

Task No: 2 Date:	To configure, monitor, and administer a Data warehouse and perform basic Query operations on the DW. <b>Tools: SQL Server Management Studio (SSMS), Microsoft Azure SQL Pool</b>	
---------------------	---	--

## TASK 2a

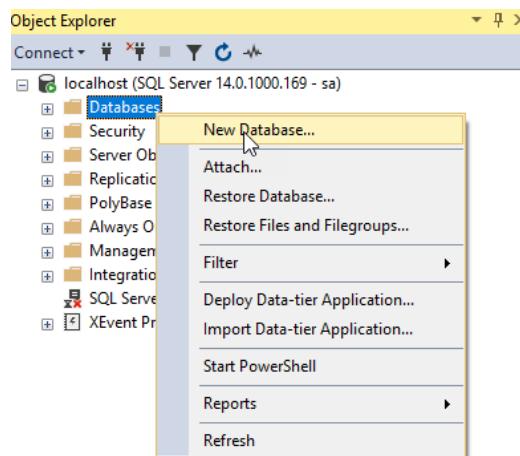
### AIM

Create a new database in SQL Server and execute the script to configure and load the sample database.

### PROCEDURE:

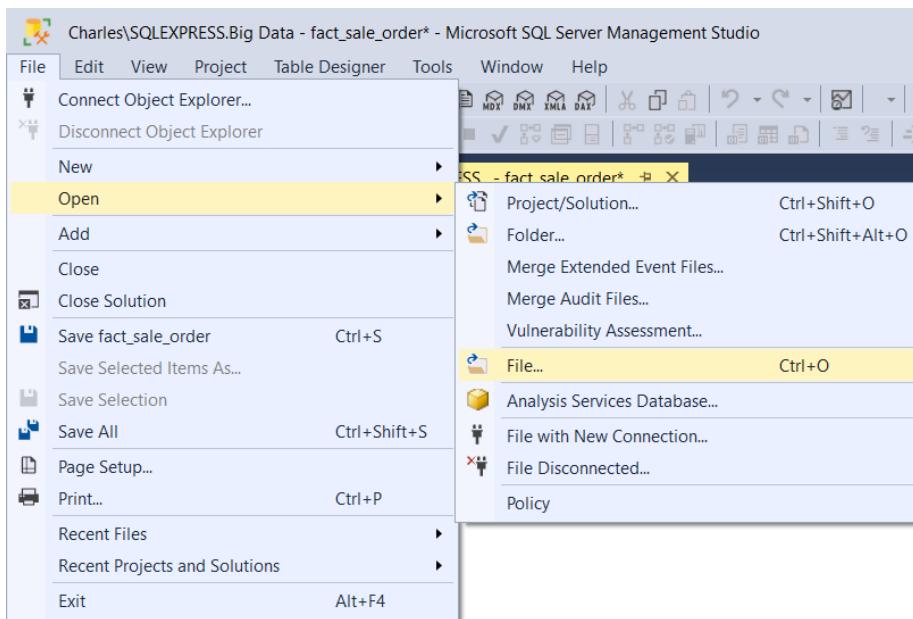
Connect to the SQL Server by choosing the server name, enter the user and password and click the Connect button.

Right-click the Databases node in the Object Explorer and select the New Database... menu item

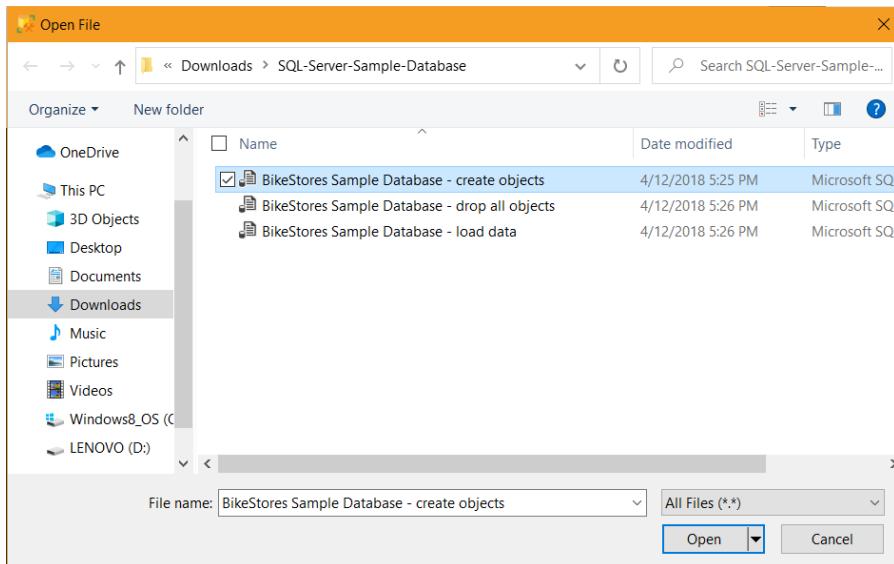


Enter the **Database name** as BikeStores and click the **OK** button to create the new database.

From the File menu, choose Open > File... menu item to open a script file.



Select the **BikeStores Sample Database – create** objects.sql file and click the Open button



Click the **Execute** button to execute the SQL script

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. In the Object Explorer on the left, under the 'CHARLES\SQLEXPRESS (SQL Server)' node, the 'Databases' folder is expanded, showing 'System Databases', 'Database Snapshots', 'AdventureWorksDW2022', and 'Big Data'. Under 'Big Data', 'Tables' is expanded, showing 'dbo.fact\_sale\_order' and 'dbo.fact\_sales\_order'. Other nodes like 'Views', 'External Resources', 'Synonyms', etc., are also visible. In the center, a query window titled 'BikeStores Sample...ARLES\Lenovo (52)' is open, displaying SQL code for creating schemas and tables. The code includes:

```
CREATE SCHEMA production;
go

CREATE SCHEMA sales;
go

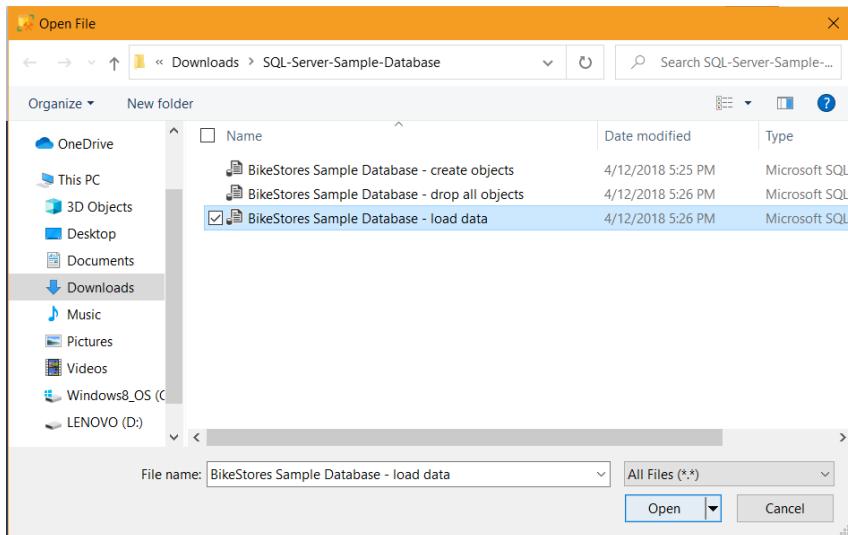
-- create tables
CREATE TABLE production.categories (
    category_id INT IDENTITY (1, 1) PRIMARY KEY,
    category_name VARCHAR (255) NOT NULL
);

CREATE TABLE production.brands (
    brand_id INT IDENTITY (1, 1) PRIMARY KEY,
    brand_name VARCHAR (255) NOT NULL
);

CREATE TABLE production.products (
    product_id INT IDENTITY (1, 1) PRIMARY KEY
```

The 'Messages' pane at the bottom right shows the message "Commands completed successfully." and the completion time: 2023-06-22T16:02:06.9671541+05:30.

Choose the **BikeStores Sample Database – load data.sql** file and click the Open button.



Click the **Execute** button to load data into the tables.

```
Messages

(1 row affected)

110 % ▶

Query executed successfully.
```

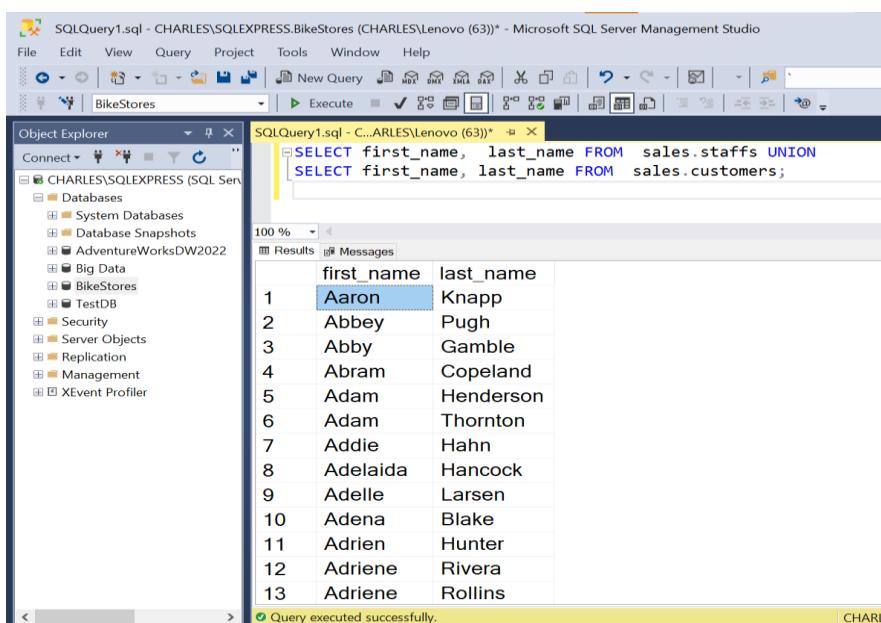
Execute query to display the values from production.products table.

## QUERY OPERATIONS ON THE DATA WAREHOUSE

### UNION OPERATION

```
SELECT first_name, last_name FROM sales.staffs UNION
```

```
SELECT first_name, last_name FROM sales.customers;
```



The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar reads "SQLQuery1.sql - CHARLES\SQLEXPRESS.BikeStores (CHARLES\Lenovo (63)) - Microsoft SQL Server Management Studio". The left pane is the Object Explorer, showing the database structure. The right pane contains a query window with the following code:

```
SELECT first_name, last_name FROM sales.staffs UNION
SELECT first_name, last_name FROM sales.customers;
```

Below the query window is a results grid displaying 13 rows of data:

	first_name	last_name
1	Aaron	Knapp
2	Abbey	Pugh
3	Abby	Gamble
4	Abram	Copeland
5	Adam	Henderson
6	Adam	Thornton
7	Addie	Hahn
8	Adelaida	Hancock
9	Adelle	Larsen
10	Adena	Blake
11	Adrien	Hunter
12	Adriene	Rivera
13	Adriene	Rollins

At the bottom of the results grid, a green status bar says "Query executed successfully."

## UNION ALL OPERATION

```
SELECT first_name, last_name FROM sales.staffs UNION ALL
```

```
SELECT first_name, last_name FROM sales.customers;
```

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including the BikeStores database. The central pane displays the results of the following query:

```
SELECT first_name, last_name FROM sales.staffs UNION ALL
SELECT first_name, last_name FROM sales.customers;
```

The results are presented in a table with columns 'first\_name' and 'last\_name'. The data consists of 13 rows, starting with Fabiola Jackson and ending with Tameka Fisher.

	first_name	last_name
1	Fabiola	Jackson
2	Mireya	Copeland
3	Genna	Serrano
4	Virgie	Wiggins
5	Jannette	David
6	Marcelene	Boyer
7	Venita	Daniel
8	Kali	Vargas
9	Layla	Terrell
10	Bernardine	Houston
11	Debra	Burks
12	Kasha	Todd
13	Tameka	Fisher

At the bottom of the results pane, a message indicates: "Query executed successfully."

## INTERSECT OPERATION

```
SELECT city FROM sales.customers INTERSECT
```

```
SELECT city FROM sales.stores ORDER BY city;
```

The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, including the BikeStores database. The central pane displays the results of the following query:

```
SELECT city FROM sales.customers INTERSECT
SELECT city FROM sales.stores ORDER BY city;
```

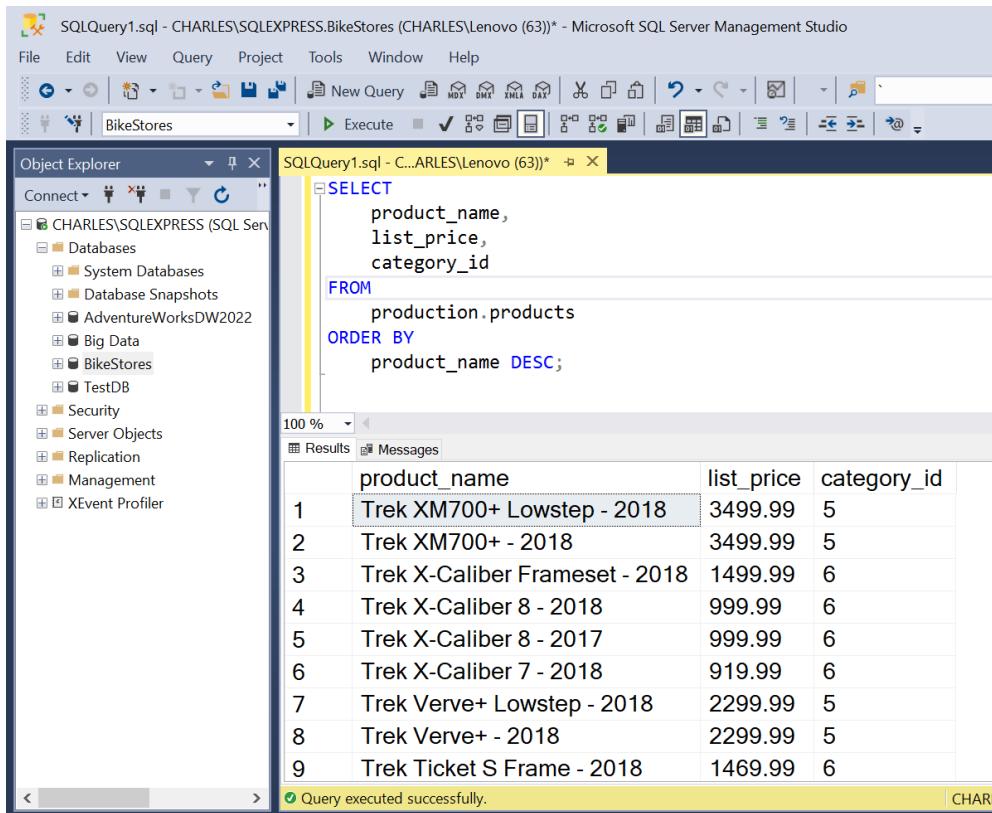
The results are presented in a table with a single column 'city'. The data consists of 3 rows, ordered by city name: Baldwin, Rowlett, and Santa Cruz.

city	
1	Baldwin
2	Rowlett
3	Santa Cruz

## JOIN OPERATION

### INNER JOIN

```
SELECT
    product_name,
    list_price,
    category_id
FROM
    production.products
ORDER BY
    product_name DESC;
```



The screenshot shows the Microsoft SQL Server Management Studio interface. The left pane displays the Object Explorer with the connection to 'CHARLES\SQLEXPRESS.BikeStores'. The right pane shows the results of the executed query:

```
SELECT
    product_name,
    list_price,
    category_id
FROM
    production.products
ORDER BY
    product_name DESC;
```

The results grid displays the following data:

	product_name	list_price	category_id
1	Trek XM700+ Lowstep - 2018	3499.99	5
2	Trek XM700+ - 2018	3499.99	5
3	Trek X-Caliber Frameset - 2018	1499.99	6
4	Trek X-Caliber 8 - 2018	999.99	6
5	Trek X-Caliber 8 - 2017	999.99	6
6	Trek X-Caliber 7 - 2018	919.99	6
7	Trek Verve+ Lowstep - 2018	2299.99	5
8	Trek Verve+ - 2018	2299.99	5
9	Trek Ticket S Frame - 2018	1469.99	6

At the bottom of the results pane, a message indicates: "Query executed successfully."

### LEFT JOIN

```
SELECT
    product_name,
    order_id
FROM
    production.products p
LEFT JOIN sales.order_items o ON o.product_id = p.product_id
ORDER BY
    order_id;
```

The screenshot shows the Microsoft SQL Server Management Studio interface. The title bar reads "SQLQuery1.sql - CHARLES\SQLEXPRESS.BikeStores (CHARLES\Lenovo (63)) - Microsoft SQL Server Management Studio". The menu bar includes File, Edit, View, Query, Project, Tools, Window, and Help. The toolbar has various icons for database management. The Object Explorer on the left shows the server structure under "CHARLES\SQLEXPRESS (SQL Server)". The central pane displays a T-SQL query:

```
SELECT
    product_name,
    order_id
FROM
    production.products p
LEFT JOIN sales.order_items o ON o.product_id = p.product_id
ORDER BY
    order_id;
```

The results pane shows the output of the query:

	product_name	order_id
1	Electra Savannah 1 (20-inch) - Girl's - 2018	NULL
2	Electra Townie Go! 8i Ladies' - 2018	NULL
3	Trek Checkpoint ALR 5 Women's - 2019	NULL
4	Trek Checkpoint ALR Frameset - 2019	NULL
5	Trek Precaliber 12 Girl's - 2018	NULL
6	Surly Krampus Frameset - 2018	NULL
7	Trek Checkpoint SL 5 Women's - 2019	NULL
8	Trek 820 - 2016	NULL
9	Trek Checkpoint ALR 4 Women's - 2019	NULL

At the bottom of the results pane, a message says "Query executed successfully.".

## RESULT:

Thus the basic SQL operation was executed using MS Server Management Studio and SQL Server.

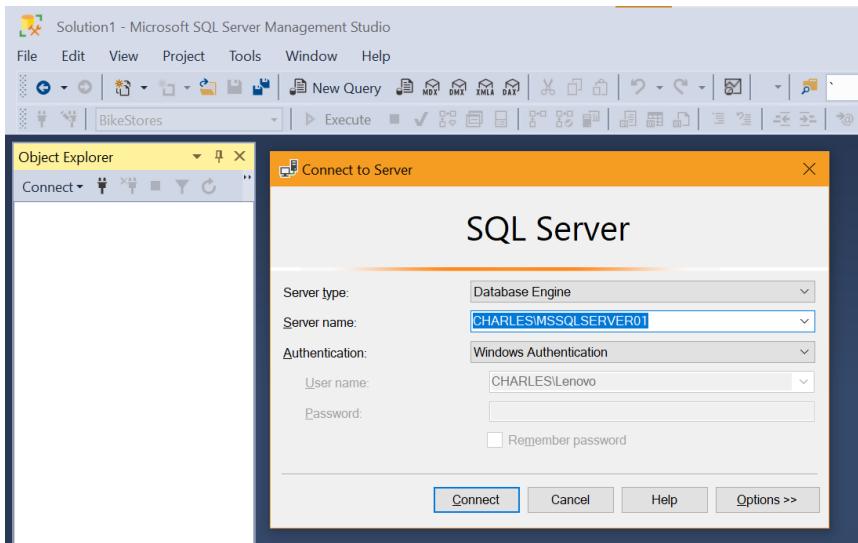
## TASK 2b:

### AIM

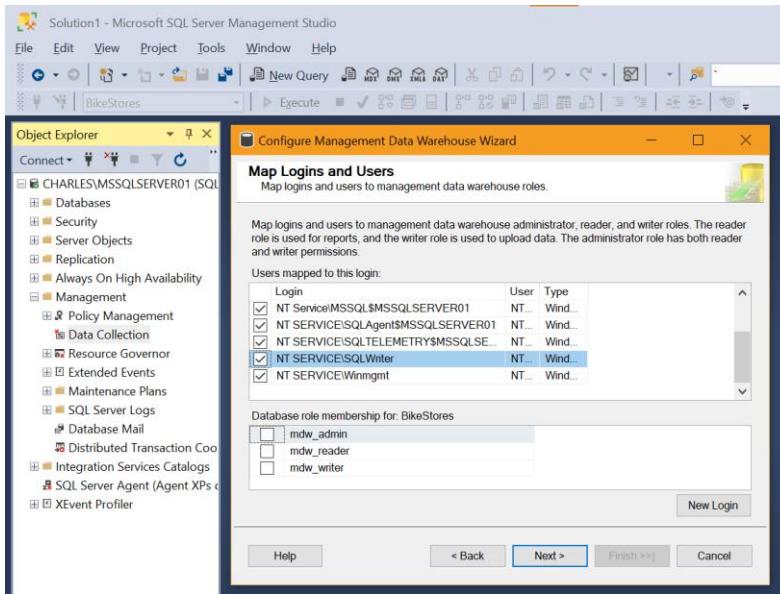
To configure, monitor, and administer a Data warehouse

### PROCEDURE

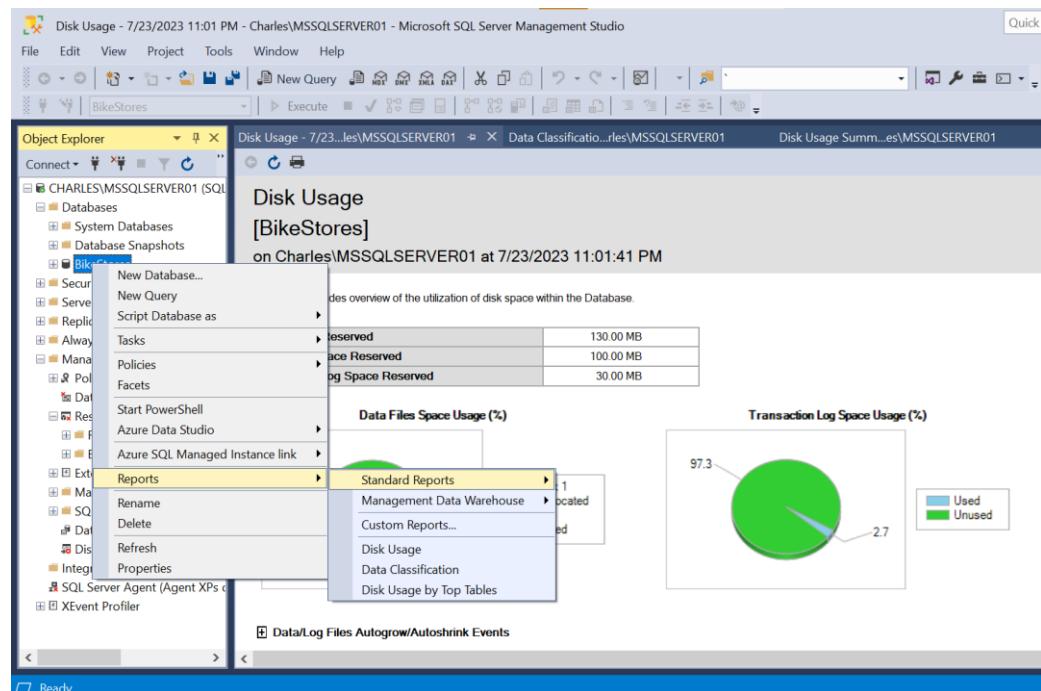
1. Install Microsoft SQL server Management Studio
2. Install MS Sql Server Developer Edition
3. Connect to the SQL Server



4. Choose Management -> Data Collection in the left side navigation bar
5. Right click on the Data collection -> Tasks -> Configure Management Data Warehouse and Configure the data warehouse settings
6. Select Server Name and Database name and click next button
7. Change the settings in map login and users
  - a. User mapped to this login
  - b. Database membership for BikeStores



8. Check the Settings and Click finish button
9. Moniter and generate the various reports for the BikeStores, Right click the Bikestores-> Reports -> Standard Reports and select the reports needed.



**Result :**

Thus a Data warehouse was configured, monitored, and administered using MS Server Management and SQL Server was completed successfully.