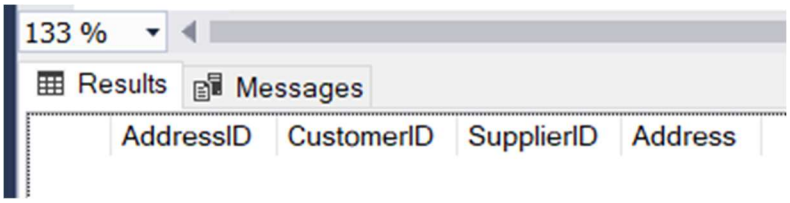
Results | Messages

	PID	PName	Size	Price	ProductID	ProductCategory
4	12	Short_Bottoms	S	18.50	Short	Bottoms
5	13	Skirt_Bottoms	L	28.00	Skirt	Bottoms
6	14	Trouser_Bottoms	X-L	40.00	Trouser	Bottoms
7	15	Jeans_Bottoms	M	38.50	Jeans	Bottoms
8	16	Jumpsuit_Bottoms	L	42.00	Jumpsuit	Bottoms
9	17	Legging_Bottoms	S	20.00	Legging	Bottoms
10	18	Dress_Dresses	M	55.00	Dress	Dresses
11	19	Gown_Dresses	X-L	75.00	Gown	Dresses
12	2	Blouse_Tops	L	25.50	Blouse	Tops
13	20	Casual_Dresses	X-L	75.00	Casual	Dresses
14	3	T-shirt_Tops	S	15.00	T-shirt	Tops
15	4	Sweater_Tops	X-L	30.00	Sweater	Tops
16	5	Jacket_Tops	M	45.00	Jacket	Tops
17	6	Tank Top_Tops	S	12.50	Tank Top	Tops

4. Create Address table

Create Address table

```
Use MCIT

DROP TABLE IF EXISTS [Address]

CREATE TABLE Address (
        AddressID INT PRIMARY KEY
        ,CustomerID INT
        ,SupplierID INT
        ,[Address] VARCHAR(200) ,--Address must consists of HouseNumber,City,State
);
```



Commands completed successfully.

Completion time: 2023-12-20T15:13:02.8896672-05:00

Insert addresses for customers and suppliers

```
INSERT INTO Address (AddressID, CustomerID, SupplierID, Address)

VALUES

(1, 1, NULL, '1234, Toronto, Ontario'), -- This means that we are inserting the CustomerAddress because SupplierId is NULL

(2, NULL, '1', '1234, Montreal, Quebec'), -- This means that we are inserting the SupplierAddress because CustomerID is NULL

(3, 2, NULL, '5678, Vancouver, British Columbia'),

(4, NULL, '2', '5678, Calgary, Alberta'),

(5, 3, NULL, '91011, Edmonton, Alberta'),

(6, NULL, '3', '91011, Ottawa, Ontario'),

(7, 4, NULL, '1213, Halifax, Nova Scotia'),

(8, NULL, '4', '1213, Quebec City, Quebec'),

(9, 5, NULL, '1415, Winnipeg, Manitoba'),

(10, NULL, '5', '1415, Saskatoon, Saskatchewan');
```

133 %

Results | Messages

	AddressID	CustomerID	SupplierID	Address
1	1	1	NULL	1234, Toronto, Ontario
2	2	NULL	1	1234, Montreal, Quebec
3	3	2	NULL	5678, Vancouver, British Columbia
4	4	NULL	2	5678, Calgary, Alberta
5	5	3	NULL	91011, Edmonton, Alberta
6	6	NULL	3	91011, Ottawa, Ontario
7	7	4	NULL	1213, Halifax, Nova Scotia
8	8	NULL	4	1213, Quebec City, Quebec
9	9	5	NULL	1415, Winnipeg, Manitoba
10	10	NULL	5	1415, Saskatoon, Saskatchewan

Address table splitting

To make it manageable, I decide to create two different address tables:

Customers address table

```
Use MCIT

DROP TABLE IF EXISTS CustomersAddress

SELECT TOP (1000) [AddressID]

                  ,[CustomerID]

                  ,[Address]

            Into CustomersAddress

  FROM [MCIT].[dbo].[Address]

Where SupplierID is null
```

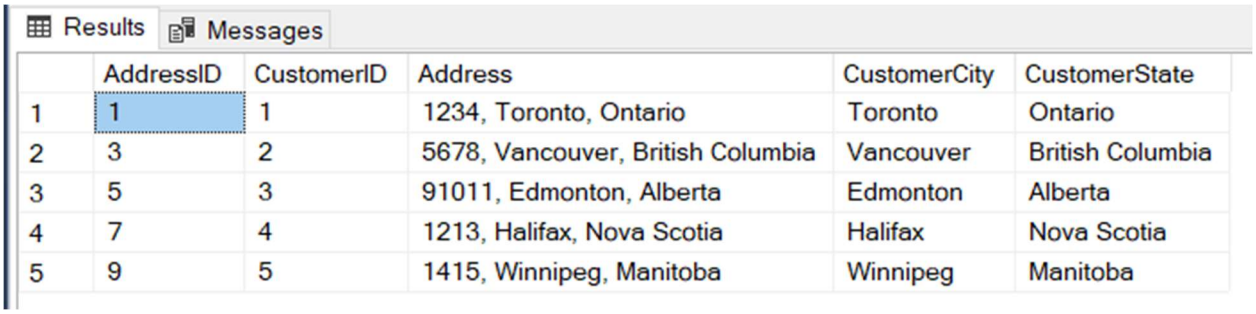
Complete Customers address table

```
 SELECT TOP (1000)

          [AddressID]

          ,[CustomerID]

          ,[Address]

     ,SUBSTRING([Address], CHARINDEX(',', [Address]) + 2, CHARINDEX(',', [Address],
CHARINDEX(',', [Address]) + 1) - CHARINDEX(',', [Address]) - 2) AS CustomerCity
```

```
    ,SUBSTRING([Address], CHARINDEX(',', [Address], CHARINDEX(',', [Address]) + 1) + 2,
LEN([Address])) AS CustomerState

 INTO Customers_Address

 FROM [MCIT].[dbo].[CustomerAddress]
```

Suppliers address table

```
SELECT TOP (1000) [AddressID]

                ,[SupplierID]

                ,[Address]

        Into CustomersAddress

 FROM [MCIT].[dbo].[Address]

Where CustomerID is null
```



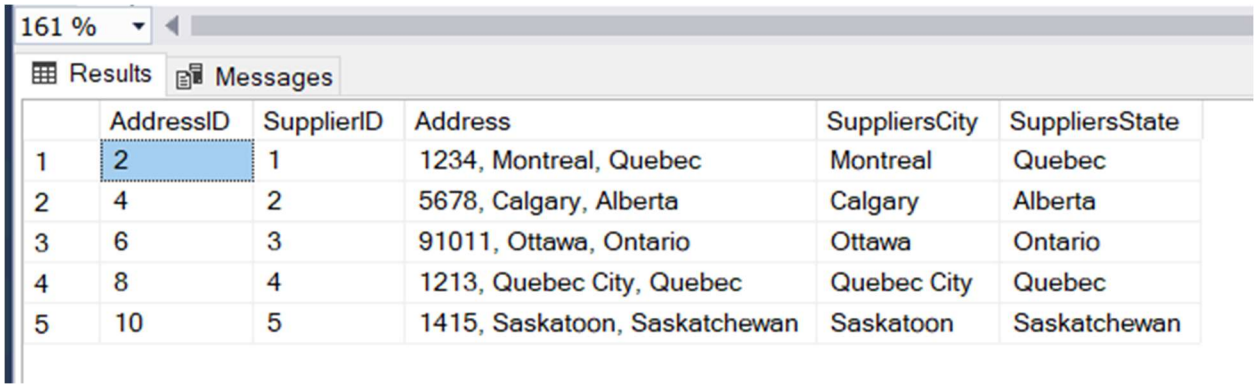
Results | Messages

	AddressID	CustomerID	Address	CustomerCity	CustomerState
1	1	1	1234, Toronto, Ontario	Toronto	Ontario
2	3	2	5678, Vancouver, British Columbia	Vancouver	British Columbia
3	5	3	91011, Edmonton, Alberta	Edmonton	Alberta
4	7	4	1213, Halifax, Nova Scotia	Halifax	Nova Scotia
5	9	5	1415, Winnipeg, Manitoba	Winnipeg	Manitoba

Complete Suppliers address table

```
SELECT TOP (1000) [AddressID]

                ,[SupplierID]

                ,[Address]
```

```
    ,SUBSTRING([Address], CHARINDEX(',', [Address]) + 2, CHARINDEX(',', [Address],
CHARINDEX(',', [Address]) + 1) - CHARINDEX(',', [Address]) - 2) AS SuppliersCity

    ,SUBSTRING([Address], CHARINDEX(',', [Address], CHARINDEX(',', [Address]) + 1) + 2,
LEN([Address])) AS SuppliersState

  INTO Suppliers_Address

  FROM [MCIT].[dbo].[SuppliersAddress]
```



161 %

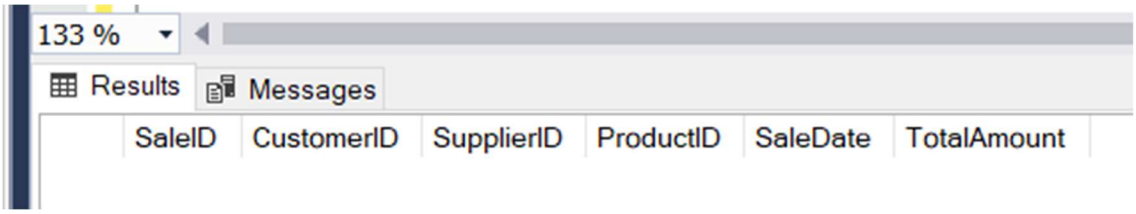
Results | Messages

	AddressID	SupplierID	Address	SuppliersCity	SuppliersState
1	2	1	1234, Montreal, Quebec	Montreal	Quebec
2	4	2	5678, Calgary, Alberta	Calgary	Alberta
3	6	3	91011, Ottawa, Ontario	Ottawa	Ontario
4	8	4	1213, Quebec City, Quebec	Quebec City	Quebec
5	10	5	1415, Saskatoon, Saskatchewan	Saskatoon	Saskatchewan

5. Create Sales table

Create the table Sales

```
Use MCIT

DROP TABLE IF EXISTS Sales

CREATE TABLE Sales (
        SaleID INT PRIMARY KEY
        ,CustomerID INT
        ,SupplierID INT
        ,ProductID VARCHAR(50)
        ,SaleDate DATE
        ,TotalAmount DECIMAL(10, 2)
```



Commands completed successfully.

Completion time: 2023-12-20T15:15:31.9709615-05:00

Insert 50 records into Sales table

```
INSERT INTO Sales (SaleID, CustomerID, SupplierID,ProductID, SaleDate, TotalAmount)

VALUES

(1,1,1,'Pant','2022-01-11',150),

(2,1,1,'Pant','2022-01-12',200.5),

(3,2,3,'Shirt','2022-02-02',100),
```

(4,2,3,'Blouse','2022-02-03',120),

(5,3,5,'Sweater','2022-02-04',80),

(6,3,5,'Jacket','2022-02-08',250),

(7,4,7,'Skirt','2022-02-09',90),

(8,4,7,'Jeans','2022-03-10',140),

(9,5,9,'Dress','2022-03-23',180),

(10,5,9,'Dress','2022-04-14',200),

(11,6,2,'T-shirt','2022-05-08',50),

(12,6,2,'Tank top','2022-05-27',40),

(13,7,4,'Cardigan','2022-05-29',70),

(14,7,4,'Blazer','2022-06-01',150),

(15,8,6,'Short','2022-06-22',60),

(16,8,6,'Trouser','2022-12-05',100),

(17,9,8,'Jumpsuit','2022-12-16',160),

(18,9,8,'Legging','2022-12-18',50),

(19,10,10,'Bustier','2022-12-23',120),

(20,10,10,'Camisole','2022-12-24',80),

(21,11,1,'Pant','2023-01-07',140),

(22,11,1,'Pant','2023-01-07',190),

(23,12,3,'Shirt','2023-01-08',90),

(24,12,3,'Blouse','2023-01-09',110),

(25,13,5,'Sweater','2023-01-16',75),

(26,13,5,'Jacket','2023-01-21',240),

(27,14,7,'Skirt','2023-01-26',85),

(28,14,7,'Jeans','2023-01-27',130),

(29,15,9,'Dress','2023-01-28',170),

(30,15,9,'Dress','2023-01-29',190),

(31,16,2,'T-shirt','2023-01-30',45),

(32,16,2,'Tank top','2023-01-31',35),

(33,17,4,'Cardigan','2023-02-01',65),

(34,17,4,'Blazer','2023-02-03',140),

(35,18,6,'Short','2023-02-04',55),

(36,18,6,'Trouser','2023-03-08',95),

(37,19,8,'Jumpsuit','2023-03-12',155),

(38,19,8,'Legging','2023-04-07',45),

(39,20,10,'Bustier','2023-04-19',115),

(40,20,10,'Camisole','2023-05-11',75),

(41,21,1,'Pant','2023-05-13',130),

(42,21,1,'Pant','2023-05-15',180),

(43,22,3,'Shirt','2023-06-01',80),

(44,22,3,'Blouse','2023-06-02',100),

(45,23,5,'Sweater','2023-06-22',70),

(46,23,5,'Jacket','2023-07-15',230),

(47,24,7,'Skirt','2023-07-22',80),

(48,24,7,'Jeans','2023-12-17',120),

(49,25,9,'Dress','2023-12-19',160),

(50,25,9,'Dress','2023-12-23',180)

(50 rows affected)

Completion time: 2023-12-20T15:17:25.1947425-05:00

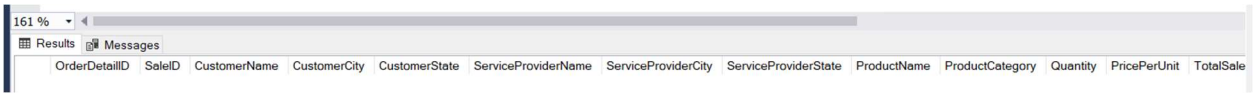
Results | Messages

	SaleID	CustomerID	SupplierID	ProductID	SaleDate	TotalAmount
1	1	1	1	Pant	2023-01-01	150.00
2	2	1	1	Pant	2023-01-02	200.50
3	3	2	3	Shirt	2023-01-03	100.00
4	4	2	3	Blouse	2023-01-04	120.00
5	5	3	5	Sweater	2023-01-05	80.00
6	6	3	5	Jacket	2023-01-06	250.00
7	7	4	7	Skirt	2023-01-07	90.00
8	8	4	7	Jeans	2023-01-08	140.00
9	9	5	9	Dress	2023-01-09	180.00
10	10	5	9	Dress	2023-01-10	200.00
11	11	6	2	T-shirt	2023-01-11	50.00
12	12	6	2	Tank top	2023-01-12	40.00
13	13	7	4	Cardigan	2023-01-13	70.00
14	14	7	4	Blazer	2023-01-14	150.00

6. SalesOrderDetail

Create SalesOrderDetail table

```
CREATE TABLE SalesOrderDetail (
      OrderDetailID INT PRIMARY KEY
      ,SaleID INT
      ,CustomerName VARCHAR(50)
      ,CustomerCity VARCHAR(50)
      ,CustomerState VARCHAR(50)
      ,ServiceProviderName VARCHAR(50)
      ,ServiceProviderCity VARCHAR(50)
      ,ServiceProviderState VARCHAR(50)
      ,ProductName VARCHAR(50)       ,--ProductName first Part
      ,ProductCategory VARCHAR(50)     ,--ProductName LastPart after _
      ,Quantity INT
      ,PricePerUnit DECIMAL(8, 2)     ,--This will come From ProductTable
      ,TotalSales DECIMAL(19, 2) - This IS Quantity * PricePerUNIT
      FOREIGN KEY (SaleID) REFERENCES Sales(SaleID)
      );
```



Commands completed successfully.

Completion time: 2023-12-20T15:22:34.1772085-05:00

Where and how can I find each information of the table SalesOrderDetail?

1. **OrderDetailID INT PRIMARY KEY :**

    Generated and unique integer

    Going with declaring the key with identity(1,1)

Note : We should not have more than 50 records

```
OrderDetailID INT PRIMARY KEY Identity(1,1)
```

2. **SaleID INT**

Foreign Key from Sales Table

Will be entered using a Join SQL query.

3. **CustomerName VARCHAR(50)**

    Composed with the FirstName and LastName of the Customers table.

    Concat(FirstsName + ‘ ‘ + LastName)

4. **CustomerCity VARCHAR(50)**

    Encapsulate in the [Address] VARCHAR(200) of the address table.

Retrieve the Cities from [Address]

This is the address format: '5678, **Vancouver** , British Columbia'

Here we need to find “Vancouver”. The cities' length is not always the same. The street number length is not always the same either.The only thing all the cities have in common, they are all between the first and second comma.

We need to find those and extract the values between them

Find the first comma :

```
CHARINDEX(',', [Address])
```

Find the Start of the City:

```
CHARINDEX(',', [Address]) + 1  – Because of the space
```

Find the second comma :

We have to use a nested CHARINDEX() function that will start from the first comma and retrieve the first comma starting from the first comma already found. That will lead to the second comma.

```
CHARINDEX(',', [Address], CHARINDEX(',', [Address]) + 1)
```

Retrieving the CustomerCity

```
SELECT SUBSTRING([Address], CHARINDEX(',', [Address]) + 2, CHARINDEX(',', [Address], CHARINDEX(',', [Address]) + 1) - CHARINDEX(',', [Address]) - 2) AS CustomerCity
FROM [dbo].[CustomerAddress];
```

5. `CustomerState VARCHAR(50)`

Retrieving the CustomerState from the CustomerAddress table

```
SELECT SUBSTRING([Address], CHARINDEX(',', [Address], CHARINDEX(',', [Address]) + 1) + 2,
LEN([Address])) AS CustomerState
FROM [dbo].[CustomerAddress];
```

6. **ServiceProviderCity VARCHAR(50)**

   This is supplier City

Retrieving the SupplierCity from the SuppliersAddress table

```
SELECT SUBSTRING([Address], CHARINDEX(',', [Address]) + 2, CHARINDEX(',', [Address],
CHARINDEX(',', [Address]) + 1) - CHARINDEX(',', [Address]) - 2) AS SuppliersCity
FROM [dbo].[SuppliersAddress] ;
```

7. **ServiceProviderState VARCHAR(50)**

   This is supplier State

Retrieving the SupplierState from the SuppliersAddress table

```
SELECT  SUBSTRING([Address], CHARINDEX(',', [Address], CHARINDEX(',', [Address]) + 1) + 2,
LEN([Address])) AS SuppliersState
FROM [dbo].[SuppliersAddress] ;
```

8. **ProductName VARCHAR(50)** **,--ProductName first Part**

The product name is as follows : Shirt_Tops

We need to extract first part of PName (before the character _)

I am using the function **LEFT** that stops at the **'_' CHARINDEX-1**

```
SELECT LEFT([PName], CHARINDEX('_',[PName]) - 1) As ProductID
 FROM [MCIT].[dbo].[Product]
```

161 %

Results | Messages

	PName	ProductID
1	Shirt_Tops	Shirt
2	Camisole_Tops	Camisole
3	Pant_Bottoms	Pant
4	Short_Bottoms	Short
5	Skirt_Bottoms	Skirt
6	Trouser_Bottoms	Trouser
7	Jeans_Bottoms	Jeans
8	Jumpsuit_Bottoms	Jumpsuit
9	Legging_Bottoms	Legging
10	Dress_Dresses	Dress
11	Gown_Dresses	Gown
12	Blouse_Tops	Blouse
13	Casual_Dresses	Casual
14	T-shirt_Tops	T-shirt

Used the alias ProductID, but it is the the product name

```
9. ProductCategory VARCHAR(50)      ,--ProductName LastPart after _
```

The product name is as follows : Shirt_Tops

To extract the category I am using the RIGHT function that will extract the number of characters equal to LEN([PName]) - CHARINDEX('_',[PName]).

LEN([PName]) measures the length of 'Shirt_Tops' which is 10

CHARINDEX('_',[PName]) : The index of '_' here is 6

So LEN([PName]) - CHARINDEX('_',[PName]) = 10 - 6 = 4

So we extract the category with RIGHT([PName,4) which is 'Shirt'

```
Select [PName]
    ,RIGHT([PName], LEN([PName]) - CHARINDEX('_',[PName]) ) As ProductCategory From
[dbo].[Product]
```

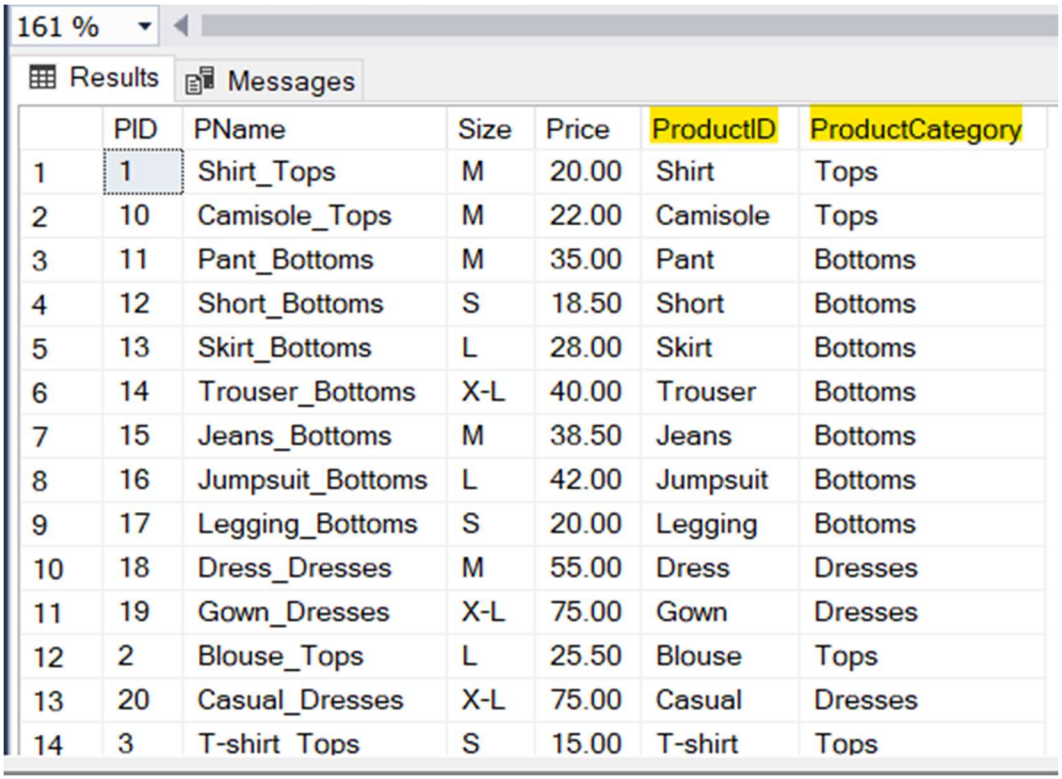
	PName	ProductCategory
1	Shirt_Tops	Tops
2	Camisole_Tops	Tops
3	Pant_Bottoms	Bottoms
4	Short_Bottoms	Bottoms
5	Skirt_Bottoms	Bottoms
6	Trouser_Bottoms	Bottoms
7	Jeans_Bottoms	Bottoms
8	Jumpsuit_Bottoms	Bottoms
9	Legging_Bottoms	Bottoms
10	Dress_Dresses	Dresses
11	Gown_Dresses	Dresses
12	Blouse_Tops	Tops
13	Casual_Dresses	Dresses
14	T-shirt_Tops	Tops

Creating a new table with the categories

```
SELECT TOP (1000) [PID]
          ,[PName]
          ,[Size]
          ,[Price]
     ,LEFT([PName], CHARINDEX('_',[PName]) - 1) As ProductID
     ,RIGHT([PName], LEN([PName]) - CHARINDEX('_',[PName]) ) As ProductCategory
```

```
INTO Products
FROM [MCIT].[dbo].[Product]
```

It is now possible retrieve the category with a join query using the Products table instead of the Product table.



161 %

Results | Messages

	PID	PName	Size	Price	ProductID	ProductCategory
1	1	Shirt_Tops	M	20.00	Shirt	Tops
2	10	Camisole_Tops	M	22.00	Camisole	Tops
3	11	Pant_Bottoms	M	35.00	Pant	Bottoms
4	12	Short_Bottoms	S	18.50	Short	Bottoms
5	13	Skirt_Bottoms	L	28.00	Skirt	Bottoms
6	14	Trouser_Bottoms	X-L	40.00	Trouser	Bottoms
7	15	Jeans_Bottoms	M	38.50	Jeans	Bottoms
8	16	Jumpsuit_Bottoms	L	42.00	Jumpsuit	Bottoms
9	17	Legging_Bottoms	S	20.00	Legging	Bottoms
10	18	Dress_Dresses	M	55.00	Dress	Dresses
11	19	Gown_Dresses	X-L	75.00	Gown	Dresses
12	2	Blouse_Tops	L	25.50	Blouse	Tops
13	20	Casual_Dresses	X-L	75.00	Casual	Dresses
14	3	T-shirt_Tops	S	15.00	T-shirt	Tops

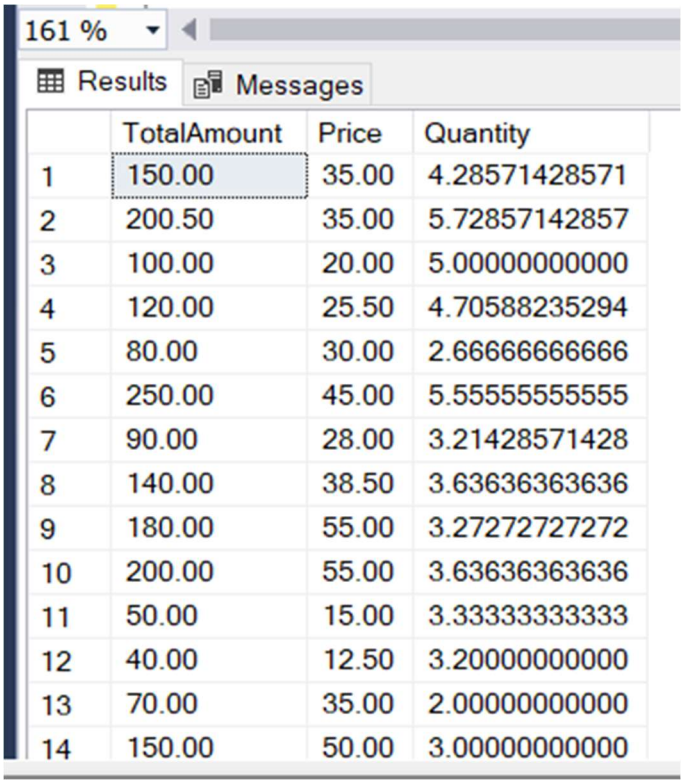
```
10.     Quantity INT
```

The quantity will be obtained with [TotalAmount]/P.[Price]

Here is the query to retrieve the quantity

```
SELECT
   S.[TotalAmount]
   ,P.[Price]
```

```
    ,(S.[TotalAmount]/P.[Price]) As Quantity
FROM [dbo].[Sales] S
INNER JOIN [dbo].[Products] P
ON S.[ProductID] = P.[ProductID];
```



161 %

Results | Messages

	TotalAmount	Price	Quantity
1	150.00	35.00	4.28571428571
2	200.50	35.00	5.72857142857
3	100.00	20.00	5.00000000000
4	120.00	25.50	4.70588235294
5	80.00	30.00	2.66666666666
6	250.00	45.00	5.55555555555
7	90.00	28.00	3.21428571428
8	140.00	38.50	3.63636363636
9	180.00	55.00	3.27272727272
10	200.00	55.00	3.63636363636
11	50.00	15.00	3.33333333333
12	40.00	12.50	3.20000000000
13	70.00	35.00	2.00000000000
14	150.00	50.00	3.00000000000

```
11.     PricePerUnit DECIMAL(8, 2)     ,--This will come From
        ProductTable
```

The unit price will be imported from the Product01 table above.

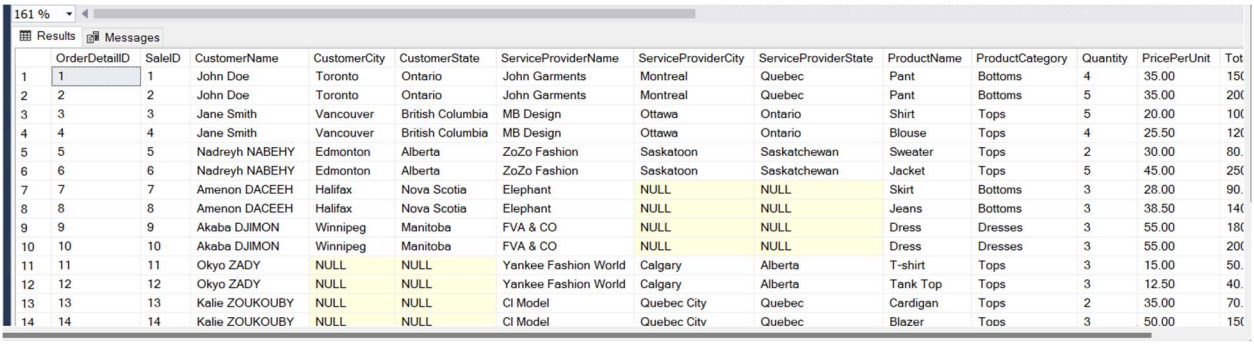
```
12.     TotalSales DECIMAL(19, 2) - This IS Quantity *
        PricePerUNIT
```

By calculating the quantity at the number10., no need to do this. We just need to import the [TotalAmount].

The Joins query

```
SELECT *

FROM [dbo].[Sales01] S
   Left Join [dbo].[Customers] C  ON C.CustomerID = S.CustomerID
   Left Join [dbo].[Suppliers] Su  ON S.SupplierID = SUBSTRING(Su.[SupplierID],5,2)
  Left Join [dbo].[Customers_Address] Ca    ON Ca.[CustomerID] =S.[CustomerID]
  Left Join [dbo].[Suppliers_Address] SA  ON SA.[SupplierID]= S.[SupplierID]
  Left Join [dbo].[Products] P ON P.[ProductID] = S.ProductID

/***  The query to insert the values into the SalesOrderDetail table ***/

INSERT INTO SalesOrderDetail(
   SaleID
   ,CustomerName
   ,CustomerCity
   ,CustomerState
   ,ServiceProviderName
   ,ServiceProviderCity
   ,ServiceProviderState
```

```
    ,ProductName
    ,ProductCategory
    ,Quantity
    ,PricePerUnit
    ,TotalSales

)
SELECT
SaleID,
C.Firstname + ' ' + C.LastName as CustomerName,
Ca.CustomerCity As CustomerCity,
Ca.CustomerState As CustomerState,
Su.SupplierName as ServiceProviderName,
SA.SuppliersCity as ServiceProviderCity,
SA.SuppliersState as ServiceProviderState,
P.ProductID as ProductName,
P.ProductCategory As ProductCategory,
S.Quantity as Quantity,
P.Price as PricePerUnit,
S.TotalAmount As TotalSales

FROM [dbo].[Sales01] S
    Left Join [dbo].[Customers] C On C.CustomerID = S.CustomerID
    Left Join [dbo].[Suppliers] Su On S.SupplierID = SUBSTRING(Su.[SupplierID],5,2)
    Left Join [dbo].[Customers_Address] Ca On Ca.[CustomerID] =S.[CustomerID]
    Left Join [dbo].[Suppliers_Address] SA On SA.[SupplierID]= S.[SupplierID]
    Left Join [dbo].[Products] P On P.[ProductID] = S.ProductID
```



161 %

Results | Messages

	OrderDetailID	SaleID	CustomerName	CustomerCity	CustomerState	ServiceProviderName	ServiceProviderCity	ServiceProviderState	ProductName	ProductCategory	Quantity	PricePerUnit	Tot
1	1	1	John Doe	Toronto	Ontario	John Garments	Montreal	Quebec	Pant	Bottoms	4	35.00	15(
2	2	2	John Doe	Toronto	Ontario	John Garments	Montreal	Quebec	Pant	Bottoms	5	35.00	20(
3	3	3	Jane Smith	Vancouver	British Columbia	MB Design	Ottawa	Ontario	Shirt	Tops	5	20.00	10(
4	4	4	Jane Smith	Vancouver	British Columbia	MB Design	Ottawa	Ontario	Blouse	Tops	4	25.50	12(
5	5	5	Nadreyh NABEHY	Edmonton	Alberta	ZoZo Fashion	Saskatoon	Saskatchewan	Sweater	Tops	2	30.00	80.
6	6	6	Nadreyh NABEHY	Edmonton	Alberta	ZoZo Fashion	Saskatoon	Saskatchewan	Jacket	Tops	5	45.00	25(
7	7	7	Amenon DACEEH	Halifax	Nova Scotia	Elephant	NULL	NULL	Skirt	Bottoms	3	28.00	90.
8	8	8	Amenon DACEEH	Halifax	Nova Scotia	Elephant	NULL	NULL	Jeans	Bottoms	3	38.50	14(
9	9	9	Akaba DJIMON	Winnipeg	Manitoba	FVA & CO	NULL	NULL	Dress	Dresses	3	55.00	18(
10	10	10	Akaba DJIMON	Winnipeg	Manitoba	FVA & CO	NULL	NULL	Dress	Dresses	3	55.00	20(
11	11	11	Okyo ZADY	NULL	NULL	Yankee Fashion World	Calgary	Alberta	T-shirt	Tops	3	15.00	50.
12	12	12	Okyo ZADY	NULL	NULL	Yankee Fashion World	Calgary	Alberta	Tank Top	Tops	3	12.50	40.
13	13	13	Kalie ZOUKOUBY	NULL	NULL	CI Model	Quebec City	Quebec	Cardigan	Tops	2	35.00	70.
14	14	14	Kalie ZOUKOUBY	NULL	NULL	CI Model	Quebec City	Quebec	Blazer	Tops	3	50.00	15(

SalesReportDetails

```
CREATE TABLE SalesReports (
    [Year] INT
    ,[Month] INT
    ,[ProductCategory] INT
    ,[Top1Customer] INT    ,--who bought the maximum , only those customers they have @gmail or @outlook emails
    [Top1Supplier] INT    ,--who's product is sold the max
    [TotalSales] DECIMAL(19, 2)
    ,TotalRunningSales(19, 2)
    );
```

There is an error in the data type :

```
    [ProductCategory] INT
    ,[Top1Customer] INT
```

Updated the data type to Varchar(50) for both

--<> Adding the [Year] and [Month] columns to the Sales01 table --

Creating the final Sales table named Sales02

```
SELECT TOP (1000)
    [SaleID]
    ,[CustomerID]
    ,[SupplierID]
    ,[ProductID]
    ,[Quantity]
    ,[SaleDate]
    ,[TotalAmount]
    ,Year([SaleDate]) as [Year]
    ,Month([SaleDate]) as [Month]
    INTO Sales02
FROM [MCIT].[dbo].[Sales01]
```

```
Select * From Sales02
```

161 %

Results | Messages

	SaleID	CustomerID	SupplierID	ProductID	Quantity	SaleDate	TotalAmount	Year	Month
7	7	4	7	Skirt	3.21428571428	2022-02-09	90.00	2022	2
8	8	4	7	Jeans	3.63636363636	2022-03-10	140.00	2022	3
9	9	5	9	Dress	3.27272727272	2022-03-23	180.00	2022	3
10	10	5	9	Dress	3.63636363636	2022-04-14	200.00	2022	4
11	11	6	2	T-shirt	3.33333333333	2022-05-08	50.00	2022	5
12	12	6	2	Tank top	3.20000000000	2022-05-27	40.00	2022	5
13	13	7	4	Cardigan	2.00000000000	2022-05-29	70.00	2022	5
14	14	7	4	Blazer	3.00000000000	2022-06-01	150.00	2022	6
15	15	8	6	Short	3.24324324324	2022-06-22	60.00	2022	6
16	16	8	6	Trouser	2.50000000000	2022-12-05	100.00	2022	12
17	17	9	8	Jumpsuit	3.80952380952	2022-12-16	160.00	2022	12
18	18	9	8	Legging	2.50000000000	2022-12-18	50.00	2022	12
19	19	10	10	Bustier	6.66666666666	2022-12-23	120.00	2022	12
20	20	10	10	Camisole	3.63636363636	2022-12-24	80.00	2022	12

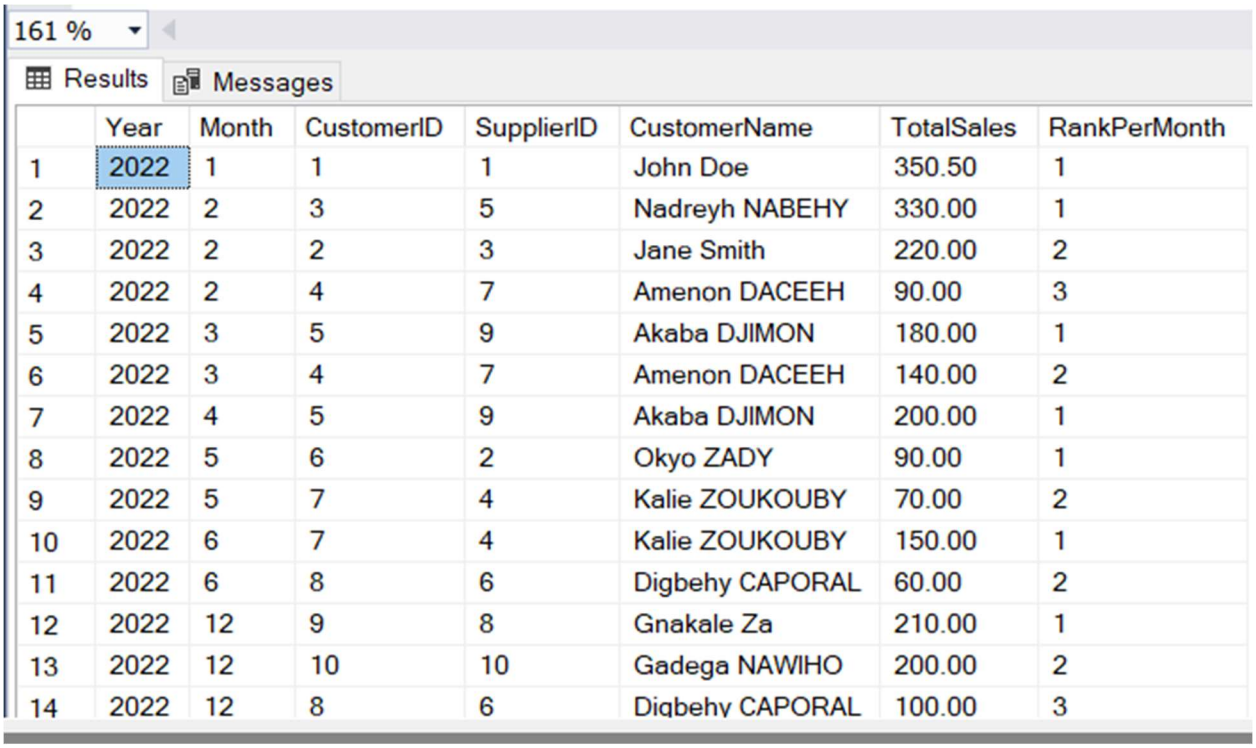
The TOP customer per year and per month

```
SELECT
S.[Year],
S.[Month],
 S.CustomerID,
 S.SupplierID,
SD.CustomerName,
SUM(TotalAmount) AS TotalSales,
RANK() OVER (PARTITION BY [Year], [Month] ORDER BY SUM(TotalAmount) DESC) AS RankPerMonth

FROM [dbo].[Sales02] S
LEFT JOIN [dbo].[Customers] C ON C.CustomerID = S.CustomerID
LEFT JOIN [dbo].[Suppliers] Su ON S.SupplierID = SUBSTRING(Su.[SupplierID],5,2)
LEFT JOIN [dbo].[Customers_Address] Ca ON Ca.[CustomerID] = S.[CustomerID]
LEFT JOIN [dbo].[Suppliers_Address] SA ON SA.[SupplierID] = S.[SupplierID]
LEFT JOIN [dbo].[Products] P ON P.[ProductID] = S.ProductID
LEFT JOIN [dbo].[SalesOrderDetail] SD ON SD.SaleID = S.SaleID
```

```
GROUP BY S.[Year], S.[Month], SD.CustomerName, S.CustomerID,
    S.SupplierID


--<> Need to use RankPerMonth to use a Where condition to select the TOP1
--<> It is not a physical column--> Creating a table from the previous query


Use MCIT
DROP TABLE IF EXISTS RankedCustomers


SELECT
    S.[Year],
    S.[Month],
     S.CustomerID,
     S.SupplierID,
    SD.CustomerName,
    SUM(TotalAmount) AS TotalSales,
    RANK() OVER (PARTITION BY [Year], [Month] ORDER BY SUM(TotalAmount) DESC) AS
RankPerMonth
    INTO RankedCustomers
  FROM [dbo].[Sales02] S
    LEFT JOIN [dbo].[Customers] C ON C.CustomerID = S.CustomerID
    LEFT JOIN [dbo].[Suppliers] Su ON S.SupplierID = SUBSTRING(Su.[SupplierID],5,2)
```

```
    LEFT JOIN [dbo].[Customers_Address] Ca ON Ca.[CustomerID] = S.[CustomerID]

    LEFT JOIN [dbo].[Suppliers_Address] SA ON SA.[SupplierID] = S.[SupplierID]

    LEFT JOIN [dbo].[Products] P ON P.[ProductID] = S.ProductID

    LEFT JOIN [dbo].[SalesOrderDetail] SD ON SD.SaleID = S.SaleID

GROUP BY S.[Year], S.[Month], SD.CustomerName, S.CustomerID, S.SupplierID
```

```
Select * FROM RankedCustomers
```



161 %

Results | Messages

	Year	Month	CustomerID	SupplierID	CustomerName	TotalSales	RankPerMonth
1	2022	1	1	1	John Doe	350.50	1
2	2022	2	3	5	Nadreyh NABEHY	330.00	1
3	2022	2	2	3	Jane Smith	220.00	2
4	2022	2	4	7	Amenon DACEEH	90.00	3
5	2022	3	5	9	Akaba DJIMON	180.00	1
6	2022	3	4	7	Amenon DACEEH	140.00	2
7	2022	4	5	9	Akaba DJIMON	200.00	1
8	2022	5	6	2	Okyo ZADY	90.00	1
9	2022	5	7	4	Kalie ZOUKOUBY	70.00	2
10	2022	6	7	4	Kalie ZOUKOUBY	150.00	1
11	2022	6	8	6	Digbehy CAPORAL	60.00	2
12	2022	12	9	8	Gnakale Za	210.00	1
13	2022	12	10	10	Gadega NAWIHO	200.00	2
14	2022	12	8	6	Digbehy CAPORAL	100.00	3

Retrieving the TOP customer per year and per month

```
SELECT

    [Year],
```

```
    [Month],
   [CustomerID],
   [SupplierID],
    [CustomerName],
    [TotalSales]
FROM RankedCustomers
WHERE RankPerMonth = 1;
```

Creating Table TOP1Customers

```
Use MCIT
DROP TABLE IF EXISTS TOP1Customers

SELECT
    ROW_NUMBER() Over(Order By Year, Month) As TopCustomerID,
    [Year],
    [Month],
     [CustomerID],
    [SupplierID],
   [CustomerName],
            [TotalSales]
  INTO TOP1Customers
```

```
FROM RankedCustomers

WHERE RankPerMonth = 1;


SELECT* FROM TOP1Customers
```

161 %

Results | Messages

	TopCustomerID	Year	Month	CustomerID	SupplierID	CustomerName	TotalSales
1	1	2022	1	1	1	John Doe	350.50
2	2	2022	2	3	5	Nadreyh NABEHY	330.00
3	3	2022	3	5	9	Akaba DJIMON	180.00
4	4	2022	4	5	9	Akaba DJIMON	200.00
5	5	2022	5	6	2	Okyo ZADY	90.00
6	6	2022	6	7	4	Kalie ZOUKOUBY	150.00
7	7	2022	12	9	8	Gnakale Za	210.00
8	8	2023	1	15	9	Eva PINK	360.00
9	9	2023	2	17	4	Mia RED	205.00
10	10	2023	3	19	8	Emma PURPLE	155.00
11	11	2023	4	20	10	Liam ORANGE	115.00
12	12	2023	5	21	1	Olivia YELLOW	310.00
13	13	2023	6	22	3	Ethan GOLD	180.00
14	14	2023	7	23	5	Sophia SILVER	230.00

This `ROW_NUMBER() Over(Order By Year, Month) As TopCustomerID,`

Was added to create a unique ID to join the tables.

Note : I found out later that I needed that column.

```
/*** From this query we can retrieve:

Year

Month
```

CustomerName As TOP1Customer per Year and Per month (Where RankPerMonth = 1

TotalSales --

Remainging : ProductCategory,TOPSupplier and TotalRunningSales ***/

The TOP Supplier per year and per month

```
SELECT S.[Year],
S.[Month],
 S.[CustomerID],
 S.[SupplierID],
SD.[ServiceProviderName],
SUM(TotalAmount) AS TotalSales,
RANK() OVER (PARTITION BY [Year], [Month] ORDER BY SUM(TotalAmount) DESC) AS RankPerMonth

FROM [dbo].[Sales02] S
LEFT JOIN [dbo].[Customers] C ON C.CustomerID = S.CustomerID
LEFT JOIN [dbo].[Suppliers] Su ON S.SupplierID = SUBSTRING(Su.[SupplierID],5,2)
LEFT JOIN [dbo].[Customers_Address] Ca ON Ca.[CustomerID] = S.[CustomerID]
LEFT JOIN [dbo].[Suppliers_Address] SA ON SA.[SupplierID] = S.[SupplierID]
LEFT JOIN [dbo].[Products] P ON P.[ProductID] = S.ProductID
LEFT JOIN [dbo].[SalesOrderDetail] SD ON SD.SaleID = S.SaleID
```

```
GROUP BY [Year], [Month], SD.[ServiceProviderName], S.[CustomerID], S.[SupplierID]


   --<> Need to use RankPerMonth to use a Where condition to select the TOP1

   --<> It is not a physical column--> Creating a table from the previous query

Use MCIT


DROP TABLE IF EXISTS RankedSuppliers


SELECT

     S.[Year],

     S.[Month],

      S.[CustomerID],

      S.[SupplierID],

     SD.[ServiceProviderName],

     SUM(TotalAmount) AS TotalSales,

     RANK() OVER (PARTITION BY [Year], [Month] ORDER BY SUM(TotalAmount) DESC) AS
RankPerMonth

     INTO RankedSuppliers

   FROM [dbo].[Sales02] S

     LEFT JOIN [dbo].[Customers] C ON C.CustomerID = S.CustomerID

     LEFT JOIN [dbo].[Suppliers] Su ON S.SupplierID = SUBSTRING(Su.[SupplierID],5,2)
```

```
LEFT JOIN [dbo].[Customers_Address] Ca ON Ca.[CustomerID] = S.[CustomerID]

LEFT JOIN [dbo].[Suppliers_Address] SA ON SA.[SupplierID] = S.[SupplierID]

LEFT JOIN [dbo].[Products] P ON P.[ProductID] = S.ProductID

LEFT JOIN [dbo].[SalesOrderDetail] SD ON SD.SaleID = S.SaleID


GROUP BY [Year], [Month], SD.[ServiceProviderName], S.[CustomerID],
S.[SupplierID]

Select * FROM RankedSuppliers
```

	Year	Month	CustomerID	SupplierID	ServiceProviderName	TotalSales	RankPerMonth
1	2022	1	1	1	John Garments	350.50	1
2	2022	2	3	5	ZoZo Fashion	330.00	1
3	2022	2	2	3	MB Design	220.00	2
4	2022	2	4	7	Elephant	90.00	3
5	2022	3	5	9	FVA & CO	180.00	1
6	2022	3	4	7	Elephant	140.00	2
7	2022	4	5	9	FVA & CO	200.00	1
8	2022	5	6	2	Yankee Fashion World	90.00	1
9	2022	5	7	4	CI Model	70.00	2
10	2022	6	7	4	CI Model	150.00	1
11	2022	6	8	6	SHAQ Garments	60.00	2
12	2022	12	9	8	Cocorico Fashion	210.00	1
13	2022	12	10	10	BATMAN Collection	200.00	2
14	2022	12	8	6	SHAQ Garments	100.00	3

Retrieving the TOP Supplier per year and per month

```
SELECT

[Year],

[Month],
```

```
    [CustomerID],
    [SupplierID],
      [ServiceProviderName],
      TotalSales
FROM RankedSuppliers
WHERE RankPerMonth = 1;
```

Creating the TOP1Suppliers table

```
Use MCIT

DROP TABLE IF EXISTS TOP1Suppliers

SELECT
    ROW_NUMBER() Over(Order By Year, Month) As TopSupplierID,
      [Year],
      [Month],
    [CustomerID],
    [SupplierID],
      [ServiceProviderName],
      TotalSales
    INTO TOP1Suppliers
```

```
FROM RankedSuppliers

WHERE RankPerMonth = 1;
```

```
Select * From TOP1Suppliers
```

Results | Messages

	TopSupplierID	Year	Month	CustomerID	SupplierID	ServiceProviderName	TotalSales
1	1	2022	1	1	1	John Garments	350.50
2	2	2022	2	3	5	ZoZo Fashion	330.00
3	3	2022	3	5	9	FVA & CO	180.00
4	4	2022	4	5	9	FVA & CO	200.00
5	5	2022	5	6	2	Yankee Fashion World	90.00
6	6	2022	6	7	4	CI Model	150.00
7	7	2022	12	9	8	Cocorico Fashion	210.00
8	8	2023	1	15	9	FVA & CO	360.00
9	9	2023	2	17	4	CI Model	205.00
10	10	2023	3	19	8	Cocorico Fashion	155.00
11	11	2023	4	20	10	BATMAN Collection	115.00
12	12	2023	5	21	1	John Garments	310.00
13	13	2023	6	22	3	MB Design	180.00
14	14	2023	7	23	5	ZoZo Fashion	230.00

```
/*** To this point we can retrieve:

    Year

    Month

    CustomerName As TOP1Customer per Year and Per month (Where RankPerMonth =
1

    TotalSales

    ServiceProviderName as TOP1Supplier--

    Remainging : ProductCategory and TotalRunningSales ***/
```

The TotalRunningSales Query

```
SELECT CustomerID
   ,Year
   ,Month
   ,SUM(TotalAmount)OVER (
            PARTITION BY CustomerID,  Year,  Month
            ORDER BY SaleDate) AS RunningSale
FROM Sales02
ORDER BY CustomerID, Year, Month, SaleDate;
```

Note : As we are retrieving the "TotalRunningSales' by year and by Month, the result will be the same for the TotalSales of the TOP Customers or Suppliers

```
/*** To this point we can retrieve:
   Year
   Month
   CustomerName As TOP1Customer per Year and Per month (Where RankPerMonth =
1
   TotalSales
   ServiceProviderName as TOP1Supplier
   TotalRunningSales--
   Remaining : ProductCategory   ***/
```

TOP ProductCategory per year and per month

```
SELECT
    [Year],
    [Month],
    ProductID,
    ROW_NUMBER() OVER (
                                PARTITION BY [Year],  [Month]
                                ORDER BY SUM(TotalAmount) DESC) AS ProductRank
    FROM [dbo].[Sales02]
    GROUP BY [Year], [Month], ProductID
```

Create a ProductRank table

```
USE MCIT

DROP TABLE IF EXISTS ProductRank

SELECT
    [Year],
    [Month],
```

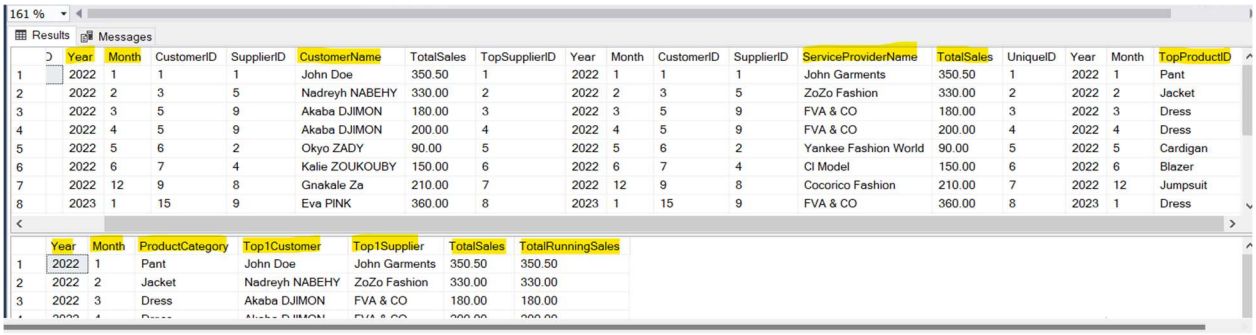
```
    ProductID,
    ROW_NUMBER() OVER (
                                PARTITION BY [Year], [Month]
                                ORDER BY SUM(TotalAmount) DESC) AS ProductRank
     INTO ProductRank
    FROM [dbo].[Sales02]
    GROUP BY [Year], [Month], ProductID
```

Create TOPProductCategory table

```
Use MCIT

DROP TABLE IF EXISTS TOPProductsCategory

SELECT
ROW_NUMBER() Over(Order By Year, Month) As UniqueID,
    [Year],
    [Month],
    ProductID AS TopProductID
  INTO TOPProductsCategory
FROM ProductRank
WHERE ProductRank = 1;
```

List of tables

```
Select * FROM Sales02

Select * FROM [dbo].[TOPProductsCategory]

Select * FROM [dbo].[TOP1Suppliers]

Select * FROM [dbo].[SalesReports]

Select * FROM [dbo].[TOP1Customers]

Select * FROM [dbo].[SalesReports]



Select *

FROM [dbo].[TOP1Customers] TC

LEFT JOIN [dbo].[TOP1Suppliers] TS ON TC.TopCustomerID = TS.TopSupplierID

LEFT JOIN [dbo].[TOPProductsCategory] TPC ON TPC.UniqueID = TC.TopCustomerID


Select * FROM [dbo].[SalesReports]
```



161 %

Results | Messages

	Year	Month	CustomerID	SupplierID	CustomerName	TotalSales	TopSupplierID	Year	Month	CustomerID	SupplierID	ServiceProviderName	TotalSales	UniqueID	Year	Month	TopProductID
1	2022	1	1	1	John Doe	350.50	1	2022	1	1	1	John Garments	350.50	1	2022	1	Pant
2	2022	2	3	5	Nadreyh NABEHY	330.00	2	2022	2	3	5	ZoZo Fashion	330.00	2	2022	2	Jacket
3	2022	3	5	9	Akaba DJIMON	180.00	3	2022	3	5	9	FVA & CO	180.00	3	2022	3	Dress
4	2022	4	5	9	Akaba DJIMON	200.00	4	2022	4	5	9	FVA & CO	200.00	4	2022	4	Dress
5	2022	5	6	2	Okyo ZADY	90.00	5	2022	5	6	2	Yankee Fashion World	90.00	5	2022	5	Cardigan
6	2022	6	7	4	Kalie ZOUKOUBY	150.00	6	2022	6	7	4	CI Model	150.00	6	2022	6	Blazer
7	2022	12	9	8	Gnakale Za	210.00	7	2022	12	9	8	Cocorico Fashion	210.00	7	2022	12	Jumpsuit
8	2023	1	15	9	Eva PINK	360.00	8	2023	1	15	9	FVA & CO	360.00	8	2023	1	Dress

	Year	Month	ProductCategory	Top1Customer	Top1Supplier	TotalSales	TotalRunningSales
1	2022	1	Pant	John Doe	John Garments	350.50	350.50
2	2022	2	Jacket	Nadreyh NABEHY	ZoZo Fashion	330.00	330.00
3	2022	3	Dress	Akaba DJIMON	FVA & CO	180.00	180.00

Side by side, we have all the information to fill the SalesReport table

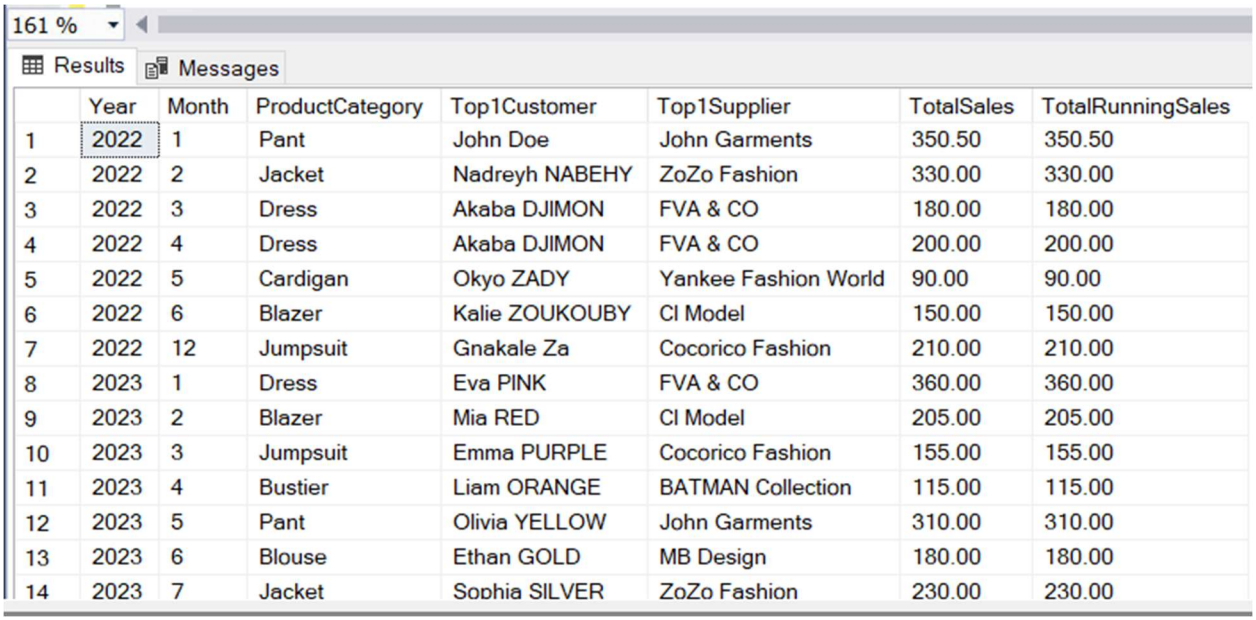
Retrieving data needed for the [dbo].[SalesReports] table

```
Select TC.Year,
    TC.Month,
    TPC.TopProductID As ProductCategory,
    TC.CustomerName as Top1Customer,
    TS.ServiceProviderName As Top1Supplier,
    TC.TotalSales As TotalSales,
    TC.TotalSales As RunningSales

FROM [dbo].[TOP1Customers] TC
LEFT JOIN [dbo].[TOP1Suppliers] TS ON TC.TopCustomerID = TS.TopSupplierID
LEFT JOIN [dbo].[TOPProductsCategory] TPC ON TPC.UniqueID = TC.TopCustomerID
```

Insert data into the [dbo].[SalesReports] table

```
INSERT INTO [dbo].[SalesReports](
      [Year]
     ,[Month]
     ,[ProductCategory]
     ,[Top1Customer]
     ,[Top1Supplier]
     ,[TotalSales]
     ,[TotalRunningSales]
     )
Select TC.Year,
      TC.Month,
      TPC.TopProductID As ProductCategory,
      TC.CustomerName as Top1Customer,
      TS.ServiceProviderName As Top1Supplier,
      TC.TotalSales As TotalSales,
      TC.TotalSales As RunningSales

FROM [dbo].[TOP1Customers] TC
LEFT JOIN [dbo].[TOP1Suppliers] TS ON TC.TopCustomerID = TS.TopSupplierID
LEFT JOIN [dbo].[TOPProductsCategory] TPC ON TPC.UniqueID = TC.TopCustomerID
```

```
Select * From [dbo].[SalesReports]
```



161 %

Results | Messages

	Year	Month	ProductCategory	Top1Customer	Top1Supplier	TotalSales	TotalRunningSales
1	2022	1	Pant	John Doe	John Garments	350.50	350.50
2	2022	2	Jacket	Nadreyh NABEHY	ZoZo Fashion	330.00	330.00
3	2022	3	Dress	Akaba DJIMON	FVA & CO	180.00	180.00
4	2022	4	Dress	Akaba DJIMON	FVA & CO	200.00	200.00
5	2022	5	Cardigan	Okyo ZADY	Yankee Fashion World	90.00	90.00
6	2022	6	Blazer	Kalie ZOUKOUBY	CI Model	150.00	150.00
7	2022	12	Jumpsuit	Gnakale Za	Cocorico Fashion	210.00	210.00
8	2023	1	Dress	Eva PINK	FVA & CO	360.00	360.00
9	2023	2	Blazer	Mia RED	CI Model	205.00	205.00
10	2023	3	Jumpsuit	Emma PURPLE	Cocorico Fashion	155.00	155.00
11	2023	4	Bustier	Liam ORANGE	BATMAN Collection	115.00	115.00
12	2023	5	Pant	Olivia YELLOW	John Garments	310.00	310.00
13	2023	6	Blouse	Ethan GOLD	MB Design	180.00	180.00
14	2023	7	Jacket	Sophia SILVER	ZoZo Fashion	230.00	230.00

The whole table:

Year Month ProductCategory Top1Customer Top1Supplier TotalSales TotalRunningSales

2022 1 Pant John Doe John Garments 350.50 350.50

2022 2 Jacket Nadreyh NABEHY ZoZo Fashion 330.00 330.00

2022 3 Dress Akaba DJIMON FVA & CO 180.00 180.00

2022 4 Dress Akaba DJIMON FVA & CO 200.00 200.00

2022 5 Cardigan Okyo ZADY Yankee Fashion World 90.00 90.00

2022 6 Blazer Kalie ZOUKOUBY CI Model 150.00 150.00

2022 12 Jumpsuit Gnakale Za Cocorico Fashion 210.00 210.00

2023 1 Dress Eva PINK FVA & CO 360.00 360.00

2023 2 Blazer Mia RED CI Model 205.00 205.00

2023 3 Jumpsuit Emma PURPLE Cocorico Fashion 155.00 155.00

2023 4 Bustier Liam ORANGE BATMAN Collection 115.00 115.00

2023 5 Pant Olivia YELLOW John Garments 310.00 310.00

2023 6 Blouse Ethan GOLD MB Design 180.00 180.00

2023 7 Jacket Sophia SILVER ZoZo Fashion 230.00 230.00

2023 12 Dress Isabella PEARL FVA & CO 340.00 340.00

Lessons to remember :

1. Data is not always clean and easy to process.
2. It is important to understand the data structure and have a clear understanding of the result before starting
3. It is not possible to join two tables, if there is no common column that contains a unique identifier for what we are looking for (not by the name, but by the data type)
4. You must know how to write your queries to do what you have to do :
    a. Create a table
    b. Insert data into the table
    c. Modify a column
    d. Add a column
    e. Extract the value you need to work with from an existing column
    f. Write a join query
    g. Use the appropriate Windows function....