

Practical 1

AIM: Import the legacy data from different sources such as Excel, SqlServer, Oracle, etc. and load in the target system.

Solution:

In Power BI:

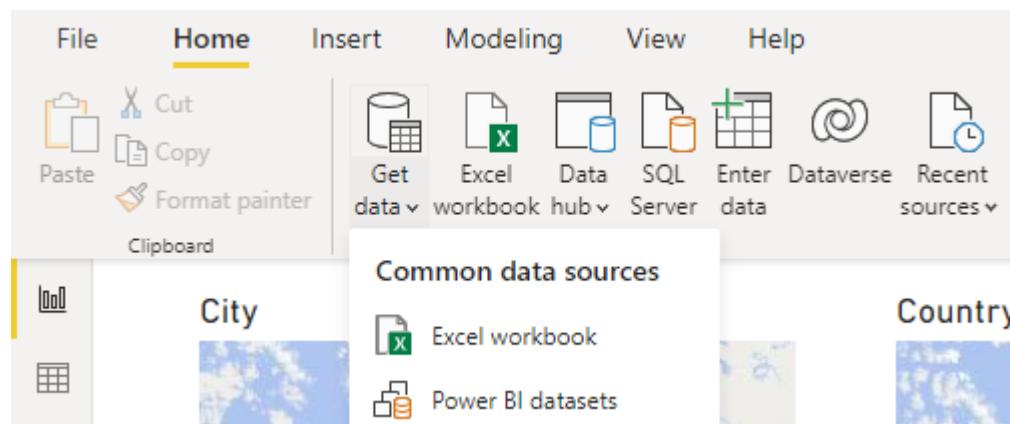
Steps:

<https://github.com/microsoft/powerbi-desktop-samples/blob/main/AdventureWorks%20Sales%20Sample/AdventureWorks%20Sales.xlsx>

Download excel sheet

Import excel sheet in Power BI

Select Get Data from home tab and select Excel workbook.



Recent Files			
AdventureWorks Sales	06-12-2022 14:21	Microsoft Excel W...	13,988 KB
SYBSc-CS, Sem 3, Reg, batch 2021-24	05-12-2022 11:55	Microsoft Excel W...	178 KB
ChapterY.1.0.6	22-11-2022 12:13	File folder	
PortableGit	11-11-2022 14:13	File folder	



Navigator

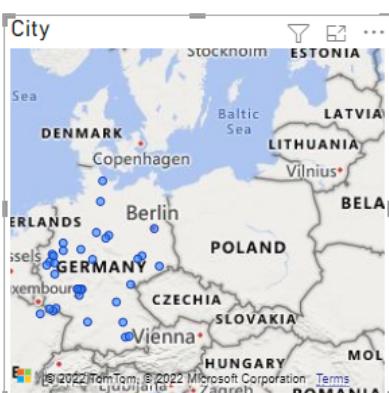
Display Options ...

- AdventureWorks Sales.xlsx [14]
- Customer
- Date
- Product
- Reseller
- Sales
- SalesOrder
- SalesTerritory
- Customer_data
- Date_data
- Product_data
- Reseller_data
- Sales Order_data
- Sales Territory_data
- Sales_data

Date

DateKey	Date	Fiscal Year	Fiscal Quarter	Month	Full
20170701	01-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170702	02-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170703	03-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170704	04-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170705	05-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170706	06-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170707	07-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170708	08-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170709	09-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170710	10-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170711	11-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170712	12-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170713	13-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170714	14-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170715	15-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170716	16-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170717	17-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170718	18-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170719	19-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170720	20-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170721	21-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170722	22-07-2017	FY2018	FY2018 Q1	01-07-2017	
20170723	23-07-2017	FY2018	FY2018 Q1	01-07-2017	

Now select the fields which you want to see.



Filters »

Search

Filters on this visual

City is (All)

Add data fields here

Filters on this page

Add data fields here

Filters on all pages

Visualizations »

Build visual

...

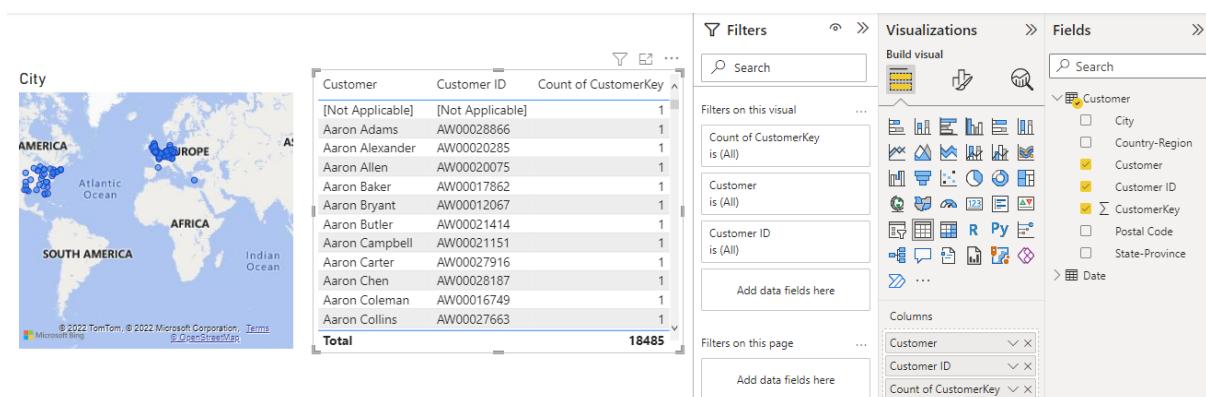
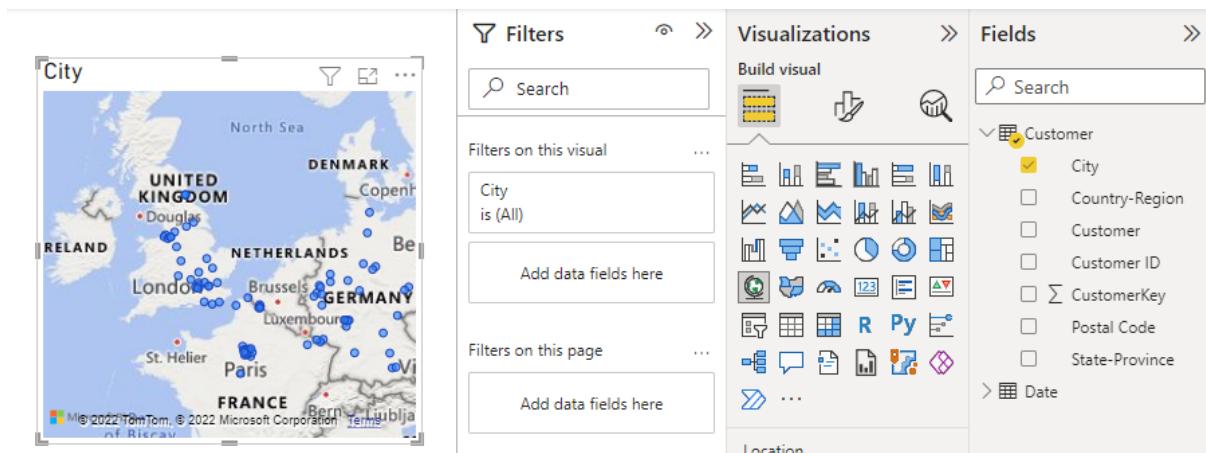
Fields »

Search

Customer

- City
- Country-Region
- Customer
- Customer ID
- CustomerKey
- Postal Code
- State-Province

Date

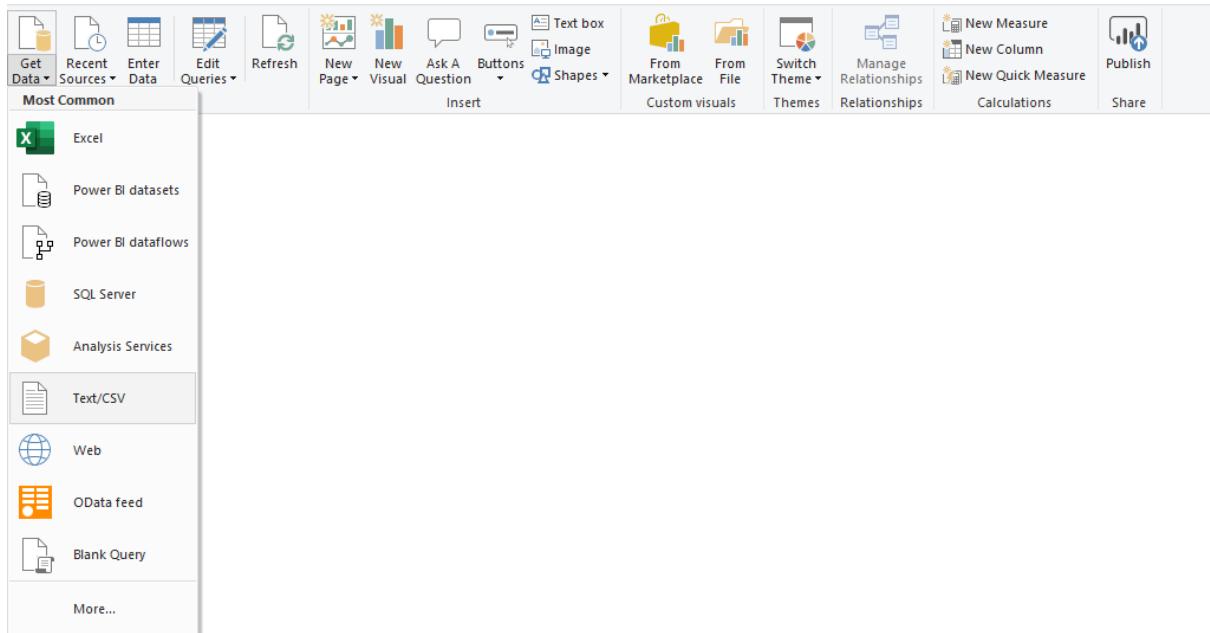


Importing through FlatFile/TextFile.

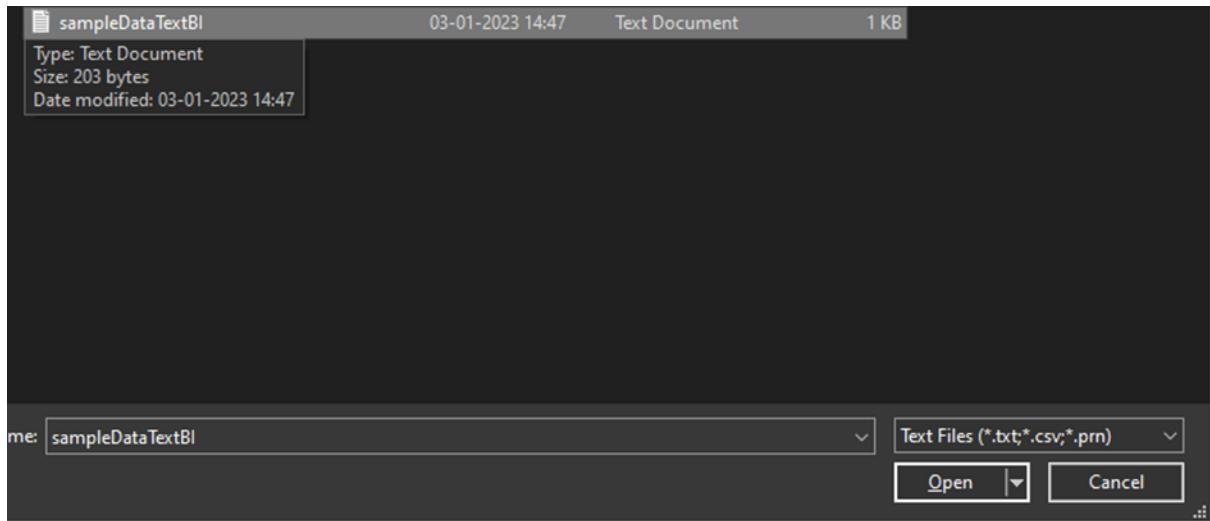
sampleDataTextBI - Notepad
File Edit Format View Help

```
name,rollNum
Adams,1
Baker,2
Clark,3
Davis,4
Evans,5
Frank,6
Ghosh,7
Hills,8
Irwin,9
Jones,10
Klein,11
Lopez,12
Mason,13
Nalty,14
Ochoa,15
Patel,16
Quinn,17
Reily,18
Smith,19
Trott,20|
```

Open Power BI and select Text/CSV option from Get Data present on Home tab.



Select the saved TextFile



sampleDataTextBl.txt

File Origin: 1252: Western European (Windows) Delimiter: Comma Data Type Detection: Based on first 200 rows

name	rollNum
Adams	1
Baker	2
Clark	3
Davis	4
Evans	5
Frank	6
Ghosh	7
Hills	8
Irwin	9
Jones	10
Klein	11
Lopez	12
Mason	13
Nalty	14
Ochoa	15
Patel	16
Quinn	17
Reily	18
Smith	19
Trott	20

Load Transform Data Cancel

Select the fields.

name	rollNum
Adams	1
Baker	2
Clark	3
Davis	4
Evans	5
Frank	6
Ghosh	7
Hills	8
Irwin	9
Jones	10
Klein	11
Total	210

Filters

Filters on this visual ...

- name is (All)
- rollNum is (All)
- Add data fields here

Filters on this page ...

Add data fields here

Visualizations

Fields

Search

sampleDataTextBl

- name
- rollNum

Values

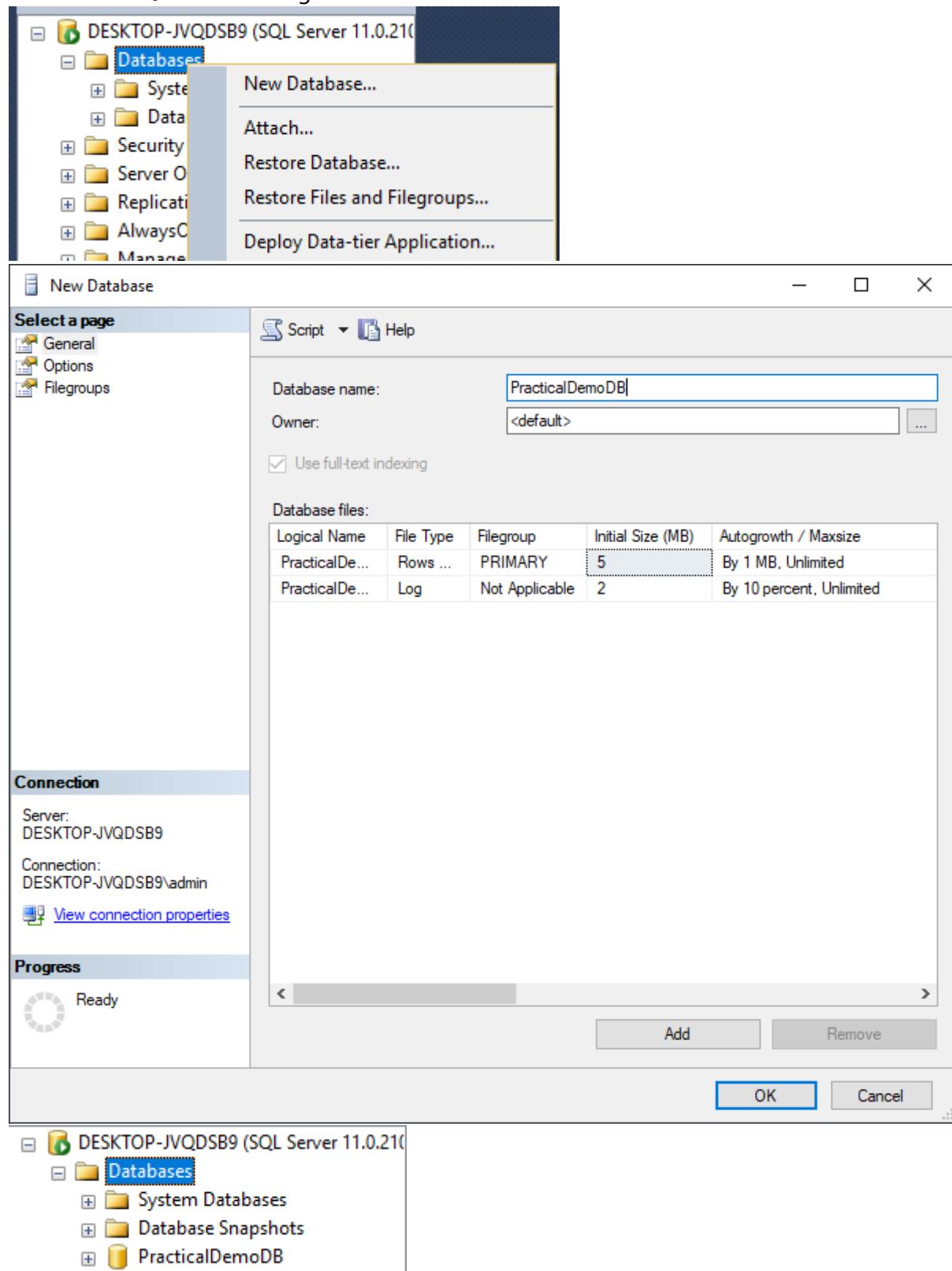
name
rollNum

In SQL Server Management Studio and SQL Server Data Tools

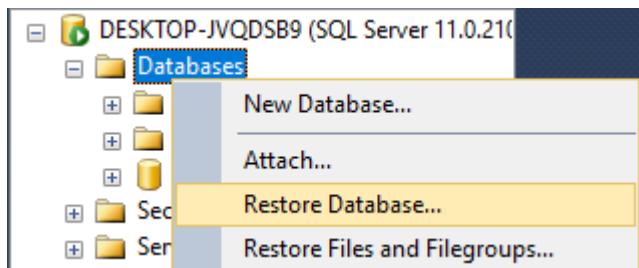
Importing data from Database

Steps:

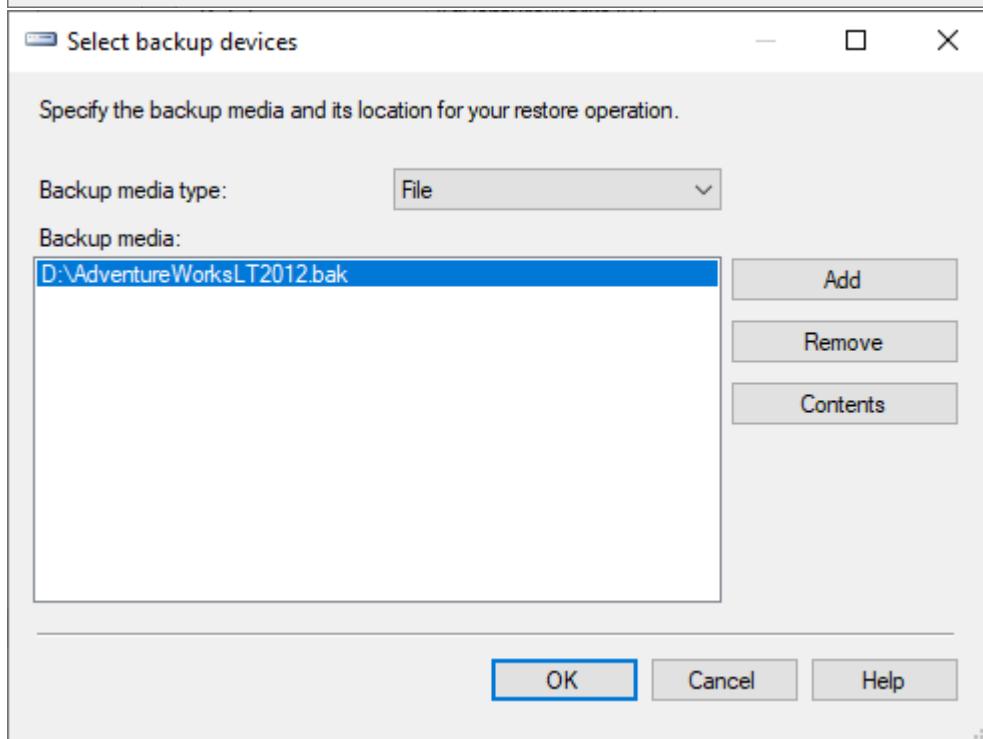
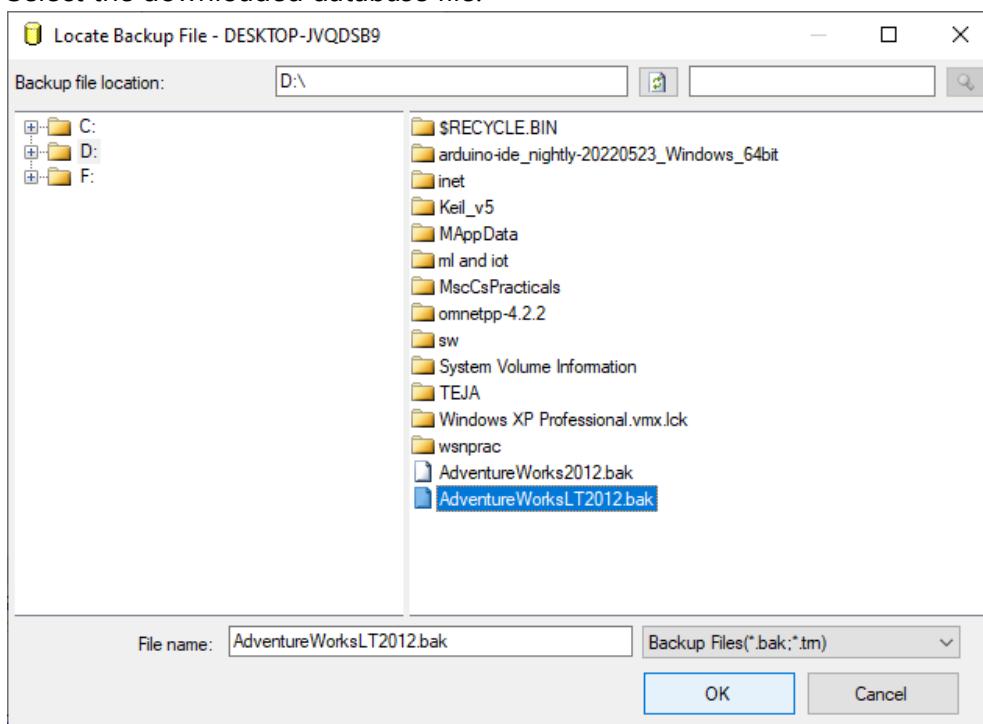
Launch the SQL server management studio and Create new database

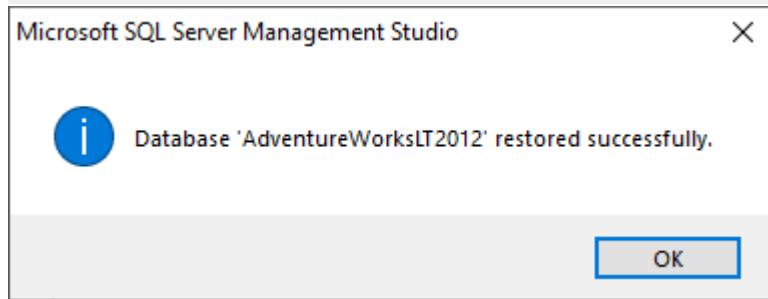
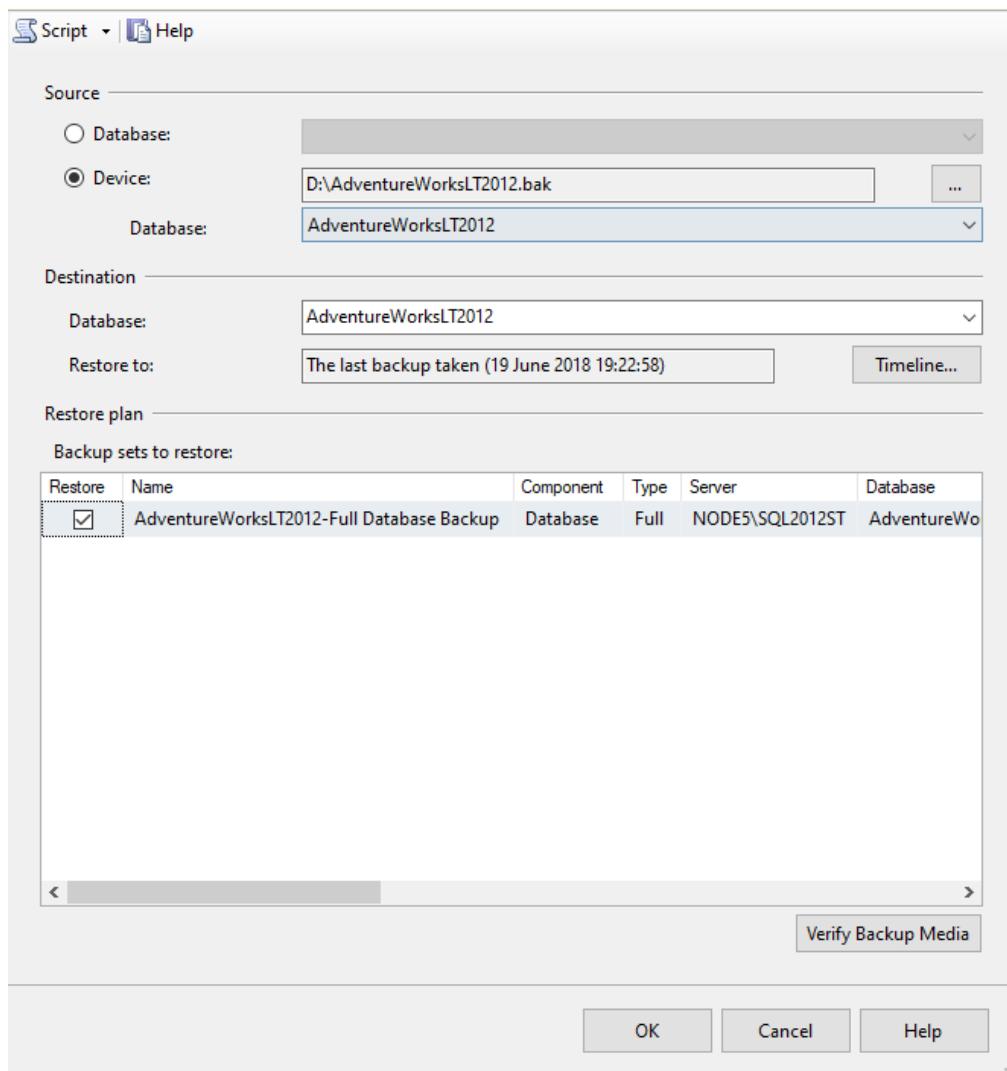


Now restore database



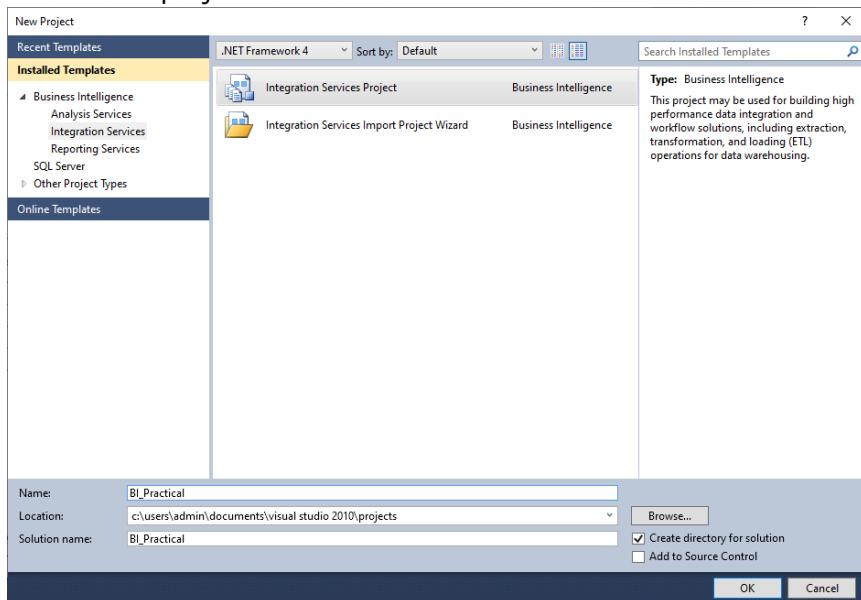
Select the downloaded database file.



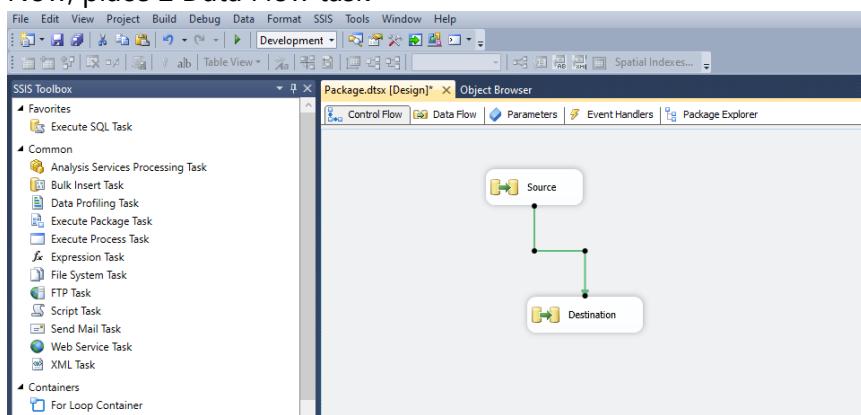


Now launch SQL server data tools

Create new project



Now, place 2 Data Flow task

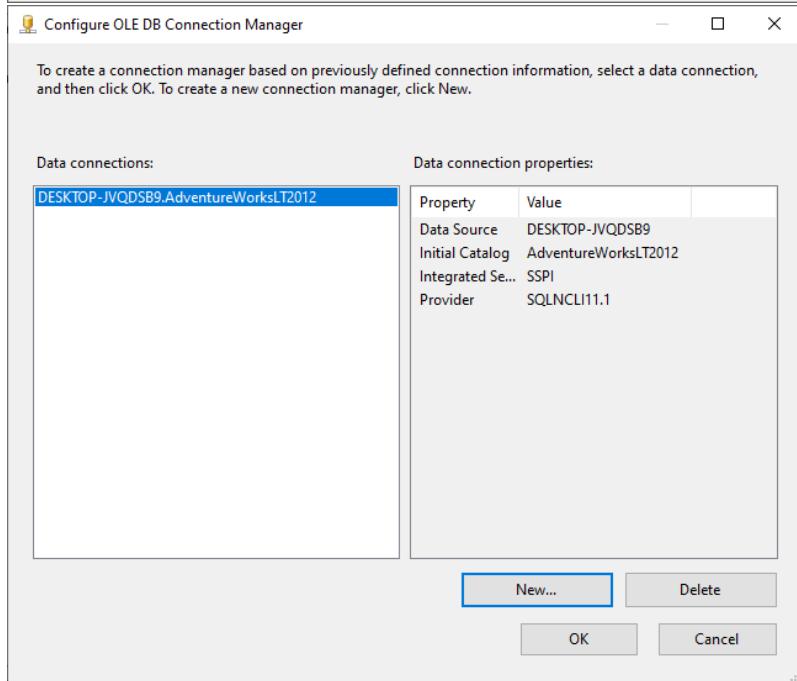
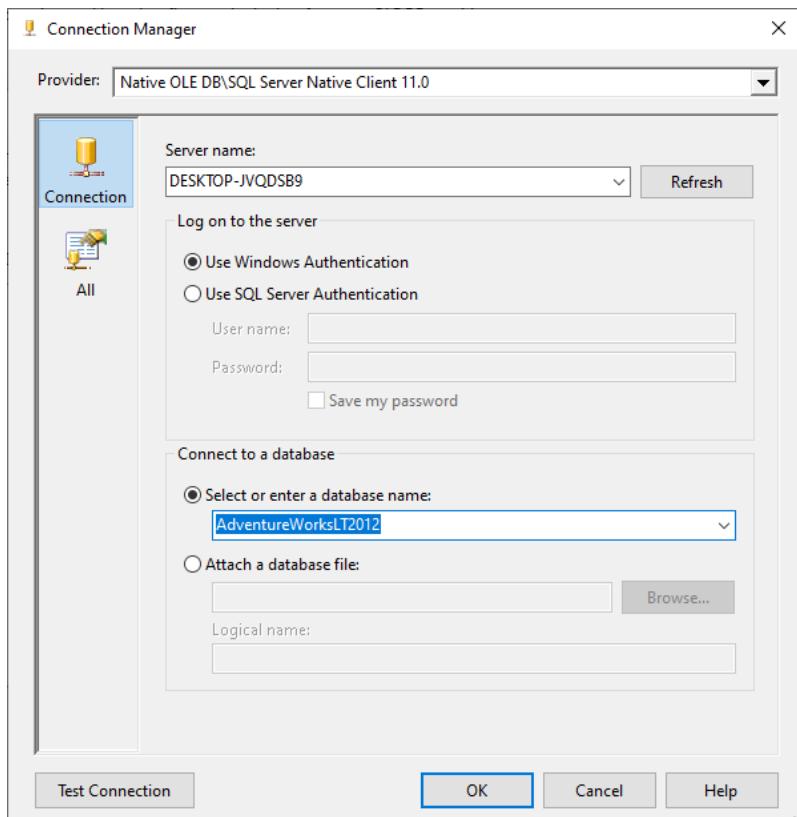


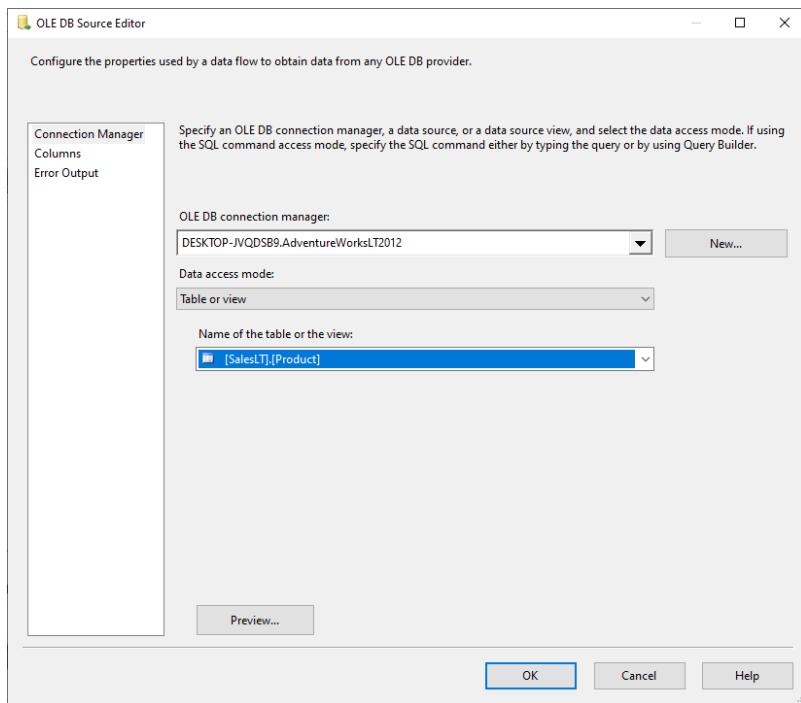
Go to source:

Add ole DB source



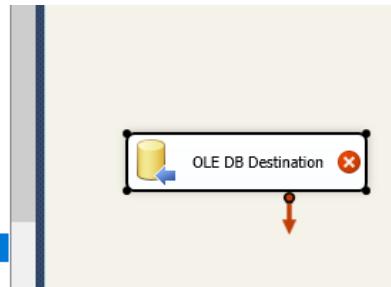
Double click on OLE DB source and add connection



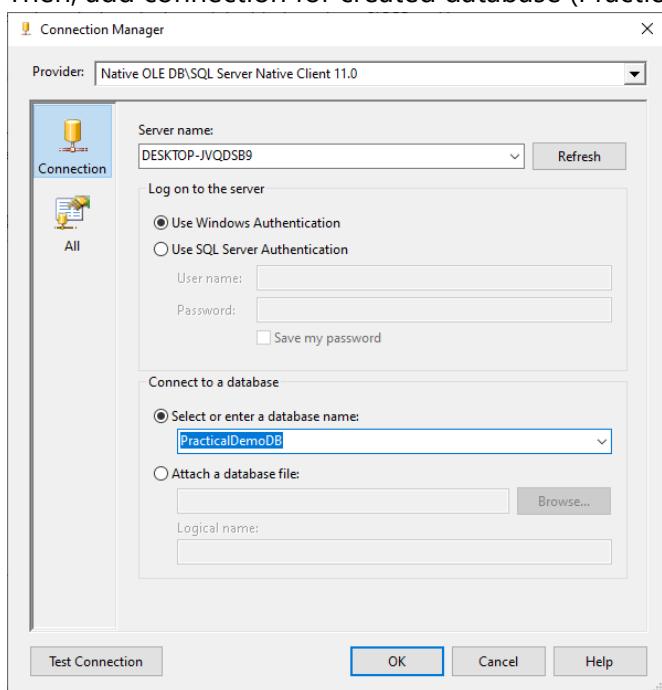


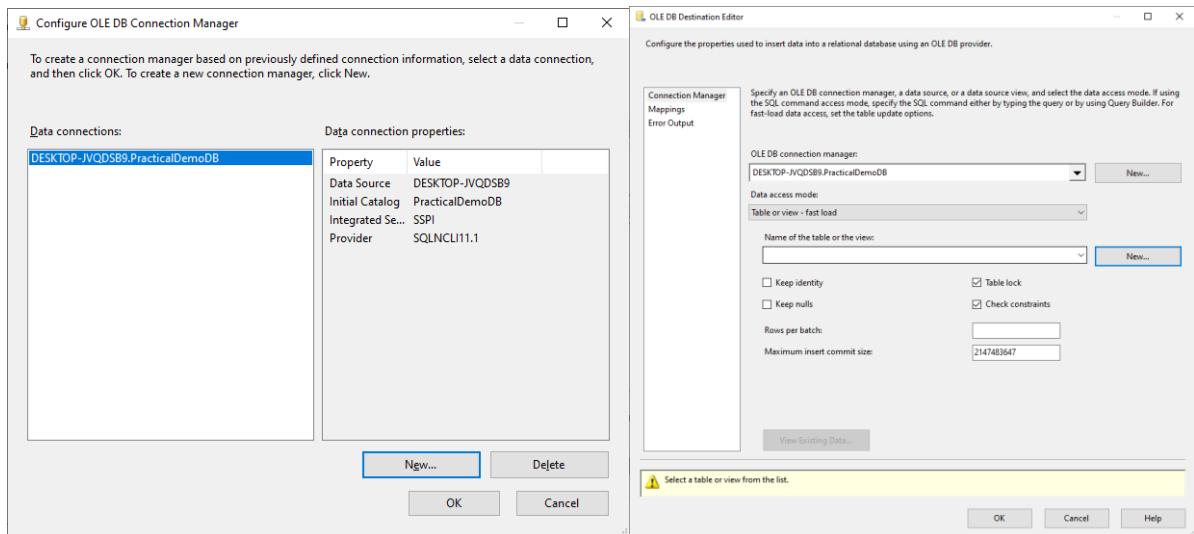
Now add one OLE DB destination

- ◀ Other Destinations
 - 📁 ADO NET Destination
 - 📝 Data Mining Model Training
 - 📁 DataReader Destination
 - 🔗 Dimension Processing
 - 📁 Excel Destination
 - 📄 Flat File Destination
 - 🌐 ODBC Destination
 - 📁 OLE DB Destination
 - 📦 Partition Processing

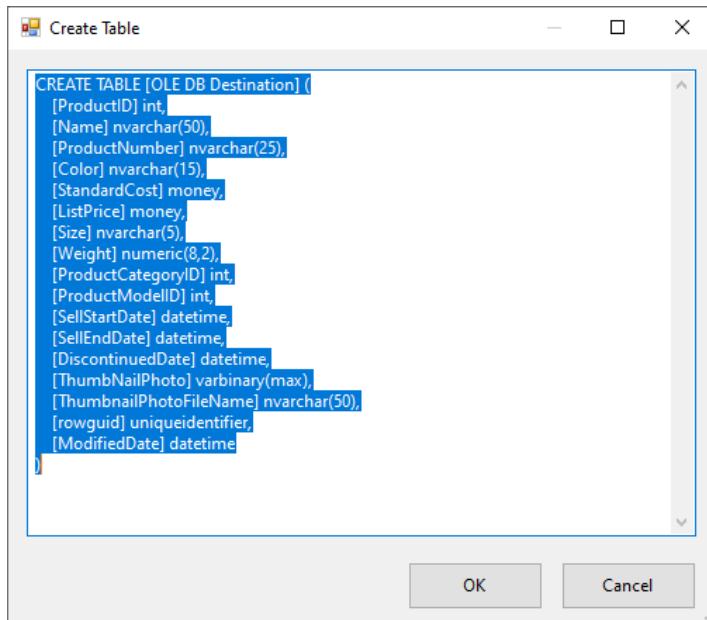


Then, add connection for created database (PracticalDemoDB)

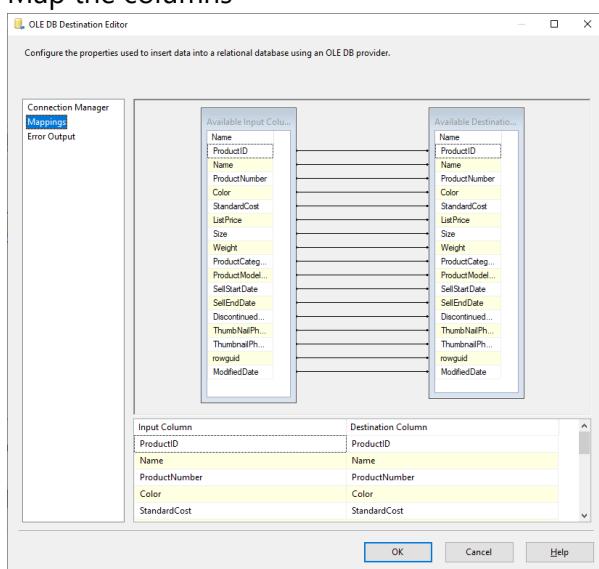


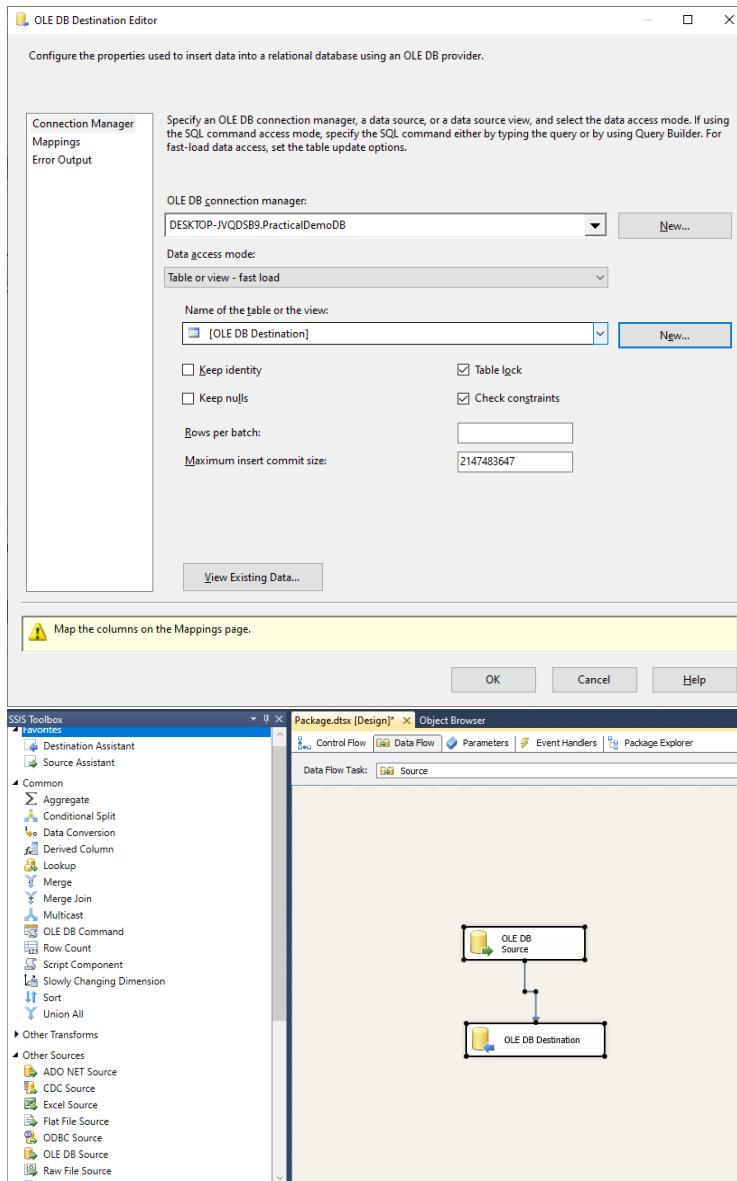


Click on "New..." to create table

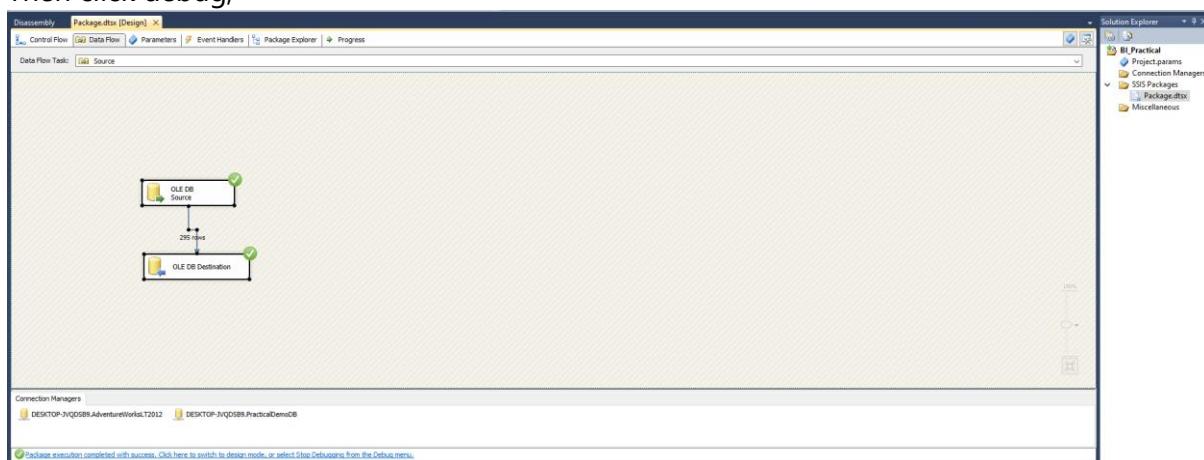


Map the columns





Then click debug,



OUTPUT (Verifying it in SQL server management studio)

SQLQuery1.sql - DESKTOP-JVQDSB9 master (DESKTOP-JVQDSB9\admin (S)) - Microsoft SQL Server Management Studio

```
/*===== Script for SelectTopNRows command from SSMS =====*/
SELECT TOP 1000 [ProductID]
      ,[Name]
      ,[ProductNumber]
      ,[Color]
      ,[StandardCost]
      ,[ListPrice]
      ,[Size]
      ,[Weight]
      ,[ProductCategoryID]
      ,[ProductModelID]
      ,[SellStartDate]
      ,[SellEndDate]
      ,[LastEditedBy]
      ,[ThumbnailPhoto]
      ,[ThumbnailPhotoFileName]
      ,[ModifiedDate]
   FROM [AdventureworksDW].[dbo].[OutputTable]
```

Results

ProductID	Name	ProductNumber	Color	StandardCost	ListPrice	Size	Weight	ProductCategoryID	ProductModelID	SellStartDate	SellEndDate	DiscontinuedDate	ThumbnailPhoto	ThumbnailPhotoFileName	ModifiedDate	
1	Holiday Hat	FR-RSHR-50	Black	1059.31	1431.50	58	1016.04	18	6	2002-01-01 00:00:00	NULL	NULL	no_image_available		2002-01-01 00:00:00	
2	Holiday Hat	FR-RSHR-48	Red	1059.31	1431.50	58	1016.04	18	6	2002-01-01 00:00:00	NULL	NULL	no_image_available		2002-01-01 00:00:00	
3	Sport-100 Helmet	HL-US09-R	Red	13.0863	34.99	NULL	NULL	35	33	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00	
4	Sport-100 Helmet	HL-US09	Black	13.0863	34.99	NULL	NULL	35	33	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00	
5	Mountain Bike Sock	M	SO-B99M-W	White	3.3963	9.50	M	NULL	27	18	2005-07-01 00:00:00	2006-06-30 00:00:00	NULL	no_image_available		2005-07-01 00:00:00
6	Mountain Bike Sock	L	SO-B99L-W	White	3.3963	9.50	L	NULL	27	18	2005-07-01 00:00:00	2006-06-30 00:00:00	NULL	no_image_available		2005-07-01 00:00:00
7	Footwear Bag	HL-YFB9-R	Blue	31.0953	34.99	NULL	NULL	35	33	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00	
8	AWC Logo Cap	CA-109S	Multi	6.5223	8.99	23	2	23	2	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00	
9	Long-Sleeve Logo Jersey	S	LJ-010S	Multi	38.4923	49.99	S	NULL	29	11	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
10	Long-Sleeve Logo Jersey	M	LJ-010M	Multi	38.4923	49.99	M	NULL	29	11	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
11	Long-Sleeve Logo Jersey	L	LJ-010L	Multi	38.4923	49.99	L	NULL	29	11	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
12	Long-Sleeve Logo Jersey	XL	LJ-010X	Multi	38.4923	49.99	XL	NULL	29	11	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
13	Hi-Roller Frame	Red	FR-RSHR-52	Red	868.5342	1431.50	52	1041.26	18	6	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
14	Hi-Roller Frame	Red	FR-RSHR-44	Red	868.5342	1431.50	52	1041.26	18	6	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
15	Hi-Roller Frame	Red	FR-RSHR-56	Red	868.5342	1431.50	52	997.90	18	6	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
16	Hi-Roller Frame	Red	FR-RSHR-48	Red	868.5342	1431.50	52	997.90	18	6	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
17	Hi-Roller Frame	Red	FR-RSHR-56	Red	868.5342	1431.50	56	1016.04	18	6	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
18	Hi-Roller Frame	Red	FR-RSHR-52	Red	868.5342	1431.50	56	1016.04	18	6	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
19	Hi-Roller Frame	Black	FR-RSHB-40	Black	204.6251	337.22	58	1115.83	18	9	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
20	Hi-Roller Frame	Black	FR-RSHB-42	Black	204.6251	337.22	60	1115.83	18	9	2005-07-01 00:00:00	NULL	NULL	no_image_available		2005-07-01 00:00:00
21	Hi-Roller Frame	Red	FR-RSHR-44	Red	187.1571	337.22	44	1052.33	18	9	2005-07-01 00:00:00	2007-06-30 00:00:00	NULL	no_image_available		2005-07-01 00:00:00
22	Hi-Roller Frame	Red	FR-RSHR-48	Red	187.1571	337.22	48	1070.47	18	9	2005-07-01 00:00:00	2007-06-30 00:00:00	NULL	no_image_available		2005-07-01 00:00:00
23	Hi-Roller Frame	Red	FR-RSHR-52	Red	187.1571	337.22	52	1088.62	18	9	2005-07-01 00:00:00	2007-06-30 00:00:00	NULL	no_image_available		2005-07-01 00:00:00
24	Hi-Roller Frame	Red	FR-RSHR-58	Red	187.1571	337.22	58	1115.83	18	9	2005-07-01 00:00:00	2007-06-30 00:00:00	NULL	no_image_available		2005-07-01 00:00:00
25	Hi-Roller Frame	Red	FR-RSHR-60	Red	187.1571	337.22	60	1124.90	18	9	2005-07-01 00:00:00	2007-06-30 00:00:00	NULL	no_image_available		2005-07-01 00:00:00

Ready

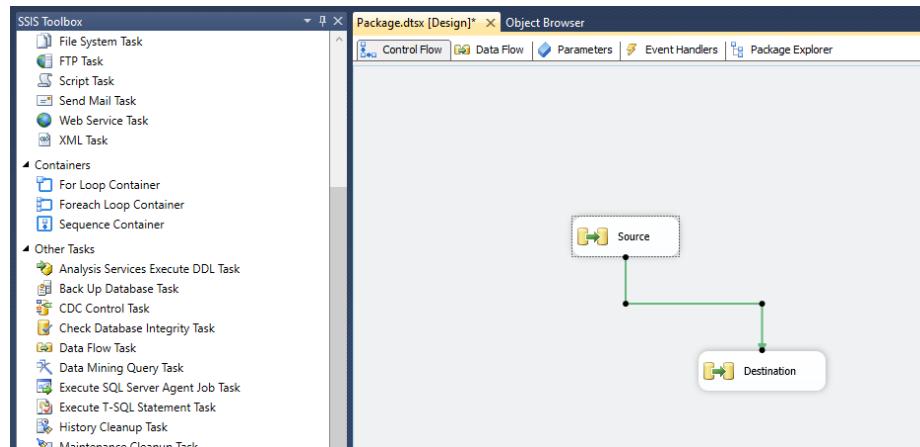
Query executed successfully.

DESKTOP-JVQDSB9 (11.0 RIM) DESKTOP-JVQDSB9\admin ... master 00:05:10 295 rows

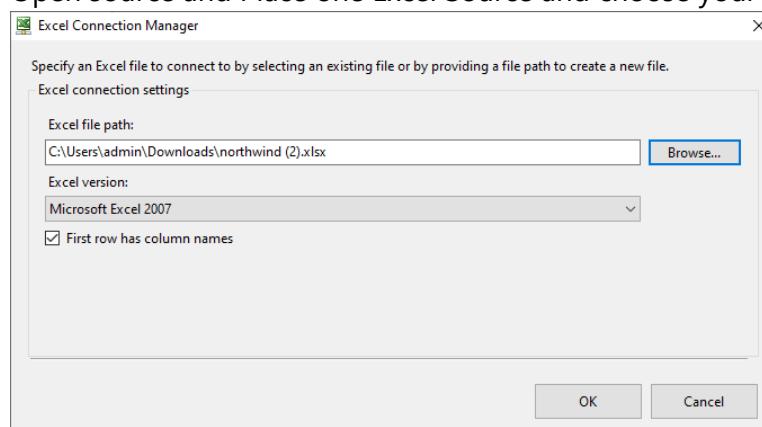
Ln 1 Col 1 Ch 1 INS 14:44 04-02-2023

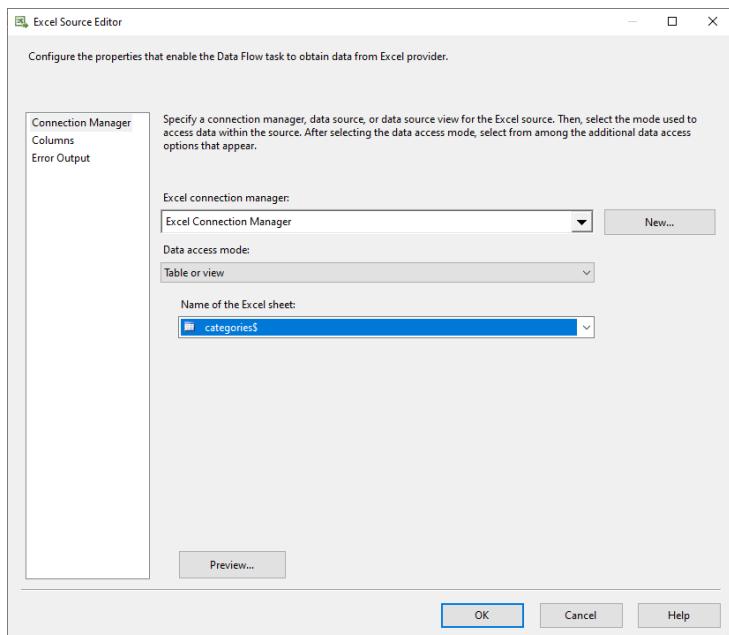
Importing Excel and loading it the system.

Place 2 Data Flow task.

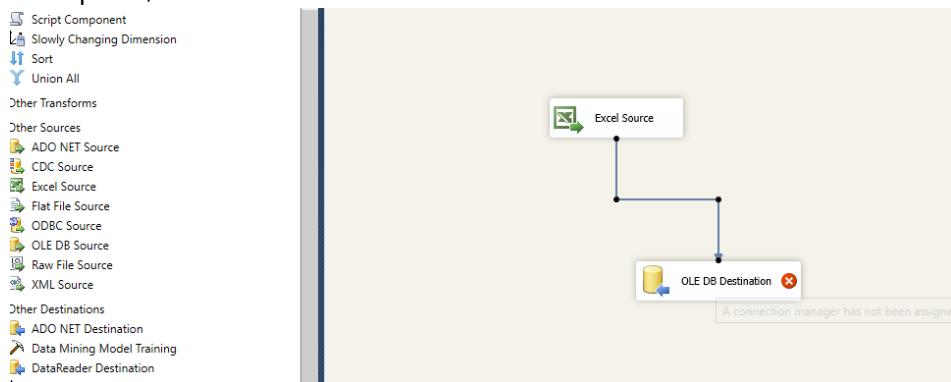


Open source and Place one Excel Source and choose your excel sheet

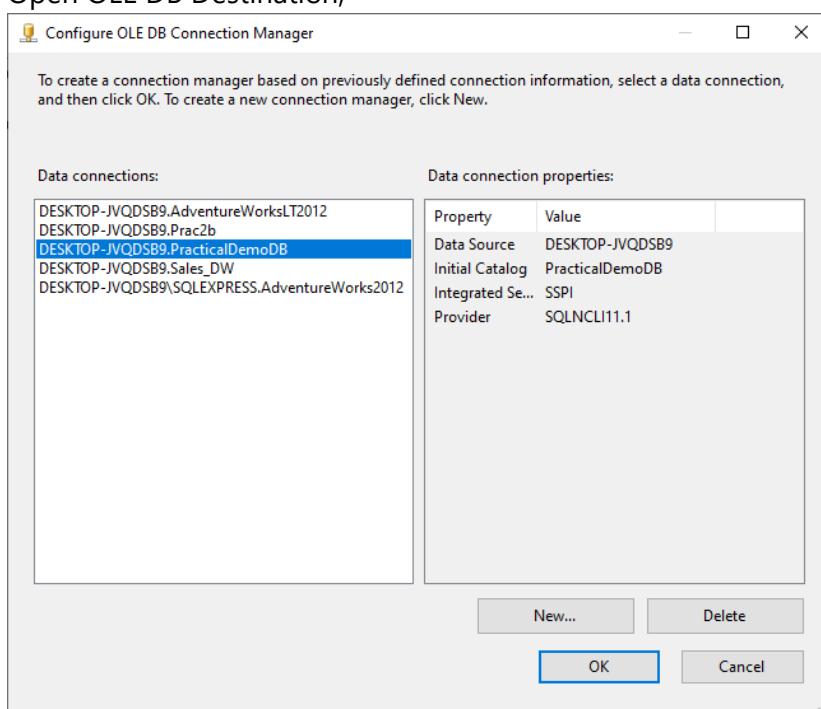


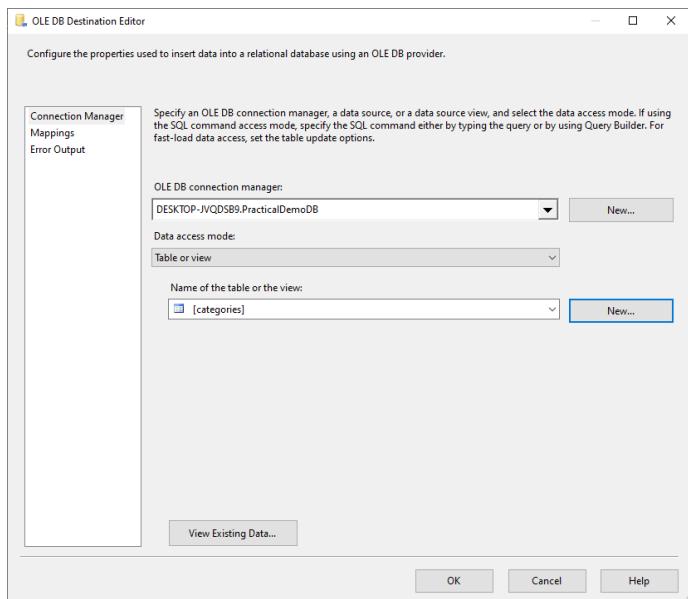


Now place, OLE DB Destination

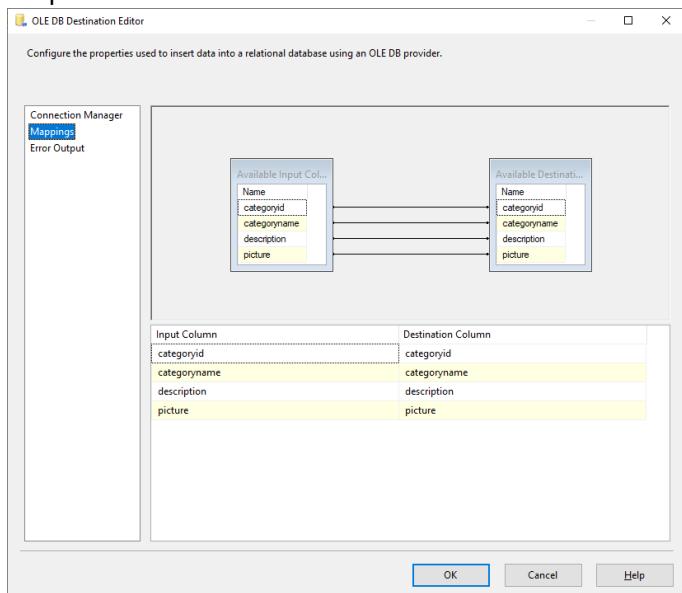


Open OLE DB Destination,



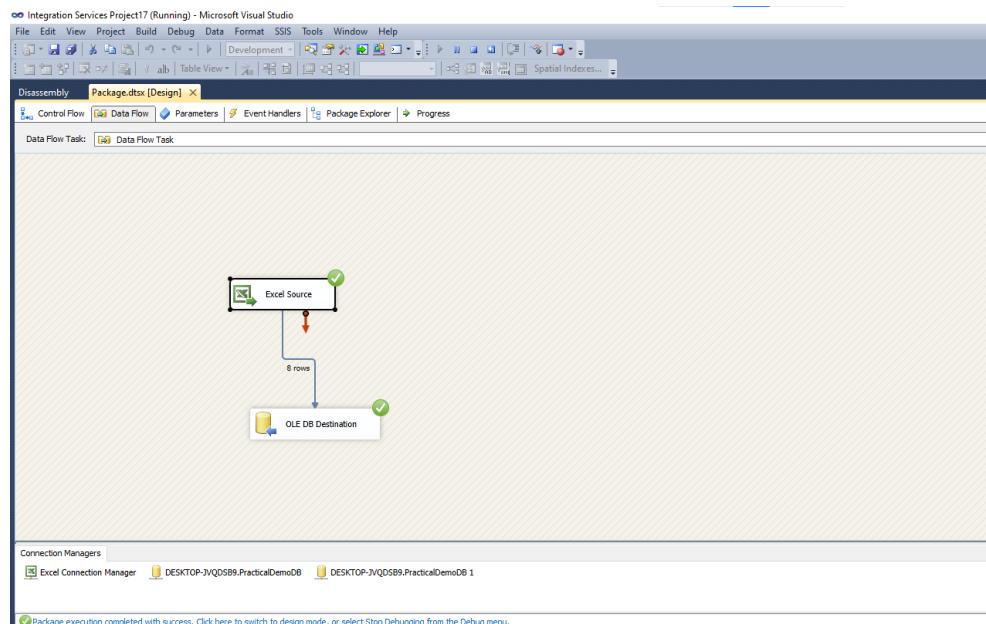


Map the columns



Now execute it,





Verifying it in Sql management studio

SQLQuery1.sql - DESKTOP-JVQDSB9.master (DESKTOP-JVQDSB9\Admin (51)) - Microsoft SQL Server Management Studio

File Edit View Query Project Debug Tools Window Help

master | Execute | Debug | A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

Object Explorer

SQLQuery1.sql - D...VQDSB9\admin (51) | **** Script for SelectTopRows command from SSMS *****

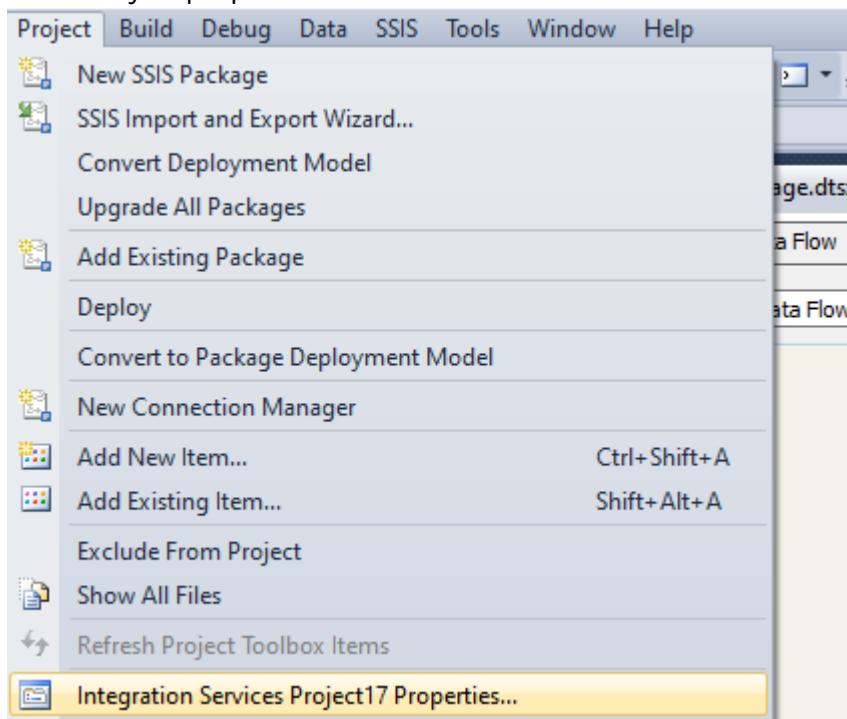
```
SELECT TOP 1000 [categoryid]
      ,[categoryname]
      ,[description]
      ,[picture]
  FROM [PracticalDemoDB].[dbo].[categories]
```

Results Messages

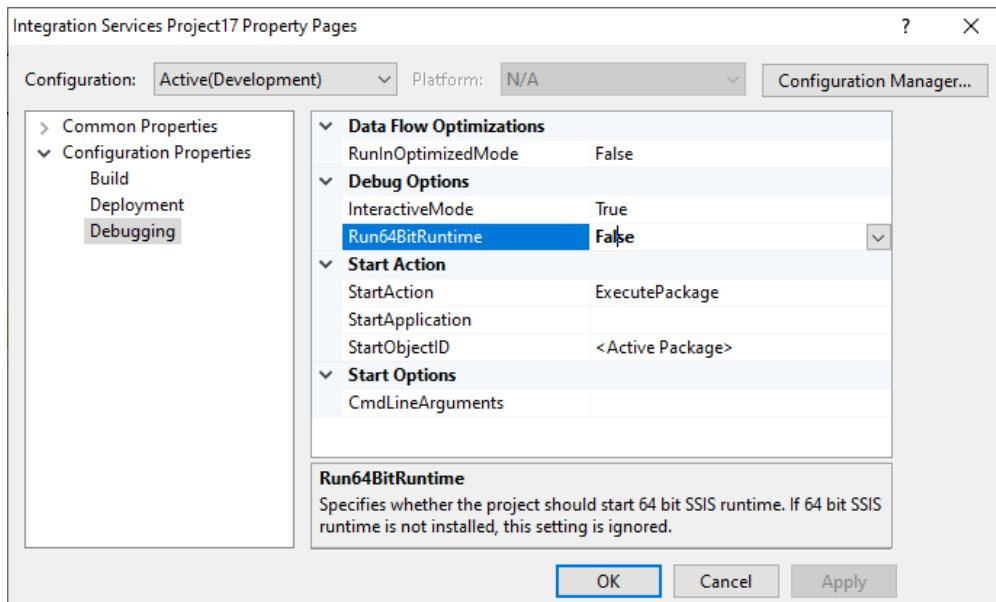
	categoryid	categoryname	description	picture
1	1	Beverages	Soft drinks, coffees, teas, beers, and ales	☒
2	2	Condiments	Sweet and savory sauces, relishes, spreads, and ...	☒
3	3	Confections	Desserts, candies, and sweet breads	☒
4	4	Dairy Products	Cheeses	☒
5	5	Grains/Cereals	Breads, crackers, pasta, and cereal	☒
6	6	Meat/Poultry	Prepared meats	☒
7	7	Produce	Dried fruit and bean curd	☒
8	8	Seafood	Seaweed and fish	☒
9	1	Beverage	Soft drinks, coffees, teas, beers, and ales	☒
10	2	Condiment	Sweet and savory sauces, relishes, spreads, and ...	☒
11	3	Confection	Desserts, candies, and sweet breads	☒
12	4	Dairy Product	Cheeses	☒
13	5	Grain/Cereal	Breads, crackers, pasta, and cereal	☒
14	6	Meat/Poultry	Prepared meats	☒
15	7	Produce	Dried fruit and bean curd	☒
16	8	Seafood	Seaweed and fish	☒

If error occurs while importing the excel data, follow below steps.

Go to Project properties.



Set Run64BitRuntime to false.



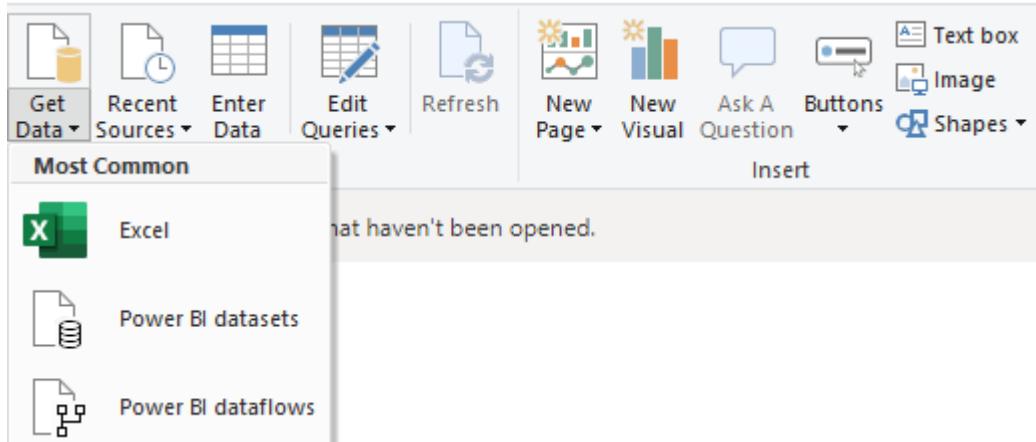
Practical 2

AIM: Perform the Extraction Transformation and Loading (ETL) process to construct the database in the Sqlserver.

Solution:

In Power Bi,

Import your desired data,



Select Tables.

A screenshot of the Power BI Navigator dialog. On the left, there's a tree view of data sources: 'AdventureWorks Sales.xlsx [14]' is expanded, showing 'Customer' (selected with a checked checkbox), 'Date', 'Product', 'Reseller', 'Sales', 'SalesOrder', 'SalesTerritory', 'Customer_data', 'Date_data', 'Product_data', 'Reseller_data', 'Sales Order_data', 'Sales Territory_data', and 'Sales_data'. On the right, a table named 'Date' is displayed with columns: DateKey, Date, Fiscal Year, Fiscal Quarter, Month, and Full. The table contains 24 rows of data from July 2017. At the bottom of the dialog are buttons for 'Load', 'Transform Data', and 'Cancel'.

Click on Transform Data

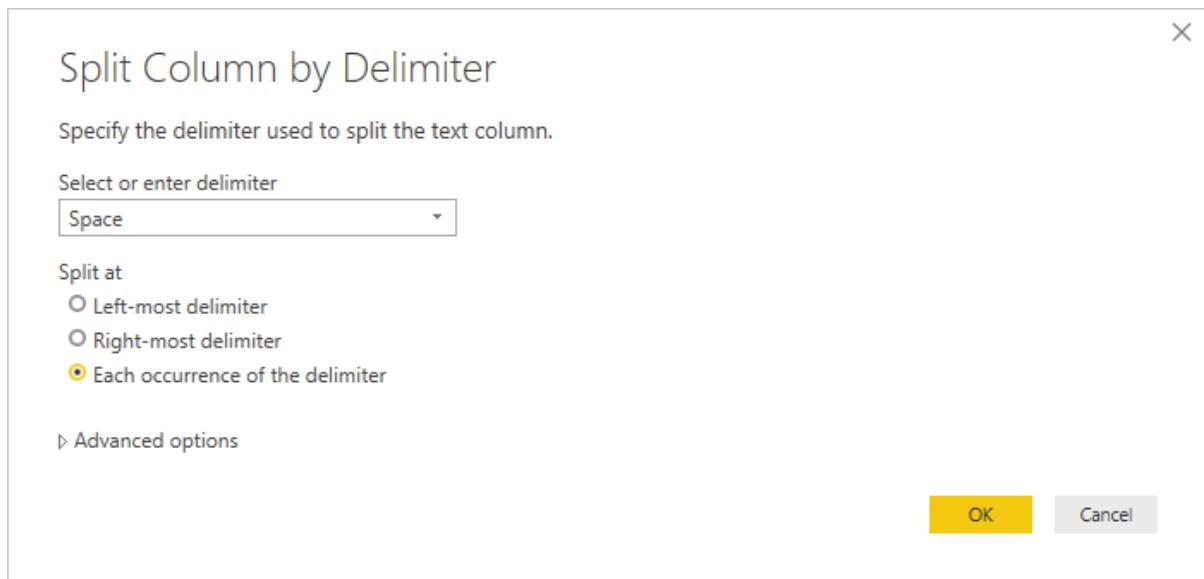
Screenshot of the Power Query Editor showing a query named "Customer". The table has columns: CustomerKey, CustomerID, Customer, City, State-Province, CountryRegion, and PostalCode. A context menu is open over the "Customer" column, specifically under the "Transform" section. The menu path is "Transform > Split Column". Other options visible include "Copy", "Remove", "Remove Other Columns", "Duplicate Column", "Add Column From Examples...", "Remove Duplicates", "Remove Errors", "Change Type", "Transform", "Replace Values...", "Replace Errors...", "Split Column", "Group By...", "Fill", "Unpivot Columns", "Unpivot Other Columns", "Unpivot Only Selected Columns", "Rename...", "Move", " Drill Down", and "Add as New Query".

Transformation 1:

Splitting customer column into FirstName and LastName

Screenshot of the Power Query Editor showing the "Customer" table. The "Customer" column is selected, and a context menu is open. The "Transform" section is expanded, and the "Split Column" option is selected. A sub-menu for "Split Column" is displayed, with "By Delimiter..." highlighted.

Customer	City	State-Province	Country-Region
[Not Applicable]			
Jon Yang	Rockhampton	Queensland	Australia
Eugene Huang	Seaford	Victoria	Australia
Ruben Torres	Hobart	Tasmania	Australia
Christy Zhu	North Ryde	New South Wales	Australia
Elizabeth Johnson	Wollongong	New South Wales	Australia
Julio Ruiz	East Brisbane	Queensland	Australia
Janet Alvarez	Metroville	New South Wales	Australia
Marco Mehta	Warrambool	Victoria	Australia
Rob Verhoff	Bendigo	Victoria	Australia
Shannon Carlson	Hervey Bay	Queensland	Australia
Jacquelyn Suarez	East Brisbane	Queensland	Australia
Curtis Lu	East Brisbane	Queensland	Australia
Lauren Walker	Bremerton	Washington	United States
Ian Jenkins	Lebanon	Oregon	United States
Sydney Bennett	Redmond	Washington	United States
Chloe Young	Burbank	California	United States
Wyatt Hill	Imperial Beach	California	United States
Shannon Wang	Sunbury	Victoria	Australia
Clarence Rai	Bendigo	Victoria	Australia
Luke Lal	Langleys	British Columbia	Canada
Jordan King	Methow	British Columbia	Canada
Destiny Wilson	Beaverton	Oregon	United States
Ethan Zhang	Bellingham	Washington	United States
Seth Edwards	Bellflower	California	United States
Russell Xie	Concord	California	United States
Alejandro Beck	Hawthorne	Queensland	Australia
Harold Sal	Goulburn	New South Wales	Australia
Jessie Zhao	Warrambool	Victoria	Australia
Jill Jimenez	St. Leonards	New South Wales	Australia



FirstName	Customer	State-Province
[Not Applicable]	Yang	[Not Applicable]
Jon	Huang	Queensland
Eugene	Torres	Victoria
Ruben	Zhu	Tasmania
Christy	Johnson	New South Wales
Elizabeth	Ruiz	New South Wales
Julio	Alvarez	Queensland
Janet	Mehta	New South Wales
Marco	Verhoff	Victoria
Rob	Carlson	Victoria
Shannon	Suarez	Queensland
Jacquelyn	Lu	Queensland
Curtis	Walker	Queensland
Lauren	Jenkins	Washington
Ian	Bennett	Oregon
Sydney	Young	Washington
Chloe	Hill	California
Wyatt	Wang	California
Shannon	Rai	Victoria
Clarence	Lal	Victoria
Luke		British Columbia

A context menu is open over the 'Customer' column, showing options like Copy, Remove, Split Column, and Rename... The 'Rename...' option is highlighted.

Untitled - Power Query Editor

File **Home** **Transform** **Add Column** **View** **Help**

Queries [1] Customer

CustomerKey	CustomerID	FirstName	LastName	City	StateProvince	CountryRegion	PostalCode
1	-1 [Not Applicable]	[Not Applicable]					
2	11000 AW00011000	Jon	Yang	Rockhampton	Queensland	Australia	4700
3	11001 AW00011001	Eugene	Huang	Seaford	Victoria	Australia	3198
4	11002 AW00011002	Ruben	Torres	Hobart	Tasmania	Australia	7001
5	11003 AW00011003	Christy	Zhu	North Ryde	New South Wales	Australia	2113
6	11004 AW00011004	Elizabeth	Johnson	Wollongong	New South Wales	Australia	2500
7	11005 AW00011005	Julio	Ruiz	East Brisbane	Queensland	Australia	4169
8	11006 AW00011006	Janet	Alvarez	Matraville	New South Wales	Australia	2036
9	11007 AW00011007	Marcos	Mehta	Warrnambool	Victoria	Australia	5280
10	11008 AW00011008	Rob	Vernhoff	Bendigo	Victoria	Australia	3550
11	11009 AW00011009	Shannon	Carlson	Hervey Bay	Queensland	Australia	4655
12	11010 AW00011010	Jacquelyn	Suarez	East Brisbane	Queensland	Australia	4168
13	11021 AW00011011	Curtis	Lu	East Brisbane	Queensland	Australia	4169
14	11022 AW00011012	Lauren	Walker	Bremerton	Washington	United States	98312
15	11023 AW00011013	Ian	Jenkins	Lebanon	Oregon	United States	97355
16	11024 AW00011014	Sydney	Bennett	Redmond	Washington	United States	98052
17	11025 AW00011015	Chloe	Young	Burbank	California	United States	91502
18	11026 AW00011016	Wyatt	Hill	Imperial Beach	California	United States	91952
19	11027 AW00011017	Shannon	Wang	Sunbury	Victoria	Australia	3429
20	11028 AW00011018	Clarence	Rai	Bendigo	Victoria	Australia	3550
21	11029 AW00011019	Luke	Lai	Langley	British Columbia	Canada	V3A 4R2
22	11030 AW00011020	Jordan	King	Methow	British Columbia	Canada	V9
23	11021 AW00011021	Destiny	Wilson	Beaverton	Oregon	United States	97005
24	11022 AW00011022	Ethan	Zhang	Bellingham	Washington	United States	98225
25	11023 AW00011023	Seth	Edwards	Bellflower	California	United States	90706
26	11024 AW00011024	Russell	Xie	Concord	California	United States	94519
27	11025 AW00011025	Alejandro	Beck	Hawthorne	Queensland	Australia	4171
28	11026 AW00011026	Harold	Sai	Goulburn	New South Wales	Australia	2580
29	11027 AW00011027	Jessie	Zhao	Warrnambool	Victoria	Australia	3280
30	11028 AW00011028	Ill	Imanou	Sr Leonards	New South Wales	Australia	2965

8 COLUMNS, 999+ ROWS Column profiling based on top 1000 rows

PREVIEW DOWNLOADED AT 1641

Transformation 2:

Replacing AW000 in Customer ID column with T20

CustomerKey CustomerID FirstName LastName City

Copy Remove Remove Other Columns Duplicate Column Add Column From Examples... Remove Duplicates Remove Errors Change Type Transform Replace Values... Replace Errors...

Replace Values

Replace one value with another in the selected columns.

Value To Find: AW000
Replace With: T20

Advanced options

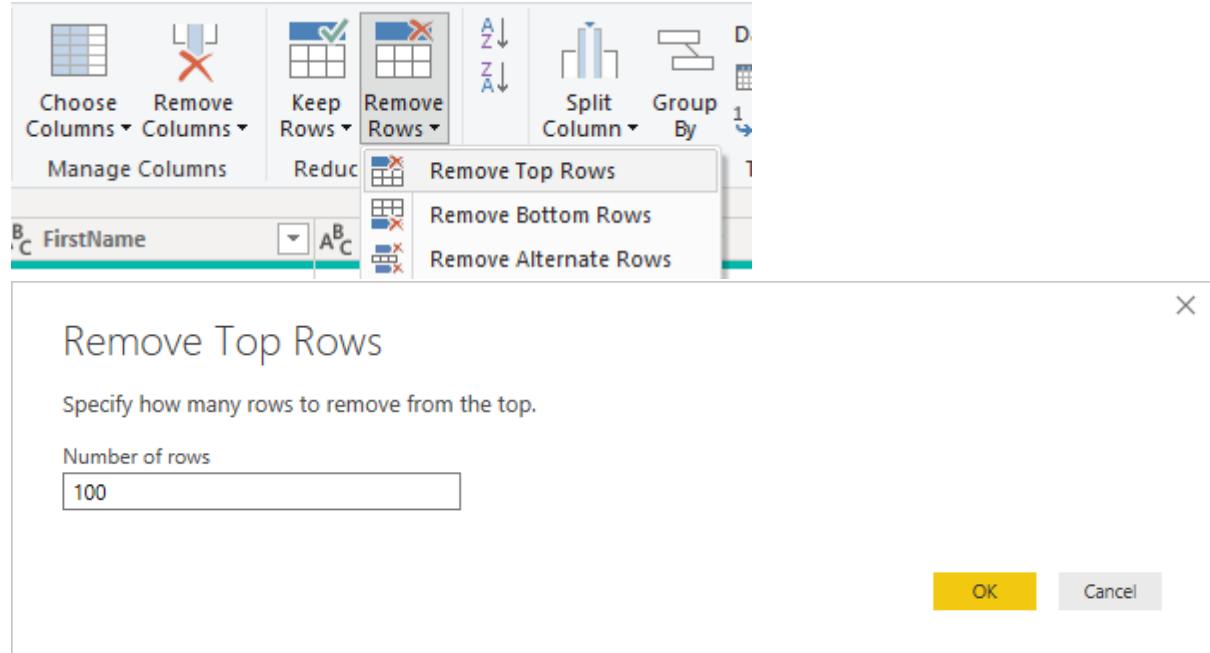
OK Cancel

Customer

	CustomerKey	CustomerID	FirstName	LastName	City	StateProvince	CountryRegion	PostalCode
1	11000	T201000	Jon	Yang	Rockhampton	Queensland	Australia	4700
2	11001	T201001	Eugene	Huang	Seaford	Victoria	Australia	3198
3	11002	T201002	Ruben	Torres	Hobart	Tasmania	Australia	7001
4	11003	T201003	Christy	Zhu	North Ryde	New South Wales	Australia	2113
5	11004	T201004	Elizabeth	Johnson	Wollongong	New South Wales	Australia	2500
6	11005	T201005	Julio	Ruiz	East Brisbane	Queensland	Australia	4169
7	11006	T201006	Janet	Alvarez	Mattaville	New South Wales	Australia	2036
8	11007	T201007	Marcos	Mehta	Warrnambool	Victoria	Australia	5280
9	11008	T201008	Rob	Vernhoff	Bendigo	Victoria	Australia	3550
10	11009	T201009	Shannon	Carlson	Hervey Bay	Queensland	Australia	4655
11	11010	T201010	Jacquelyn	Suarez	East Brisbane	Queensland	Australia	4169
12	11011	T201011	Curtis	Lu	East Brisbane	Queensland	Australia	4169
13	11012	T201012	Lauren	Walker	Bremerton	Washington	United States	98312
14	11013	T201013	Ian	Jenkins	Lebanon	Oregon	United States	97355
15	11014	T201014	Sydney	Bennett	Redmond	Washington	United States	98052
16	11015	T201015	Chloe	Young	Burbank	California	United States	91502
17	11016	T201016	Wyatt	Hill	Imperial Beach	California	United States	91952
18	11017	T201017	Shannon	Wang	Sunbury	Victoria	Australia	3429
19	11018	T201018	Clarence	Rai	Bendigo	Victoria	Australia	3550
20	11019	T201019	Luke	Lai	Langley	British Columbia	Canada	V3A 4R2
21	11020	T201020	Jordan	King	Methow	British Columbia	Canada	V9
22	11021	T201021	Destiny	Wilson	Beaverton	Oregon	United States	97005
23	11022	T201022	Ethan	Zhang	Bellingham	Washington	United States	98225
24	11023	T201023	Seth	Edwards	Bellflower	California	United States	90706
25	11024	T201024	Russell	Xie	Concord	California	United States	94519
26	11025	T201025	Alejandro	Beck	Hawthorne	Queensland	Australia	4171
27	11026	T201026	Harold	Sai	Goulburn	New South Wales	Australia	2580
28	11027	T201027	Jessie	Zhao	Warrnambool	Victoria	Australia	3280
29	11028	T201028	Ill	Imaneer	Sr Leonard	New South Wales	Australia	2965

Transformation 3:

Deleting first 100 records of all columns



Untitled - Power Query Editor

File Home Transform Add Column View Help

Close & Apply Close New Source... Data Sources Data source settings Manage Parameters Refresh Preview Advanced Editor Properties Choose Columns... Remove Columns... Keep Rows... Remove Rows... Sort Split Column... Group By Replace Values Data Type: Text Merge Queries Use First Row as Headers Append Queries Combine Files

Queries [1] Customer

	CustomerKey	CustomerID	FirstName	LastName	City	StateProvince	CountryRegion	PostalCode
1	1109	T201109	Adam	Ross	Port Macquarie	New South Wales	Australia	2444
2	1110	T201100	Latache	Navarro	Cloverdale	South Australia	Australia	6105
3	1110	T201101	Abby	Sai	Cranbourne	Victoria	Australia	3977
4	1110	T201102	Julia	Nelson	Sydney	New South Wales	Australia	1002
5	1110	T201103	Cassie	Chande	Darlinghurst	New South Wales	Australia	2010
6	1110	T201104	Edgar	Sara	Gold Coast	Queensland	Australia	4217
7	1110	T201105	Candace	Fernandez	Mattaville	New South Wales	Australia	2036
8	1110	T201106	Jessie	Liu	Warrnambool	Victoria	Australia	3280
9	1110	T201107	Bianca	Lin	St Leonards	New South Wales	Australia	2065
10	1108	T201108	Kari	Alvarez	Port Macquarie	New South Wales	Australia	2444
11	1109	T201109	Ruben	Kapoor	South Melbourne	Victoria	Australia	3205
12	1110	T201110	Curtis	Yang	Darlinghurst	New South Wales	Australia	2010
13	1112	T201111	Meredith	Gutierrez	Geelong	Victoria	Australia	3220
14	1112	T201112	Crystal	Wang	Hervey Bay	Queensland	Australia	4655
15	1113	T201113	Michael	Bianco	Perth	South Australia	Australia	6006
16	1114	T201114	Leslie	Moreno	Rhodes	New South Wales	Australia	2138
17	1115	T201115	Alvin	Cal	St Leonards	New South Wales	Australia	2065
18	1116	T201116	Clinton	Carlson	Sydney	New South Wales	Australia	1002
19	1117	T201117	April	Deng	Mattaville	New South Wales	Australia	2036
20	1118	T201118	Alvin	Zeng	Lavender Bay	New South Wales	Australia	2060
21	1119	T201119	Evan	James	Port Macquarie	New South Wales	Australia	2444
22	1120	T201120	Beth	Jimenez	Sydney	New South Wales	Australia	1002
23	1121	T201121	Orlando	Suarez	Newcastle	New South Wales	Australia	2300
24	1122	T201122	Byron	Vazquez	North Sydney	New South Wales	Australia	2055
25	1124	T201123	Philip	Alvarez	Sunbury	Victoria	Australia	3429
26	1124	T201124	Ross	Jordan	Warrnambool	Victoria	Australia	3280
27	1125	T201125	Dana	Navarro	Seaford	Victoria	Australia	3198
28	1126	T201126	Shaun	Carson	North Ryde	New South Wales	Australia	2113
29	1127	T201127	Jan	Edwards	San Gabriel	California	United States	91776
30	1128	T201128	Samantha	Ione	Brenton	Washington	United States	98055

8 COLUMNS, 999+ ROWS Column profiling based on top 1000 rows

PREVIEW DOWNLOADED AT 1641

Transformation 4:

Converting Country-Region column values to uppercase

The screenshot shows the Power Query Editor interface with a table containing the 'Country-Region' column. The column values are all set to 'Australia'. A context menu is open over this column, with the 'Transform' option selected. Under 'Transform', the 'Change Type' option is also selected. In the 'Applied Steps' pane, there is a step labeled 'Country-Region' with a dropdown arrow. The dropdown shows several options: 'lowercase', 'UPPERCASE' (which is highlighted), 'Capitalize Each Word', and 'Trim'. This indicates that the transformation step applied to the column converts the text to uppercase.

Untitled - Power Query Editor

File Home Transform Add Column View Help

Queries [1] Customer

	CustomerKey	CustomerID	FirstName	LastName	City	State-Province	Country-Region	Postal Code
1	11099	T2011099	Adam	Ross	Port Macquarie	New South Wales	AUSTRALIA	2444
2	11100	T2011100	Larsa	Navarro	Cloverdale	South Australia	AUSTRALIA	6105
3	11101	T2011101	Abby	Sai	Cranbourne	Victoria	AUSTRALIA	3977
4	11102	T2011102	Julia	Nelson	Sydney	New South Wales	AUSTRALIA	1002
5	11103	T2011103	Cassie	Chande	Darlinghurst	New South Wales	AUSTRALIA	2010
6	11104	T2011104	Edgar	Sara	Gold Coast	Queensland	AUSTRALIA	4217
7	11105	T2011105	Candace	Fernandez	Matraville	New South Wales	AUSTRALIA	2036
8	11106	T2011106	Jessie	Liu	Warrnambool	Victoria	AUSTRALIA	3280
9	11107	T2011107	Bianca	Lin	St. Leonards	New South Wales	AUSTRALIA	2065
10	11108	T2011108	Kari	Alvarez	Port Macquarie	New South Wales	AUSTRALIA	2444
11	11109	T2011109	Ruben	Kapoor	South Melbourne	Victoria	AUSTRALIA	3205
12	11110	T2011110	Curtis	Yang	Darlinghurst	New South Wales	AUSTRALIA	2010
13	11121	T2011111	Meredith	Gutierrez	Geelong	Victoria	AUSTRALIA	3220
14	11122	T2011112	Crystal	Wang	Hervey Bay	Queensland	AUSTRALIA	4655
15	11123	T2011113	Michael	Bianco	Perth	South Australia	AUSTRALIA	6006
16	11124	T2011114	Leslie	Moreno	Rhodes	New South Wales	AUSTRALIA	2138
17	11125	T2011115	Alvin	Cal	St. Leonards	New South Wales	AUSTRALIA	2065
18	11126	T2011116	Clinton	Carlson	Sydney	New South Wales	AUSTRALIA	1002
19	11127	T2011117	April	Deng	Matraville	New South Wales	AUSTRALIA	2036
20	11128	T2011118	Alvin	Zeng	Lavender Bay	New South Wales	AUSTRALIA	2060
21	11129	T2011119	Evan	James	Port Macquarie	New South Wales	AUSTRALIA	2444
22	11120	T2011120	Beth	Jimenez	Sydney	New South Wales	AUSTRALIA	1002
23	11121	T2011121	Orlando	Suarez	Newcastle	New South Wales	AUSTRALIA	2300
24	11122	T2011122	Byron	Vazquez	North Sydney	New South Wales	AUSTRALIA	2055
25	11123	T2011123	Philip	Alvarez	Sunbury	Victoria	AUSTRALIA	3429
26	11124	T2011124	Ross	Jordan	Warrnambool	Victoria	AUSTRALIA	3280
27	11125	T2011125	Dana	Navarro	Seaford	Victoria	AUSTRALIA	3198
28	11126	T2011126	Shaun	Carson	North Ryde	New South Wales	AUSTRALIA	2113
29	11127	T2011127	Jan	Edwards	San Gabriel	California	UNITED STATES	91776
30	11128	T2011128	Samantha	Ione	Renton	Washington	UNITED STATES	98055

8 COLUMNS, 999+ ROWS Column profiling based on top 1000 rows

PREVIEW DOWNLOADED AT 1641

Finally click on "close and apply" to import

Untitled - Power Query Editor

File Home Transform Add Column View Help

Queries [1] Customer

City	LastName	Postal Code	Country-Region	Customer ID	CustomerKey	FirstName
Ballard	Adams	98107	UNITED STATES	T2016877	1	Katelyn
Ballard	Allen	98107	UNITED STATES	T2019663	1	Alex
Ballard	Baker	98107	UNITED STATES	T2029036	1	Rebecca
Ballard	Brown	98107	UNITED STATES	T2027036	1	Hunter
Ballard	Cai	98107	UNITED STATES	T2025898	1	Brittney
Ballard	Chande	98107	UNITED STATES	T2016229	1	Wayne
Ballard	Clark	98107	UNITED STATES	T2028255	1	Robert
Ballard	Cox	98107	UNITED STATES	T2012366	1	Stephanie
Ballard	Cox	98107	UNITED STATES	T2027730	1	Isabella
Ballard	Davis	98107	UNITED STATES	T2014404	1	Rachel
Ballard	Diaz	98107	UNITED STATES	T2011785	1	Theodore
Ballard	Flores	98107	UNITED STATES	T2023193	1	Alexis
Ballard	Gill	98107	UNITED STATES	T2023246	1	Renee
Ballard	Green	98107	UNITED STATES	T2027548	1	Mason
Ballard	Hall	98107	UNITED STATES	T2021296	1	Luis
Ballard	Hayes	98107	UNITED STATES	T2015512	1	Jose
Ballard	He	98107	UNITED STATES	T2022741	1	Darryl
Ballard	Hernandez	98107	UNITED STATES	T2011144	1	Edward
Ballard	Hernandez	98107	UNITED STATES	T2016022	1	Devin
Ballard	Hernandez	98107	UNITED STATES	T2018813	1	Kristy
Ballard	Hernandez	98107	UNITED STATES	T2018887	1	Kari
Ballard	Hill	98107	UNITED STATES	T2013926	1	Dalton
Ballard	Hill	98107	UNITED STATES	T2023381	1	Rianca
Total					18385	

Filters on this visual

- City is (All)
- Country-Region is (All)
- Customer ID is (All)
- CustomerKey is (All)
- CustomerName is (All)
- FirstName is (All)
- LastName is (All)
- Postal Code is (All)
- Country-Region is (All)
- Customer ID is (All)
- CustomerKey is (All)
- FirstName is (All)

Values

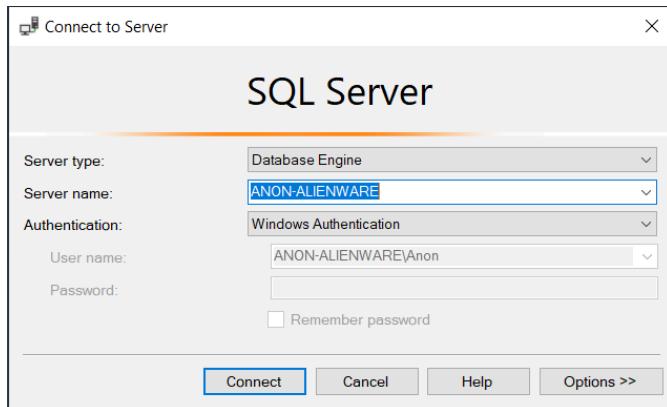
- City
- LastName
- Postal Code
- Country-Region
- Customer ID
- CustomerKey
- FirstName

Drillthrough

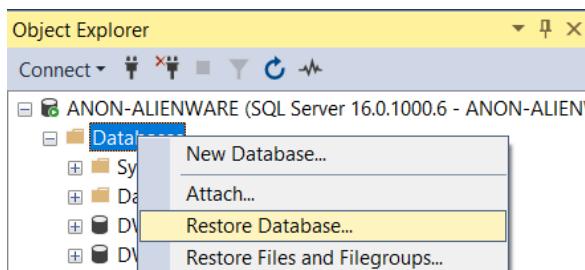
Cross-report

In SQL Server Data Tools.

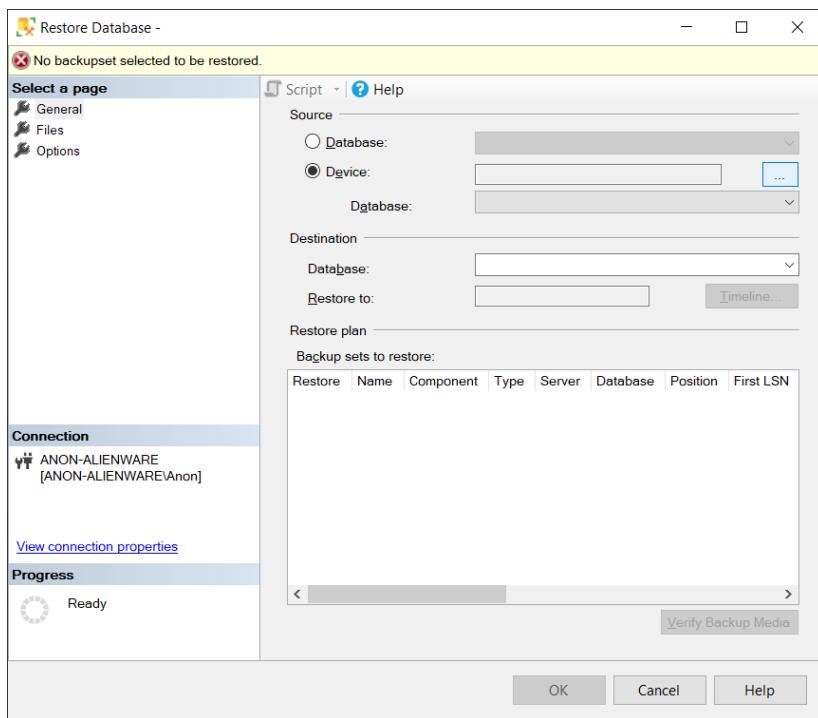
Step 1: Open SQL server management studio to restore backup file.

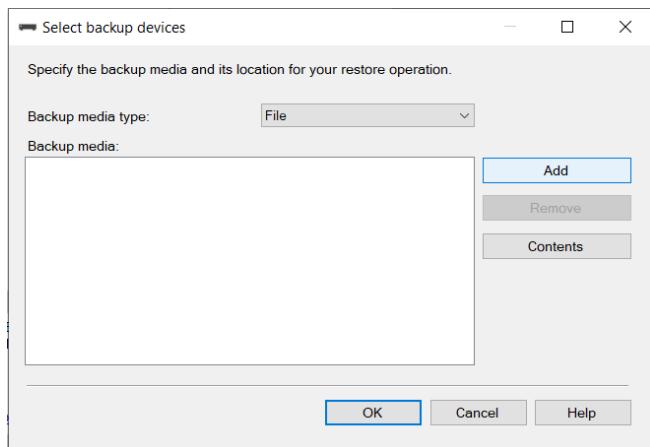


Step 2: Right click on the databases -> restore database

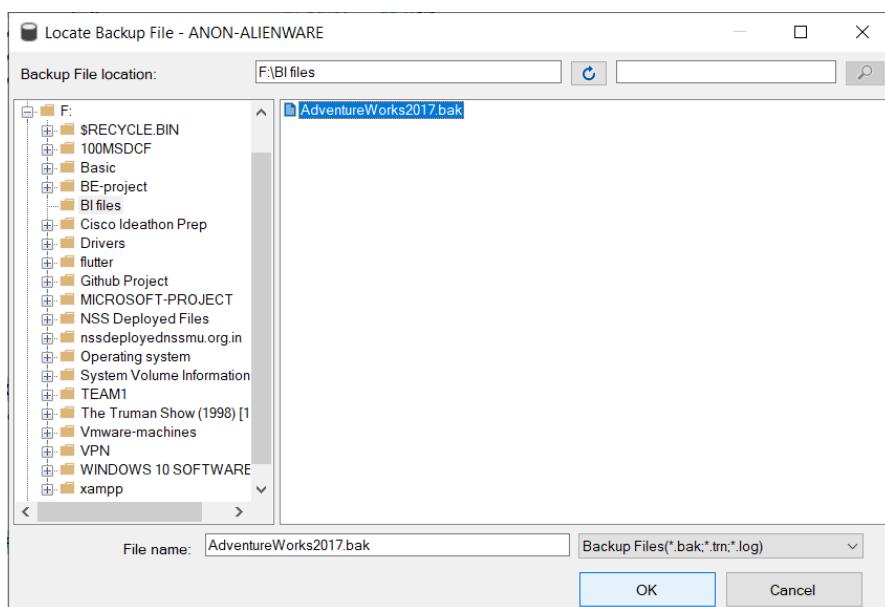


Step 3 : Select device -> click on 3 dots icon towards the end of device box

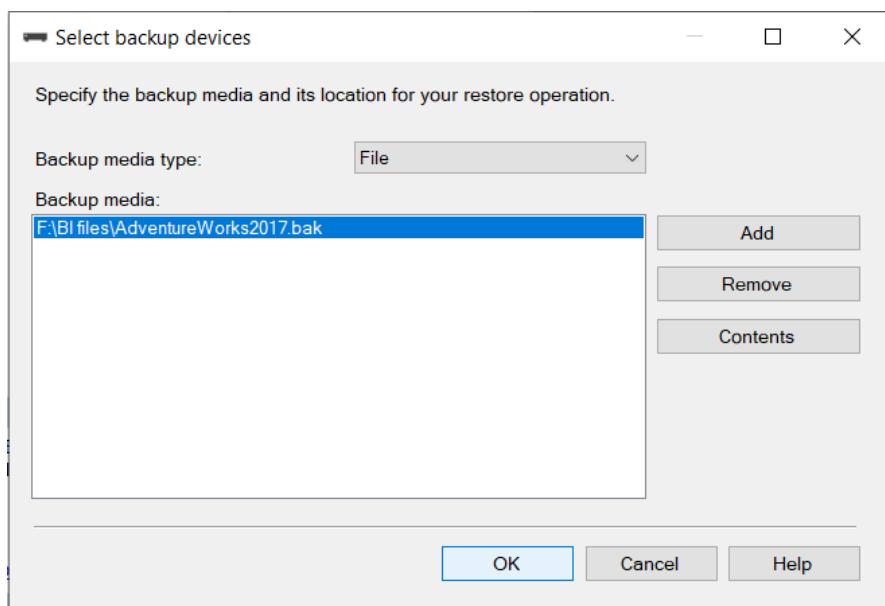




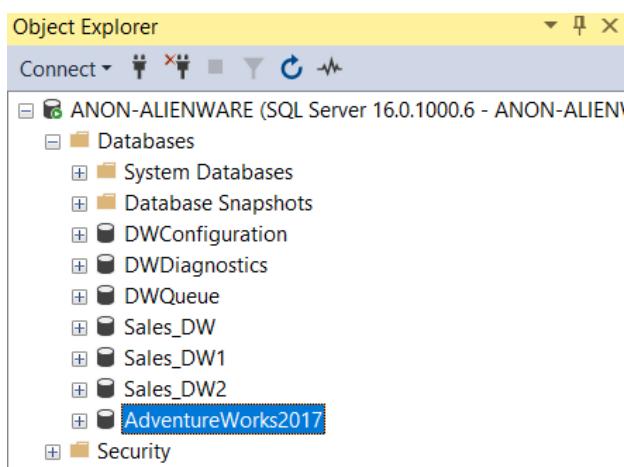
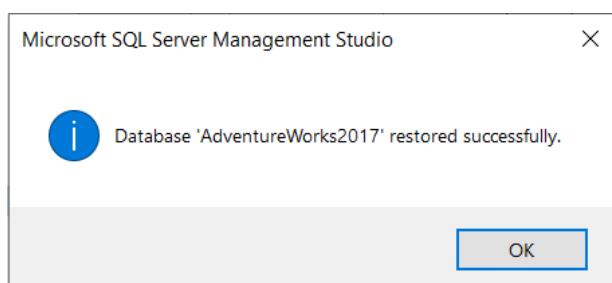
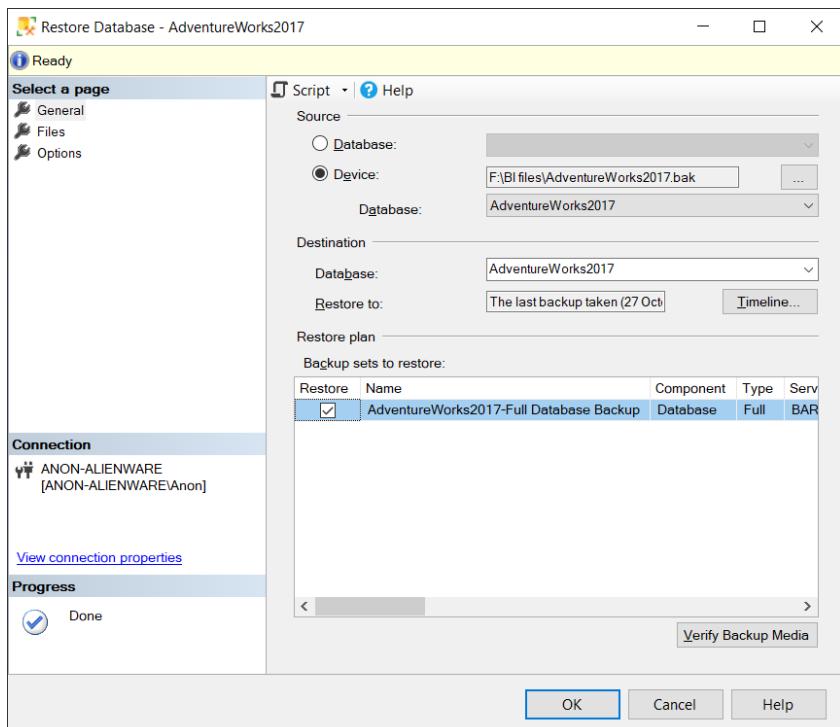
Step 4: Click on Add -> the path of backup file



Step 5: Select the file



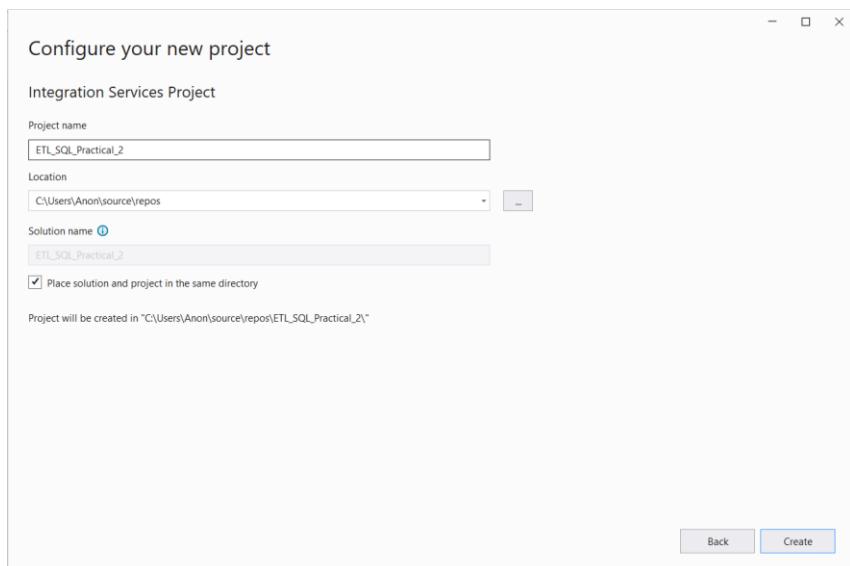
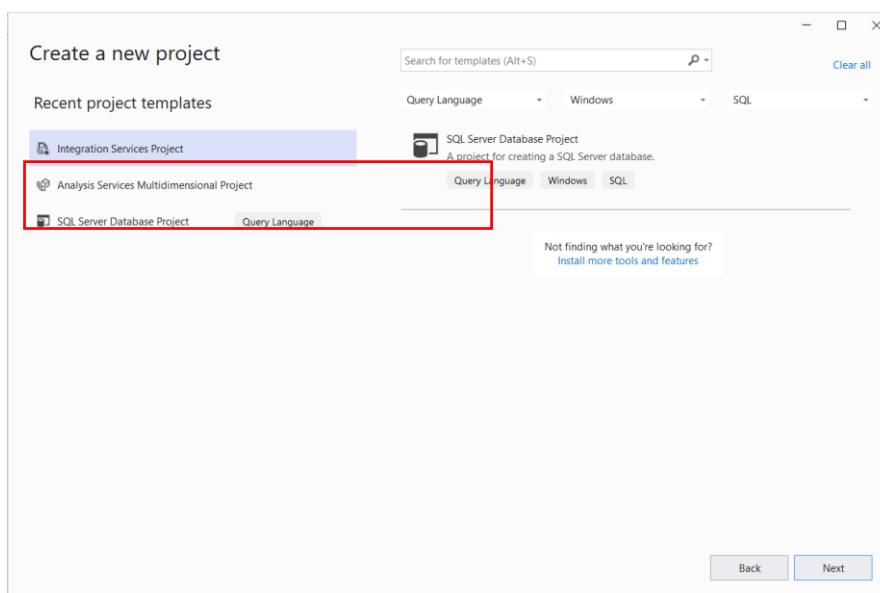
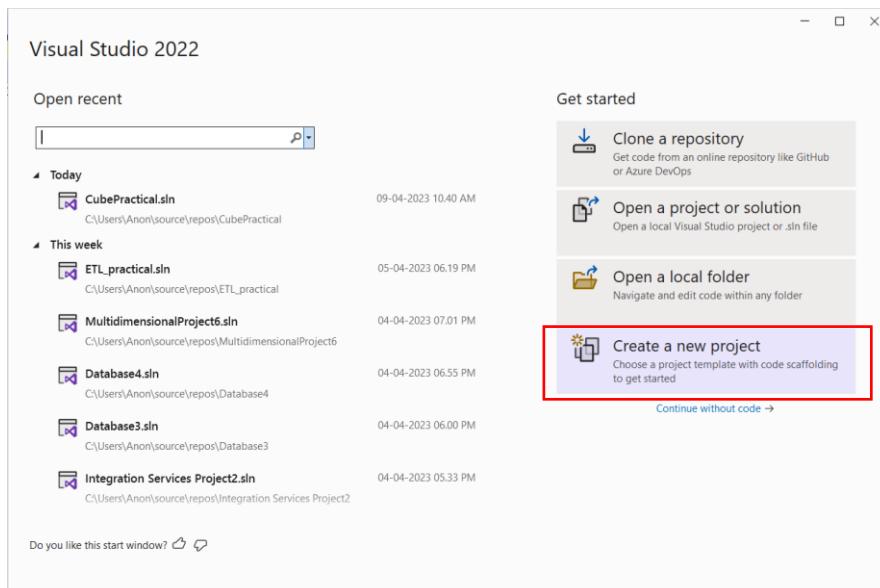
Step 6: Click on OK



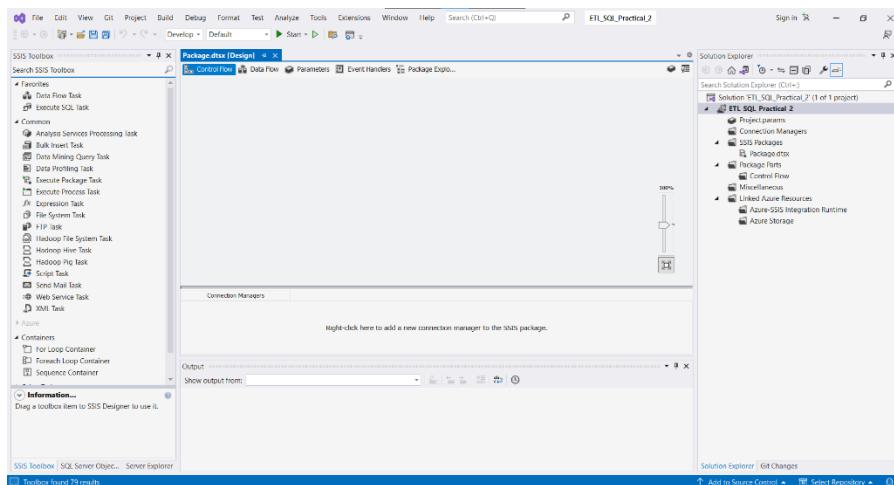
The database is restored successfully.

Step 7: Open SQL server data tools

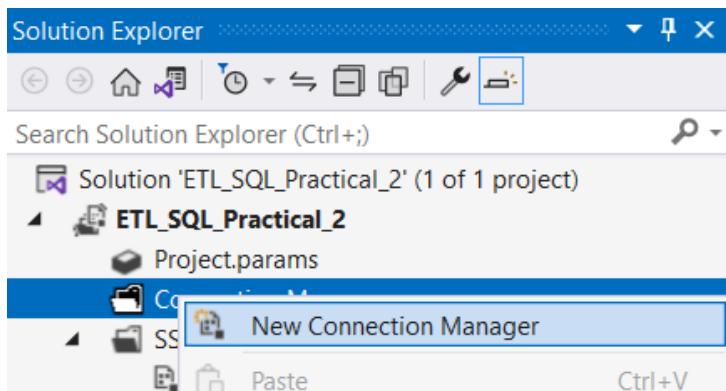
Select File -> new -> project -> Business Intelligence -> **Integration services project** & give project name



Environment consist of SQL Server Integration Services (SSIS)

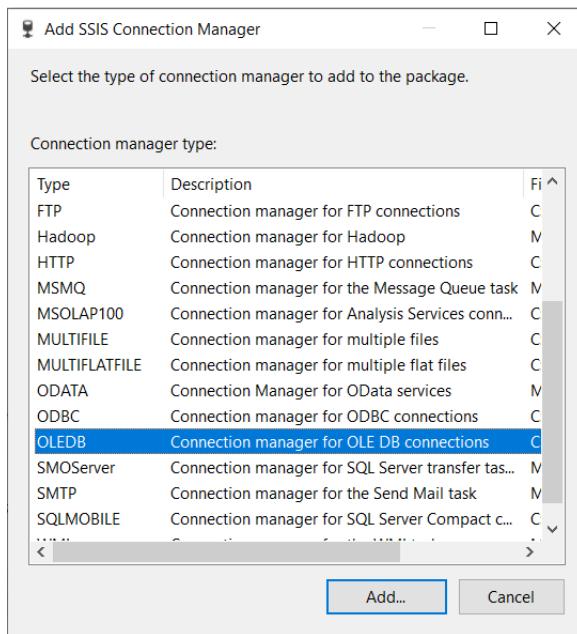


Step 8: Right click on connection manager in solution explorer and click on new connection manager

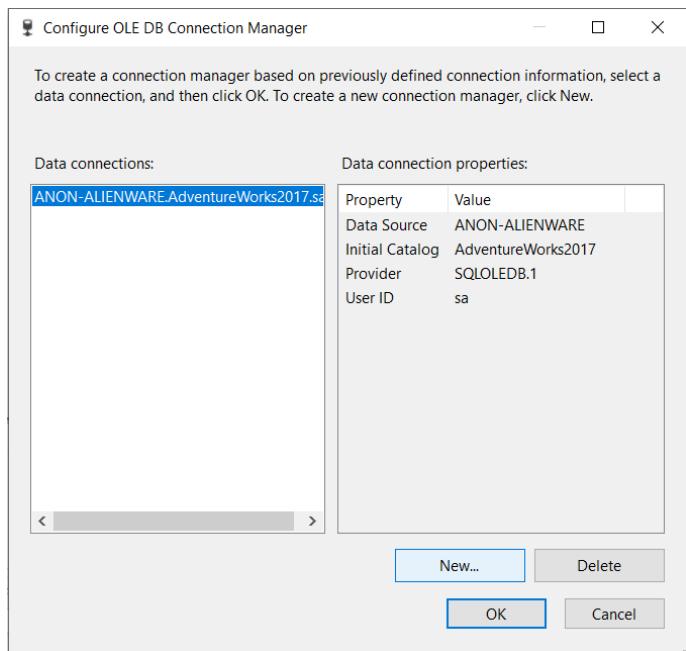


Add SSIS connection manager window appears

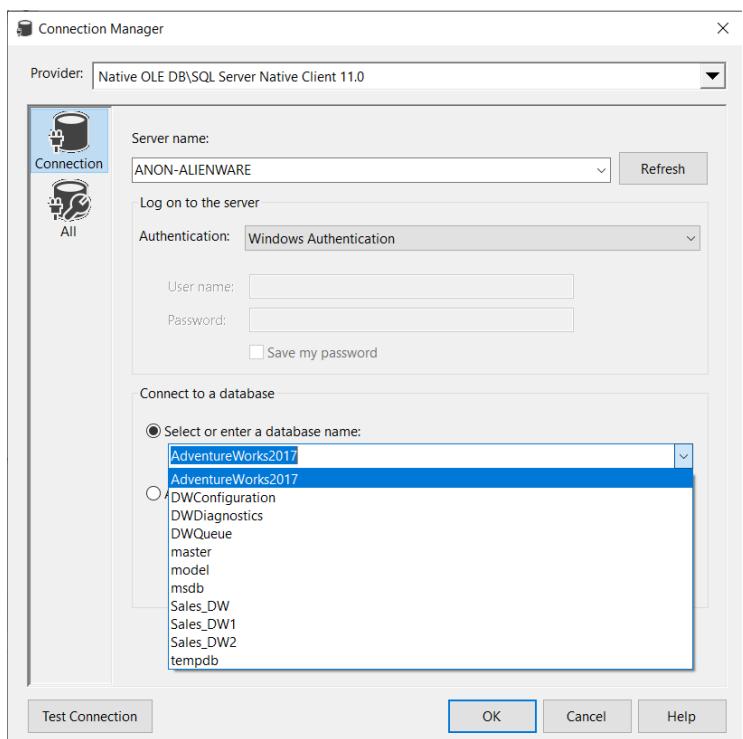
Step 9: Select OLEDB Connection Manager and Click on Add



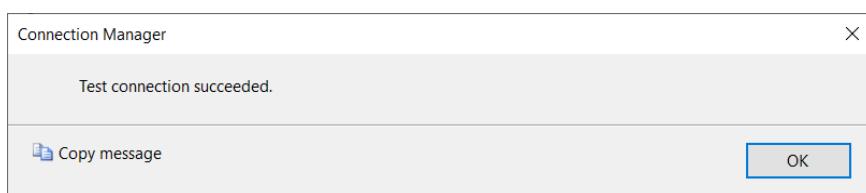
Step 10: Configure OLE DB Connection Manager window appears -> click on new



Step 11: Select Server name from drop down and database name and click on test connection

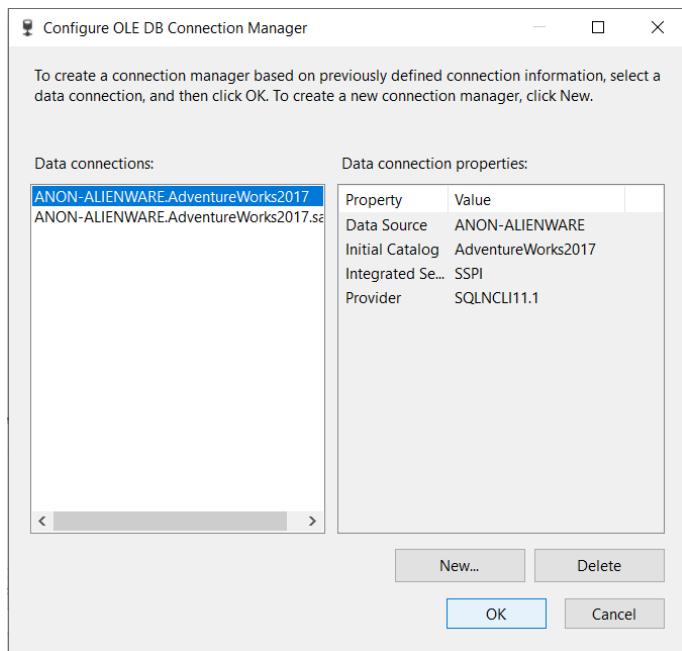


The 'Connection Manager' dialog box shows the configuration for a connection named 'ANON-ALIENWARE'. The provider is set to 'Native OLE DB\SQL Server Native Client 11.0'. Under 'Log on to the server', 'Authentication' is set to 'Windows Authentication'. The 'User name' and 'Password' fields are empty, and the 'Save my password' checkbox is unchecked. In the 'Connect to a database' section, the 'Select or enter a database name:' radio button is selected, and the dropdown menu shows several database options, with 'AdventureWorks2017' currently selected. Buttons at the bottom include 'Test Connection', 'OK', 'Cancel', and 'Help'.

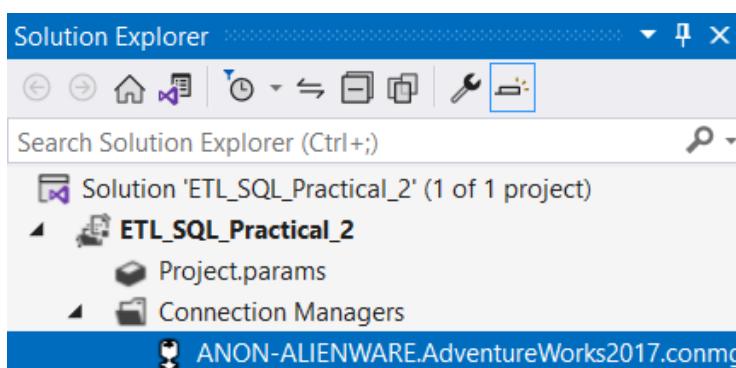


The 'Connection Manager' dialog box displays a message: 'Test connection succeeded.' A 'Copy message' button is available, and an 'OK' button is at the bottom right.

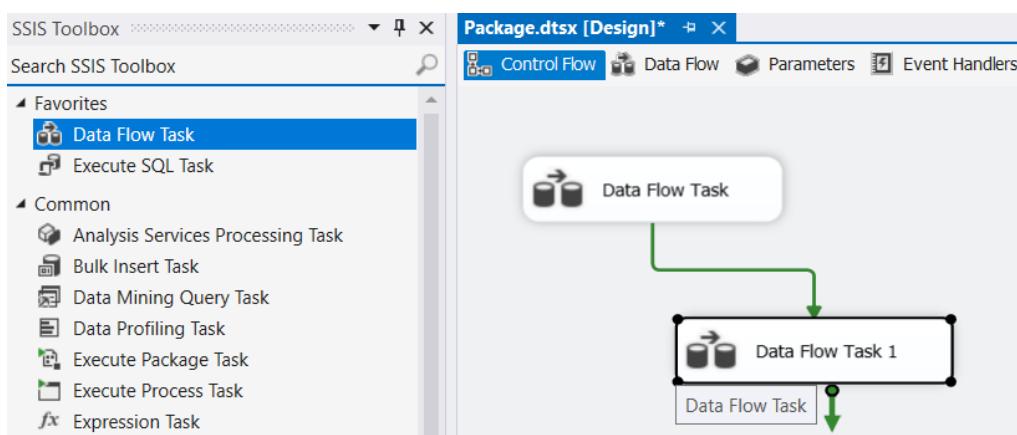
Step 12: Click on OK



Connection is added in connection manager

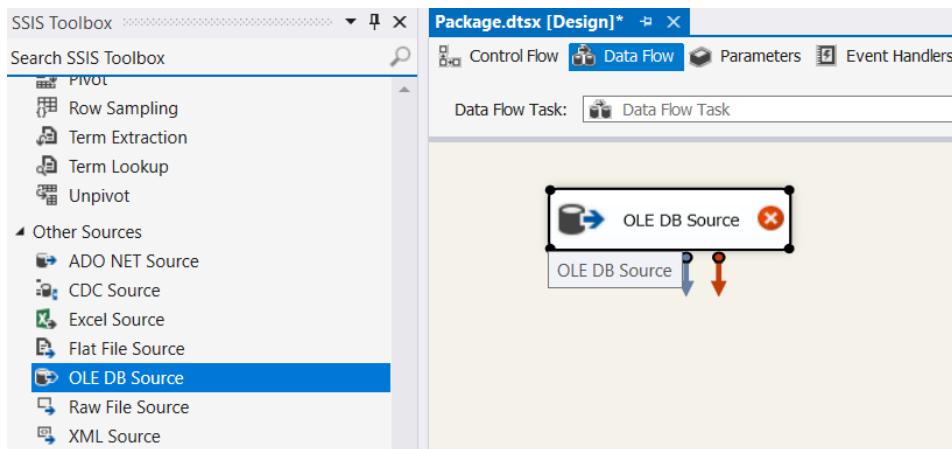


Step 13: Drag and drop **Data Flow Task** in **Control Flow Tab**

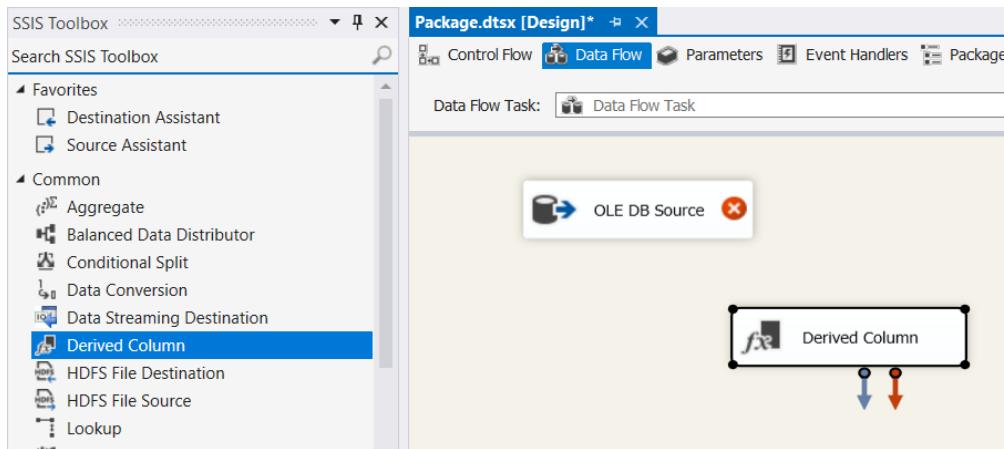


Double click on data flow task

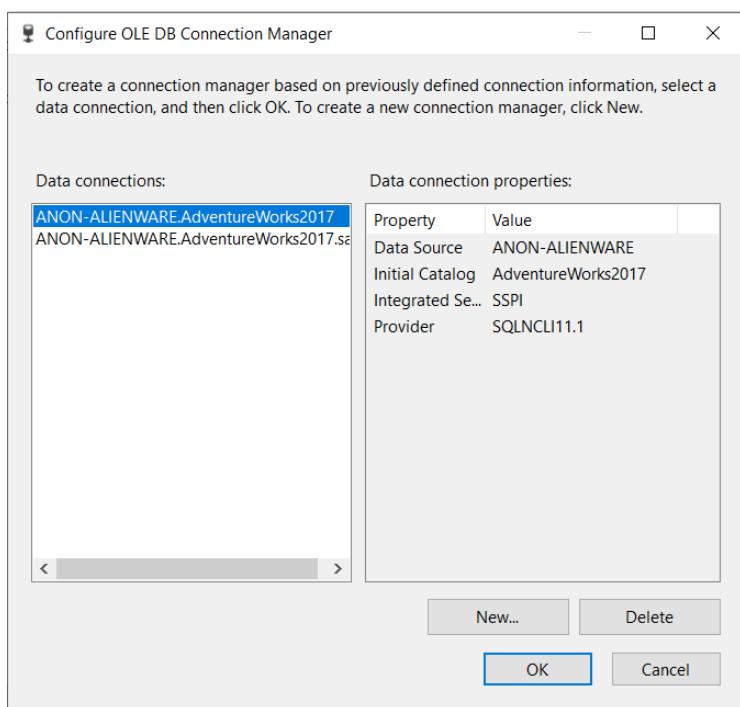
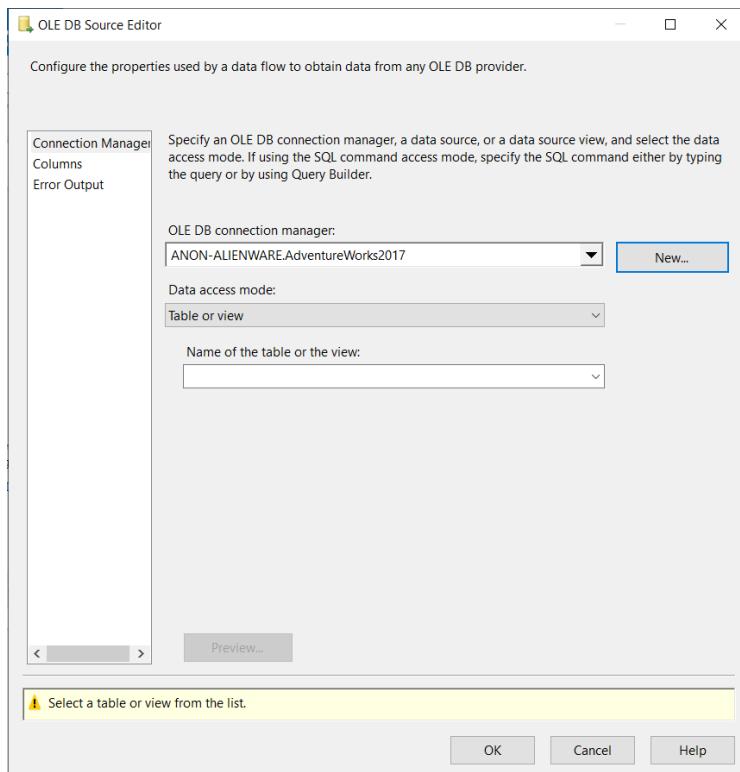
Step 14: Drag and drop **OLE DB Source** from other source into **Data Flow tab**



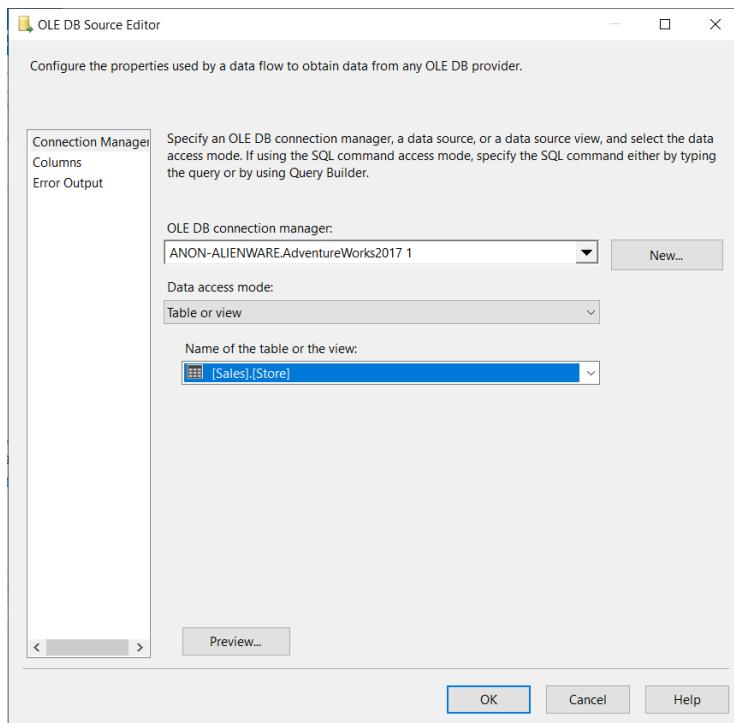
Add derived column from SSIS toolbox -> common



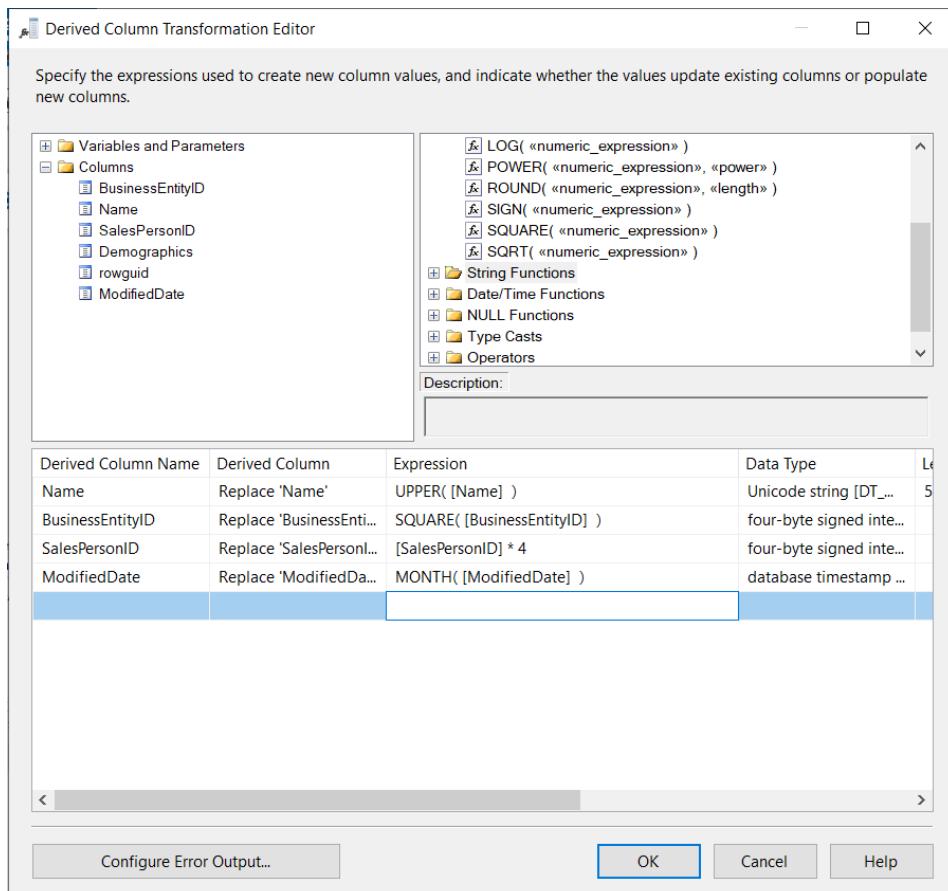
Step 15: Double click on OLE DB Source -> OLE DB source editor appears -> click on new to add connection manager



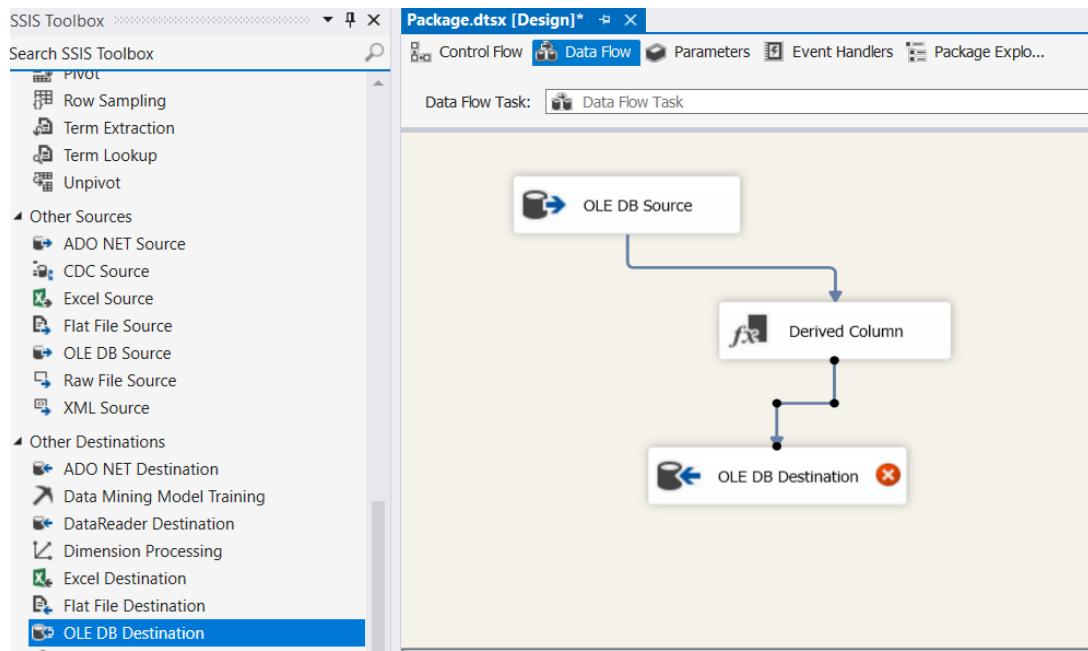
Select [Sales].[Store] table from drop down -> OK



Step 16: Connect OLE DB Source to derived column and Double click on derived column and add the expressions of ETL

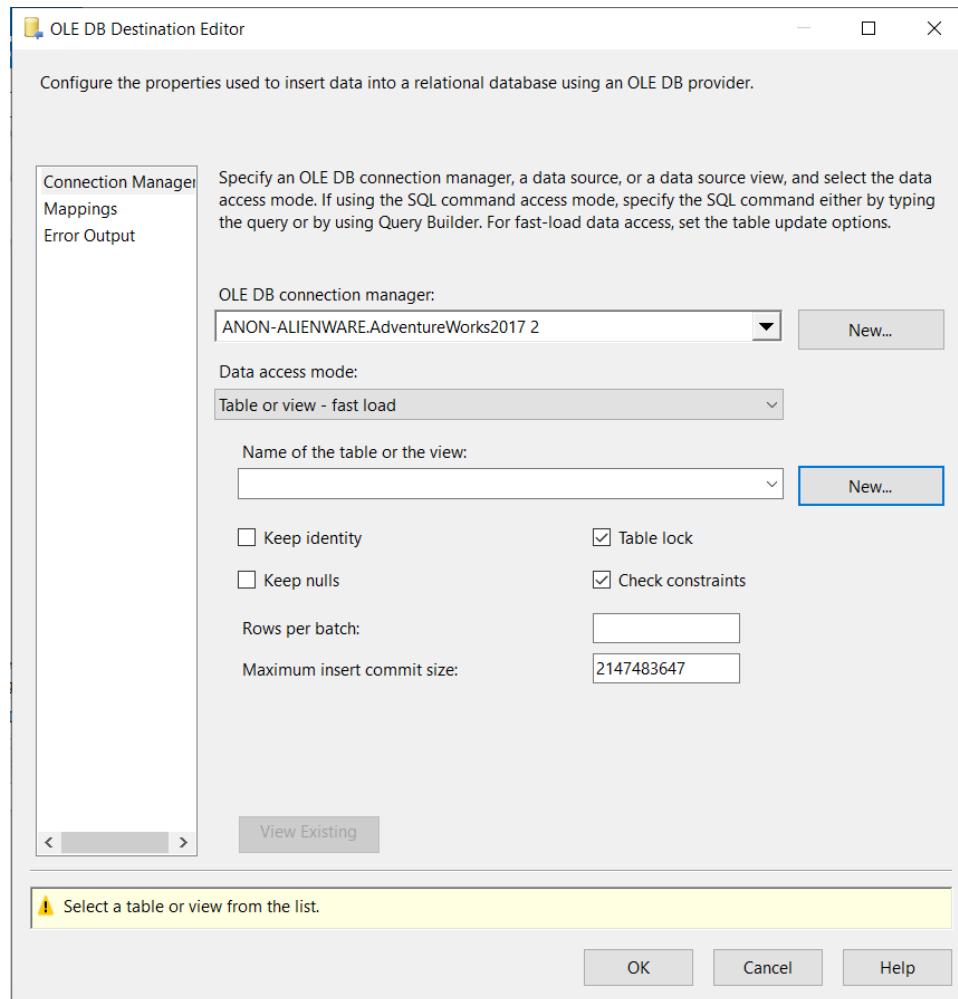


Step 17: Add OLE DB Destination in data flow and connect the derived column to it



Step 18: Double click on OLE DB Destination

Click on new to run the query to get [OLE DB DESTINATION] in name of the table or the view



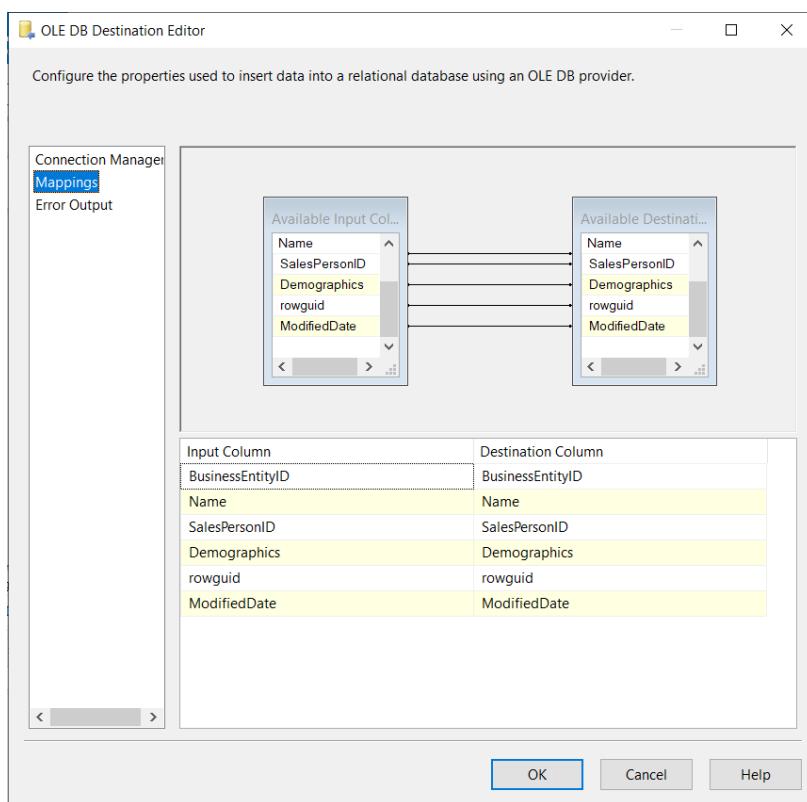
Change the name of table

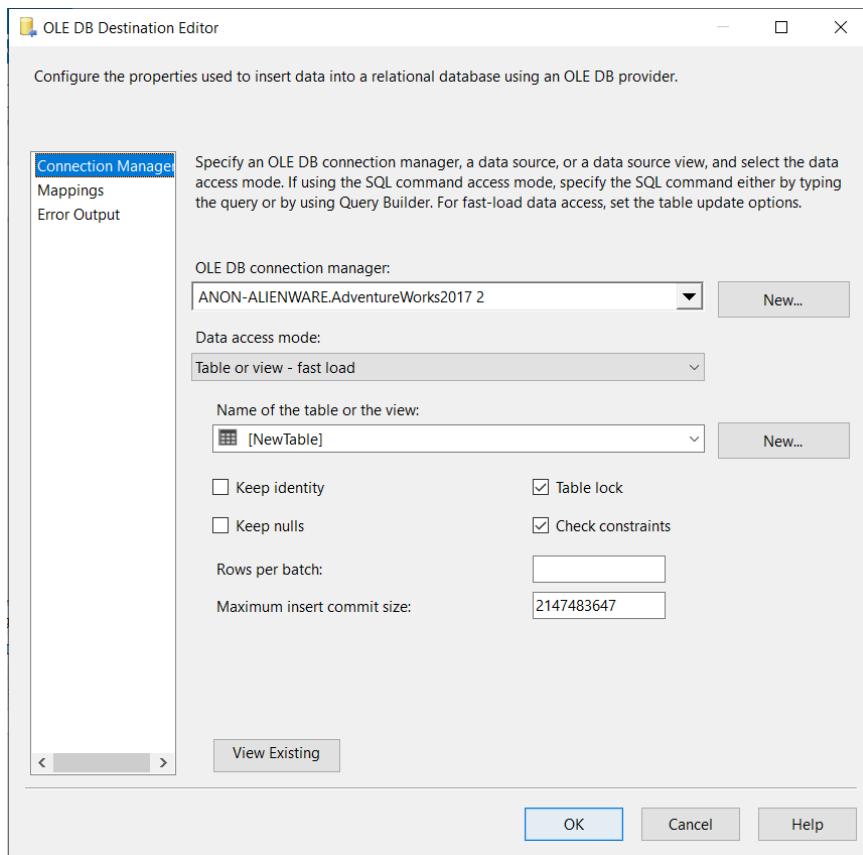
Create Table

```
CREATE TABLE [NewTable] (
    [BusinessEntityID] int,
    [Name] nvarchar(50),
    [SalesPersonID] int,
    [Demographics] nvarchar(max),
    [rowguid] uniqueidentifier,
    [ModifiedDate] datetime
)
```

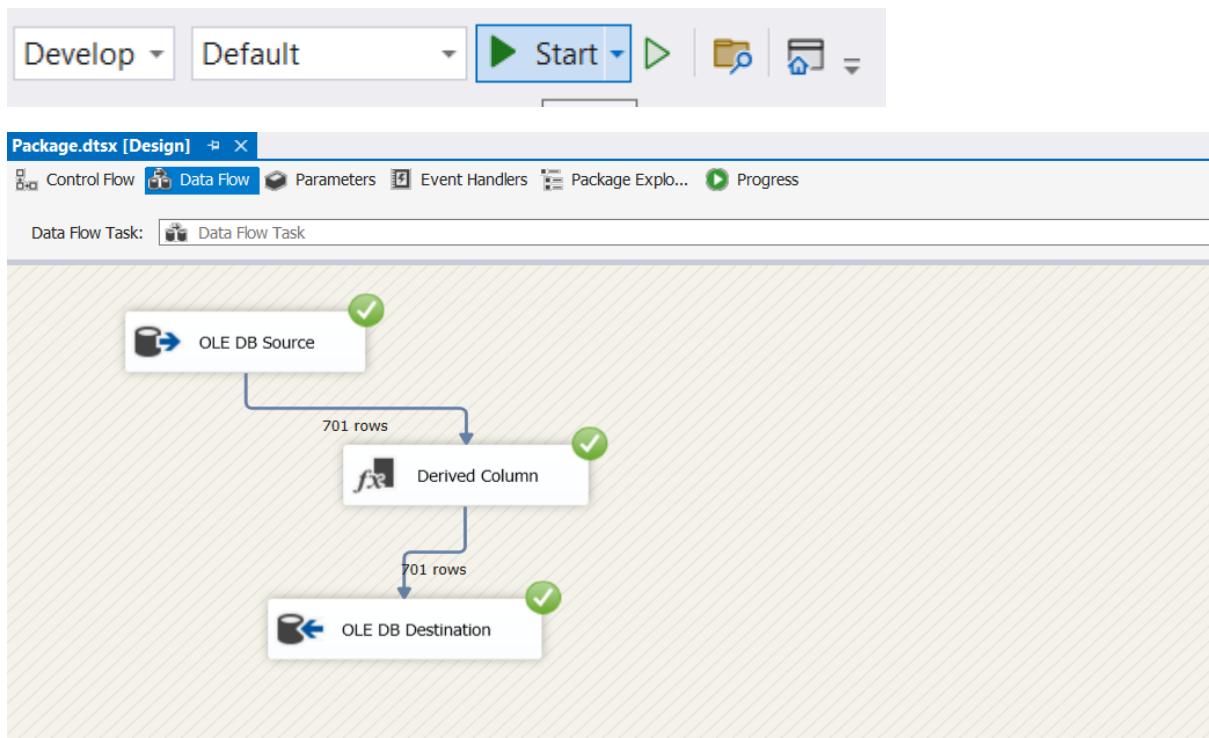
OK Cancel

Map the columns

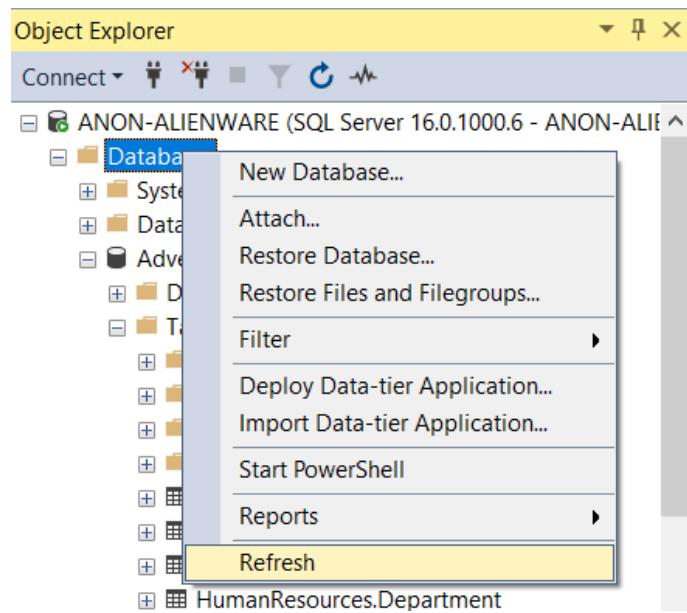




Click on Start button

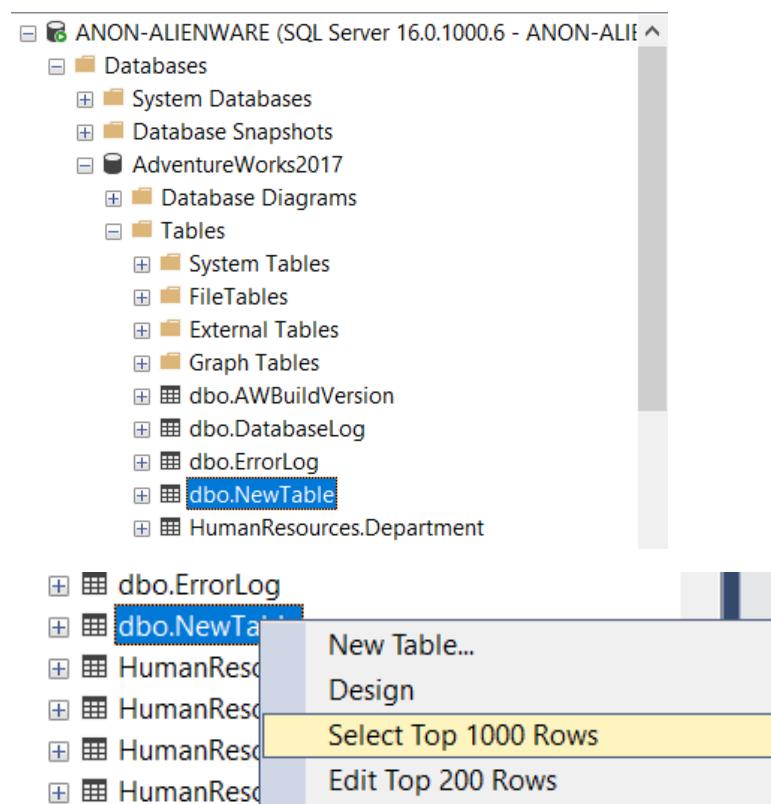


Refresh the Database



New table will be added to database

Right click on new table -> select top 100 rows



Final Output

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar indicates the session is for 'SQLQuery6.sql - ANON-ALIENWARE\AdventureWorks2017 (ANON-ALIENWARE\Anon (60))'. The Object Explorer sidebar lists various database objects like Tables, System Tables, and Views. The main pane displays a T-SQL script:

```
***** Script for SelectTopNRows command from SSMS *****
SELECT TOP (1000) [BusinessEntityID]
      ,[Name]
      ,[SalesPersonID]
      ,[Demographics]
      ,[rowguid]
      ,[ModifiedDate]
  FROM [AdventureWorks2017].[dbo].[NewTable]
```

Below the script, the Results tab shows the output of the query, which is a table with columns: BusinessEntityID, Name, SalesPersonID, Demographics, rowguid, and ModifiedDate. The table contains 701 rows of data, such as 'NEXT-DOOR BIKE STORE' and 'PROFESSIONAL SALES AND SERVICE'. The status bar at the bottom right shows 'Query executed successfully.' and 'ANON-ALIENWARE (16.0 RTM) - ANON-ALIENWARE\Anon (60) - AdventureWorks2017 00:00:00 701 rows'.

Practical 3

AIM: Create the cube with suitable dimensions and flat table olap/holap

Solution:

Steps

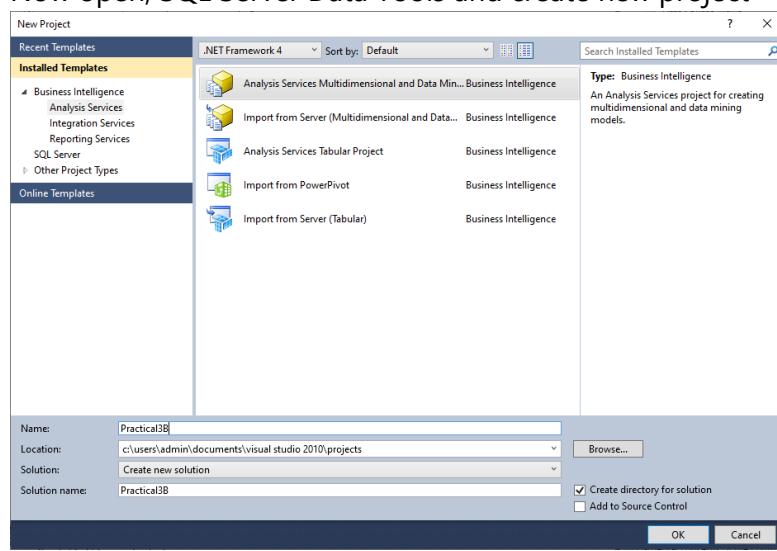
Download the script file:

https://www.codeproject.com/KB/database/652108/Data_WareHouse_SQLScript.zip

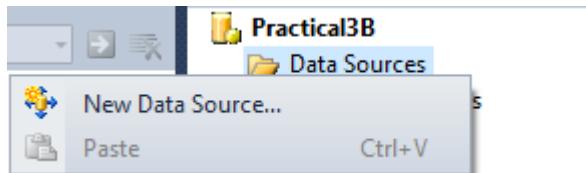
And execute it in the Sql server management tools;

```
File Database Project Tools Window Help
Object Explorer
Connect to... DESKTOP-JVQDSB99 [SQL Server 11.0.2180]
Databases Security Scripts Objects Replication AlwaysOn High Availability Management Integration Services Catalogs SQL Server Agent (Agent IP is disabled)
Data Warehouse [S_QDSB99\admin] X
-- SalesInvoicedata, SalesInvoiceKey, SalesTimeKey, StoreID, CustomerID, ProductID ,SalesPersonID,Quantity,ProductActualCost,SalesTotalCost,Deviation)
4,203130102,44347,121987,1,1,1,2,11,13,2),
4,203130102,44347,121987,1,2,1,1,22,59,24,1,5),
5,203130102,44519,122159,1,2,3,1,42,43,5,1,5),
5,203130102,44519,122159,1,2,4,1,3,54,60,6),
6,203130102,52415,143335,1,3,2,2,11,13,2),
6,203130102,52415,143335,1,3,5,2,1,139,139,4),
7,203130102,44347,121987,2,1,4,3,54,60,6),
7,203130102,44347,121987,2,1,5,3,1,139,139,4),
8,203130103,59326,162846,1,1,3,1,2,84,87,3),
8,203130103,59326,162846,1,1,4,3,54,60,3),
9,203130103,59349,162999,1,2,1,1,5,5,6,5,1),
10,203130103,59349,162999,1,2,1,1,5,5,6,5,1),
Results Messages
Database Date FullDateUK FullDateUSA DayMonth DaySuffix DayName DayOfWeekUSA DayOfWeekUK DayOfWeekInMonth DayOfWeekInYear DayOfQuarter DayOfYear WeekOfMonth WeekOfQuarter WeekOfYear Month MonthName MonthOfQuarter
1 2013-01-01 00:00:00.000 01/01/2013 01/01/2013 1st 1st Tuesday 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2013-01-02 00:00:00.000 02/01/2013 02/01/2013 2nd Wednesday 2 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1
3 2013-01-03 00:00:00.000 03/01/2013 03/01/2013 3rd Thursday 3 1 1 1 1 1 3 1 1 1 1 1 1 1 1 1 1 1
4 2013-01-04 00:00:00.000 04/01/2013 04/01/2013 4th Friday 4 1 1 1 1 1 4 1 1 1 1 1 1 1 1 1 1 1
5 2013-01-05 00:00:00.000 05/01/2013 05/01/2013 5th Saturday 5 1 1 1 1 1 5 1 1 1 1 1 1 1 1 1 1 1
6 2013-01-06 00:00:00.000 06/01/2013 06/01/2013 6th Sunday 6 1 1 1 1 1 6 1 1 1 1 1 1 1 1 1 1 1
7 2013-01-07 00:00:00.000 07/01/2013 07/01/2013 7th Monday 7 1 1 1 1 1 7 1 1 1 1 1 1 1 1 1 1 1
8 2013-01-08 00:00:00.000 08/01/2013 08/01/2013 8th Tuesday 8 1 2 2 2 2 8 2 2 2 2 2 1 2 1 1 1 1
9 2013-01-09 00:00:00.000 09/01/2013 09/01/2013 9th Wednesday 9 1 2 2 2 2 9 2 2 2 2 2 1 2 1 1 1 1
10 2013-01-10 00:00:00.000 10/01/2013 10/01/2013 10th Thursday 10 1 2 2 2 2 10 2 2 2 2 2 1 2 1 1 1 1
11 2013-01-11 00:00:00.000 11/01/2013 11/01/2013 11th Friday 11 1 2 2 2 2 11 2 2 2 2 2 1 2 1 1 1 1
12 2013-01-12 00:00:00.000 12/01/2013 12/01/2013 12th Saturday 12 1 2 2 2 2 12 2 2 2 2 2 1 2 1 1 1 1
13 2013-01-13 00:00:00.000 13/01/2013 13/01/2013 13th Sunday 13 1 2 2 2 2 13 3 2 2 3 2 3 1 1 1 1
< C
Query completed with errors.
DESKTOP-JVQDSB99 [11.0 RTM] DESKTOP-JVQDSB99\admin ... SalesDW [00:00:46] 736 rows
```

Now open, SQL Server Data Tools and create new project



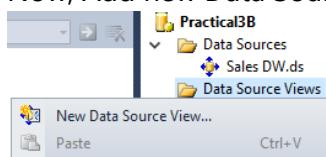
Right click on data sources and click add new Data Source

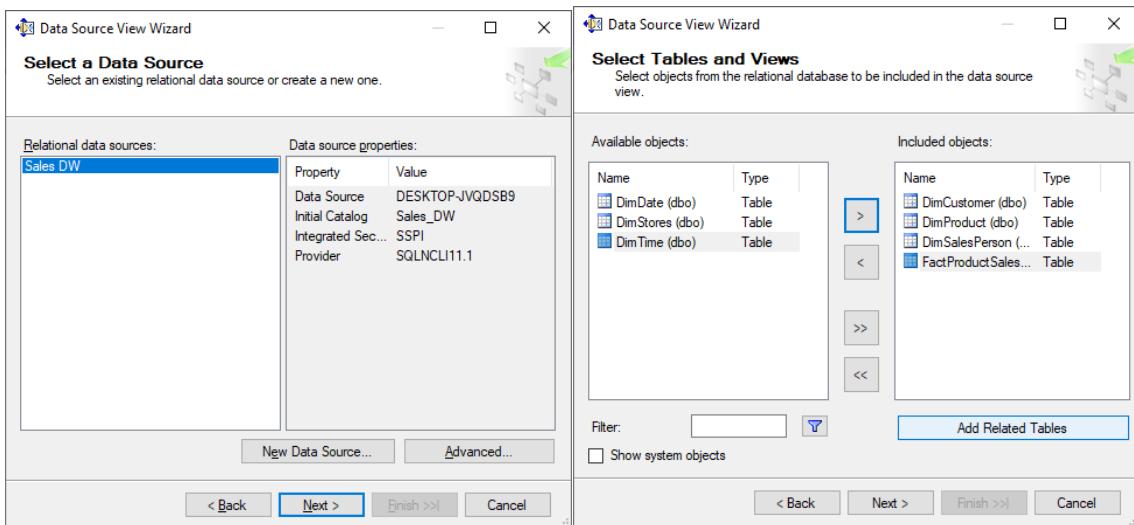


And click on New.. to add database

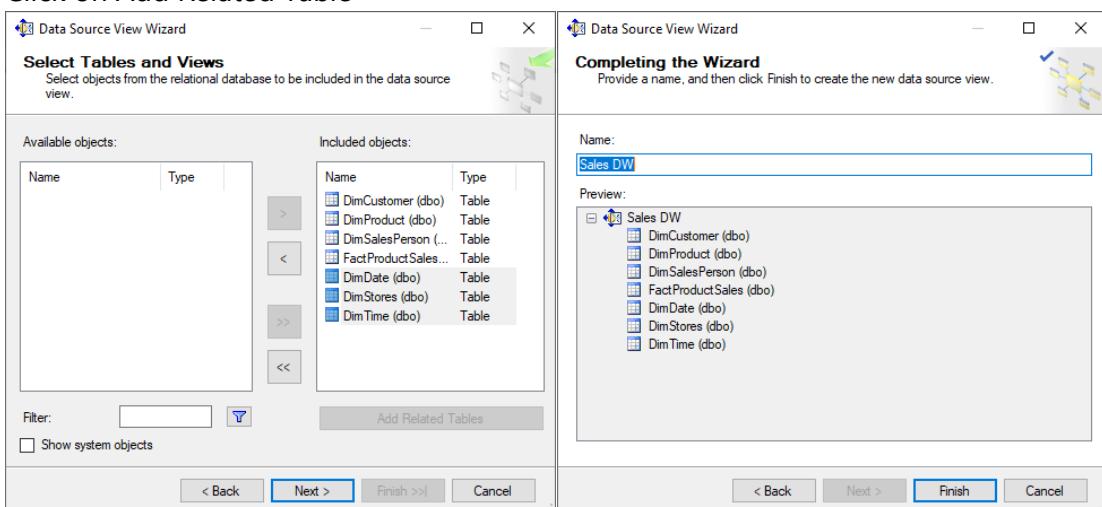
Select inherit

Now, Add new Data Source View

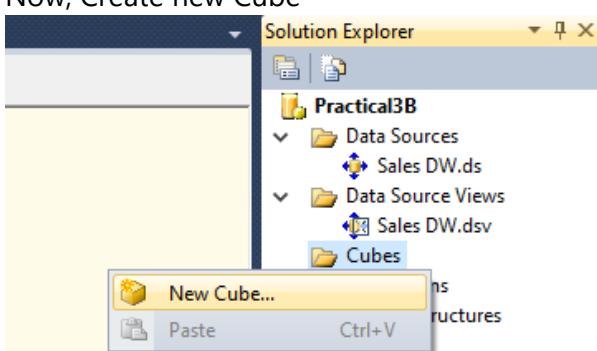




Click on Add Related Table



Now, Create new Cube



Cube Wizard

Select Creation Method
Cubes can be created by using existing tables, creating an empty cube, or generating tables in the data source.

How would you like to create the cube?

Use existing tables
 Create an empty cube
 Generate tables in the data source

Template:
(None)

Description:
Create a cube based on one or more tables in a data source.

< Back Next > Finish >> Cancel

Cube Wizard

Select Measure Group Tables
Select a data source view or diagram and then select the tables that will be used for measure groups.

Data source view:
Sales DW

Measure group tables:

- DimCustomer
- DimProduct
- DimSalesPerson
- FactProductSales
- DimDate
- DimStores
- DimTime

Suggest

< Back Next > Finish >> Cancel

Cube Wizard

Select Measures
Select measures that you want to include in the cube.

Measure

- Dim Customer
 - Dim Customer Count
- Dim Product
 - Product Actual Cost
 - Product Sales Cost
 - Dim Product Count
- Dim Sales Person
 - Store ID
 - Dim Sales Person Count

< Back Next > Finish >> Cancel

Cube Wizard

Select New Dimensions
Select new dimensions to be created, based on available tables.

Dimension

- Dim Customer
 - DimCustomer
- Dim Product
 - DimProduct
- Dim Sales Person
 - DimSalesPerson

< Back Next > Finish >> Cancel

Moves to the next wizard page

Cube Wizard

Completing the Wizard
Name the cube, review its structure, and then click Finish to save the cube.

Cube name:
Sales DW

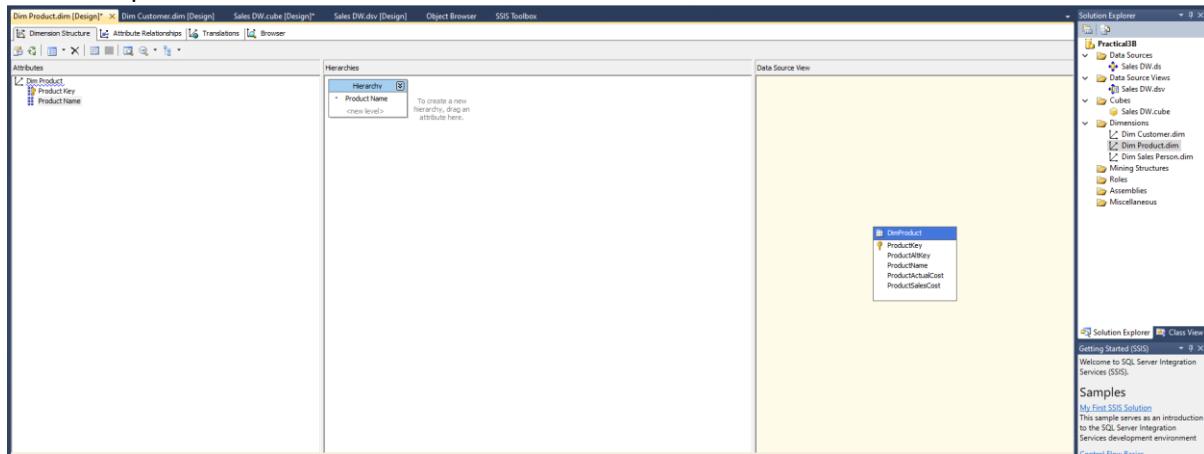
Preview:

- Measure groups
 - Dim Customer
 - Dim Customer Count
 - Dim Product
 - Product Actual Cost
 - Product Sales Cost
 - Dim Product Count
 - Dim Sales Person
 - Store ID
 - Dim Sales Person Count
- Dimensions
 - Dim Customer
 - Dim Product
 - Dim Sales Person

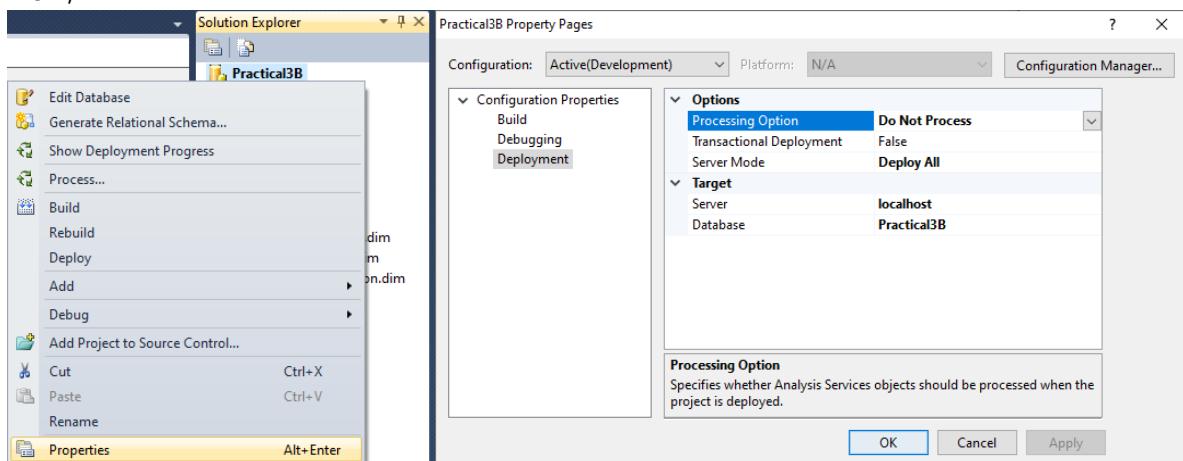
< Back Next > Finish Cancel

Now, open any dimensions

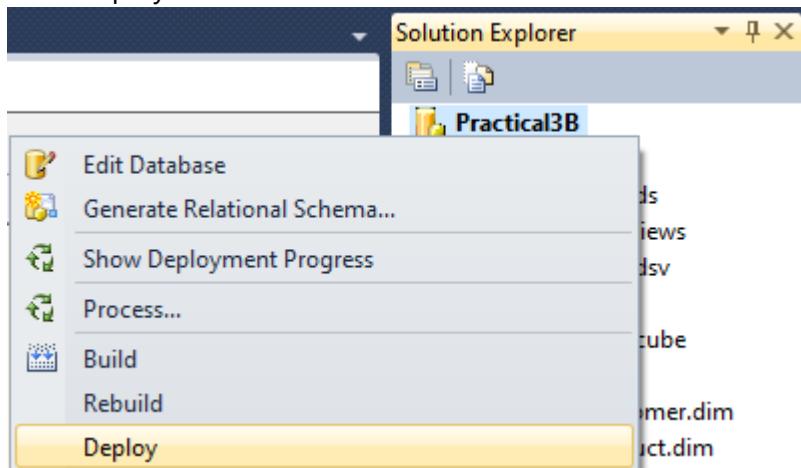
Drag any attribute from Data Source View to Attribute panel then, Drag any attribute from Attribute panel to Hierarchies Panel.



Now,



Now Deploy



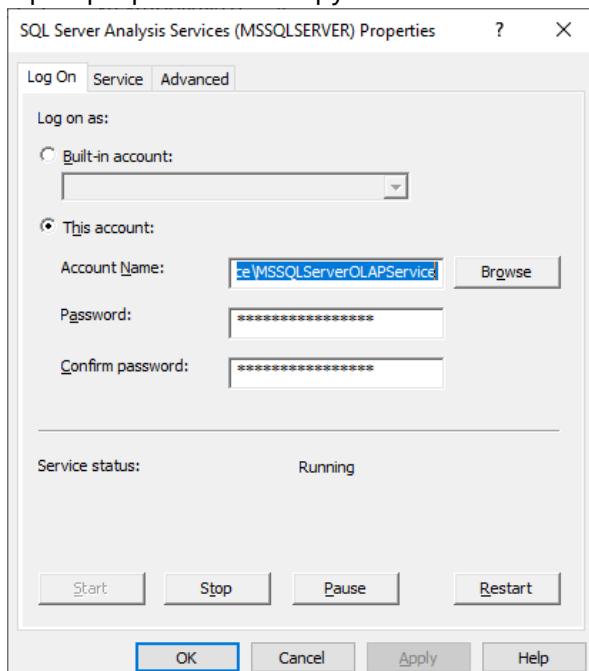
To solve this error,

- 8 Database [Practical3B] : The database has no Time dimension. Consider creating one.
 - 9 Either the 'DESKTOP-JVQDSB9\administrator' user does not have permission to create a new object in 'DESKTOP-JVQDSB9', or the object does not exist.
- [Error List](#) [Task List](#) [Code Definition Window](#) [Bookmarks](#)

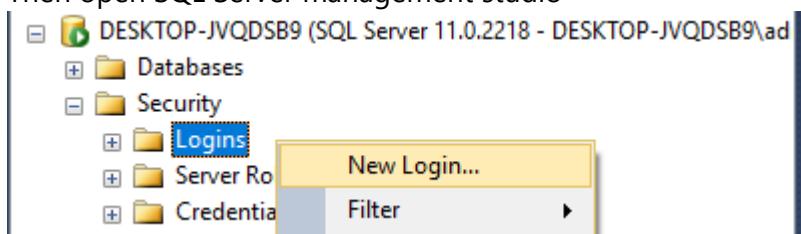
Open Sql server configuration manager:

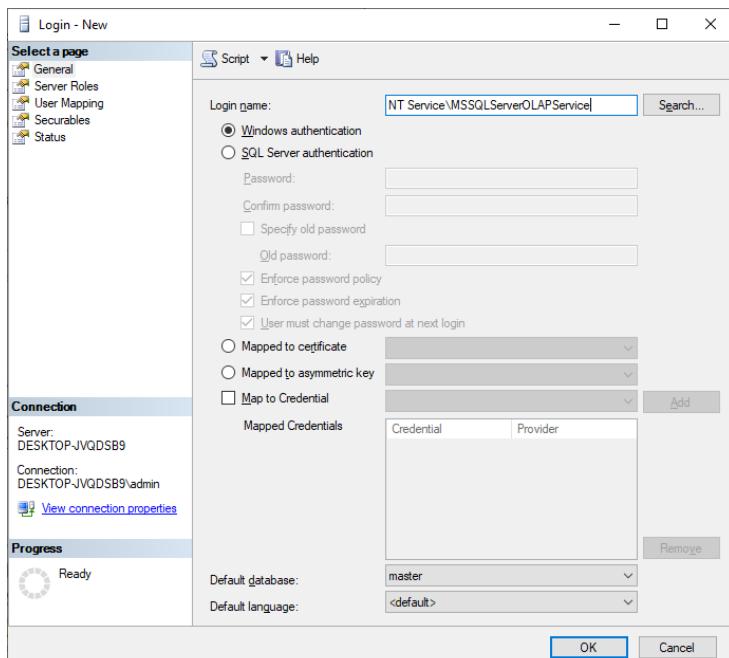
Name	State	Start Mode	Log On As	Process ID	Service Type
SQL Server Integration Services 11.0	Running	Automatic	NT Service\MsDtsS... 6276		
SQL Server (SQLEXPRESS)	Running	Automatic	NT Service\MSSQL\$... 9988		SQL Server
SQL Full-text Filter Daemon Launcher (MSSQLSERV...)	Running	Manual	NT Service\MSSQLF... 7220		
SQL Full-text Filter Daemon Launcher (SQLEXPRE...)	Running	Manual	NT Service\MSSQLF... 10936		
SQL Server Launchpad (SQLEXPRESS)	Running	Automatic	NT Service\MSSQLL... 9368		
SQL Server (MSSQLSERVER)	Running	Automatic	NT Service\MSSQL\$... 6284		SQL Server
SQL Server Analysis Services (MSSQLSERVER)	Running	Automatic	NT Service\MSSQL\$... 6316		Analysis Server
SQL Server Reporting Services (MSSQLSERVER)	Starting	Automatic	NT Service\ReportS... 6296		Report Server
SQL Server Agent (SQLEXPRESS)	Running	Manual	NT AUTHORITY\NE... 0		SQL Agent
SQL Server Browser	Running	Manual	NT AUTHORITY\LO... 0		
SQL Server Agent (MSSQLSERVER)	Running	Manual	NT Service\SQLSER... 0		SQL Agent

Open properties and copy account name

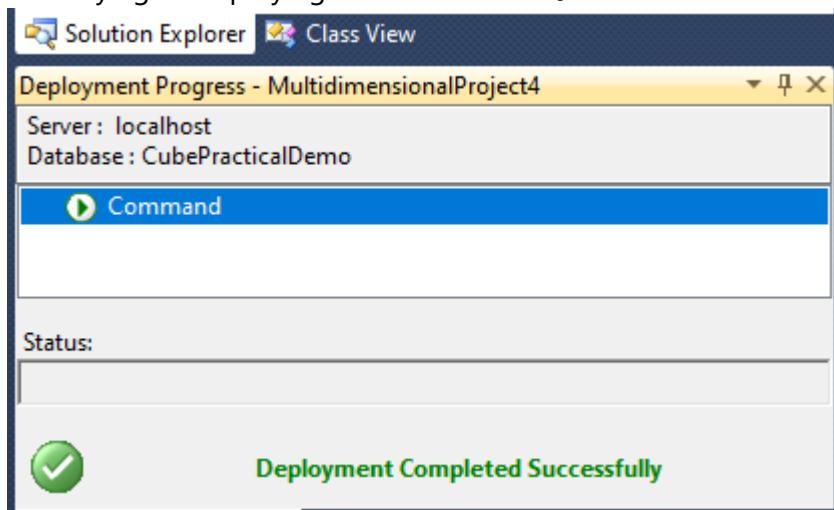


Then open SQL Server management studio

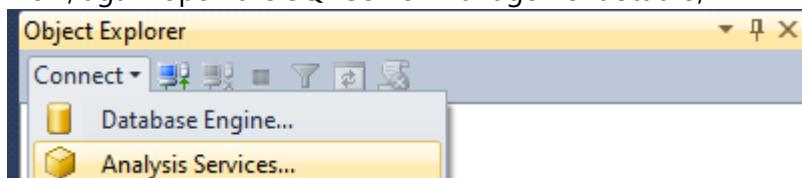


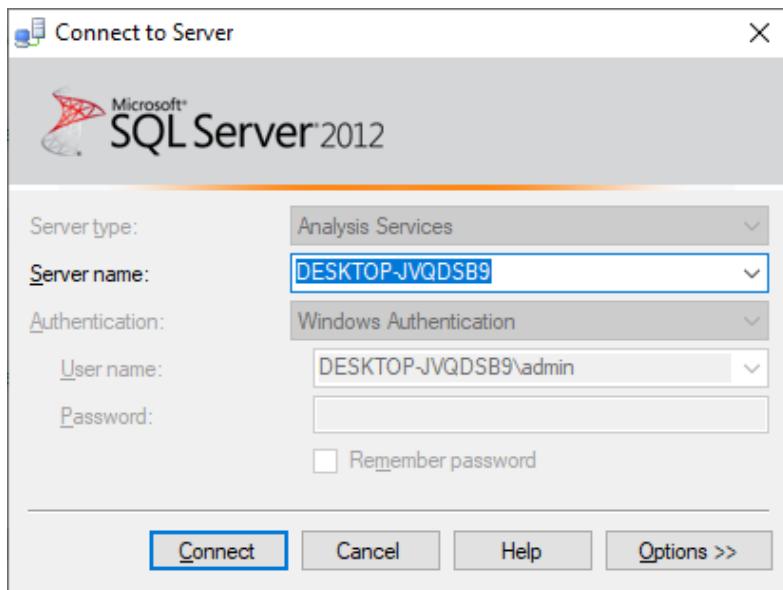


Now try again deploying the cube from SQL server data tools.



Now, again open the SQL Server Management Studio,





OUTPUT

Object Explorer

Connect ▾

DESKTOP-JVQDSB9 (Microsoft Analysis Server 11.0.2218.0 - DESKT...)

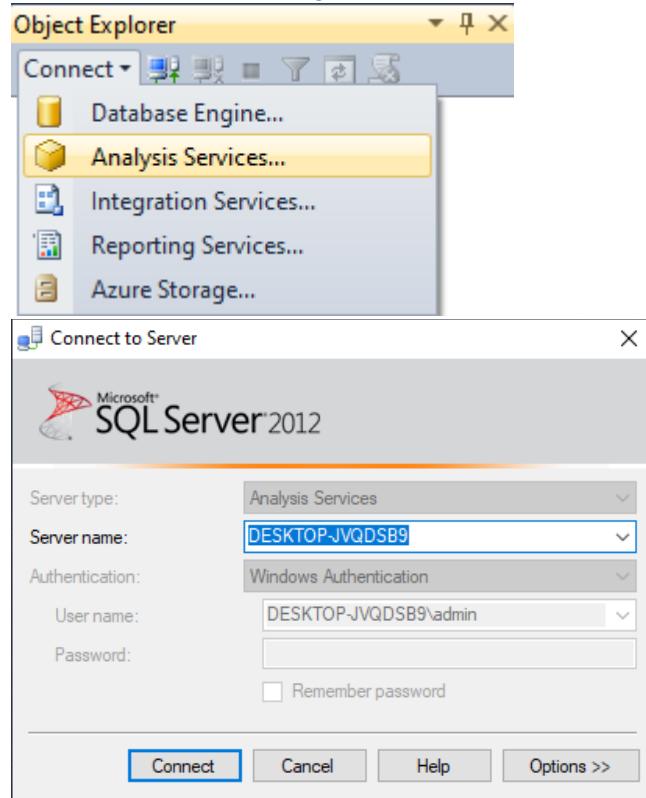
- Databases
 - cube
 - CubePracticalDemo
 - MultidimensionalProject4
 - Practical3B
 - Data Sources
 - Data Source Views
 - Cubes
 - Dimensions
 - Mining Structures
 - Roles
 - Assemblies

Practical 4

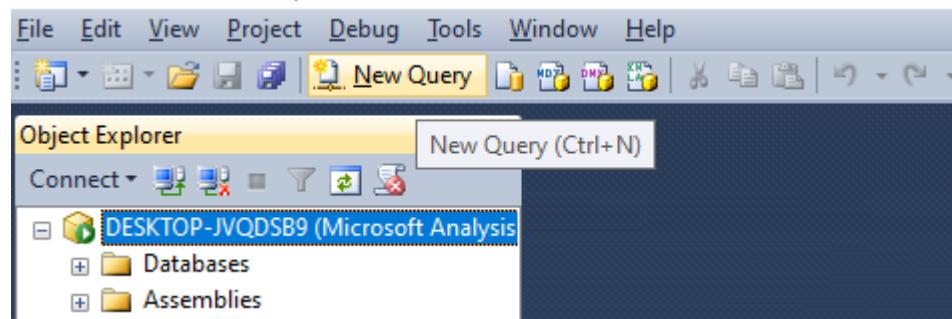
AIM: Execute the mdx queries to extract the data from data warehouse

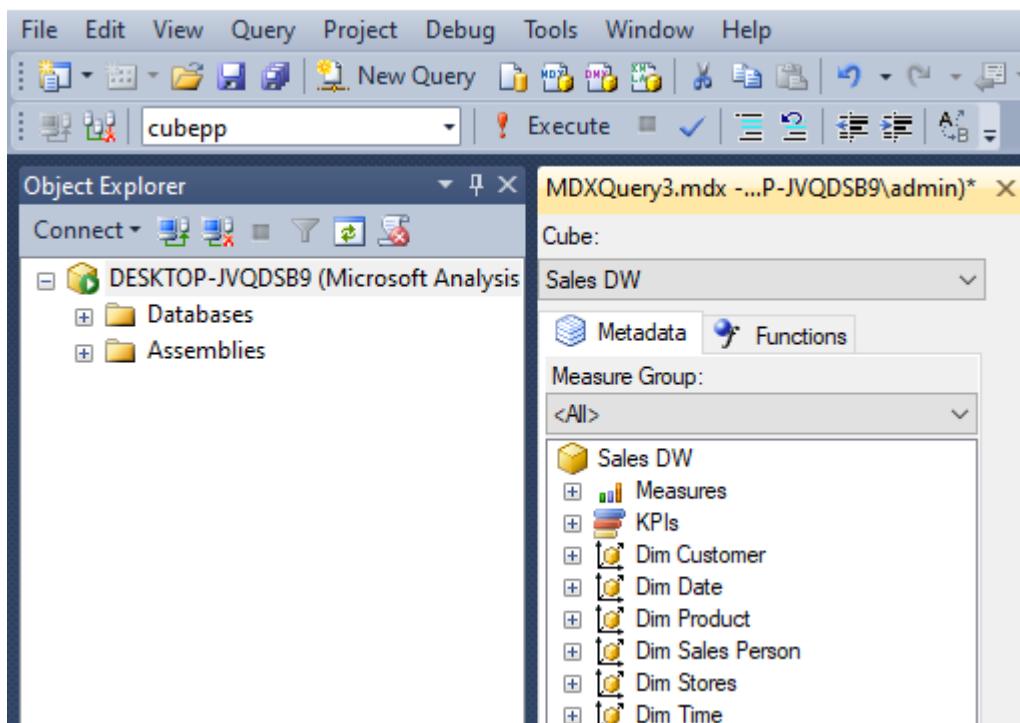
Solution:

Open SQL Server Management Studio and connect to analysis services



Now, create new query,



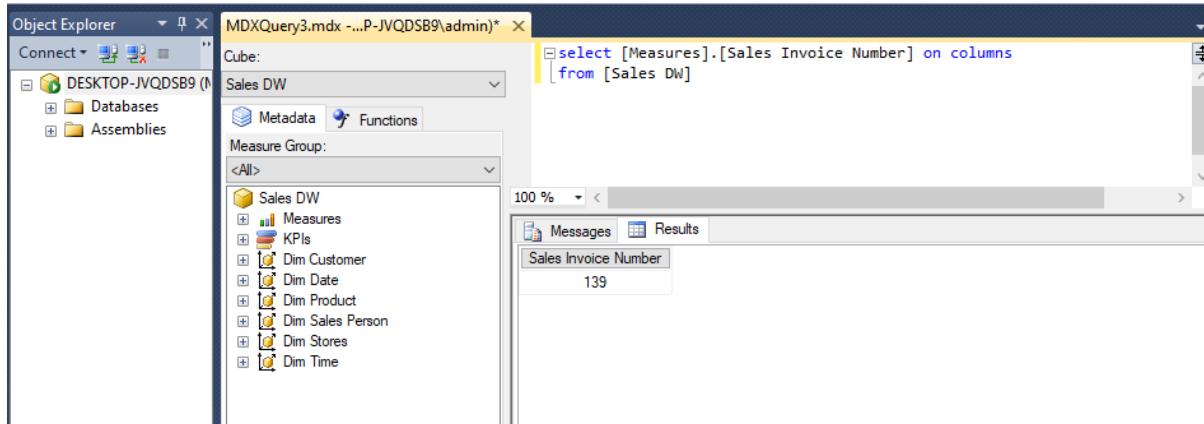


1st Query

select [Measures].[Sales Invoice Number] on columns

from [Sales DW]

Click on execute



2nd Query

select [Measures].[Sales Total Cost] on columns
from [Sales DW]

The screenshot shows the SSMS interface with the Object Explorer on the left and the MDXQuery3.mdx query window on the right. The query is:

```
select [Measures].[Sales Total Cost] on columns
from [Sales DW]
```

The results pane shows a single row:

Sales Total Cost
1231.5

3rd Query

select [Measures].[Sales Total Cost] on columns
, [Dim Date].[Year].[Year] on rows
from [Sales DW]

The screenshot shows the SSMS interface with the Object Explorer on the left and the MDXQuery3.mdx query window on the right. The query is:

```
select [Measures].[Sales Total Cost] on columns
, [Dim Date].[Year].[Year] on rows
from [Sales DW]
```

The results pane shows the following data:

	Sales Total Cost
2013	1231.5
2014	(null)
Unknown	(null)

4th Query

select [Measures].[Sales Total Cost] on columns
, NONEMPTY ([Dim Date].[Year].[Year]) on rows
from [Sales DW]

The screenshot shows the SSMS interface with the Object Explorer on the left and the MDXQuery3.mdx query window on the right. The query is:

```
select [Measures].[Sales Total Cost] on columns
, NONEMPTY ([Dim Date].[Year].[Year]) on rows
from [Sales DW]
```

The results pane shows the following data:

	Sales Total Cost
2013	1231.5

5th Query

select [Measures].[Sales Total Cost] on columns

from [Sales DW]
Where [Dim Date].[Year].[Year].&[2013]

The screenshot shows the Object Explorer on the left with a connection to 'DESKTOP-JVQDSB9' containing 'cube', 'cubapp', and 'CubePractical' under 'Database'. The MDXQuery3.mdx window on the right displays the following query and result:

```
select [Measures].[Sales Total Cost] on columns
from [Sales DW]
Where [Dim Date].[Year].[Year].&[2013]
```

The result pane shows a single row with the value '1231.5' under the heading 'Sales Total Cost'.

Practical 5:

AIM:

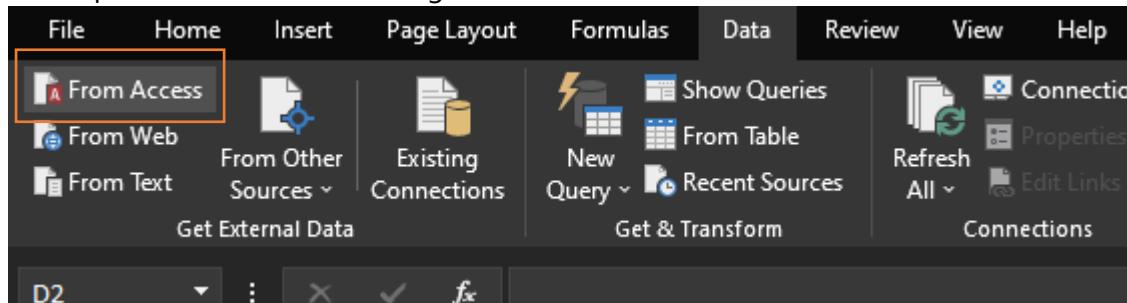
- a. Import the datawarehouse data in Microsoft Excel and create the Pivot table and Pivot Chart.
- b. Import the cube in Microsoft Excel and create the Pivot table and Pivot Chart to perform data analysis

Solution:

a)

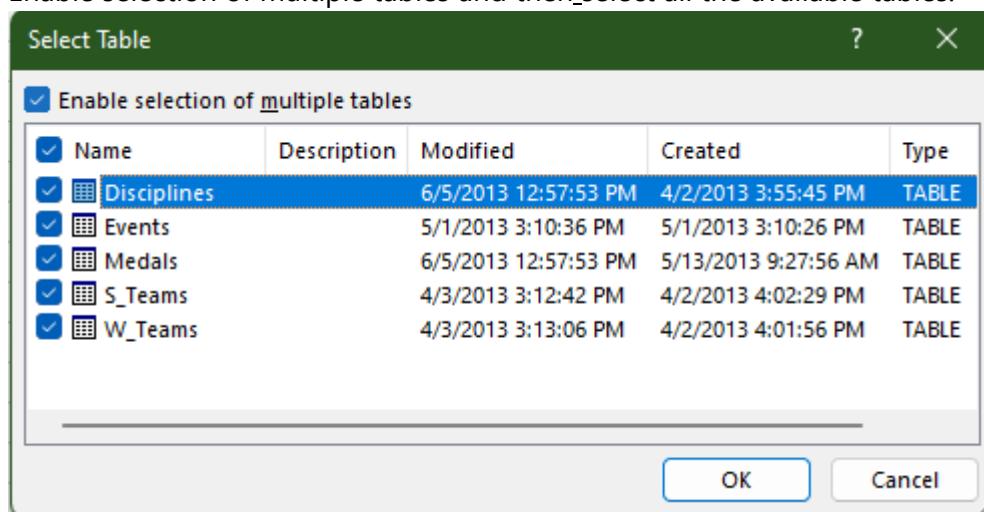
Download [OlympicMedal.accdl](#)

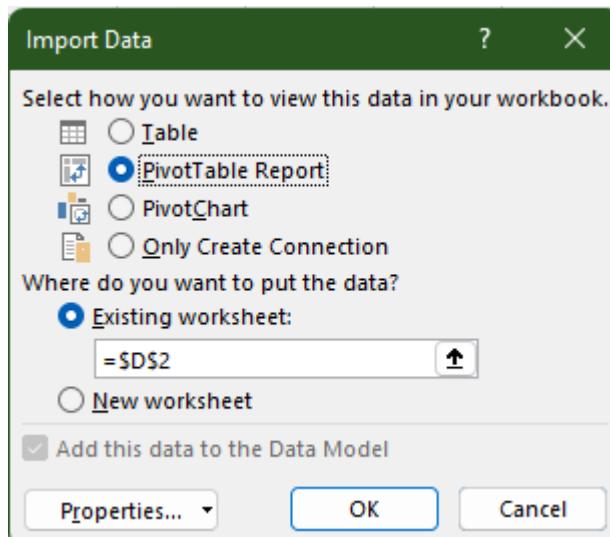
Now open Microsoft Excel, then go to Data tab and select "From Access"



Then, select the downloaded [OlympicMedal.accdl](#) file.

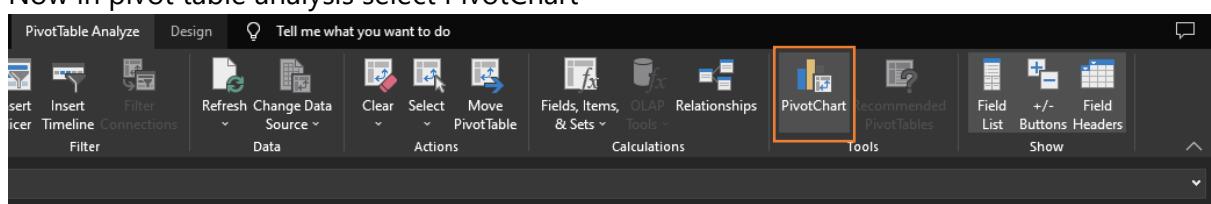
Enable selection of multiple tables and then select all the available tables.

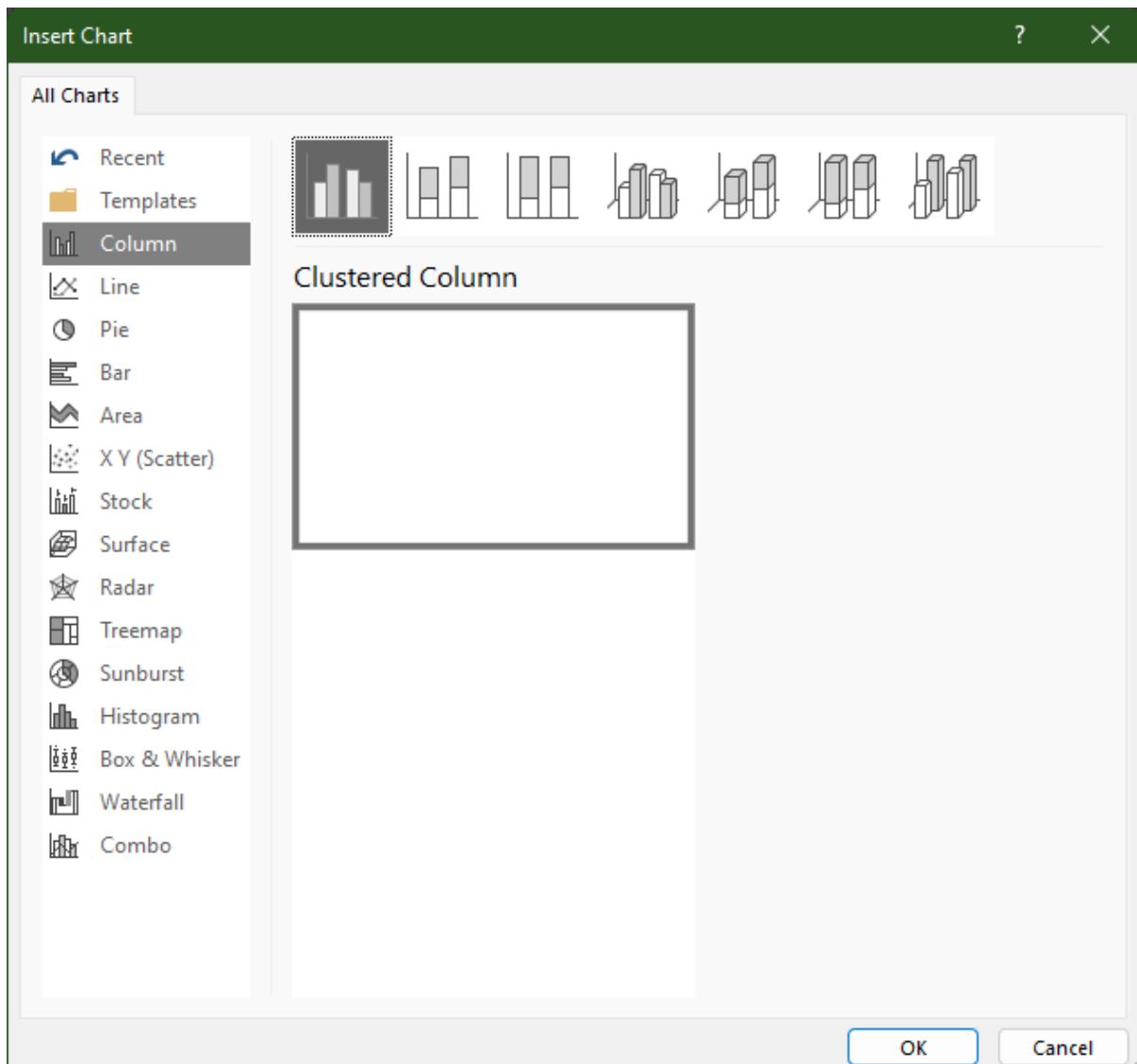




A screenshot of Microsoft Excel showing a PivotTable being created on 'Sheet1'. The PivotTable Fields pane on the right lists fields: EventID, Event, Sport, DisciplineID, Discipline, SportID, and DisciplineEvent. The 'Rows' area of the PivotTable Fields pane is currently empty. The main worksheet area shows the PivotTable structure starting at cell A1.

Now in pivot table analysis select PivotChart





Book1 - Excel (Product Activation Failed)

File Home Insert Page Layout Formulas Data Review View Help PivotChart Analyze Design Format Tell me what you want to do

Format Selection Reset to Match Style Current Selection Insert Shapes Shape Styles WordArt Styles Size

Shape Fill Shape Outline Shape Effects

A A A Text Fill Text Outline Text Effects

Bring Forward Send Backward Group Rotate

Height: 7.62 cm Width: 12.7 cm

OK Cancel

Chart1

To build a report, choose fields from the PivotTable Field List.

To build a PivotChart, choose fields from the PivotChart Field List.

PivotTable1

EventID Event Sport DisciplinedD Discipline SportID DisciplinedEvent

Filters Legend (Series)

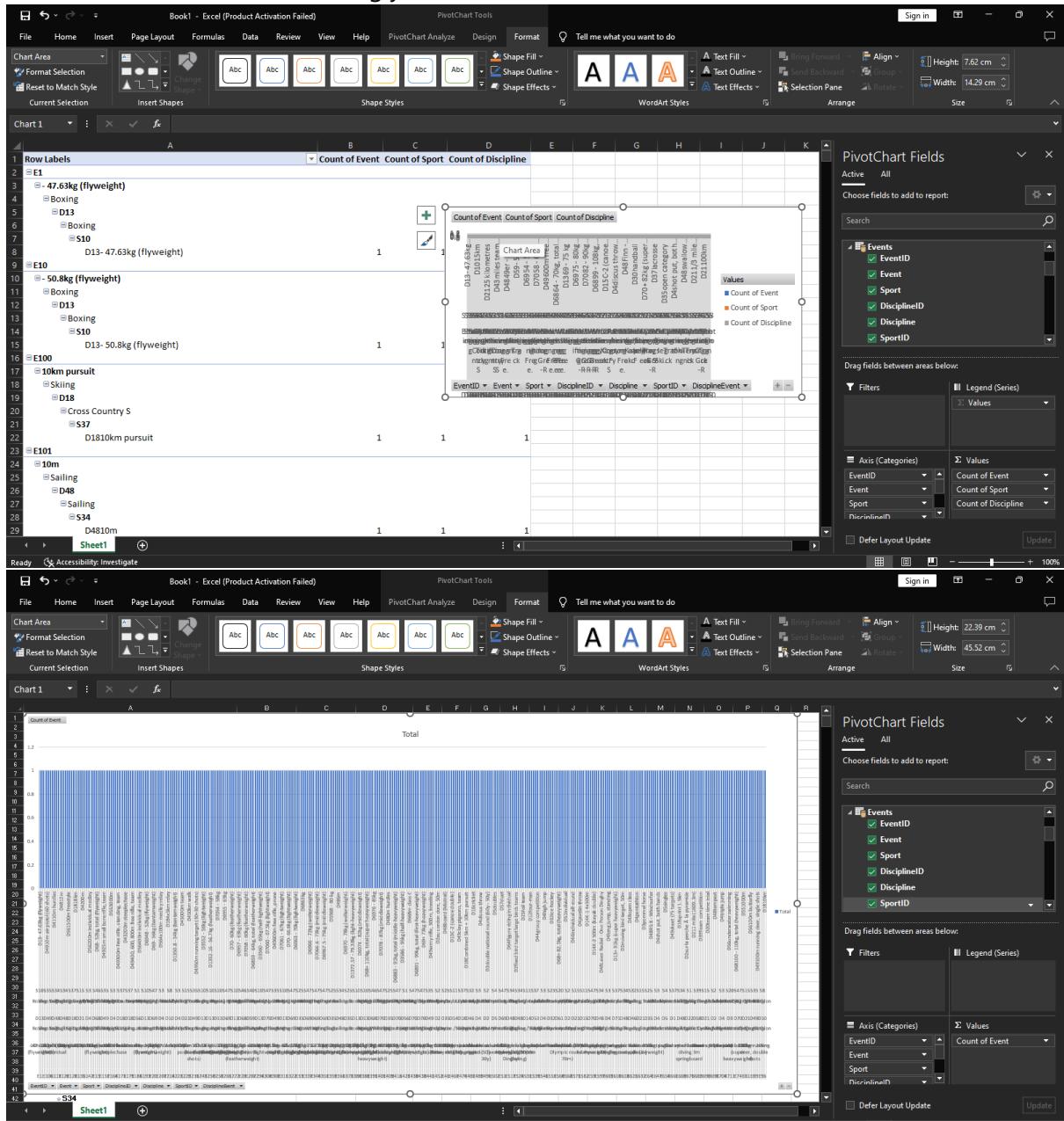
Axis (Categories) Values

Defer Layout Update Update

Sheet1

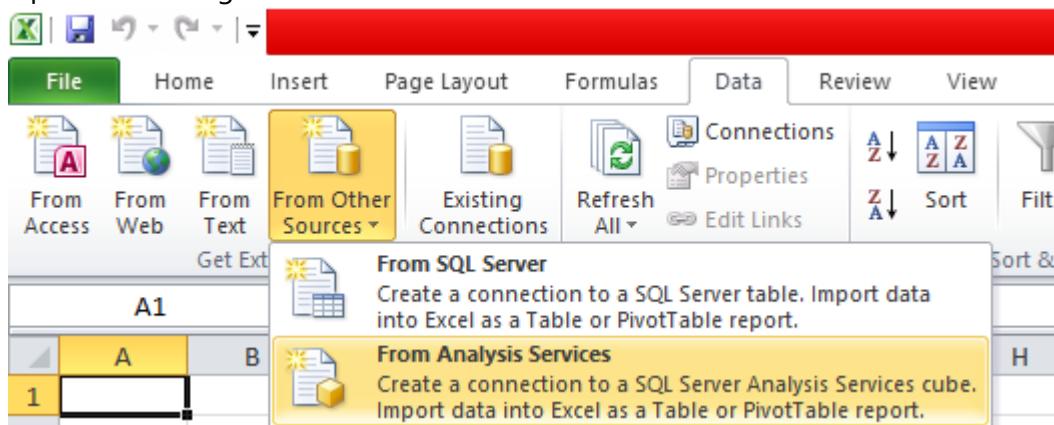
Ready Accessibility: Investigate

Now, choose the fields accordingly.

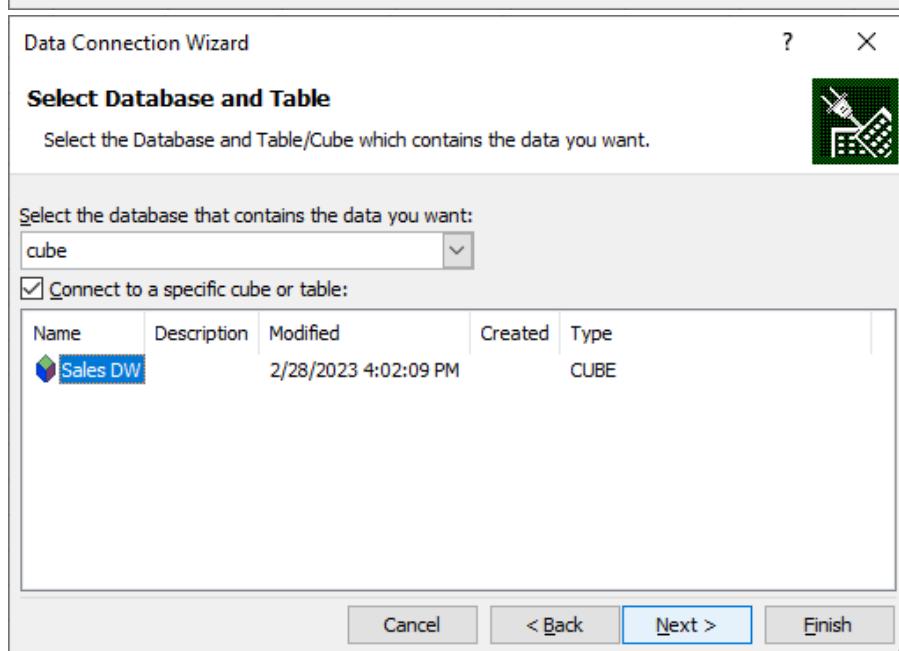
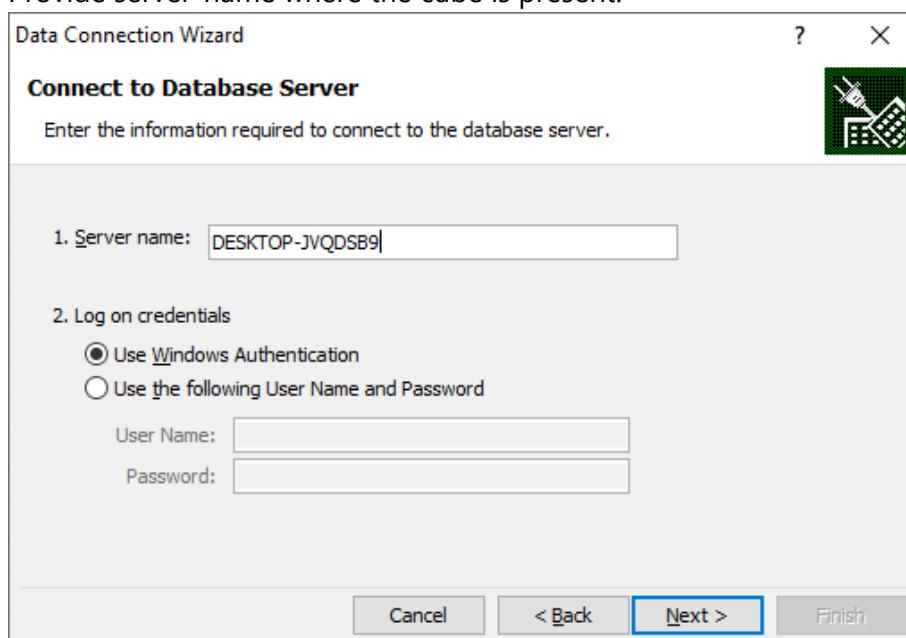


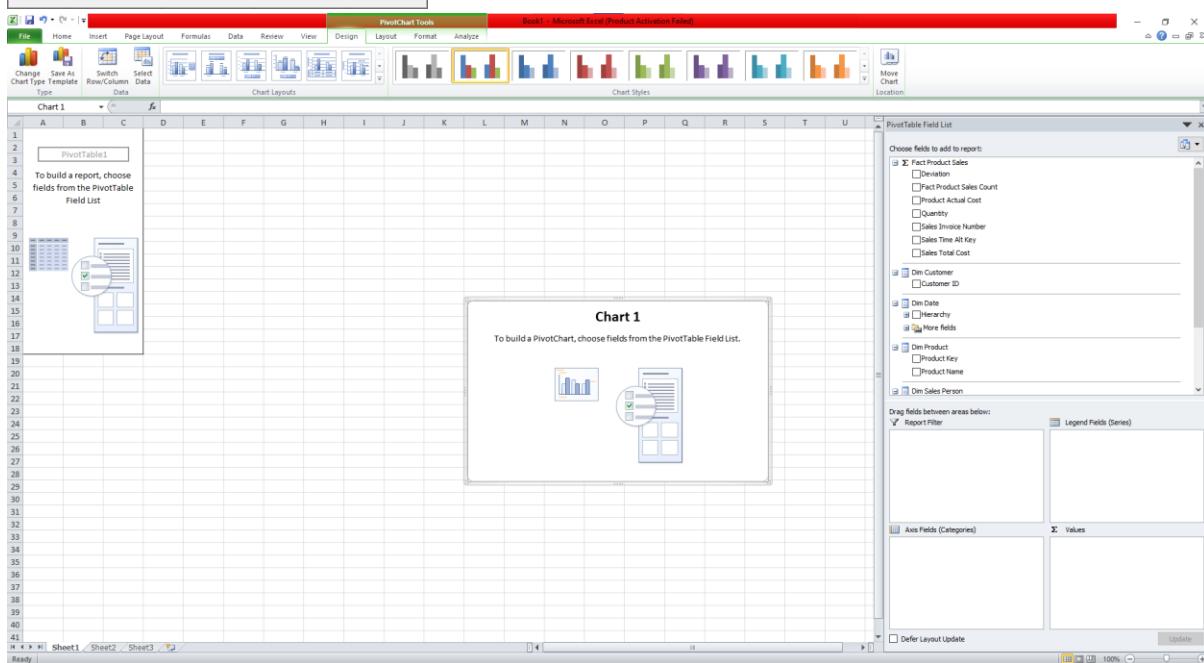
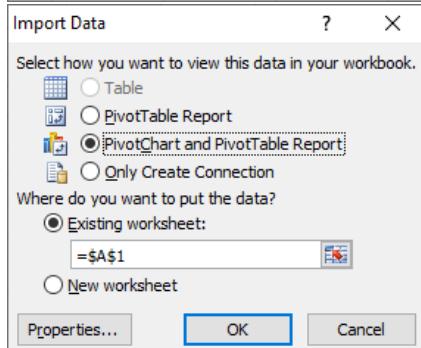
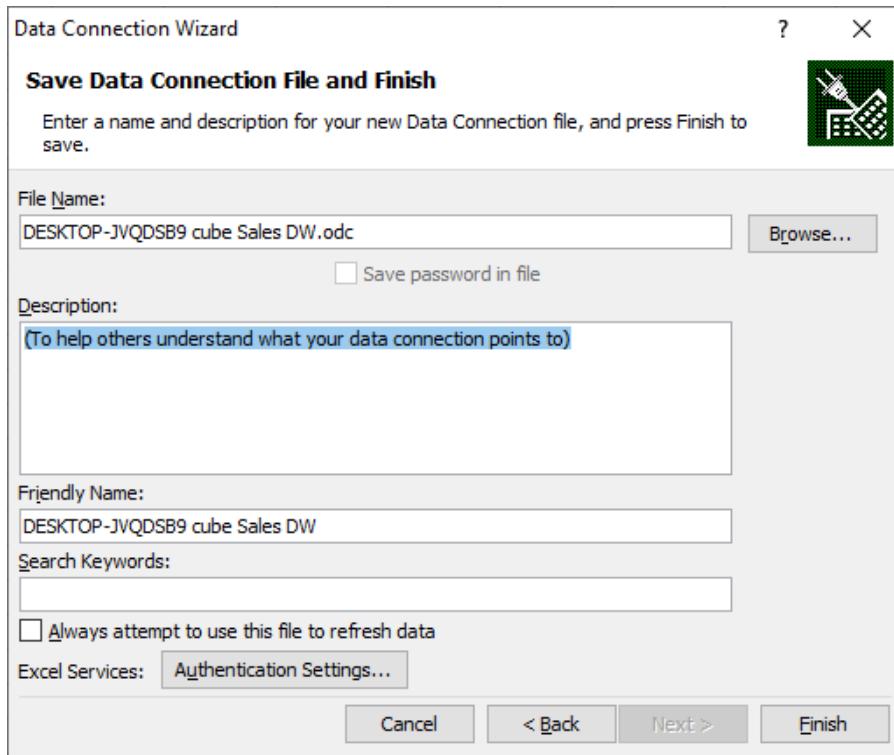
b)

Open excel and go to data:



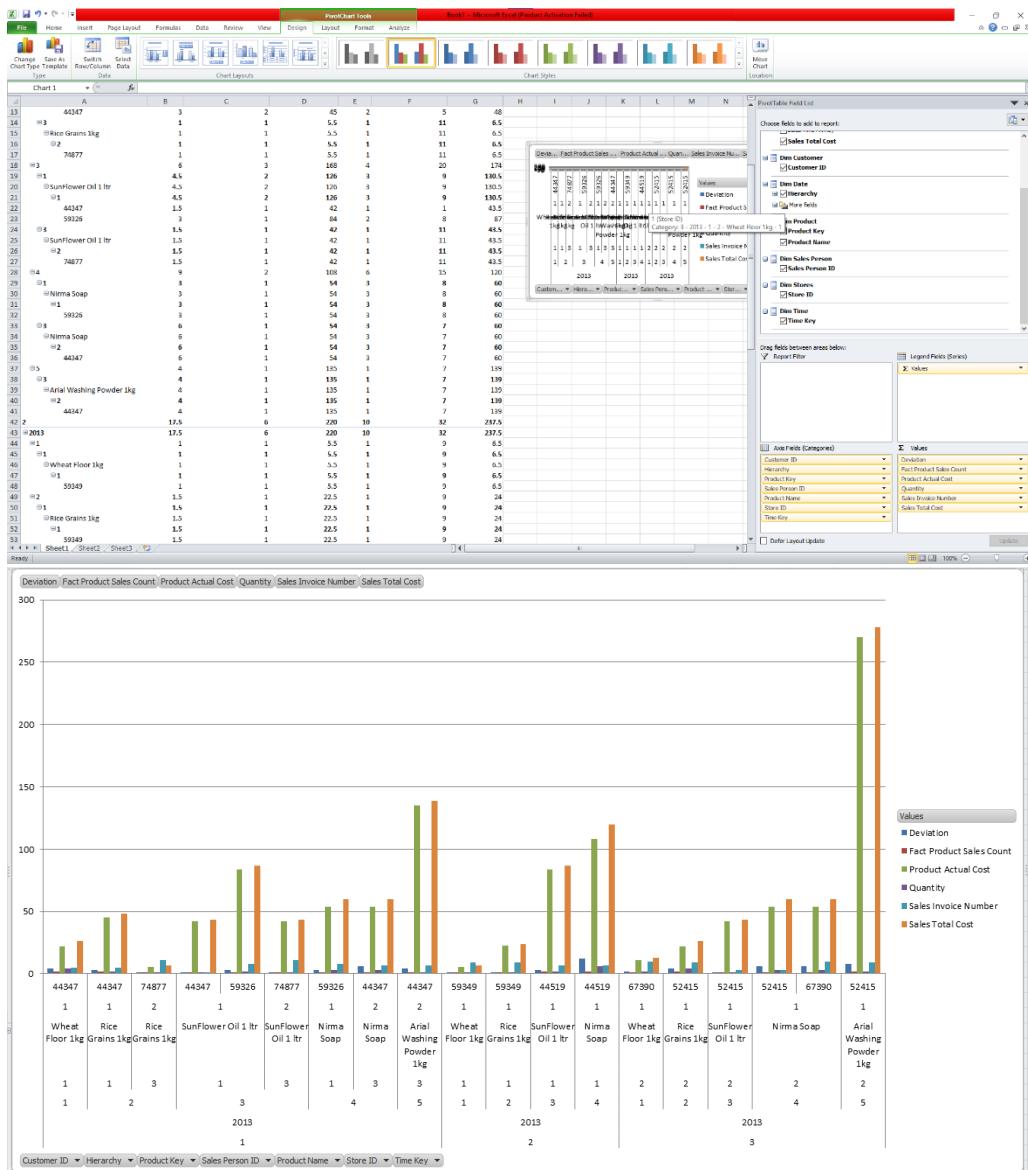
Provide server-name where the cube is present:





OUTPUT

Sweta Prajapati T.20.112



Practical 7

AIM: Apply the what – if Analysis for data visualization. Design and generate necessary reports based on the data warehouse data.

Solution:

Cell calculations

		C7	f _x	=B4*C4		
1	A	B	C	D	E	
1	Grain Store					
2						
3	total weight of wheat	% sold for the highest price				
4	150	68%				
5						
6		weight of wheat	unit profit			
7	highest price	102	50			
8	lower price	48	35			
9						
10		total profit	6780			
		C8	f _x	=B4-C7		
1	A	B	C	D	E	F
1	Grain Store					
2						
3	total weight of wheat	% sold for the highest price				
4	150	68%				
5						
6		weight of wheat	unit profit			
7	highest price	102	50			
8	lower price	48	35			
9						
10		total profit	6780			
		D10	f _x	=C7*D7+C8*D8		
1	A	B	C	D	E	
1	Grain Store					
2						
3	total weight of wheat	% sold for the highest price				
4	150	68%				
5						
6		weight of wheat	unit profit			
7	highest price	102	50			
8	lower price	48	35			
9						
10		total profit	6780			

	A	B	C	D	E
1	Grain Store				
2					
3		total weight of wheat	% sold for the highest price		
4		150	68%		
5					
6			weight of wheat	unit profit	
7		highest price		102	50
8		lower price		48	35
9					
10			total profit		6780
11					
12					

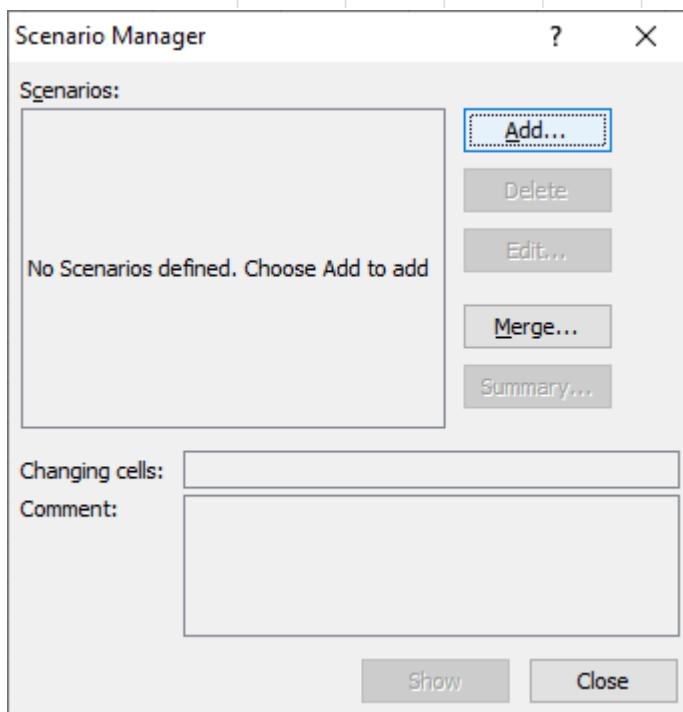
What if analysis (Scenario manager configuration)

Formulas Data Review View

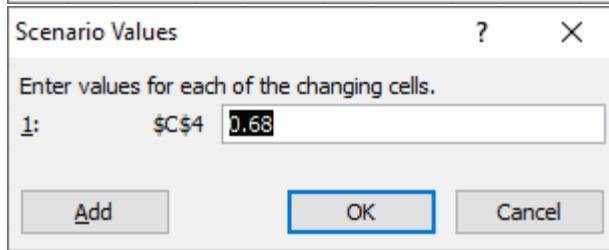
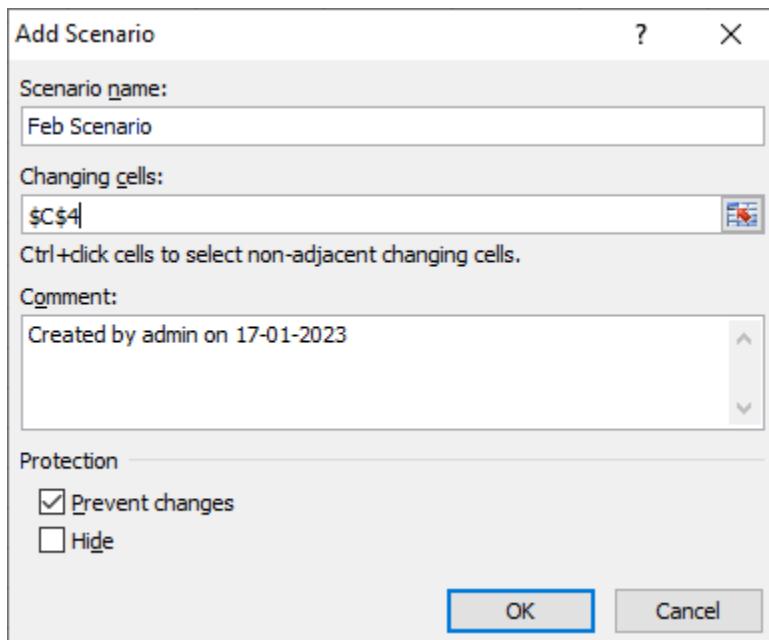
Connections Properties Refresh All Connections Edit Links Z A Sort Filter Advanced Text to Columns Remove Duplicates Data Validation Consolidate What-If Analysis Group Ungroup Subl Scenario Manager... Goal Seek... Data Table... M

7*D7+C8*D8

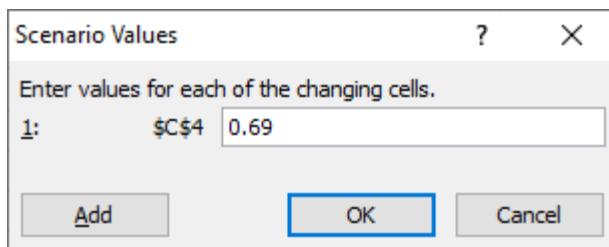
C	D	E	F	G	H	I	J
---	---	---	---	---	---	---	---



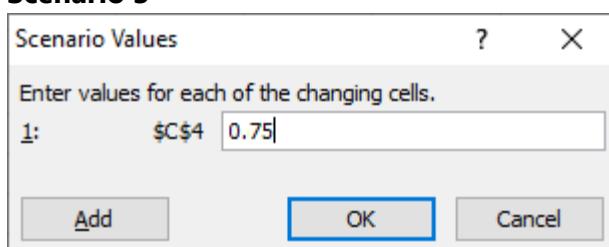
Scenario 1



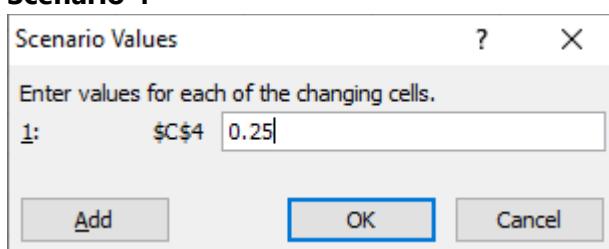
Scenario 2

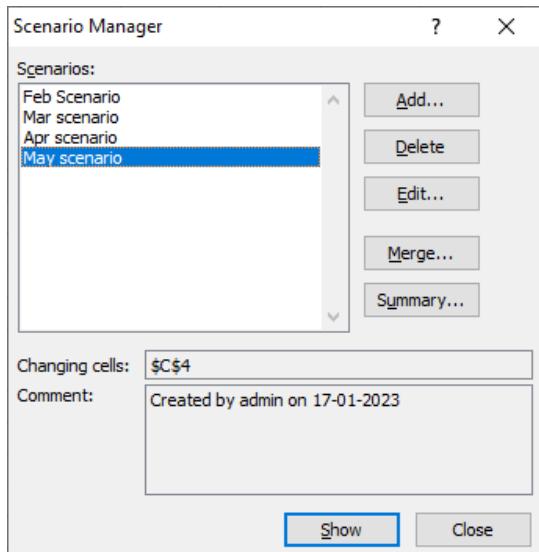


Scenario 3



Scenario 4





Click show

	A	B	C	D	E	F	G	H	I	J
1	Grain Store									
2										
3	total weight of wheat		% sold for the highest price							
4	150		75%							
5										
6	weight of wheat			unit profit						
7	highest price			112.5	50					
8	lower price			37.5	35					
9										
10	total profit			6937.5						
11										
12										
13										
14										
15										
16										
17										
18										

	A	B	C	D	E	F	G	H	I	J
1	Grain Store									
2										
3	total weight of wheat		% sold for the highest price							
4	150		25%							
5										
6	weight of wheat			unit profit						
7	highest price			37.5	50					
8	lower price			112.5	35					
9										
10	total profit			5812.5						
11										
12										
13										
14										
15										
16										
17										
18										

Scenario Manager

Scenarios:

- Feb Scenario
- Mar scenario
- Apr scenario
- May scenario

Buttons: Add..., Delete, Edit..., Merge..., Summary...

Changing cells: \$C\$4

Comment: Created by admin on 17-01-2023

Show Close

Scenario Manager

Scenarios:

- Feb Scenario
- Mar scenario
- Apr scenario
- May scenario

Buttons: Add..., Delete, Edit..., Merge..., Summary...

Changing cells: \$C\$4

Comment: Created by admin on 17-01-2023

Show Close

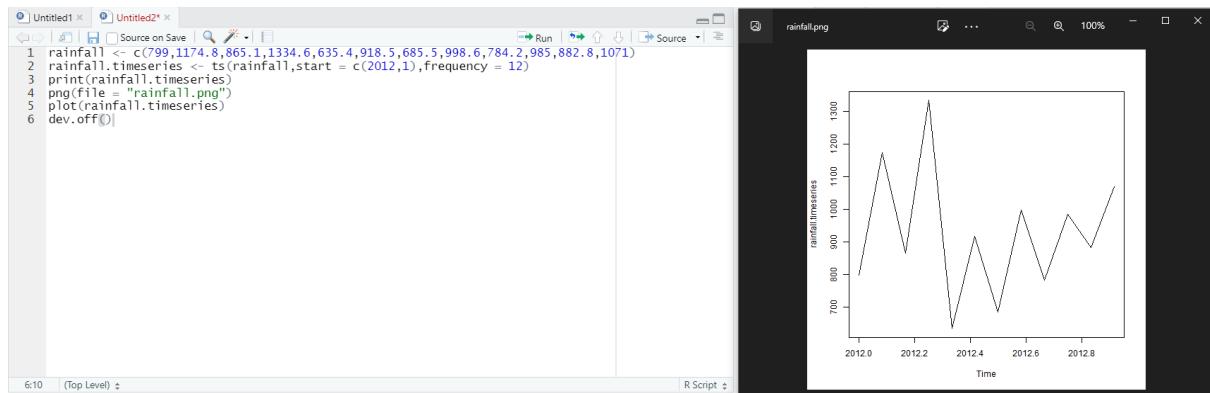
Practical 7

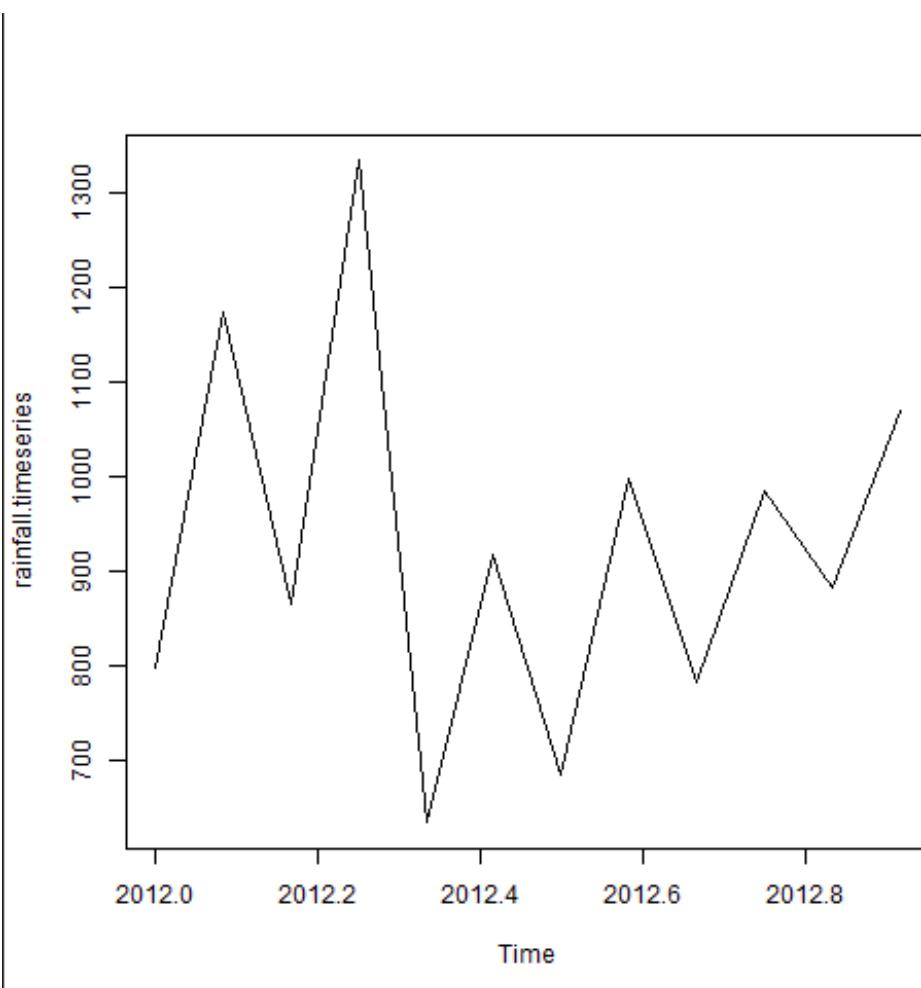
AIM: Perform the data classification using classification algorithm.

Solution:

R studio code:

```
# Get the data points in form of a R vector.  
rainfall <- c(799,1174.8,865.1,1334.6,635.4,918.5,685.5,998.6,784.2,985,882.8,1071)  
# Convert it to a time series object.  
rainfall.timeseries <- ts(rainfall,start = c(2012,1),frequency = 12)  
# Print the timeseries data.  
print(rainfall.timeseries)  
# Give the chart file a name.  
png(file = "rainfall.png")  
# Plot a graph of the time series.  
plot(rainfall.timeseries)  
# Save the file.  
dev.off()
```





Practical 8

AIM: Perform the data clustering using clustering algorithm.

Solution:

Step 1: In RStudio run below command

```
print(iris)
> print(iris)
   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1       3.5        1.4       0.2    setosa
2          4.9       3.0        1.4       0.2    setosa
3          4.7       3.2        1.3       0.2    setosa
4          4.6       3.1        1.5       0.2    setosa
5          5.0       3.6        1.4       0.2    setosa
6          5.4       3.9        1.7       0.4    setosa
7          4.6       3.4        1.4       0.3    setosa
8          5.0       3.4        1.5       0.2    setosa
9          4.4       2.9        1.4       0.2    setosa
10         4.9       3.1        1.5       0.1    setosa
11         5.4       3.7        1.5       0.2    setosa
12         4.8       3.4        1.6       0.2    setosa
13         4.8       3.0        1.4       0.1    setosa
14         4.3       3.0        1.1       0.1    setosa
15         5.8       4.0        1.2       0.2    setosa
16         5.7       4.4        1.5       0.4    setosa
17         5.4       3.9        1.3       0.4    setosa
18         5.1       3.5        1.4       0.3    setosa
19         5.7       3.8        1.7       0.3    setosa
20         5.1       3.8        1.5       0.3    setosa
21         5.4       3.4        1.7       0.2    setosa
22         5.1       3.7        1.5       0.4    setosa
23         4.6       3.6        1.0       0.2    setosa
24         5.1       3.3        1.7       0.5    setosa
25         4.8       3.4        1.9       0.2    setosa
```

Step 2: Creating copy of iris dataset and removing the Species column

```
newiris <- iris
```

```
newiris$Species <- NULL
```

```
print(newiris)
```

```
> newiris <- iris
> newiris$Species <- NULL
> print(newiris)
   Sepal.Length Sepal.Width Petal.Length Petal.Width
1          5.1       3.5        1.4       0.2
2          4.9       3.0        1.4       0.2
3          4.7       3.2        1.3       0.2
4          4.6       3.1        1.5       0.2
5          5.0       3.6        1.4       0.2
6          5.4       3.9        1.7       0.4
7          4.6       3.4        1.4       0.3
8          5.0       3.4        1.5       0.2
9          4.4       2.9        1.4       0.2
10         4.9       3.1        1.5       0.1
11         5.4       3.7        1.5       0.2
12         4.8       3.4        1.6       0.2
13         4.8       3.0        1.4       0.1
14         4.3       3.0        1.1       0.1
15         5.8       4.0        1.2       0.2
16         5.7       4.4        1.5       0.4
17         5.4       3.9        1.3       0.4
18         5.1       3.5        1.4       0.3
19         5.7       3.8        1.7       0.3
20         5.1       3.8        1.5       0.3
21         5.4       3.4        1.7       0.2
22         5.1       3.7        1.5       0.4
23         4.6       3.6        1.0       0.2
24         5.1       3.3        1.7       0.5
25         4.8       3.4        1.9       0.2
```

Step 3: Applying K-means to iris and storing result

```
print(iris)
newiris <- iris
newiris$Species <- NULL
print(newiris)
(kc <- kmeans(newiris,4))
```

Step 4: Compare the species label with clustering result

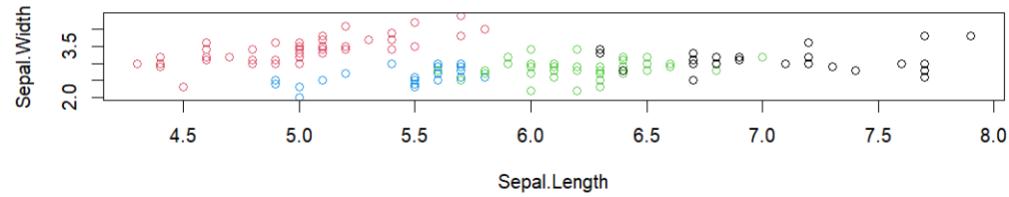
```
table(iris$Species, kc$cluster)
```

```
> table(iris$Species, kc$cluster)
```

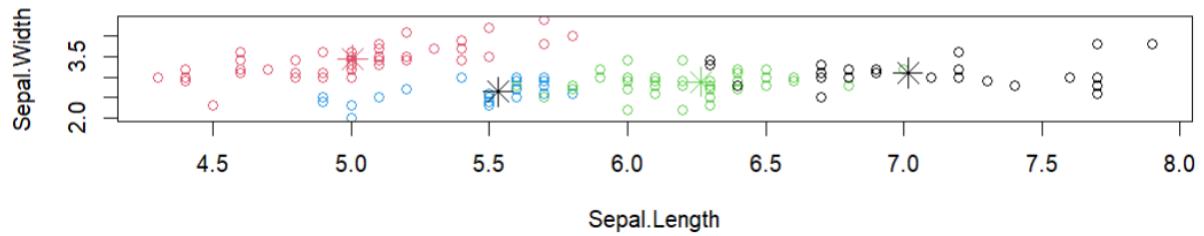
	1	2	3	4
setosa	0	50	0	0
versicolor	0	0	23	27
virginica	27	0	22	1

Step 5: Plot the cluster and their centers

```
plot(newiris[c("Sepal.Length","Sepal.Width")],col=kc$cluster)
```



```
plot(newiris[c("Sepal.Length", "Sepal.Width")], col=kc$cluster)
points(kc$centers[, c("Sepal.Length", "Sepal.Width")], col=1:3, pch=8, cex=2)
```



Complete code

```

print(iris)
newiris <- iris
newiris$Species <- NULL
print(newiris)
(kc <- kmeans(newiris,4))
table(iris$Species, kc$cluster)
plot(newiris[c("Sepal.Length","Sepal.Width")],col=kc$cluster)
points(kc$centers[,c("Sepal.Length", "Sepal.Width")],col=1:3,pch=8,cex=2)

```

Practical 9

AIM: Perform the Linear regression on the given data warehouse data.

Solution:

Step 1: Setting up data

```
x <- c(151,174,138,186,128,136,179,163,152,131) # height  
y <- c(63,81,56,91,47,57,76,72,62,48) # weight
```

Step 2: printing relation

```
relation <- lm(y~x)
```

```
print(relation)
```

```
1 x <- c(151,174,138,186,128,136,179,163,152,131) # height  
2 y <- c(63,81,56,91,47,57,76,72,62,48) # weight  
3  
4 relation <- lm(y~x)  
5 print(relation)
```

4:1 (Top Level)

Console Terminal Background Jobs

R 4.1.3 · ~/ ◊
> relation <- lm(y~x)
> print(relation)

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept) x
-38.4551 0.6746

Step 3: printing summary of the relation

```
print(summary(relation))
```

```
> print(summary(relation))  
  
Call:  
lm(formula = y ~ x)  
  
Residuals:  
Min 1Q Median 3Q Max  
-6.3002 -1.6629 0.0412 1.8944 3.9775  
  
Coefficients:  
Estimate Std. Error t value Pr(>|t|)  
(Intercept) -38.45509 8.04901 -4.778 0.00139 **  
x 0.67461 0.05191 12.997 1.16e-06 ***  
---  
Signif. codes: 0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
  
Residual standard error: 3.253 on 8 degrees of freedom  
Multiple R-squared: 0.9548, Adjusted R-squared: 0.9491  
F-statistic: 168.9 on 1 and 8 DF, p-value: 1.164e-06
```

Step 4: Predicting the weight of new person

```
a <- data.frame(x = 170)  
result <- predict(relation,a)  
print(result)
```

```

7 a <- data.frame(x = 170)
8 result <- predict(relation,a)
9 print(result)

7:1 (Top Level) ⇡

Console Terminal × Background Jobs ×
R 4.1.3 · ~/Documents/RStudioProjects/Untitled Project

> a <- data.frame(x = 170)
> result <- predict(relation,a)
> print(result)
  1
76.22869
> |

```

Step 5: Visualize the Regression Graphically

```

png(file = "linearregression.png")
# Plot the chart.
plot(y,x,col = "blue",main = "Height & Weight Regression",
abline(lm(x~y)),cex = 1.3,pch = 16,xlab = "Weight in Kg",ylab = "Height in
cm")
# Save the file.
dev.off()

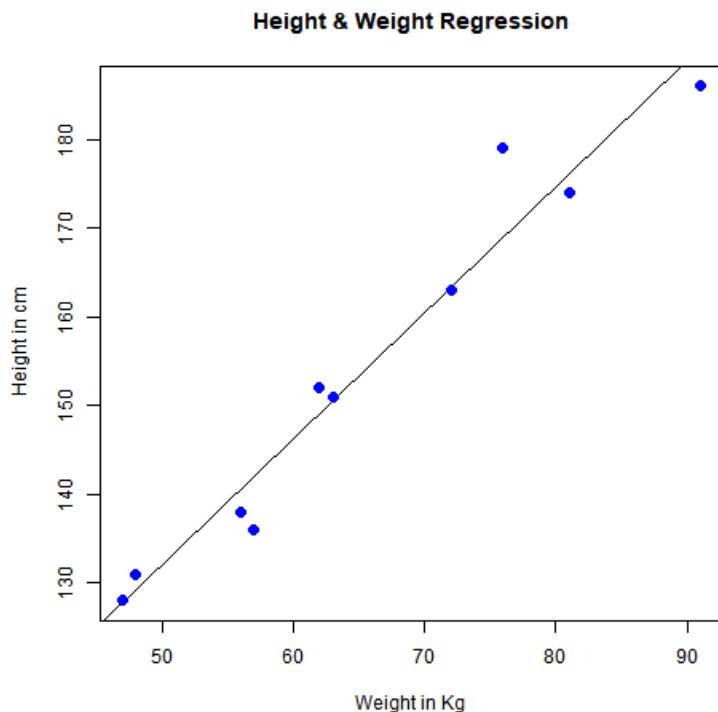
```

```

Console Terminal × Background Jobs ×
R 4.1.3 · ~/Documents/RStudioProjects/Untitled Project

> png(file = "linearregression.png")
> # Plot the chart.
> plot(y,x,col = "blue",main = "Height & Weight Regression", abline(lm(x~y)),cex = 1.3,pch = 16,xlab = "Weight in Kg",ylab = "Height in cm")
> # Save the file.
> dev.off()
RStudioGD
2
> |

```



Complete Code

```
x <- c(151,174,138,186,128,136,179,163,152,131) # height
y <- c(63,81,56,91,47,57,76,72,62,48) # weight
relation <- lm(y~x)
print(summary(relation))
a <- data.frame(x = 170)
result <- predict(relation,a)
print(result)
png(file = "linearregression.png")
# Plot the chart.
plot(y,x,col = "blue",main = "Height & Weight Regression", abline(lm(x~y)),cex = 1.3,pch =
16,xlab = "Weight in Kg",ylab = "Height in cm")
# Save the file.
dev.off()
```

Practical 10

AIM: Perform the logistic regression on the given data warehouse data.

Solution:

To perform this download the data from the link given below:

<https://github.com/TarekDib03/Analytics/blob/master/Week3%20-%20Logistic%20Regression/Data/quality.csv>

Code:

```
> quality <- read.csv('C:/users/Anon/Desktop/BI Practical/quality.csv')
> str(quality)
```

Output:

```
'data.frame': 131 obs. of 14 variables:
 $ MemberID      : int  1 2 3 4 5 6 7 8 9 10 ...
 $ InpatientDays : int  0 1 0 0 8 2 16 2 2 4 ...
 $ ERVisits       : int  0 1 0 1 2 0 1 0 1 2 ...
 $ OfficeVisits   : int  18 6 5 19 19 9 8 8 4 0 ...
 $ Narcotics      : int  1 1 3 0 3 2 1 0 3 2 ...
 $ DaysSinceLastERVisit: num 731 411 731 158 449 ...
 $ Pain           : int 10 0 10 34 10 6 4 5 5 2 ...
 $ TotalVisits    : int 18 8 5 20 29 11 25 10 7 6 ...
 $ ProviderCount  : int 21 27 16 14 24 40 19 11 28 21 ...
 $ MedicalClaims  : int 93 19 27 59 51 53 40 28 20 17 ...
 $ ClaimLines     : int 222 115 148 242 204 156 261 87 98 66 ...
 .
 $ StartedOnCombination: logi FALSE FALSE FALSE FALSE FALSE ...
 .
 $ AcuteDrugGapSmall : int 0 1 5 0 0 4 0 0 0 0 ...
 $ PoorCare         : int 0 0 0 0 0 1 0 0 1 0 ...

> table(quality$PoorCare)

 0 1
98 33

> 98/131
[1] 0.7480916

> install.packages("caTools")

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/users/Anon/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
also installing the dependency 'bitops'

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/bitops_1.0-7.zip'
Content type 'application/zip' length 31679 bytes (30 KB)
downloaded 30 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/caTools_1.18.2.zip'
Content type 'application/zip' length 245985 bytes (240 KB)
downloaded 240 KB
```

```
package 'bitops' successfully unpacked and MD5 sums checked
package 'caTools' successfully unpacked and MD5 sums checked
```

```
The downloaded binary packages are in
C:\Users\Anon\AppData\Local\Temp\RtmpCyEvnb\downloaded_packages
```

```
> library(caTools)
> set.seed(88)
> split = sample.split(quality$PoorCare, SplitRatio = 0.75)
> split
 [1] TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE  T
RUE FALSE
 [13] TRUE  T
RUE TRUE
 [25] FALSE TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE TRUE  T
RUE TRUE
 [37] FALSE TRUE  TRUE  TRUE FALSE FALSE TRUE  TRUE FALSE TRUE  FA
LSE TRUE
 [49] FALSE TRUE  TRUE FALSE FALSE TRUE  TRUE TRUE  TRUE TRUE  T
RUE TRUE
 [61] TRUE  TRUE  TRUE  TRUE FALSE TRUE  TRUE TRUE  TRUE TRUE  FA
LSE TRUE
 [73] TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE TRUE  T
RUE FALSE
 [85] TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE T
RUE TRUE
 [97] TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE TRUE  TRUE  T
RUE FALSE
[109] TRUE FALSE FALSE TRUE  TRUE FALSE TRUE  TRUE TRUE  TRUE FALSE T
RUE TRUE
[121] FALSE TRUE  TRUE FALSE TRUE  TRUE FALSE TRUE  TRUE TRUE  FA
LSE

> qualityTrain = subset(quality, split == TRUE)
> qualityTest = subset(quality, split == FALSE)
> nrow(qualityTrain)
[1] 99
> nrow(qualityTest)
[1] 32

> QualityLog = glm(PoorCare ~ OfficeVisits + Narcotics, data = qualityTrain, family = binomial)
> summary(QualityLog)

Call:
glm(formula = PoorCare ~ OfficeVisits + Narcotics, family = binomial
, data = qualityTrain)

Deviance Residuals:
    Min      1Q      Median      3Q      Max 
-2.06303 -0.63155 -0.50503 -0.09689  2.16686 

Coefficients:
            Estimate Std. Error z value Pr(>|z|)    
(Intercept) -2.64613   0.52357 -5.054 4.33e-07 ***
OfficeVisits  0.08212   0.03055  2.688  0.00718 ** 
Narcotics     0.07630   0.03205  2.381  0.01728 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)
```

```

Null deviance: 111.888 on 98 degrees of freedom
Residual deviance: 89.127 on 96 degrees of freedom
AIC: 95.127

Number of Fisher Scoring iterations: 4

> predictTrain = predict(QualityLog, type = "response")
> summary(predictTrain)
   Min. 1st Qu. Median Mean 3rd Qu. Max.
0.06623 0.11912 0.15967 0.25253 0.26765 0.98456

> tapply(predictTrain, qualityTrain$PoorCare, mean)
0          1
0.1894512 0.4392246
> table(qualityTrain$PoorCare, predictTrain > 0.5)

    FALSE TRUE
0      70    4
1      15   10
> 10/25
[1] 0.4
> 70/74
[1] 0.9459459
> table(qualityTrain$PoorCare, predictTrain > 0.7)

    FALSE TRUE
0      73    1
1      17    8
> 8/25
[1] 0.32
> 73/74
[1] 0.9864865
> table(qualityTrain$PoorCare, predictTrain > 0.2)

    FALSE TRUE
0      54   20
1       9   16
> 16/25
[1] 0.64
> 54/74
[1] 0.7297297

> install.packages("ROCR")

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/Anon/AppData/Local/R/win-library/4.2'
(as 'lib' is unspecified)
also installing the dependencies 'gtools', 'gplots'

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/gtools_3.9.4.zip'
Content type 'application/zip' length 359411 bytes (350 KB)
downloaded 350 KB

trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/gplots_3.1.3.zip'
Content type 'application/zip' length 603183 bytes (589 KB)
downloaded 589 KB

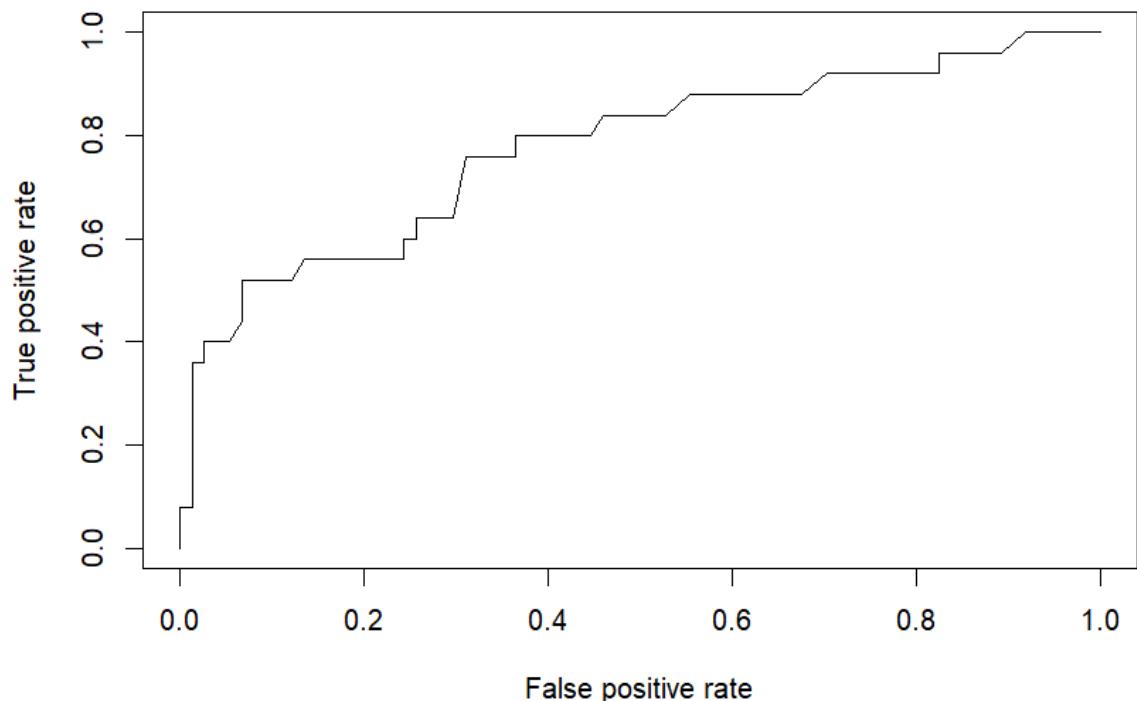
```

```
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/ROCR_1.0-11.zip'
Content type 'application/zip' length 453649 bytes (443 KB)
downloaded 443 KB
```

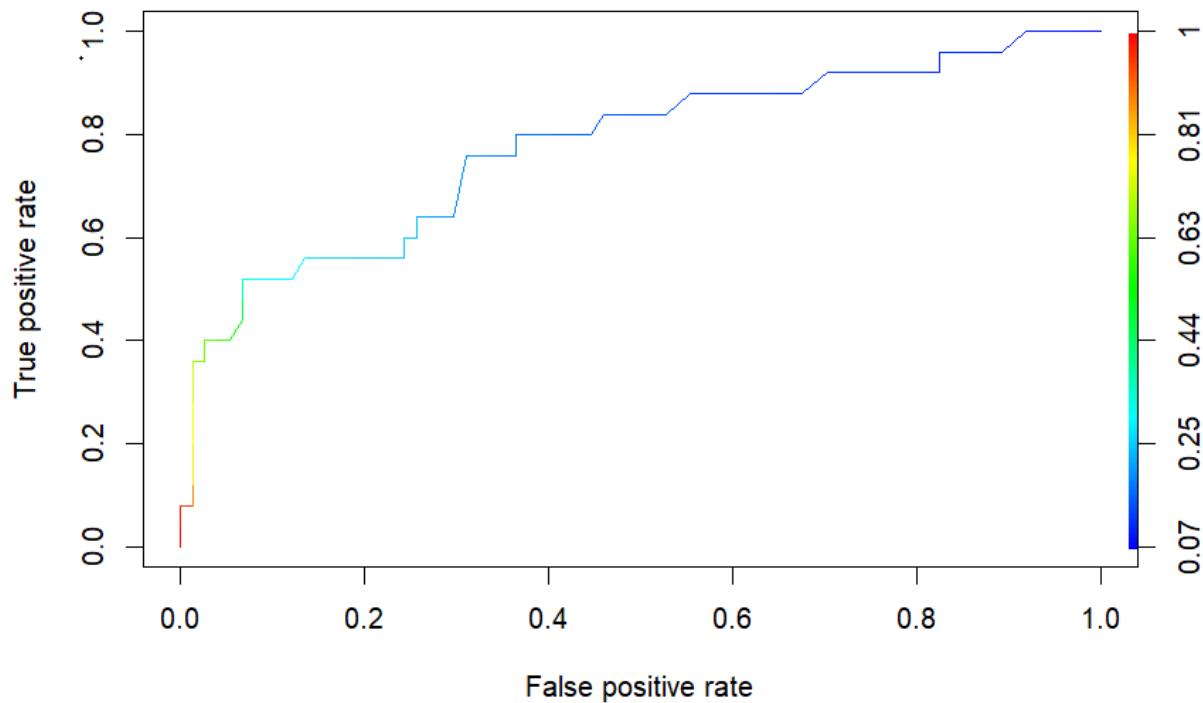
```
package 'gtools' successfully unpacked and MD5 sums checked
package 'gplots' successfully unpacked and MD5 sums checked
package 'ROCR' successfully unpacked and MD5 sums checked
```

```
The downloaded binary packages are in
  C:\Users\Anon\AppData\Local\Temp\RtmpCyEvnb\downloaded_packages
```

```
> library(ROCR)
> ROCRpred = prediction(predictTrain, qualityTrain$PoorCare)
> ROCRpref = performance(ROCRpred, "tpr", "fpr")
> plot(ROCRpref)
```



```
> plot(ROCRpref, colorize=TRUE)
```



Complete code:

```

quality <- read.csv('C:/Users/Anon/Desktop/BI Practical/quality.csv')
str(quality)
table(quality$PoorCare)
98/131
install.packages("caTools")
library(caTools)
set.seed(88)
split = sample.split(quality$PoorCare, SplitRatio = 0.75)
split
qualityTrain = subset(quality, split == TRUE)
qualityTest = subset(quality, split == FALSE)
nrow(qualityTrain)
nrow(qualityTest)
QualityLog = glm(PoorCare ~ OfficeVisits + Narcotics, data = qualityTrain, family = binomial)
summary(QualityLog)
predictTrain = predict(QualityLog, type = "response")
summary(predictTrain)
tapply(predictTrain, qualityTrain$PoorCare, mean)
table(qualityTrain$PoorCare, predictTrain > 0.5)
10/25
70/74
table(qualityTrain$PoorCare, predictTrain > 0.7)
8/25
73/74
table(qualityTrain$PoorCare, predictTrain > 0.2)
16/25

```

54/74

```
install.packages("ROCR")
library(ROCR)
ROCRpred = prediction(predictTrain, qualityTrain$PoorCare)
ROCRpref = performance(ROCRpred, "tpr", "fpr")
plot(ROCRpref)
plot(ROCRpref, colorize=TRUE)
plot(ROCRpref, colorize=TRUE, print, print.cutoffs.at = seq(0,1,by =
0.1), text.adj = c(-0.2,0.7))
```

Practical 11

AIM: Practical implementation of Decision tree using R tool

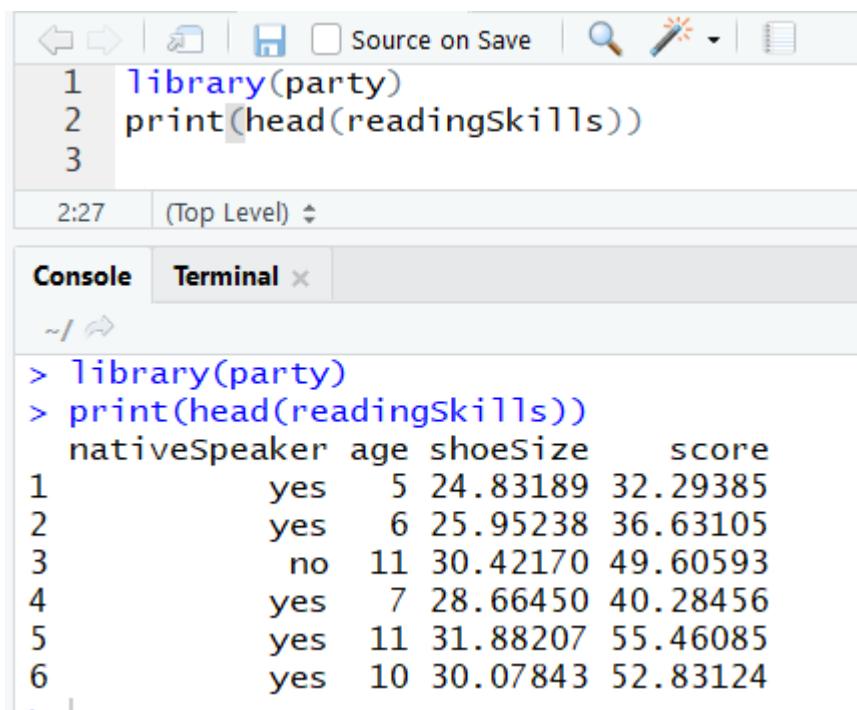
Solution:

Step 1: Install "party package"

```
>install.packages("party")
```

Step 2: Code

```
library(party)  
print(head(readingSkills))
```



The screenshot shows the RStudio interface. The top panel displays the following R code:

```
1 library(party)  
2 print(head(readingSkills))  
3
```

The bottom panel shows the Console tab with the following output:

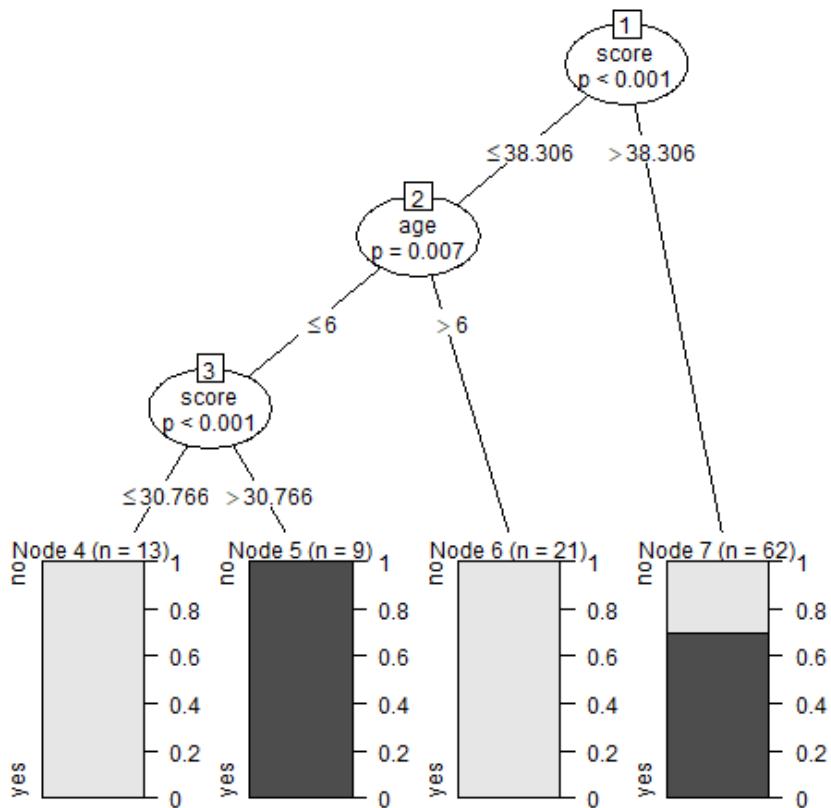
```
> library(party)  
> print(head(readingSkills))  
 nativeSpeaker age shoeSize score  
 1 yes 5 24.83189 32.29385  
 2 yes 6 25.95238 36.63105  
 3 no 11 30.42170 49.60593  
 4 yes 7 28.66450 40.28456  
 5 yes 11 31.88207 55.46085  
 6 yes 10 30.07843 52.83124
```

```
input.dat <- readingSkills[c(1:105),]  
png(file = "decision_tree.png")  
output.tree <- ctree(nativeSpeaker ~ age + shoeSize + score, data = input.dat)  
plot(output.tree)  
dev.off()
```

OUTPUT

```
4 input.dat <- readingSkills[c(1:105),]
5 png(file = "decision_tree.png")
6 output.tree <- ctree(nativeSpeaker ~ age + shoeSize + score,data = input.dat)
7 plot(output.tree)
8 dev.off()
```

```
8:10 (Top Level) ⇧
Console Terminal ✘
~/
> input.dat <- readingSkills[c(1:105),]
> png(file = "decision_tree.png")
> output.tree <- ctree(nativeSpeaker ~ age + shoeSize + score,data = input.dat)
> plot(output.tree)
> dev.off()
null device
1
> |
```



Complete Code

```
library(party)
print(head(readingSkills))
input.dat <- readingSkills[c(1:105),]
png(file = "decision_tree.png")
output.tree <- ctree(nativeSpeaker ~ age + shoeSize + score,data = input.dat)
plot(output.tree)
dev.off()
```