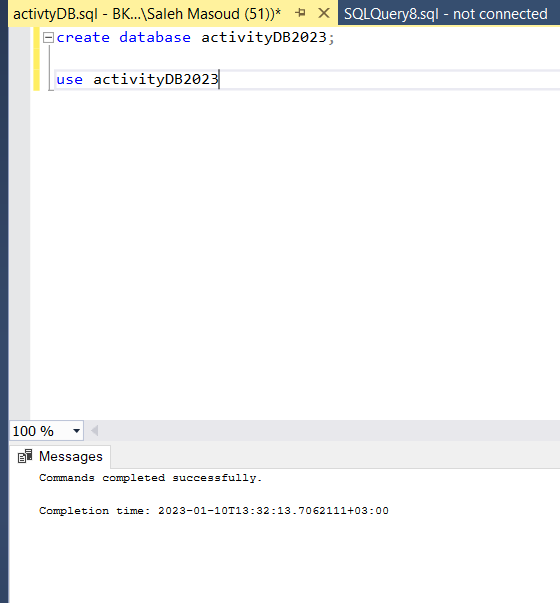
**QUESTION ONE: Queries that create a database and all the tables.**

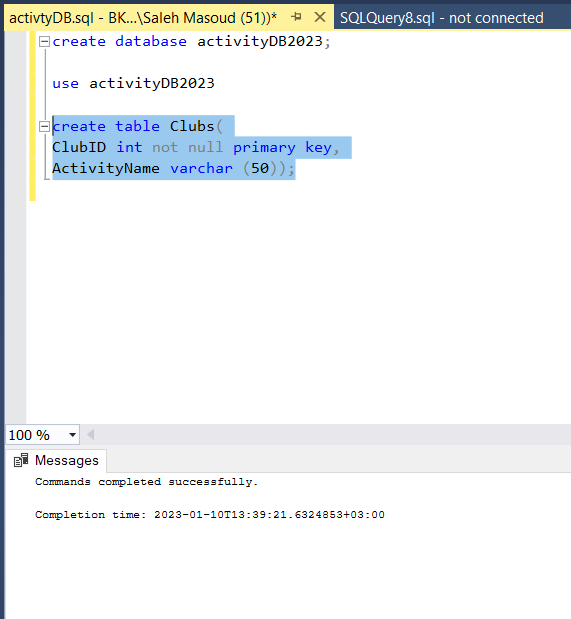
Clubs (ClubID, ActivityName)

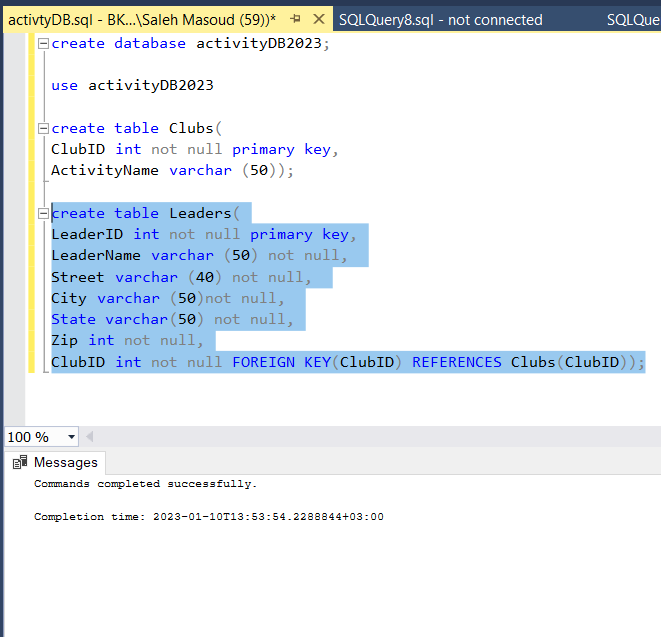
Leaders (LeaderID, LeaderName, Street, City, State, Zip, ClubID)

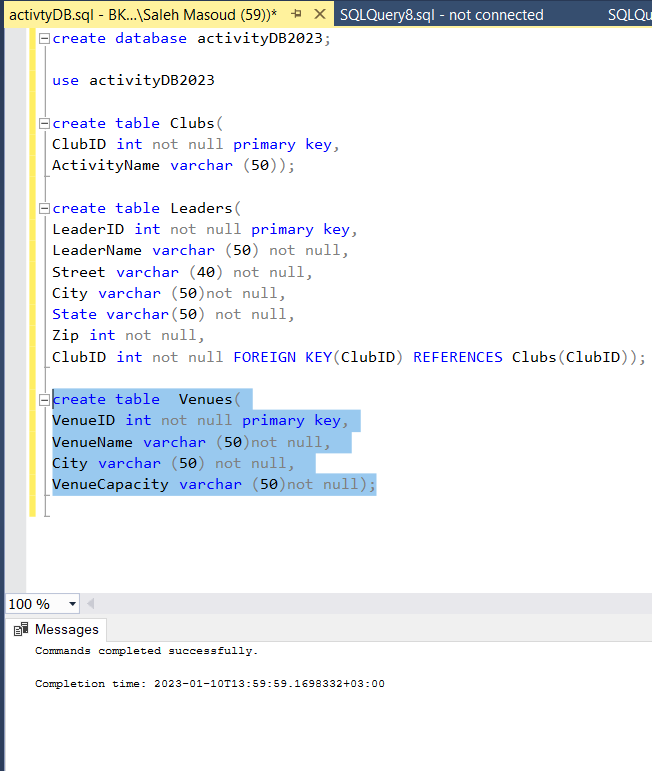
Venues (VenueID, VenueName, City, VenueCapacity)

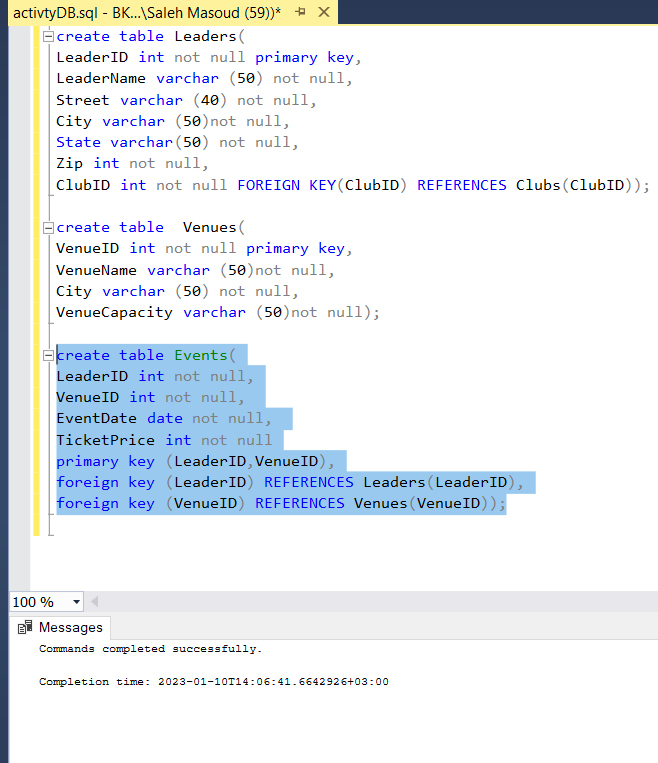
Events (LeaderID, VenueID, EventDate, TicketPrice)

****

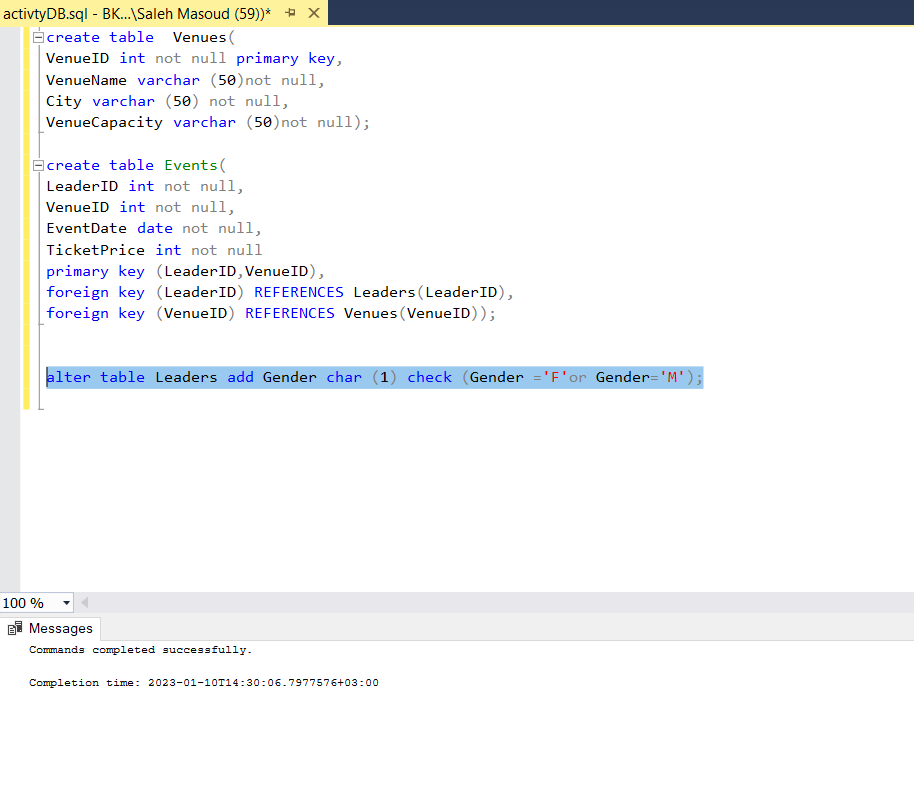
****

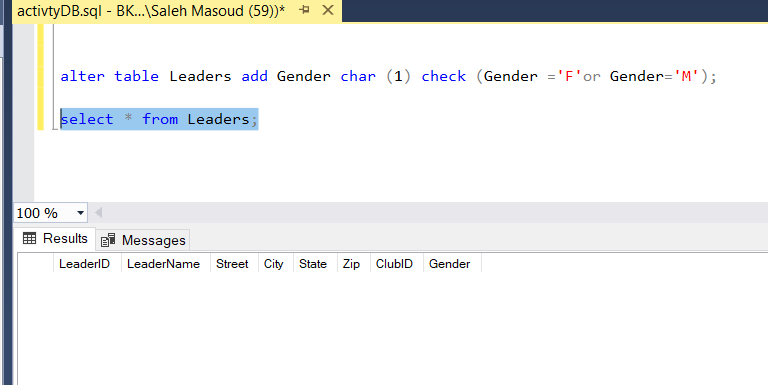
****

****

****

**QUESTION TWO: Add a gender column to the Leaders table which only accepts two values: “M” or “F” only.**

****

****

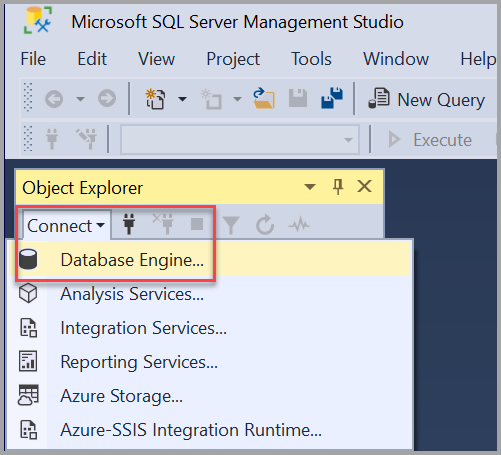
**QUESTION THREE: How to connect to an SQL Server, and the information required.**

Prerequisites:

* Install [SQL Server Management Studio](https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16).
* [Install](https://learn.microsoft.com/en-us/sql/database-engine/install-windows/install-sql-server-from-the-installation-wizard-setup?view=sql-server-ver16) and configure a [SQL Server instance](https://www.microsoft.com/sql-server/sql-server-downloads)

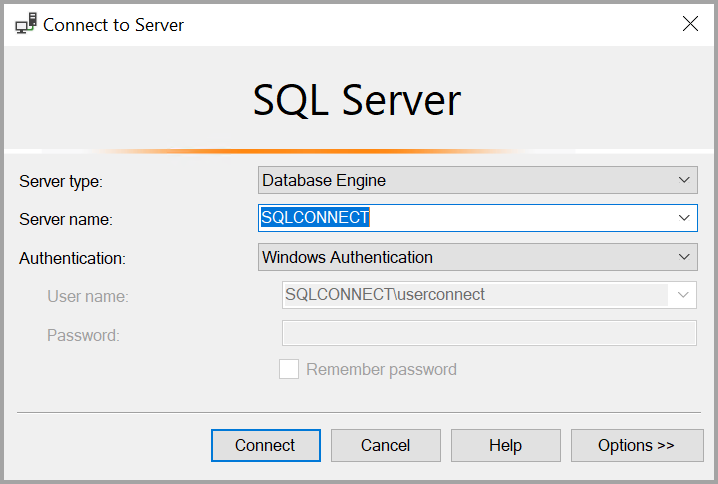
Step to connect your SQL Server;

1. Start SQL Server Management Studio. The first time you run SSMS, the **Connect to Server** window opens. If it doesn't open, you can open it manually by selecting **Object Explorer** > **Connect** > **Database Engine**.

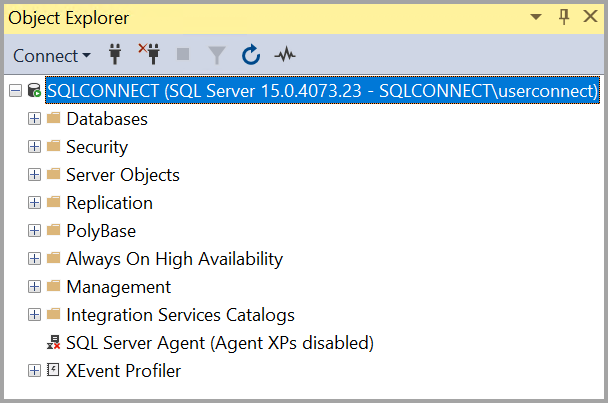


**The information required.**

Server type , another information is server name, you can use window authentication or SQL server authentication and login server account by entered user ID required when using SQL server authentication and server account password used to log in when using SQL server authentication.



1. After you have completed all the fields, select connect You can also modify additional connection options by selecting **Options**. Examples of connection options are the database you're connecting to, the connection timeout value, and the network protocol. This article uses the default values for all the fields.
2. To verify that your SQL Server connection succeeded, expand and explore the objects within **Object Explorer** where the server name, the SQL Server version, and the username are displayed. These objects are different depending on the server type.



**QUESTION FOUR: Advantages of T-SQL over SQL.**

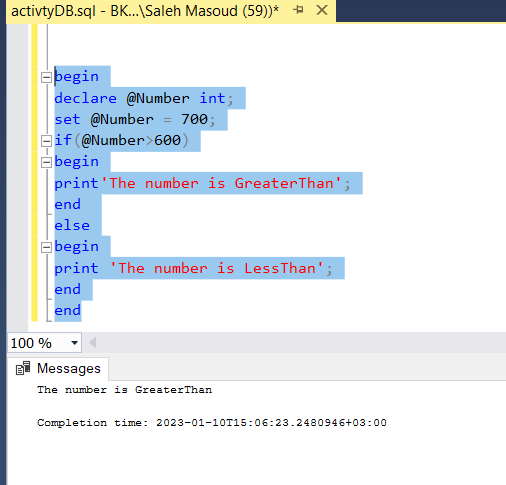
TSQL has several advantages that make this language popular and easy to use. T\_SQL is reliable in nature and an efficient language that helps to establish communication with the relational database.

1. **Processing of query is fast**: T-SQL language is capable of processing huge chunks of data very fast and also helps the data administrator to fetch the data from the database. All the data-related operations like deletion, insertion, and manipulation can be easily performed without taking much time.
2. **Secure transaction**: Data access from the database is secure in nature and also uniform in nature. All the business policies and business rules are present in encapsulated form.
3. **No extra knowledge**: T-SQL language is easy to use and no additional effort is required by the data administrator for the data retrieval process from the database.T-SQL language supports all the necessary operations like INSERT, SELECT, UPDATE, and many more.
4. **Interactive in nature**: T-SQL language is interactive in nature as the language is capable of writing complex business rules to fetch the data from the database. T-SQL language is easy to use and easy to understand.
5. **User-defined function**: One of the major advantages of the T-SQL language is providing functionality to defined user-defined functions that help to form the structure for the transactions which can be used in different platforms like e-commerce platform that helps to fetch the data in an efficient manner.

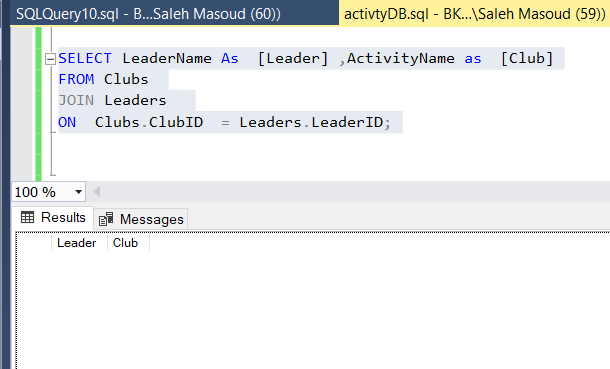
SQL has many advantages which makes it popular and highly demanded. It is a reliable and efficient language used for communicating with the database.

1. **Faster Query Processing;** Large amount of data is retrieved quickly and efficiently. Operations like Insertion, deletion, manipulation of data is also done in almost no time.
2. **No Coding Skills;** For data retrieval, large number of lines of code is not required. All basic keywords such as SELECT, INSERT INTO, UPDATE, are used and also the syntactical rules are not complex in SQL, which makes it a user-friendly language.
3. **Standardized Language;** Due to documentation and long establishment over years, it provides a uniform platform worldwide to all its users.
4. **Portable;** It can be used in programs in PCs, server, laptops independent of any platform (Operating System,). Also, it can be embedded with other applications as per need/requirement/use.
5. **Interactive Language;** Easy to learn and understand, answers to complex queries can be received in seconds.

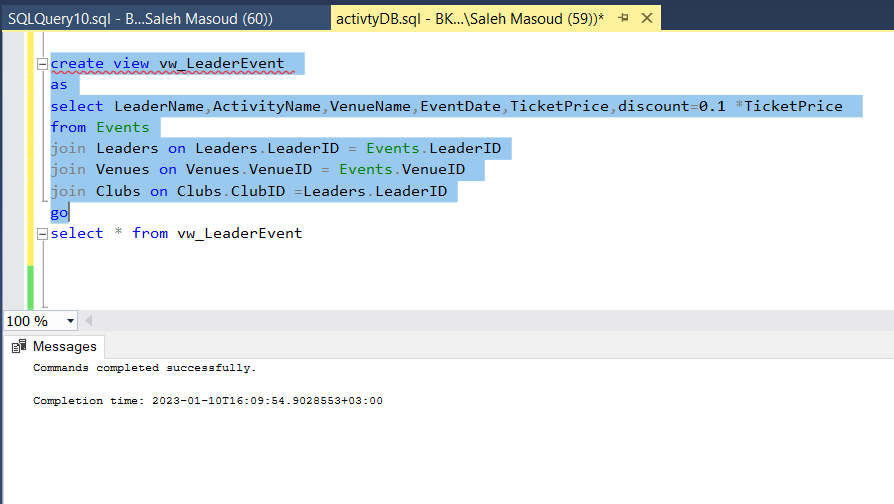
**QUESTION FIVE: T-SQL Block to declare a variable and print if a number is either less than 600 or greater than 600 to the user.**

****

**QUESTION SIX: SQL query to list all the Leaders’ name and the name of the club the leader is in. The titles should be Leader and Club respectively.**

****

**QUESTION SEVEN: View called vw\_LeaderEvent which shows the name of the Leader, name of the Activity, Venue Name, the date of the Event and discount Ticket price. The discount price is 10%**

****

**QUESTION EIGHT: Differentiate between,**

1. **Stored Procedure and User Defined Function;**

The main difference is that the idea is**a user-defined function is going to stand on its own while the stored procedure can run all command that affects the database.**

* Stored Procedure can return zero or n values, where user defined function can return one value which is mandatory.
* Stored Procedure can have input/output parameters for it, where user defined functions can have only input parameters.
* Stored Procedure allows select as well as DML statement in it, where user defined function allows only select statement in it.
* User defined Functions can be called from procedure, but stored procedures cannot be called from function.
* Stored Procedures can not be utilized in a select statement, where user defined function can be embedded in a select statement.
* UDF can be used in the SQL statements anywhere in the WHERE/HAVING/SELECT section, where Stored procedures cannot be.

1. **Trigger and Stored Procedure;**

**Triggers**

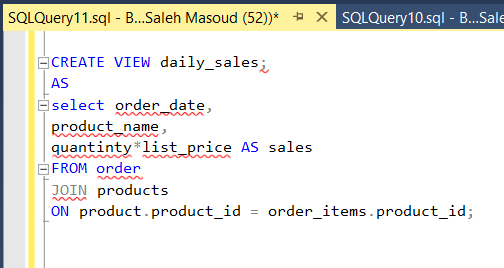
* Trigger is a stored procedure that runs automatically when various events happen (eg update, insert, delete).
* It can execute automatically based on the events.
* It can not take input as parameter.
* We can't use transaction statements inside a trigger.
* Triggers can not return values.

**Stored procedures**

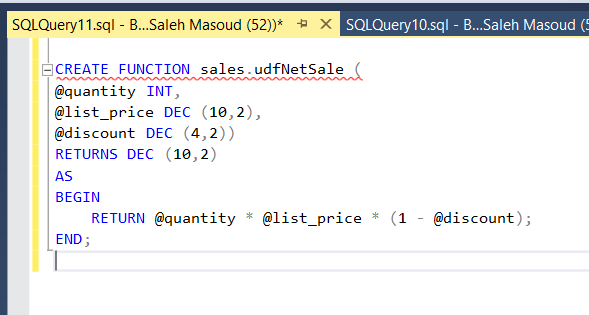
* Stored procedures are a pieces of the code in written in PL/SQL to do some specific task.
* It can be invoked explicitly by the user.
* It can take input as a parameter.
* We can use transaction statements like begin transaction, commit transaction, and rollback inside a stored procedure.
* Stored procedures can return values.

**QUESTION NINE: The diagram below to answer the questions that follow:**

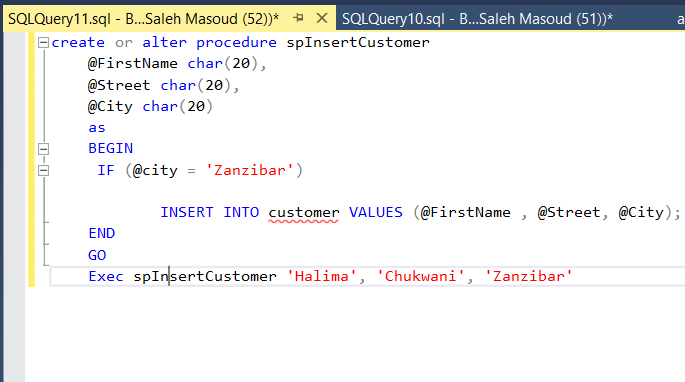
1. **Create a VIEW daily\_sales which displays the order date, product\_name and sales (which is equal to quantity \* list-price (REMEMBER DERIVED ATTRIBUTES)).**

****

1. **Create a function in the Sales Schema that calculates the net sales based on the quantity, list price, and discount: net-sale = quantity \* list\_price \* (1 - discount).**

****

1. **Create a stored procedure called spInsertCustomer that inserts a new customer into the database, under some conditions. The stored procedure has as input parameters the customer first name, city and customer street. The stored procedure should insert a row in the Customer table only if the city is equal to ‘Zanzibar”. If the condition is not satisfied, the stored procedure just terminates (no errors generated).**

****