Creating a database-

1. Open Visual Studio.
2. Go to Server Explorer. If Server Explorer is not opened go to View/ Server Explorer.
3. Right click on Data Connections and select Create New SQL Server Database.
4. Fill in details like this.

a. Server Name:.(**localdb)\mssqllocaldb**
b. Select Windows Authentication
c. Database Name: ComputerShop
d. Click OK

Database Created

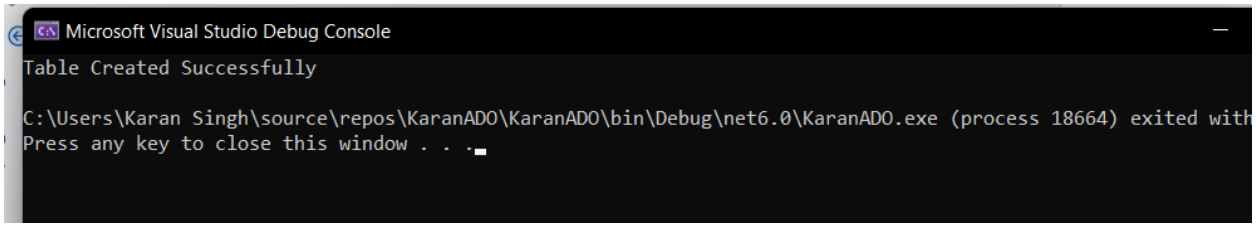
1. Tables Creation

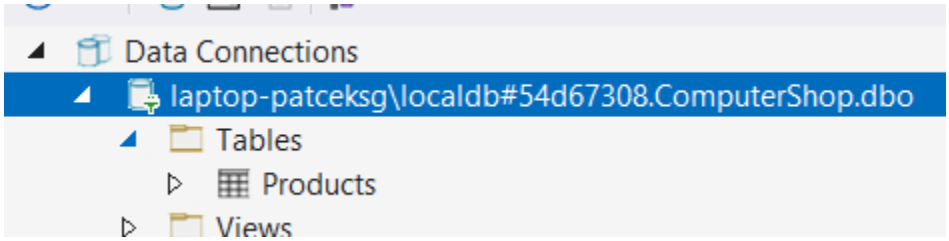
1- Product Table

```
using System;
using System.Data.SqlClient;

namespace CreateTable
{
   class Program
   {
      static void Main(string[] args)
      {
         SqlConnection con = new SqlConnection(@"Data Source=(localdb)\mssqllocaldb;Initial
Catalog=ComputerShop;Integrated Security=True;Pooling=False");
         // Creating a table using C# code
         string query =
         @"CREATE TABLE dbo.Products
            (
               ID int IDENTITY(1,1) NOT NULL,
               Name nvarchar(50) NULL,
               Price nvarchar(50) NULL,
               Date datetime NULL,
               CONSTRAINT pk_id PRIMARY KEY (ID)
            );";
         SqlCommand cmd = new SqlCommand(query, con);
```

```
        try
        {
            con.Open();
            cmd.ExecuteNonQuery();
            Console.WriteLine("Table Created Successfully");
        }
        catch (SqlException e)
        {
            Console.WriteLine("Error Generated. Details: " + e.ToString());
        }
        finally
        {
            con.Close();
            Console.ReadKey();
        }
    }
  }
}
```



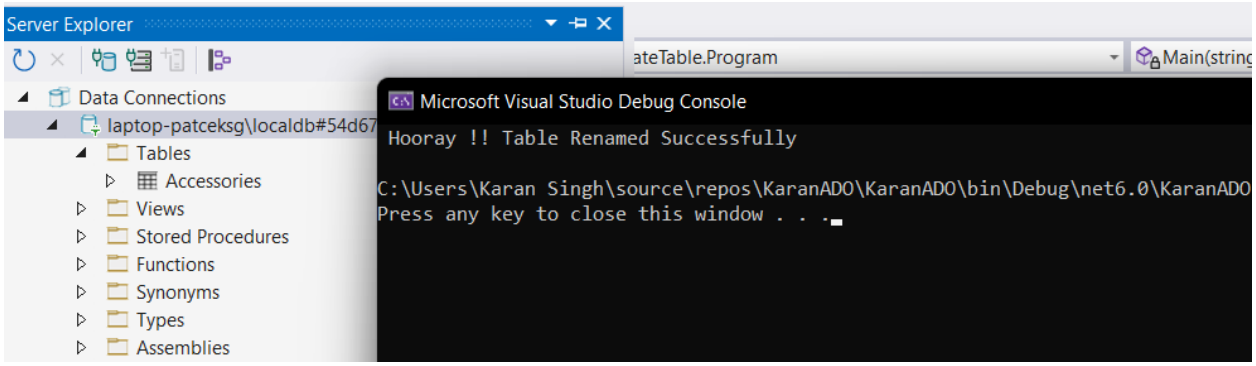


2. RENAME A TABLE USING C# ADO.NET FROM PRODUCTS TO ACCESSORIES

For this change the sql query to

```
using System;
using System.Data.SqlClient;

namespace CreateTable
{
   class Program
   {
      static void Main(string[] args)
      {
         SqlConnection con = new SqlConnection(@"Data Source=(localdb)\mssqllocaldb;Initial
Catalog=ComputerShop;Integrated Security=True;Pooling=False");
         // Creating a table using C# code
         string query = @"EXEC sp_rename 'Products', 'Accessories'";
         SqlCommand cmd = new SqlCommand(query, con);
         try
         {
            con.Open();
            cmd.ExecuteNonQuery();
            Console.WriteLine(" Hooray !! Table Renamed Successfully");
         }
         catch (SqlException e)
         {
            Console.WriteLine("Error Generated. Details: " + e.ToString());
         }
         finally
         {
            con.Close();
            Console.ReadKey();
         }
      }
   }
}
```

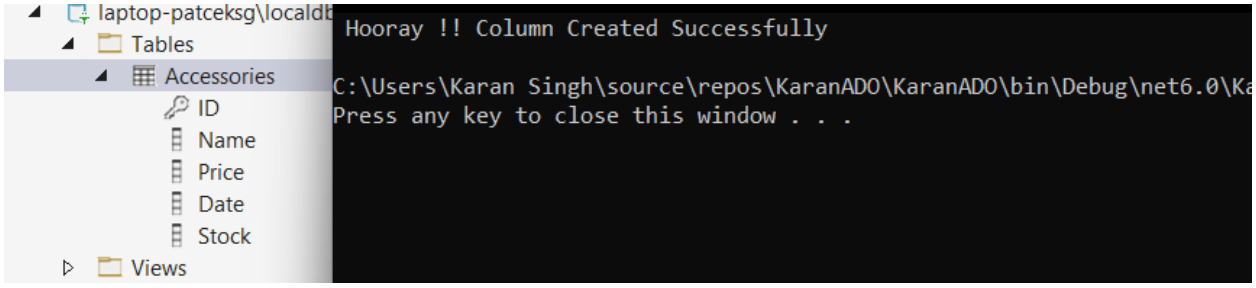

Server Explorer
ateTable.Program
Main(string
Data Connections
laptop-patceksg\localdb#54d67
Tables
Accessories
Views
Stored Procedures
Functions
Synonyms
Types
Assemblies
Microsoft Visual Studio Debug Console
Hooray !! Table Renamed Successfully
C:\Users\Karan Singh\source\repos\KaranADO\KaranADO\bin\Debug\net6.0\KaranADO
Press any key to close this window . . .

3. Updating A TABLE BY ADDING COLUMN TO IT

Can be done using sql or program
Change string query to

```
ALTER TABLE Accessories
ADD Stock nvarchar(50);
```

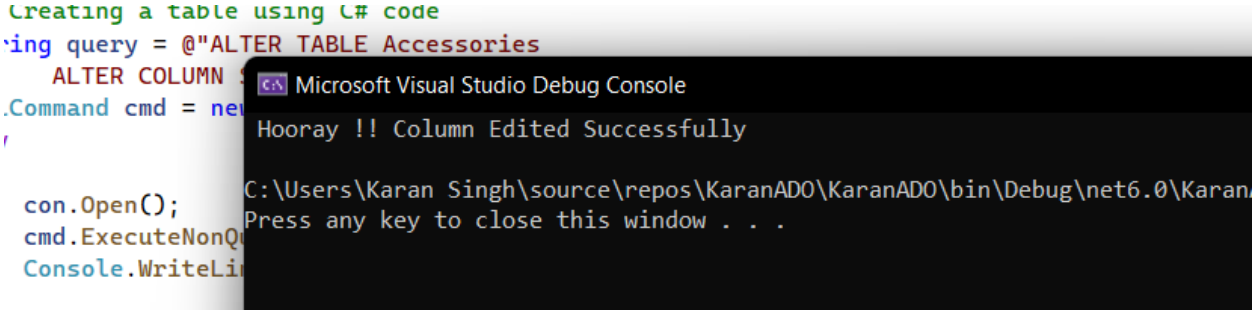
On adding column Stock



Now changing column Stock nvarchar(50) to Stock int.

ALTER TABLE Accessories
ALTER COLUMN Stock int;

Changing sql query in Program.cs file

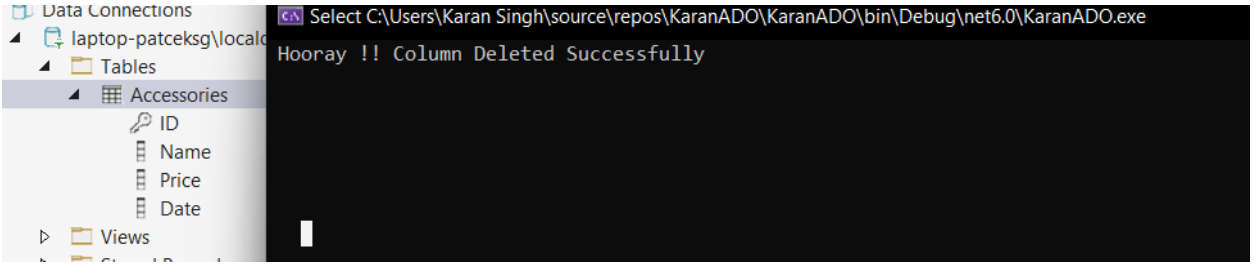


4. Now Deleting a column here

Using SQL query

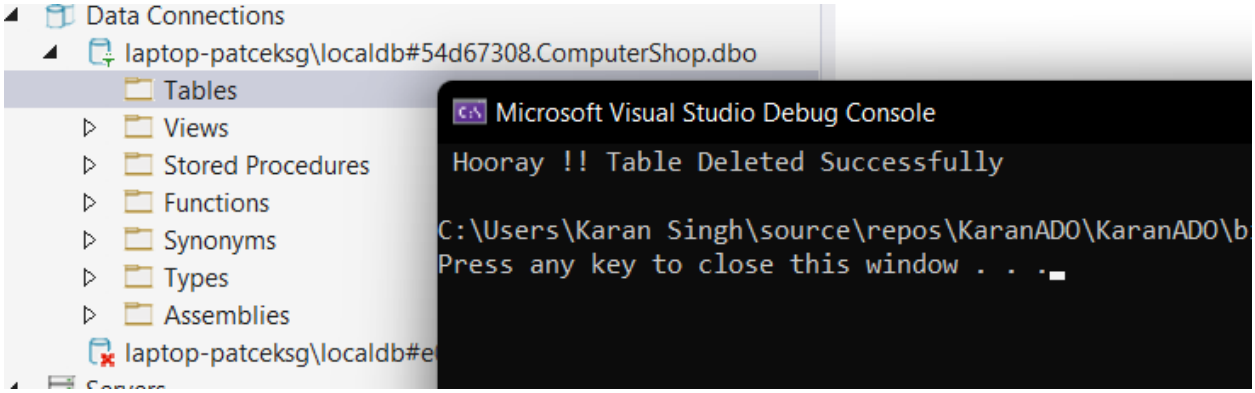
ALTER TABLE Accessories
DROP COLUMN Stock;

Changing sql query in Program.cs file

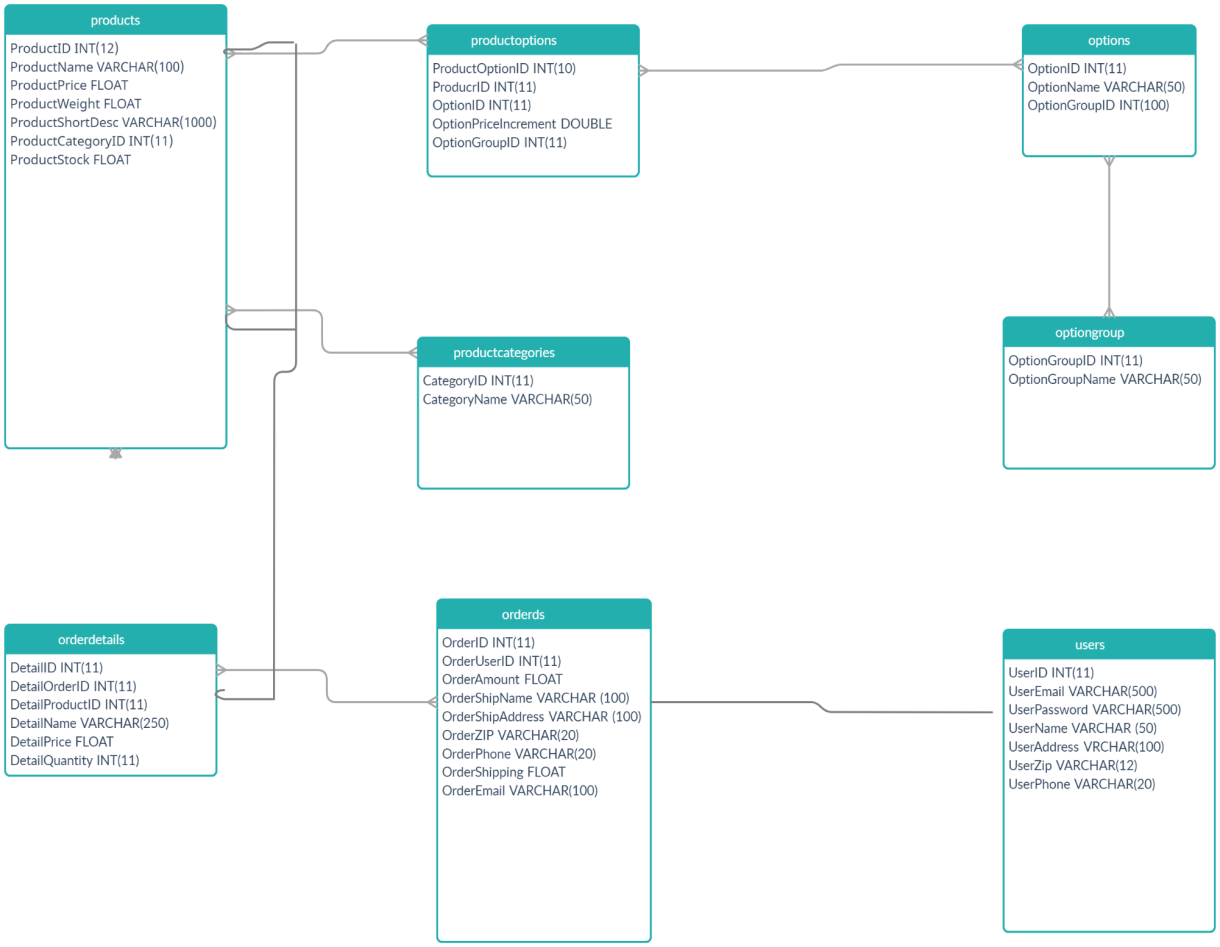


5. Deleting a table

DROP TABLE Accessories



Now again creating the table for Ecommerce operations



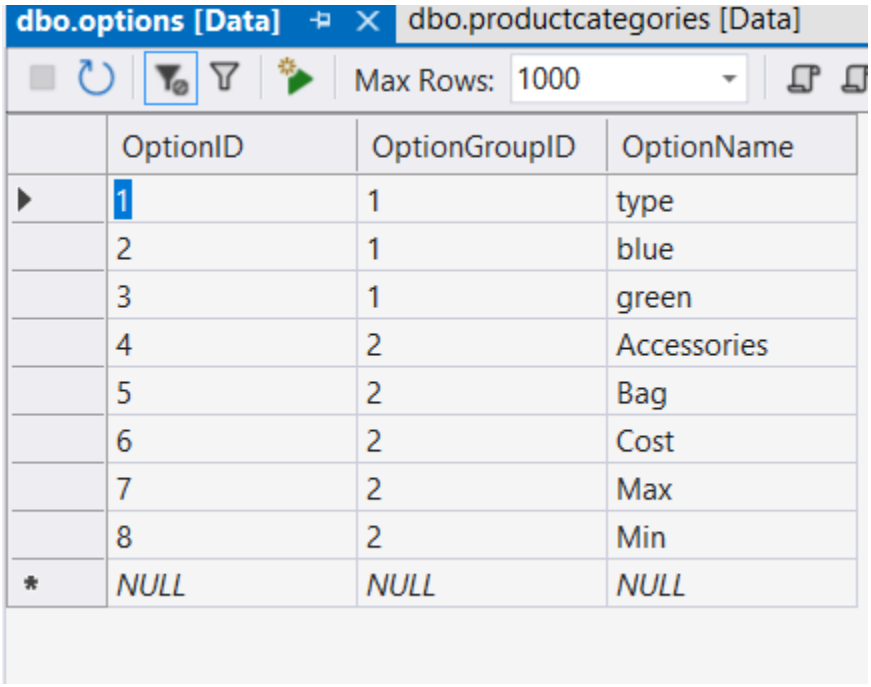

```
CREATE TABLE dbo.optiongroups
(
   OptionGroupID int(11) NOT NULL,
   OptionGroupName nvarchar(50) NULL,
 CONSTRAINT pk_id PRIMARY KEY (OptionGroupID)
 );
```

```
CREATE TABLE dbo.options  (
  OptionID int NOT NULL,
  OptionGroupID int DEFAULT NULL,
  OptionName varchar(50) DEFAULT NULL,
  PRIMARY KEY (OptionID)
)
```

Adding data in options table

Changing sql query in Program to:

```
INSERT INTO options (OptionID, OptionGroupID, OptionName) VALUES
(1, 1, 'type'),
(2, 1, 'blue'),
(3, 1, 'green'),
(4, 2, 'Accessories'),
(5, 2, 'Bag'),
(6, 2, 'Cost'),
(7, 2, 'Max'),
(8, 2, 'Min');
```

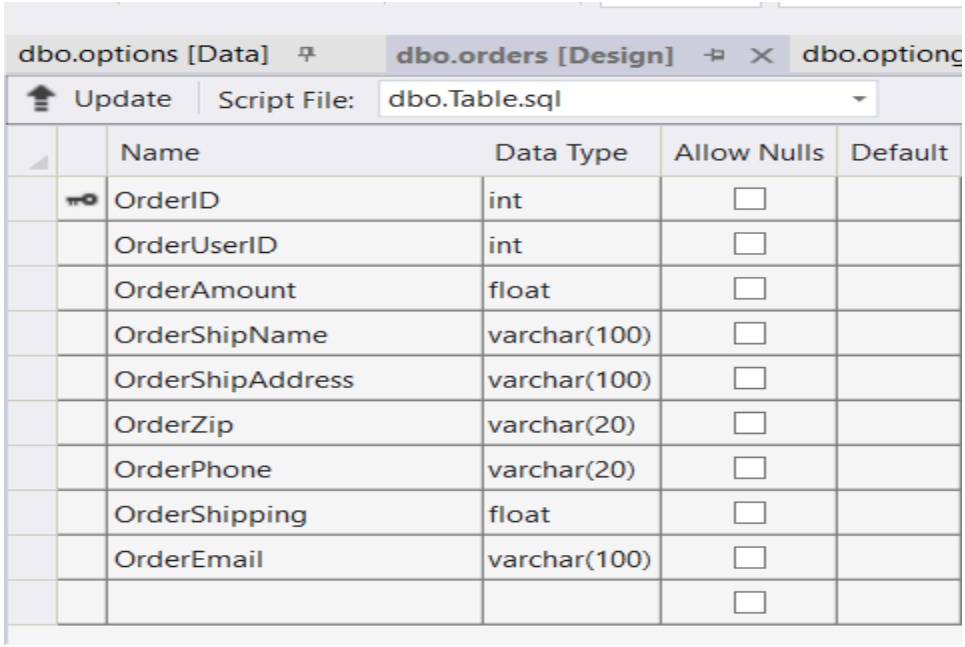


	OptionID	OptionGroupID	OptionName
▶	1	1	type
	2	1	blue
	3	1	green
	4	2	Accessories
	5	2	Bag
	6	2	Cost
	7	2	Max
	8	2	Min
*	*NULL*	*NULL*	*NULL*

```
CREATE TABLE dbo.orderdetails  (
```

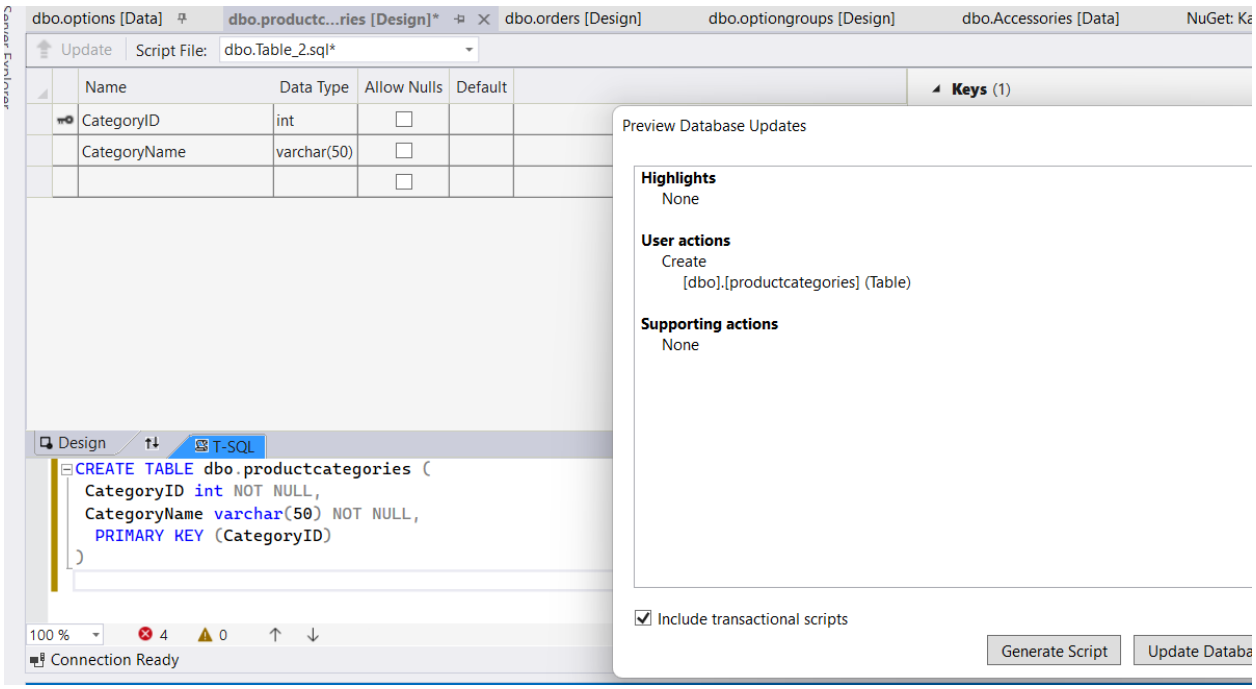
```
DetailID int NOT NULL,
  DetailOrderID int NOT NULL,
  DetailProductID int NOT NULL,
  DetailName varchar(250) NOT NULL,
  DetailPrice float NOT NULL,
 DetailQuantity int NOT NULL,
  PRIMARY KEY (DetailID)
)
```

```
CREATE TABLE dbo.orders (
OrderID int NOT NULL,
  OrderUserID int NOT NULL,
  OrderAmount float NOT NULL,
  OrderShipName varchar(100) NOT NULL,
  OrderShipAddress varchar(100) NOT NULL,
  OrderZip varchar(20) NOT NULL,
  OrderPhone varchar(20) NOT NULL,
  OrderShipping float NOT NULL,
   OrderEmail varchar(100) NOT NULL,
  PRIMARY KEY (OrderID)
)
```

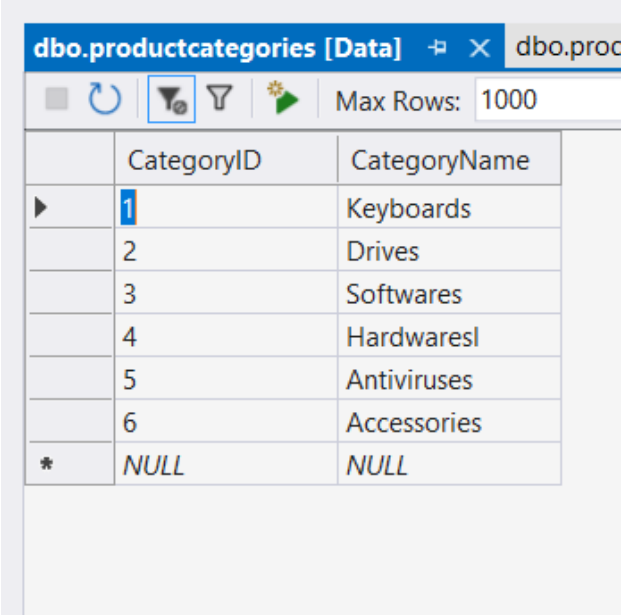


	Name	Data Type	Allow Nulls	Default
🔑	OrderID	int	☐	
	OrderUserID	int	☐	
	OrderAmount	float	☐	
	OrderShipName	varchar(100)	☐	
	OrderShipAddress	varchar(100)	☐	
	OrderZip	varchar(20)	☐	
	OrderPhone	varchar(20)	☐	
	OrderShipping	float	☐	
	OrderEmail	varchar(100)	☐	
			☐	

```
CREATE TABLE dbo.productcategories (
 CategoryID int NOT NULL,
 CategoryName varchar(50) NOT NULL,
  PRIMARY KEY (CategoryID)
)
```



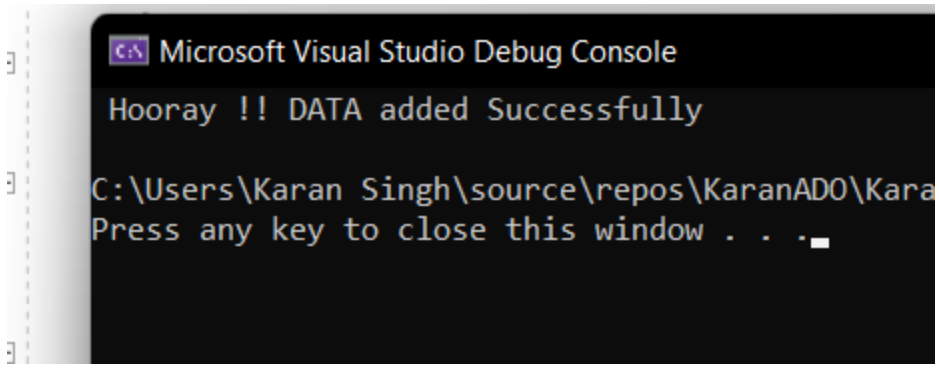

```
INSERT INTO productcategories (CategoryID, CategoryName) VALUES
(1, ‘Keyboards’),
(2, ‘Drives’),
(3, 'Softwares'),
(4, 'Hardwaresl'),
(5, 'Antiviruses'),
(6, 'Accessories');
```




```
CREATE TABLE dbo.productoptions (
  ProductOptionID  int NOT NULL,
  ProductID int NOT NULL,
  OptionID int NOT NULL,
  OptionPriceIncrement double DEFAULT NULL,
  OptionGroupID int NOT NULL,
  PRIMARY KEY (ProductOptionID)
)
```

INSERTING USING PROGRAM.CS FILE

```
INSERT INTO productoptions (ProductOptionID, ProductID, OptionID, OptionPriceIncrement,
OptionGroupID) VALUES
(1, 1, 1, 0, 1),
(2, 1, 2, 0, 1),
(3, 1, 3, 0, 1),
(4, 1, 4, 0, 2),
(5, 1, 5, 0, 2),
(6, 1, 6, 0, 2),
(7, 1, 7, 2, 2),
(8, 1, 8, 2, 2);
```




```
CREATE TABLE dbo.users (
  UserID int NOT NULL,
  UserEmail varchar(500) DEFAULT NULL,
  UserPassword varchar(500) DEFAULT NULL,
  UserName varchar(50) DEFAULT NULL,
  UserAddress varchar(90) DEFAULT NULL,
  UserZip varchar(12)  DEFAULT NULL,
   UserPhone varchar(20) DEFAULT NULL,
  PRIMARY KEY (UserID)
)
```

```
CREATE TABLE dbo.products (
  ProductID int NOT NULL,
  ProductName varchar(100) NOT NULL,
  ProductPrice float NOT NULL,
  ProductWeight float NOT NULL,
 ProductShortDesc varchar(1000) NOT NULL,
 ProductCategoryID int DEFAULT NULL,
  ProductStock float DEFAULT NULL,
   PRIMARY KEY (ProductID)
)
```

```
INSERT INTO products (ProductID,ProductName, ProductPrice, ProductWeight,
ProductShortDesc,ProductCategoryID, ProductStock) VALUES
(1, 'Keyboardt', 9.99, 3, 'A black colour keyboard.', 5, 100),
```

(2, 'Mouse', 5.99, 2, 'A brown colour mouse', 8, 100),
(3, 'Antivirus', 6.99, 1, 'Avast Antivirus at 1999', 5, 100),
(4, 'Laptop Bag', 7.99, 5, 'A green bag made with 100% real cotton.', 9, 100),
(5, 'Hard Drive', 3.99, 3, '500 gb hard drive WD', 4, 100),
(6, 'Pen Drive', 1.99, 1, 'Sandisk 32 gb pendrive', 5, 100),
(7, 'Data Cable', 1.09,1, 'white colour cable.', 6, 100);

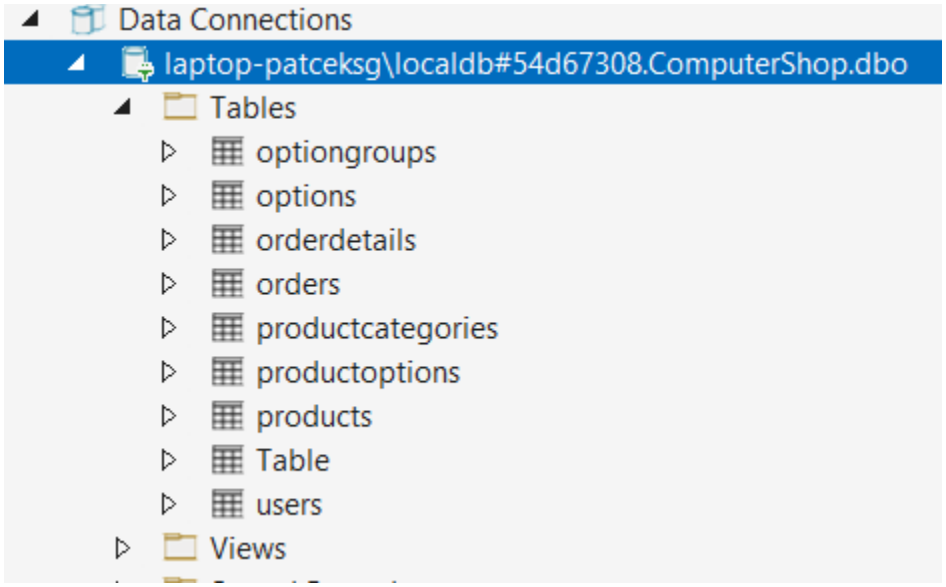
	ProductID	ProductName	ProductPrice	ProductWeight	ProductShortD...	ProductCatego...	ProductStock
▶	1	Keyboardt	9.99	3	A black colour k...	5	100
	2	Mouse	5.99	2	A brown colour ...	8	100
	3	Antivirus	6.99	1	Avast Antivirus ...	5	100
	4	Laptop Bag	7.99	5	A green bag m...	9	100
	5	Hard Drive	3.99	3	500 gb hard dri...	4	100
	6	Pen Drive	1.99	1	Sandisk 32 gb p...	5	100
	7	Data Cable	1.09	1	white colour ca...	6	100
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Deleting data from productcategories

```
// Creating a table using C# code
string query = @"DELETE FROM productcategories;

SqlCommand cmd = new SqlCommand(query, con);
```

Tables



READING DATA AND OUTPUT ON CONSOLE

1- PRODUCT CATEGORIES

```
using System;
using System.Data.SqlClient;

namespace CreateTable
{
   class Program
   {
      static void Main(string[] args)
      {
         SqlConnection con = new SqlConnection(@"Data Source=(localdb)\mssqllocaldb;Initial
Catalog=ComputerShop;Integrated Security=True;Pooling=False");
         // Creating a table using C# code
         string query = @"SELECT * FROM productcategories;
```

```
";
        SqlCommand cmd = new SqlCommand(query, con);
        try
        {
            con.Open();
            SqlCommand cmdR = new SqlCommand(query, con);
            SqlDataReader reader = cmdR.ExecuteReader();
            while (reader.Read())
            {
                Console.WriteLine(reader[0].ToString() + " " + reader[1].ToString());
            }

            Console.WriteLine(" Hooray !! Task Done Successfully");
        }
        catch (SqlException e)
        {
            Console.WriteLine("Error Generated. Details: " + e.ToString());
        }
        finally
        {
            con.Close();
            Console.ReadKey();
        }
    }
  }
}
```

```
// Creating a table using C# code
string query = @"SELECT * FROM productcategories;


SqlCommand cmd = new SqlCommand(query, con);
try
{
    con.Open();
    SqlCommand cmdR = new SqlCommand(query, con);
    SqlDataReader reader = cmdR.ExecuteReader();
    while (reader.Read())
    {
        Console.WriteLine(reader[0].ToString() + " "
    }

    Console.WriteLine(" Hooray !! Task Done Success
```

```
Microsoft Visual Studio Debug Console
1 Keyboards
2 Drives
3 Softwares
4 Hardwaresl
5 Antiviruses
6 Accessories
 Hooray !! Task Done Successfully

C:\Users\Karan Singh\source\repos\KaranADO\KaranADO\b
Press any key to close this window . . .
```

2- OPTIONS

```
atic void Main(string[] args)

   SqlConnection con = new SqlConnection(@"Da
   // Creating a table using C# code
   string query = @"SELECT * FROM options;


   SqlCommand cmd = new SqlCommand(query, con
   try
   {
       con.Open();
       SqlCommand cmdR = new SqlCommand(query
       SqlDataReader reader = cmdR.ExecuteRea
       while (reader.Read())
       {
           Console.WriteLine(reader[0].ToStri
       }
```

```
Select Microsoft Visual Studio Debug Console
1 1 type
2 1 blue
3 1 green
4 2 Accessories
5 2 Bag
6 2 Cost
7 2 Max
8 2 Min
 Hooray !! Task Done Successfully

C:\Users\Karan Singh\source\repos\KaranADO\KaranA
Press any key to close this window . . .
```

3- Products

```
Microsoft Visual Studio Debug Console
1 Keyboardt 9.99 3 A black colour keyboard. 5 100
2 Mouse 5.99 2 A brown colour mouse 8 100
3 Antivirus 6.99 1 Avast Antivirus at 1999 5 100
4 Laptop Bag 7.99 5 A green bag made with 100% real cotton. 9 100
5 Hard Drive 3.99 3 500 gb hard drive WD 4 100
6 Pen Drive 1.99 1 Sandisk 32 gb pendrive 5 100
7 Data Cable 1.09 1 white colour cable. 6 100
 Hooray !! Task Done Successfully

C:\Users\Karan Singh\source\repos\KaranADO\KaranADO\bin\Debug\net6.0\Ka
Press any key to close this window . . .
```

```
// code change
Console.WriteLine(reader[0].ToString() + " " + reader[1].ToString() + " "+
                  reader[2].ToString() + " " + reader[3].ToString() + " " + reader[4].ToString() + " "
                  + reader[5].ToString() + " " + reader[6].ToString());
```

4- Productoptions