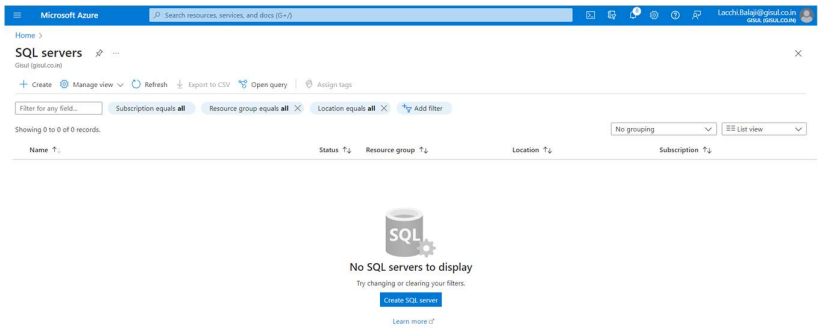
Context

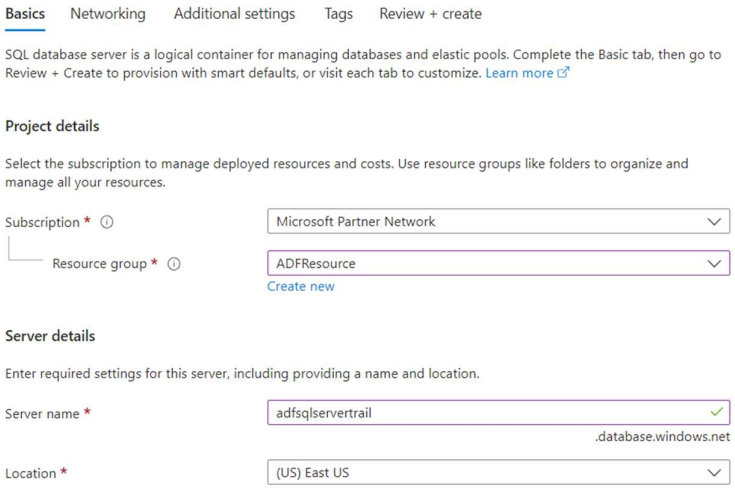
Create SQL Server

1. Login to Microsoft Azure Portal Account
2. On the Home page click on SQL servers. If you don't see the SQL Server click on Create a resource and search for SQL Server.

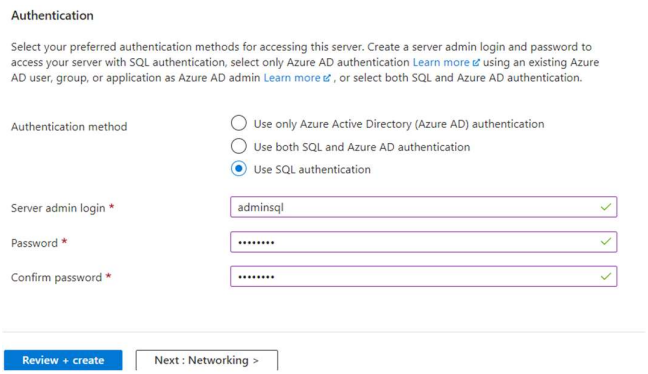
3. Click on Create SQL Server as shown below.



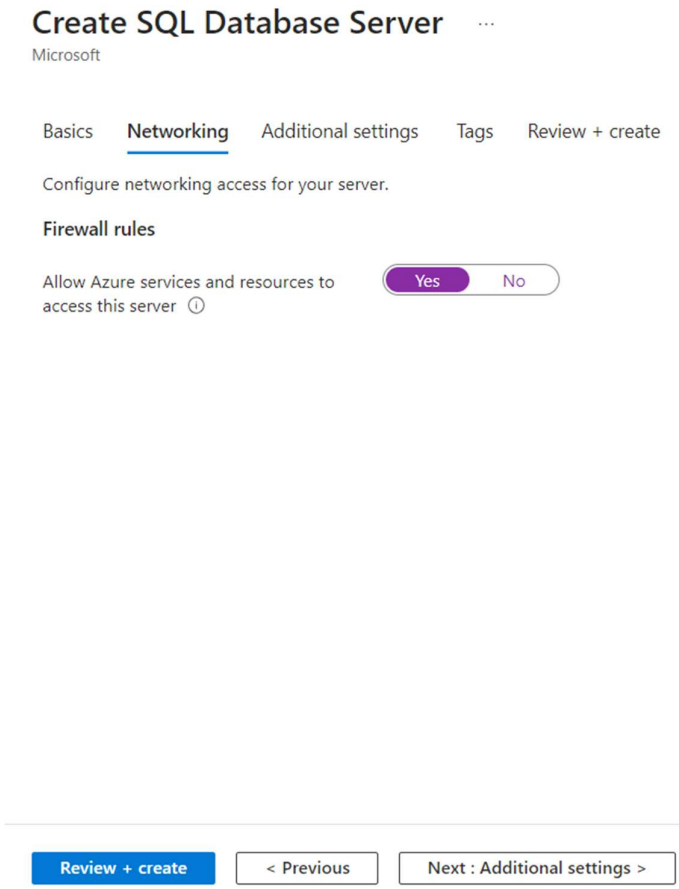
4. Under basic give the below properties as shown below.



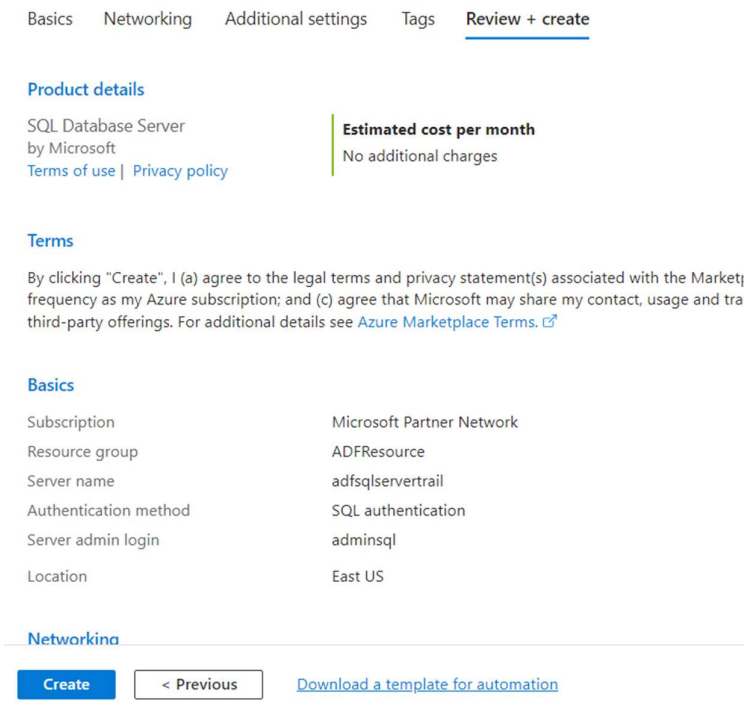
5. Scroll down and give the login name and password then click on the Next option.



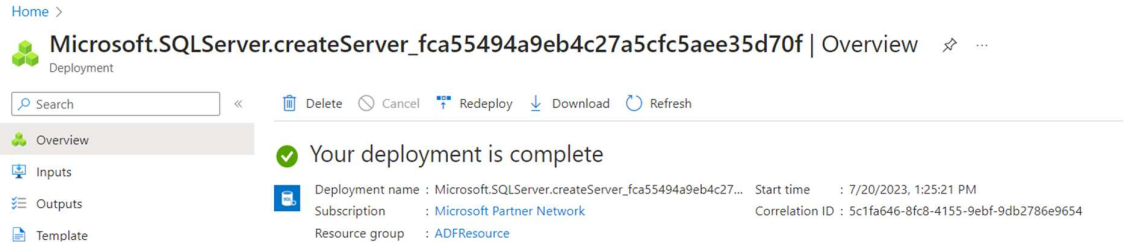
6. Set the below properties and click on Review+Create



7. Next click on Create.

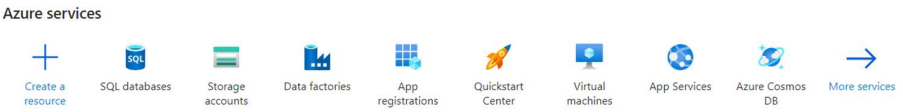


8. After some time you will see a screen like below.

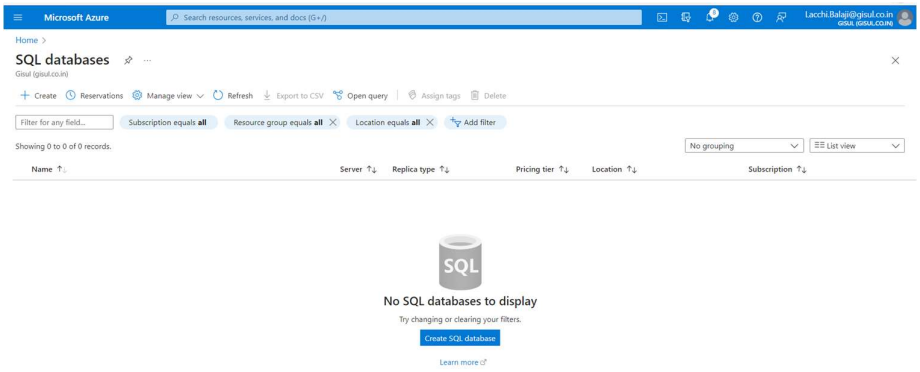


Create SQL database

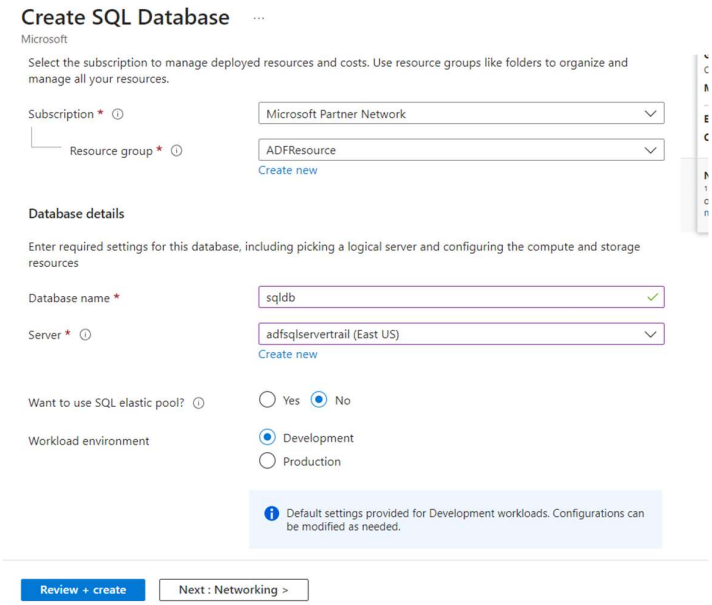
1. On the Home page click on SQL database.



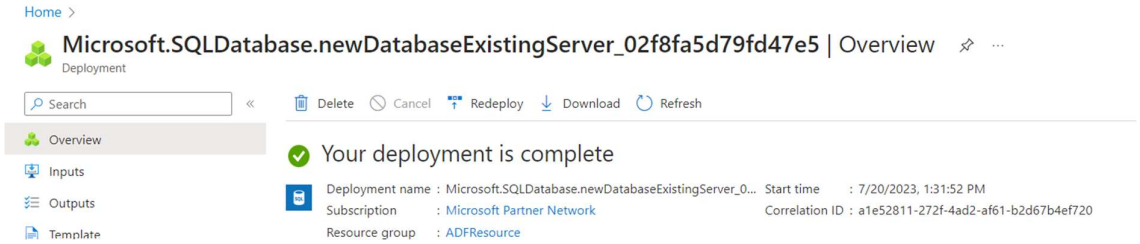
2. Next click on Create SQL database.



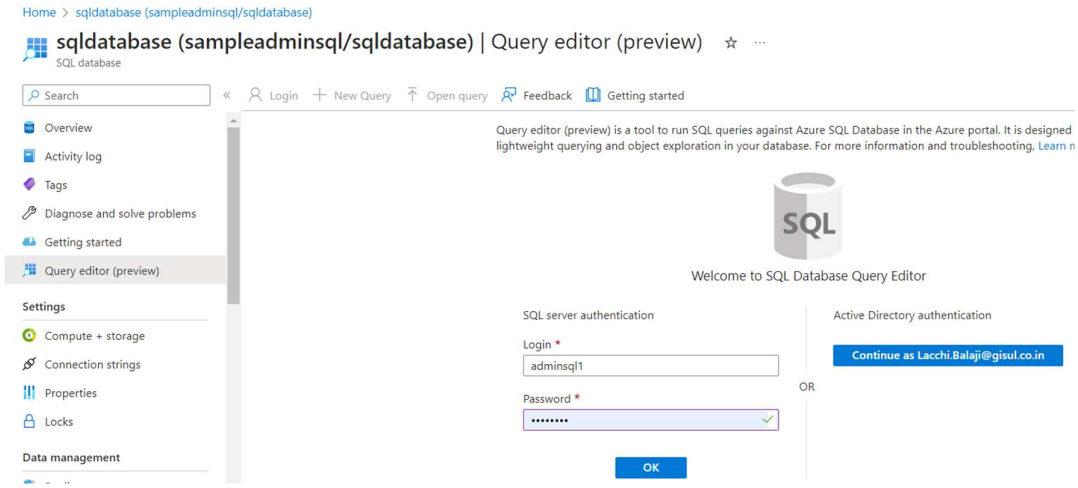
3. Give the Database name as sqldb and select the server that we created before then set the properties as shown below. And click on Reviw+Create.



4. Next click on Create.
5. After some time you will see the below screen as shown.



6. Now click on the SQL Database that we created just now.
7. Go to Query editor, give login credentials that we created in the sql server click on Ok.
8. If it asks to allow access to an IP address allow it.



9. Next you will see a Query window like below where we create our tables, views stored procedure etc.

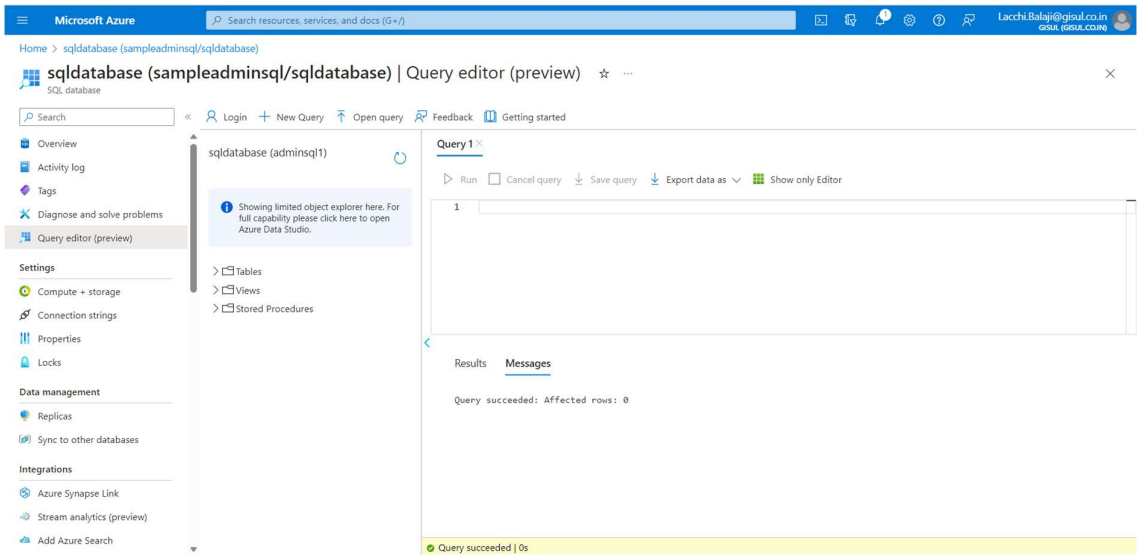
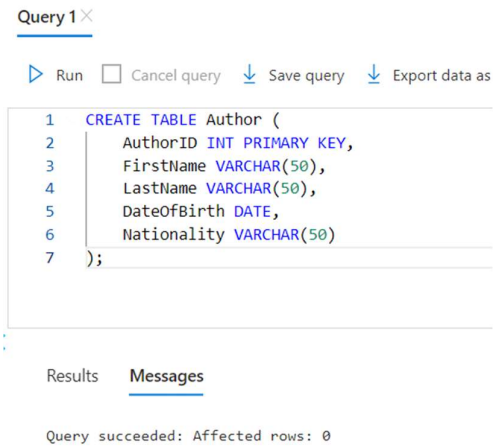


Table Creations

Exercise: Create and implement schema as per the ER Model design in the previous module and apply Constraints.

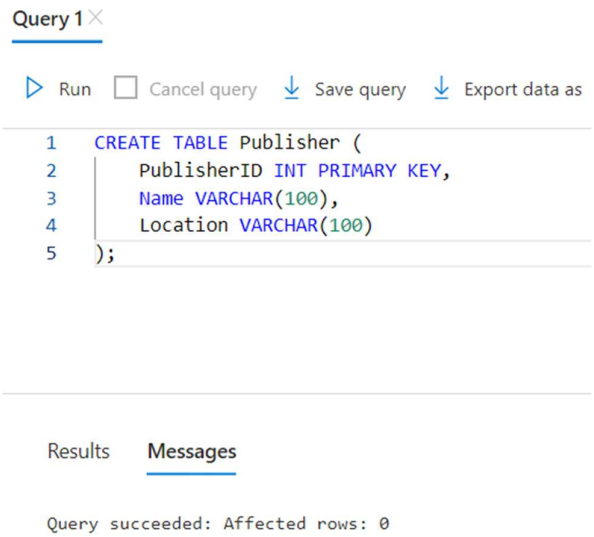
1. Now here we are creating an Author Table. Use the below query to create.

Query: CREATE TABLE Author (
AuthorID INT PRIMARY KEY,
FirstName VARCHAR(50),
LastName VARCHAR(50),
DateOfBirth DATE,
Nationality VARCHAR(50)
);



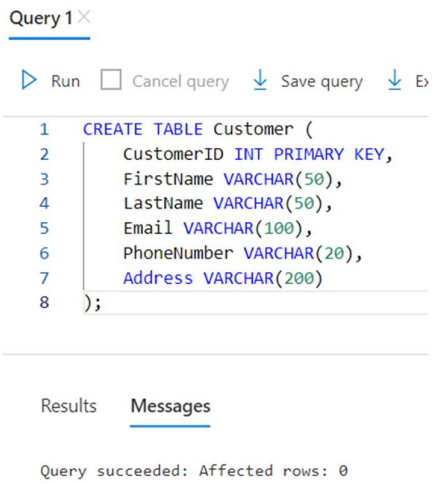
2. Next we will create a Publisher table by using the below Query.

Query: CREATE TABLE Publisher (
PublisherID INT PRIMARY KEY,
Name VARCHAR(100),
Location VARCHAR(100)
);



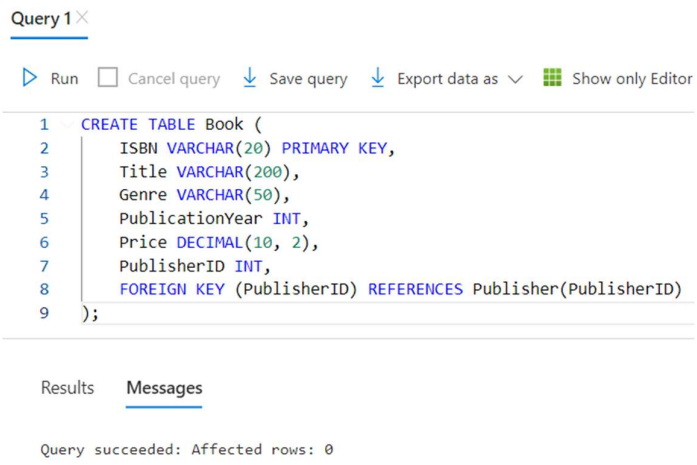
3. Next create a customer table by using the below query.

Query: CREATE TABLE Customer (
CustomerID INT PRIMARY KEY,
FirstName VARCHAR(50),
LastName VARCHAR(50),
Email VARCHAR(100),
PhoneNumber VARCHAR(20),
Address VARCHAR(200)
);



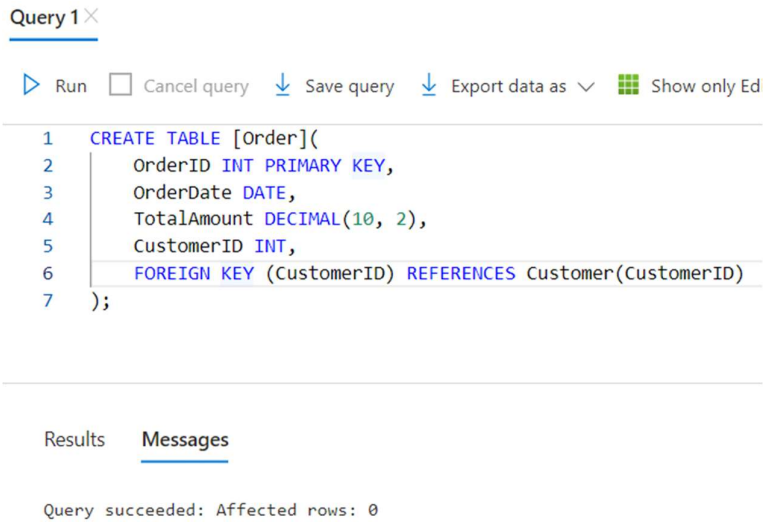
4. Create Book table and use Foreign key to make connection with Publisher table by using below Query.

Query: CREATE TABLE Book (
ISBN VARCHAR(20) PRIMARY KEY,
Title VARCHAR(200),
Genre VARCHAR(50),
PublicationYear INT,
Price DECIMAL(10, 2),
PublisherID INT,
FOREIGN KEY (PublisherID) REFERENCES Publisher(PublisherID)
);



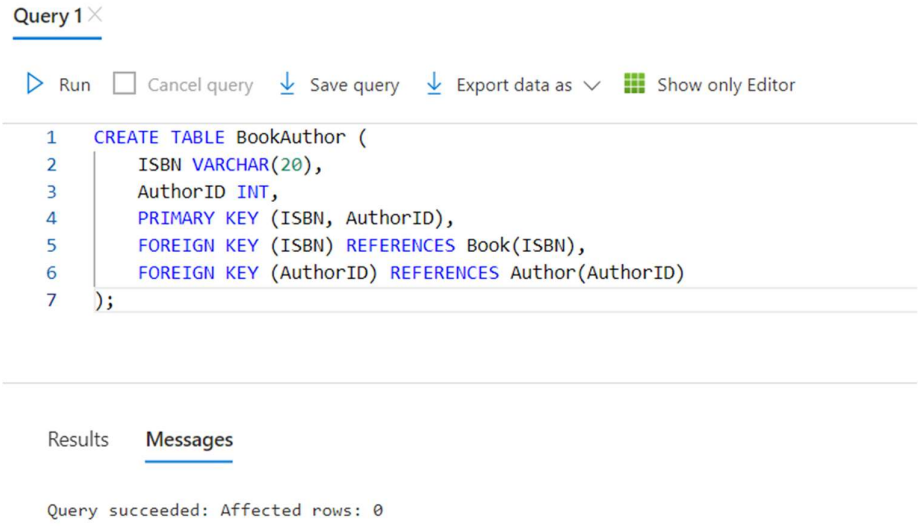
5. Create an Order table by using the below Query.

Query: CREATE TABLE [Order] (
OrderID INT PRIMARY KEY,
OrderDate DATE,
TotalAmount DECIMAL(10, 2),
CustomerID INT,
FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID)
);



6. Next create BookAuthor table by using below query.

Query: CREATE TABLE BookAuthor (
ISBN VARCHAR(20),
AuthorID INT,
PRIMARY KEY (ISBN, AuthorID),
FOREIGN KEY (ISBN) REFERENCES Book(ISBN),
FOREIGN KEY (AuthorID) REFERENCES Author(AuthorID)
);



7. Create our last table OrderBook by using the below query.

Query: CREATE TABLE OrderBook (
  OrderID INT,
  ISBN VARCHAR(20),
  PRIMARY KEY (OrderID, ISBN),
  FOREIGN KEY (OrderID) REFERENCES Order(OrderID),
  FOREIGN KEY (ISBN) REFERENCES Book(ISBN)
);

Insert Data

1. Insert data into the Author table by using the below query.

Query: INSERT INTO Author (AuthorID, FirstName, LastName, DateOfBirth, Nationality)
VALUES
  (1, 'J.K.', 'Rowling', '1965-07-31', 'British'),
  (2, 'George', 'Orwell', '1903-06-25', 'British')

2. To check the Data, use the below query.

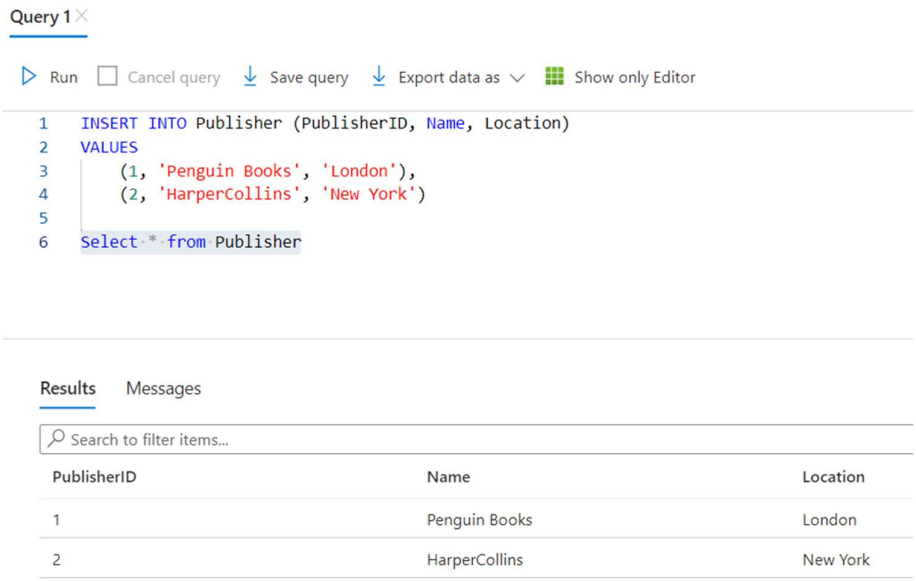
Query: Select * from Author;

3. Insert data into the Publisher table by using the below Query.

Query: INSERT INTO Publisher (PublisherID, Name, Location)
VALUES
(1, 'Penguin Books', 'London'),
(2, 'HarperCollins', 'New York')

4. To check the Data, use the below query.

Query: Select * from Publisher;

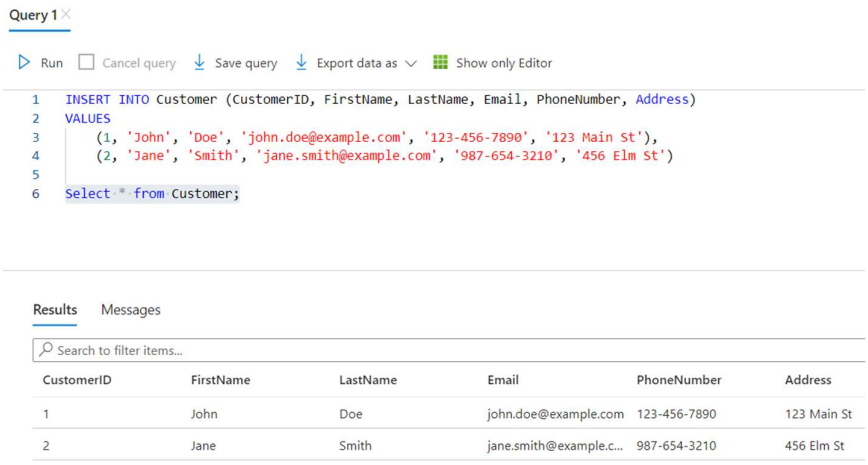


5. Insert data into the Customer table by using the below Query.

Query: INSERT INTO Customer (CustomerID, FirstName, LastName, Email, PhoneNumber, Address)
VALUES
(1, 'John', 'Doe', 'john.doe@example.com', '123-456-7890', '123 Main St'),
(2, 'Jane', 'Smith', 'jane.smith@example.com', '987-654-3210', '456 Elm St')

6. To check the Data, use the below query.

Query: Select * from Customer;

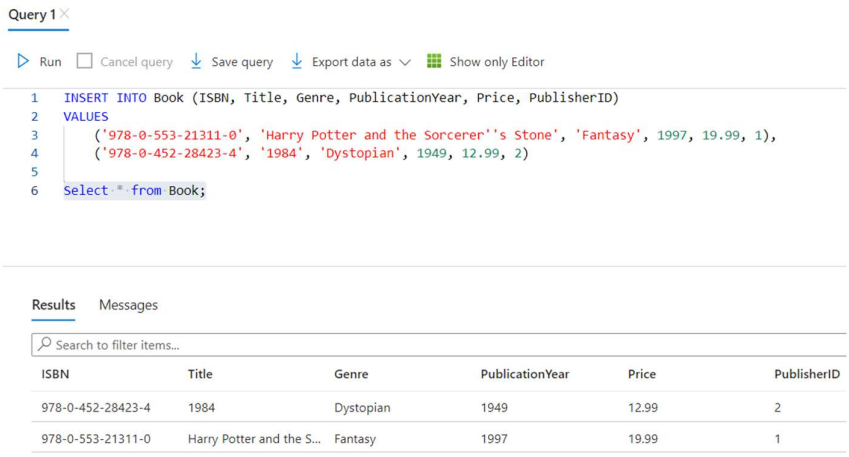


7. Insert data into the Book table by using the below query.

Query: INSERT INTO Book (ISBN, Title, Genre, PublicationYear, Price, PublisherID)
VALUES
('978-0-553-21311-0', 'Harry Potter and the Sorcerer''s Stone', 'Fantasy', 1997, 19.99, 1),
('978-0-452-28423-4', '1984', 'Dystopian', 1949, 12.99, 2)

8. To check the Data, use the below query.

Query: Select * from Book;

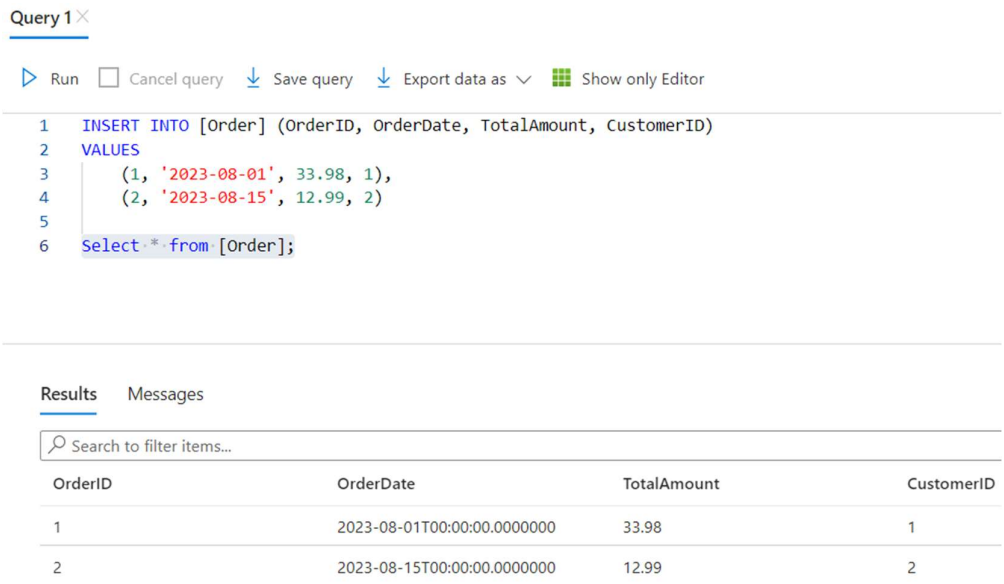


9. Insert data into the Order table by using the below query.

Query: INSERT INTO [Order] (OrderID, OrderDate, TotalAmount, CustomerID)
VALUES
(1, '2023-08-01', 33.98, 1),
(2, '2023-08-15', 12.99, 2)

10. To check the Data, use the below query.

Query: Select * from [Order];

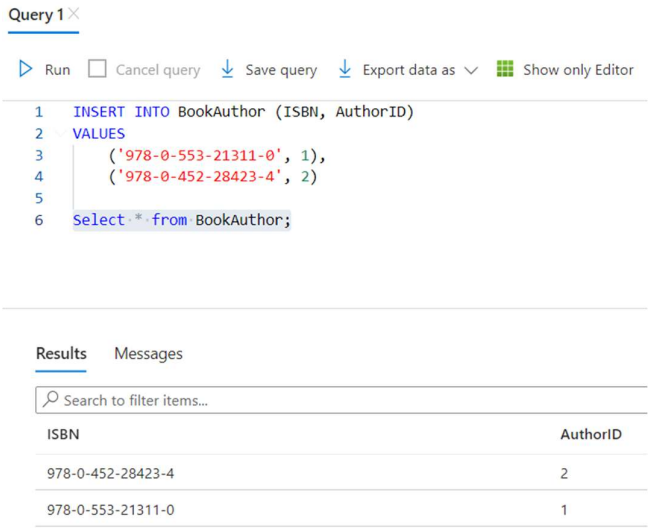


11. Insert data into the BookAuthor table by using the below query.

Query: INSERT INTO BookAuthor (ISBN, AuthorID)
VALUES
('978-0-553-21311-0', 1),
('978-0-452-28423-4', 2)

12. To check the Data, use the below query.

Query: Select * from BookAuthor;

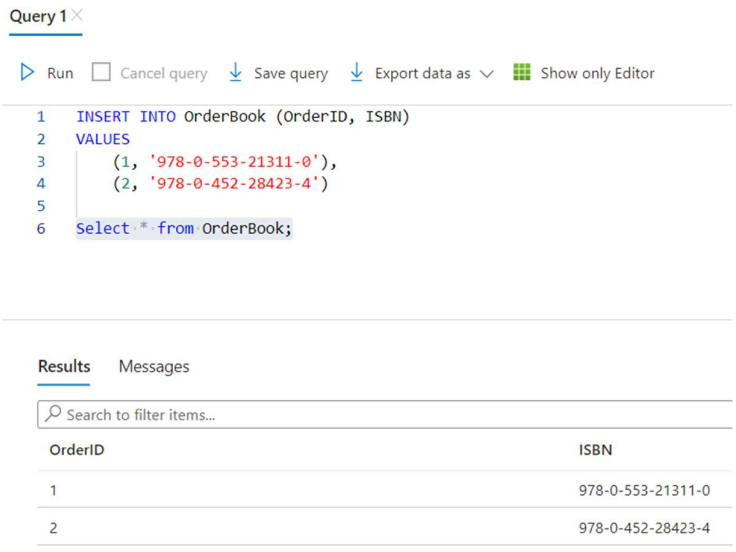


13. Insert data into the OrderBook table by using the below query.

Query: INSERT INTO OrderBook (OrderID, ISBN)
VALUES
(1, '978-0-553-21311-0'),
(2, '978-0-452-28423-4')

14. To check the Data, use the below query.

Query: Select * from OrderBook;



Grant

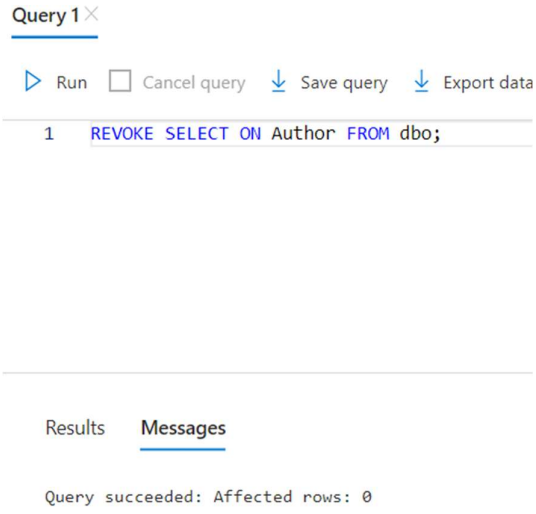
1. Grant SELECT permission on the Author table to a user.

Query: GRANT SELECT ON Author TO dbo;

Revoke

1. Revoke SELECT permission on the Author table from a user.

Query: REVOKE SELECT ON Author FROM dbo;



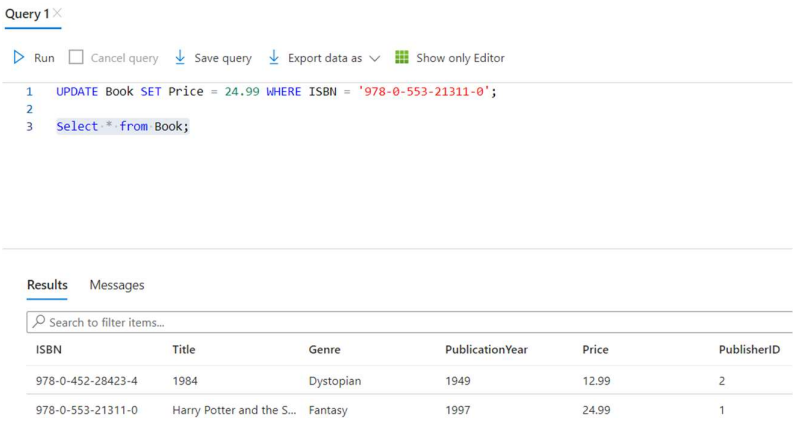
Update

1. The UPDATE operation is used to modify existing records in a table.
2. Update the price of a book with ISBN '978-0-553-21311-0'

Query: UPDATE Book SET Price = 24.99 WHERE ISBN = '978-0-553-21311-0';

3. To check the data use the query below.

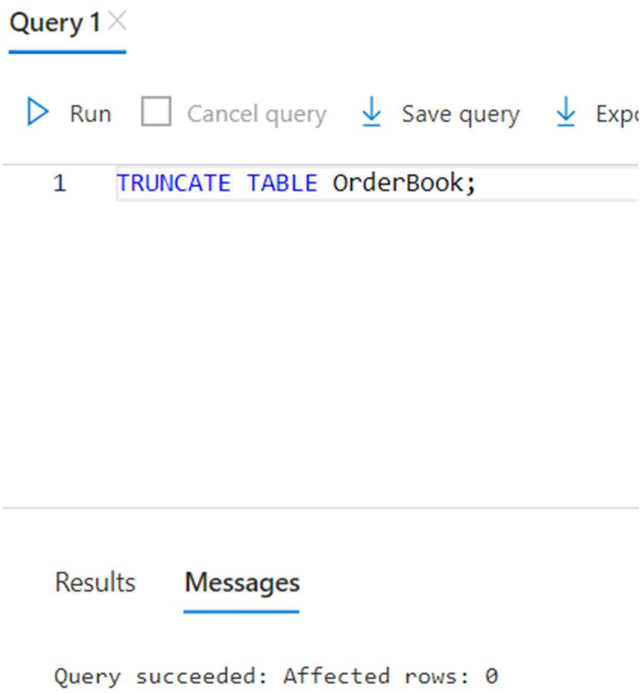
Query: Select * from Book;



Truncate

1. The TRUNCATE operation is used to remove all records from a table while keeping the table structure intact.

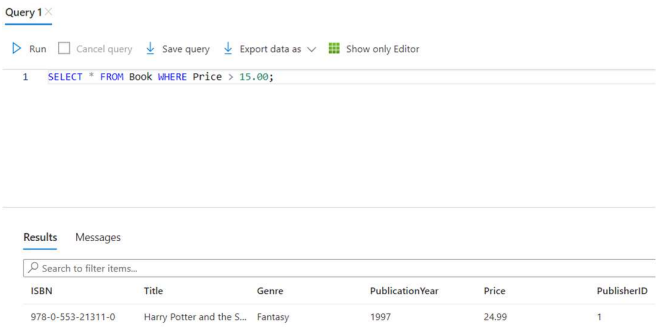
Query: TRUNCATE TABLE OrderBook;



Filtering Data

1. Filtering data involves selecting specific rows based on certain conditions.
2. Retrieve books with a price greater than $15

Query: SELECT * FROM Book WHERE Price > 15.00;



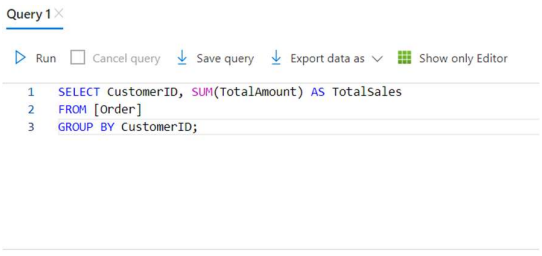
3. Retrieve customers from a specific city.

Query: SELECT * FROM Customer WHERE Address LIKE '%Main St%';

Grouping Data using GROUP BY

1. Grouping data allows you to aggregate data based on specific columns.
2. Get the total sales amount for each customer.

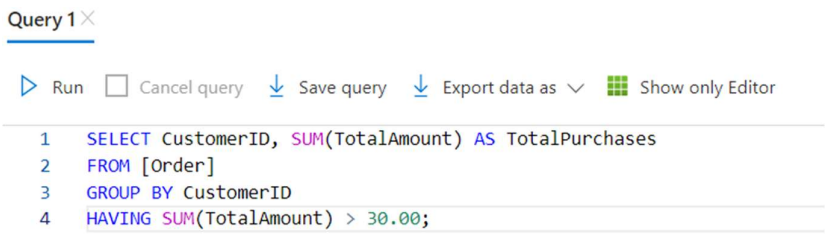
Query: SELECT CustomerID, SUM(TotalAmount) AS TotalSales
FROM [Order]
GROUP BY CustomerID;



Filtering Aggregated Data using HAVING

1. The HAVING clause is used with GROUP BY to filter aggregated data.
2. Get customers who have made total purchases greater than $30.

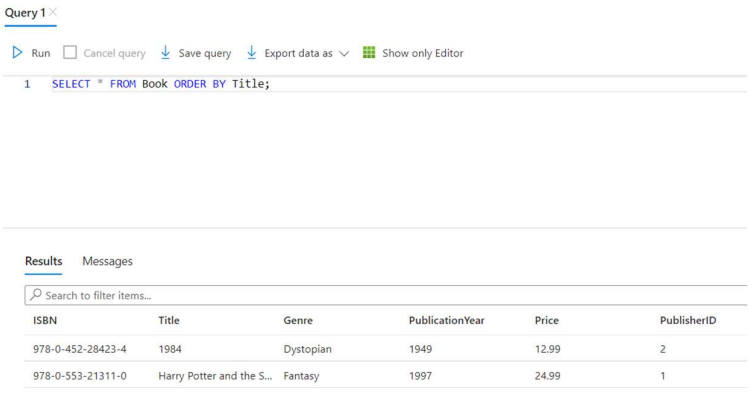
Query: SELECT CustomerID, SUM(TotalAmount) AS TotalPurchases
FROM [Order]
GROUP BY CustomerID
HAVING SUM(TotalAmount) > 30.00;



Sorting Data

1. Sorting data allows you to order the result set based on specific columns.
2. Retrieve books sorted by title in ascending order

Query: SELECT * FROM Book ORDER BY Title;



3. Retrieve customers sorted by last name in descending order.

Query: SELECT * FROM Customer ORDER BY LastName DESC;