# BCIT

**Comp 3951 Topics in Application Development**

**Technical Programming Option**

# Option Head Mirela Gutica

**Winter 2022**

Mark: \_\_\_\_\_\_\_\_ /100

Lab 8

**Introduction to ADO.NET**

**Requirements**:

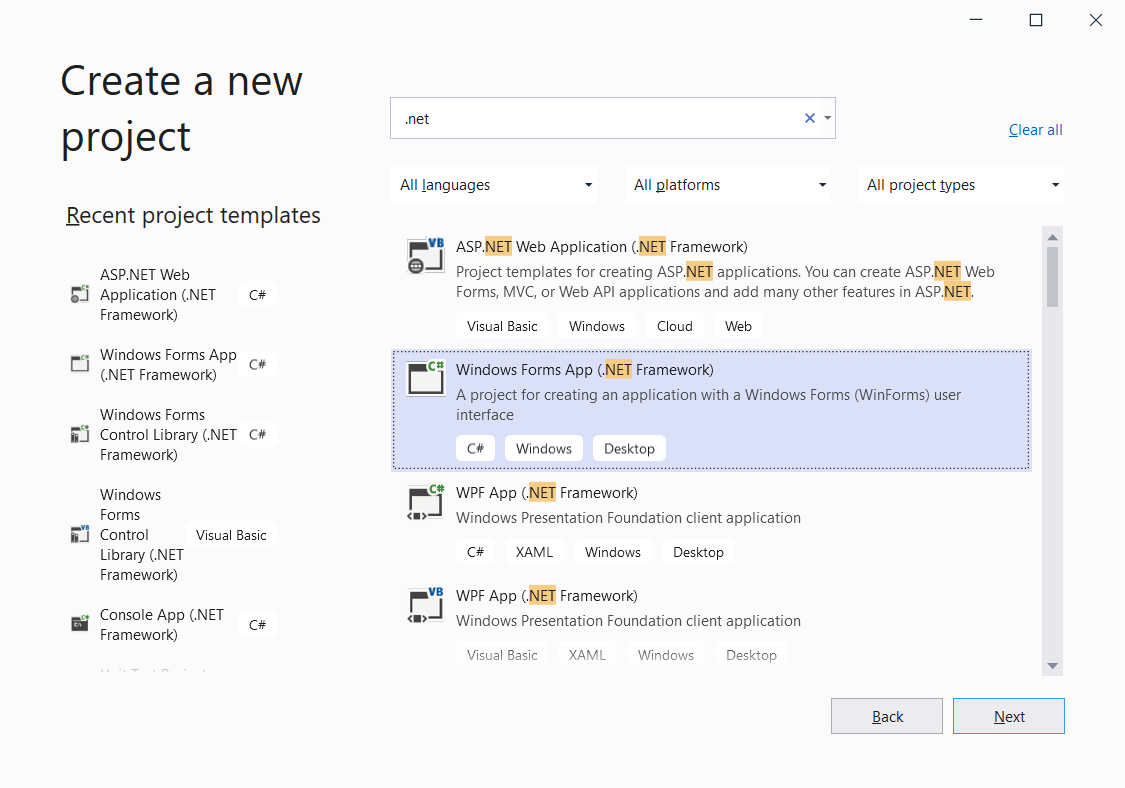
Part 1: Read Chapter 19 from your textbook.

Follow the instructions and complete the application described below.

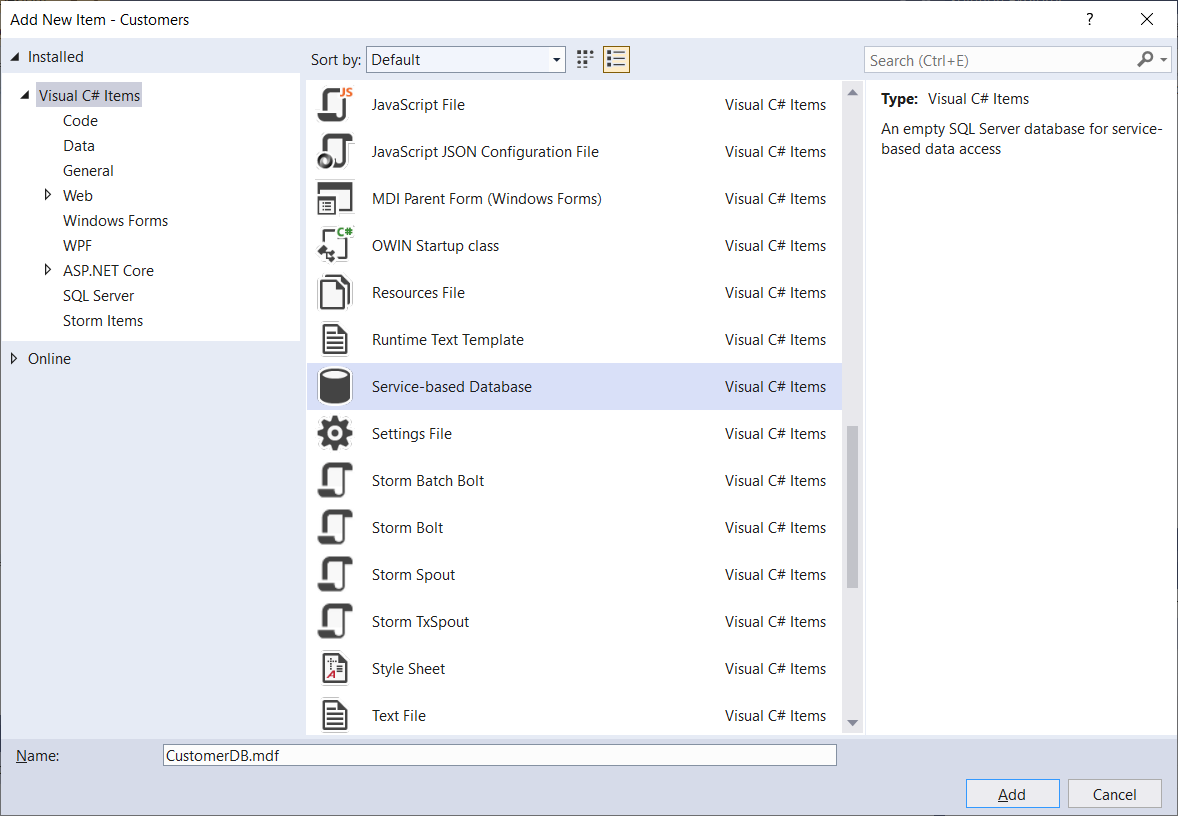
Part 2 Bonus (20%): Add to the application described below three queries to select and display, add, update data from database (see pages 628 – 631).

Part 1:

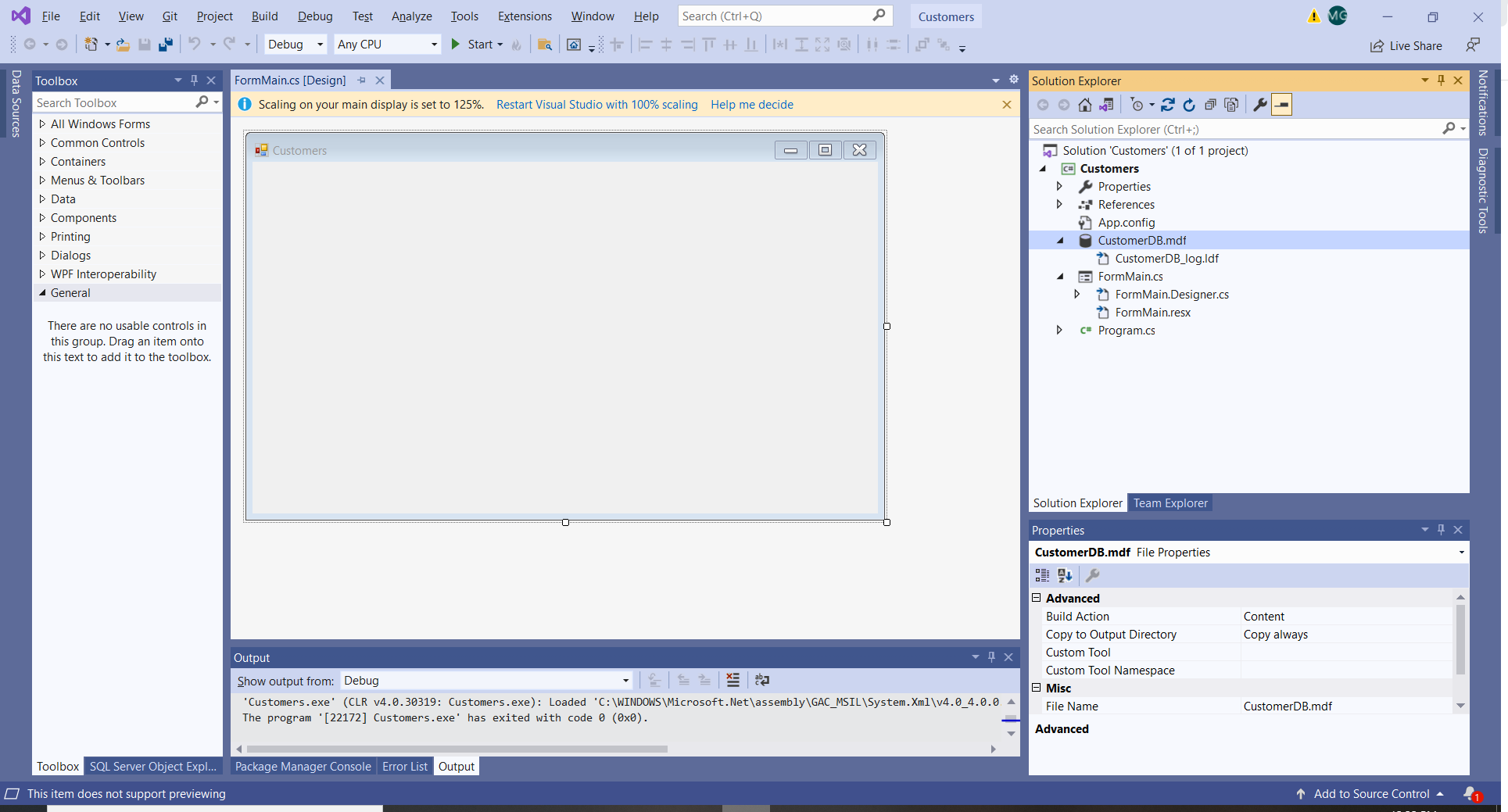
1. You will design and implement a simple application with a local DB.
2. You will create a local DB containing two tables and will retrieves data from the DB to populate a DataGridView control in a Windows form.
3. Create a .NET Windows Application:



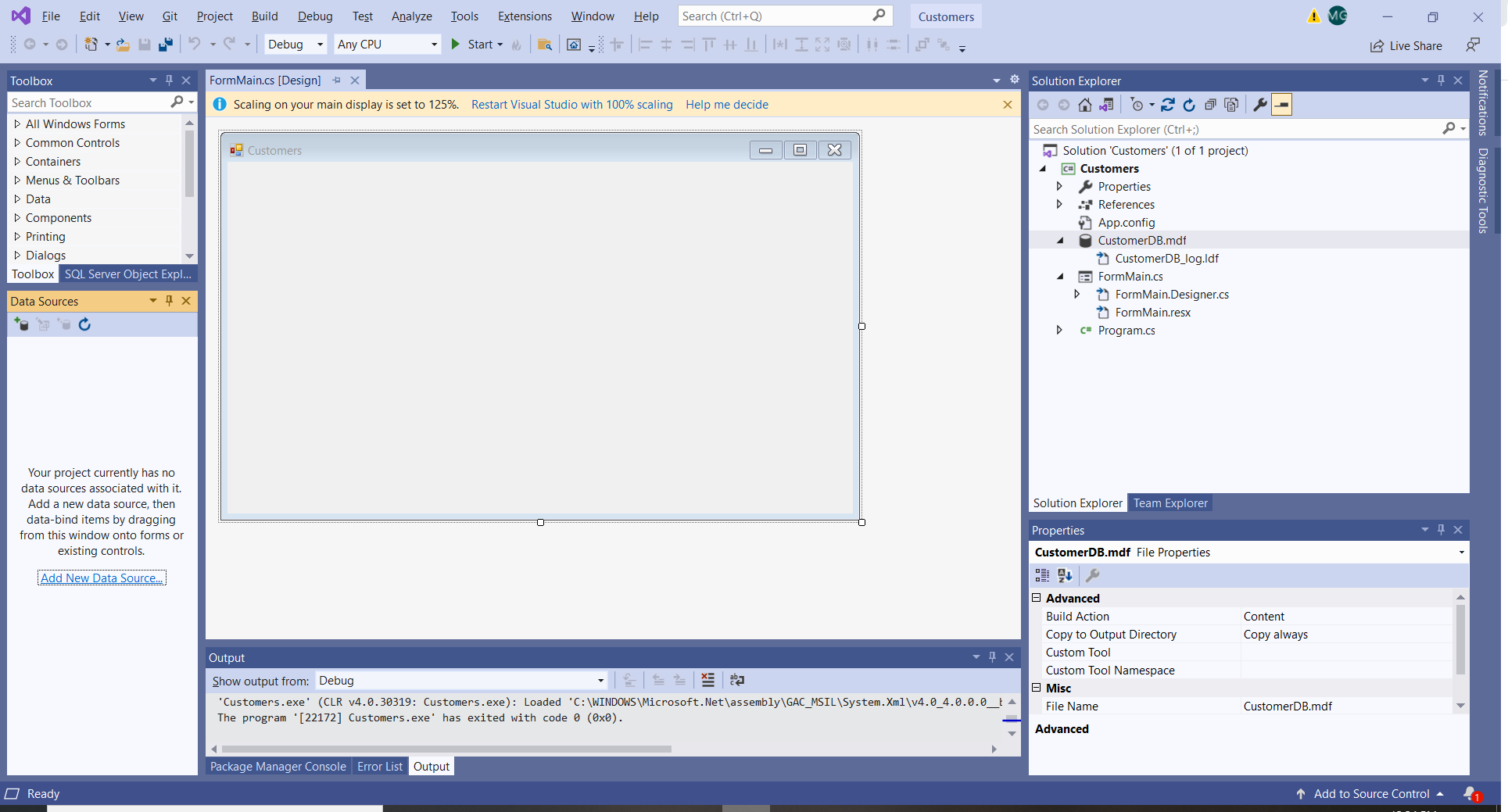
1. Add to the project (Project > Add New Item) a Service-based Database:



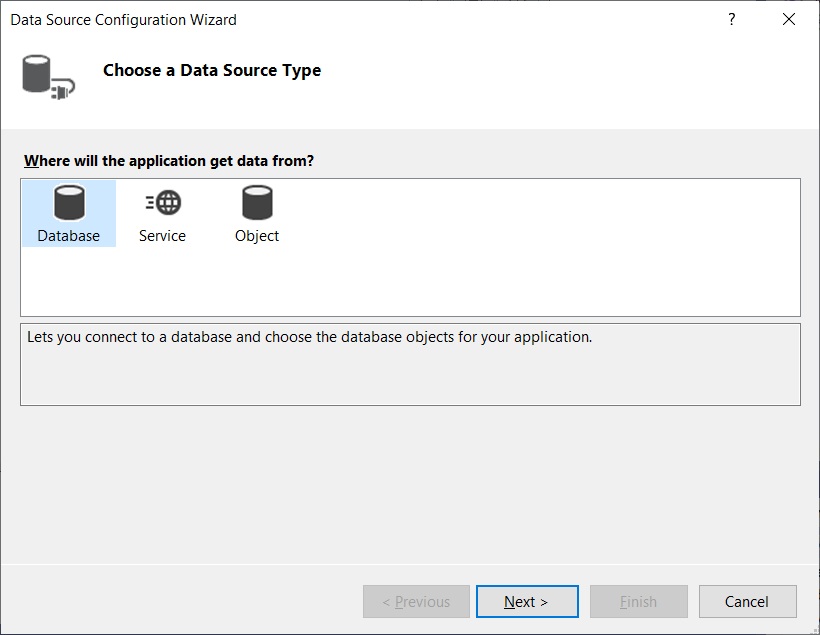
1. The DB is created part of your solution:

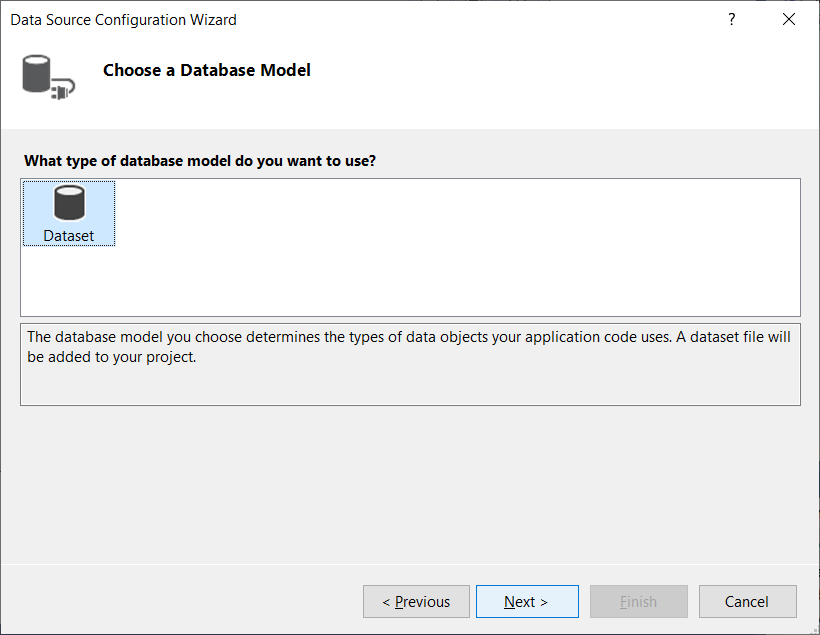


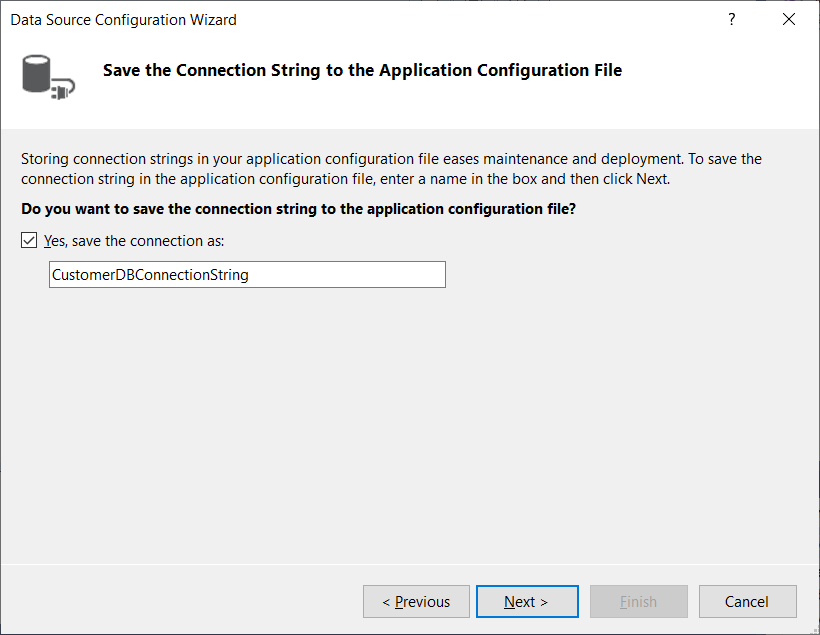
1. Add a Data Source:
   1. Open the Data Source window using Shift-Alt-D (or from the View menu)
   2. Select to Add a New Data Source

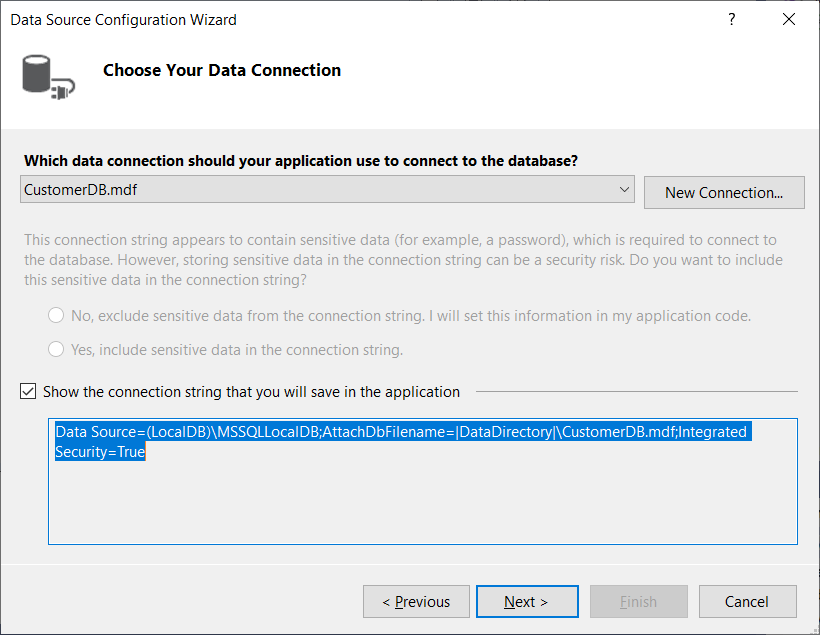


* 1. Follow the wizard (Data Source Configuration Wizard):

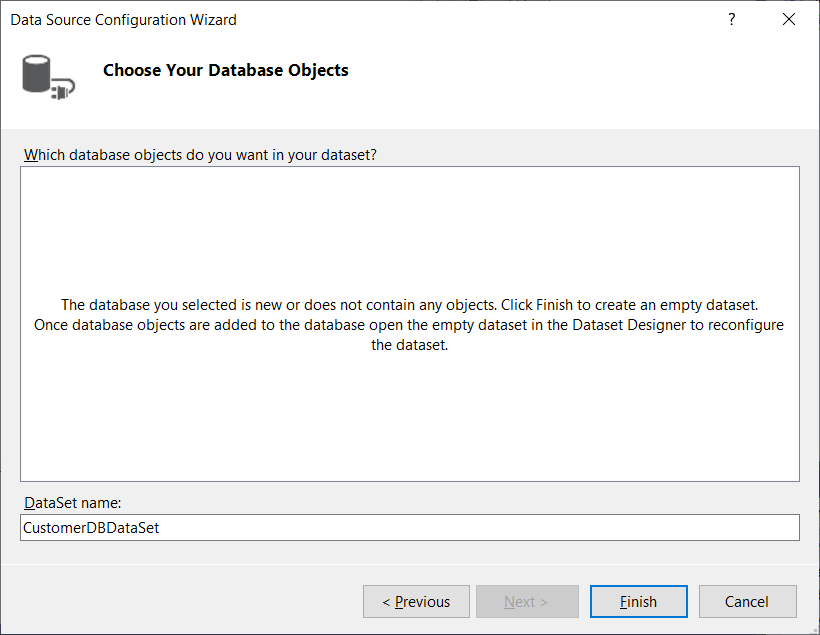




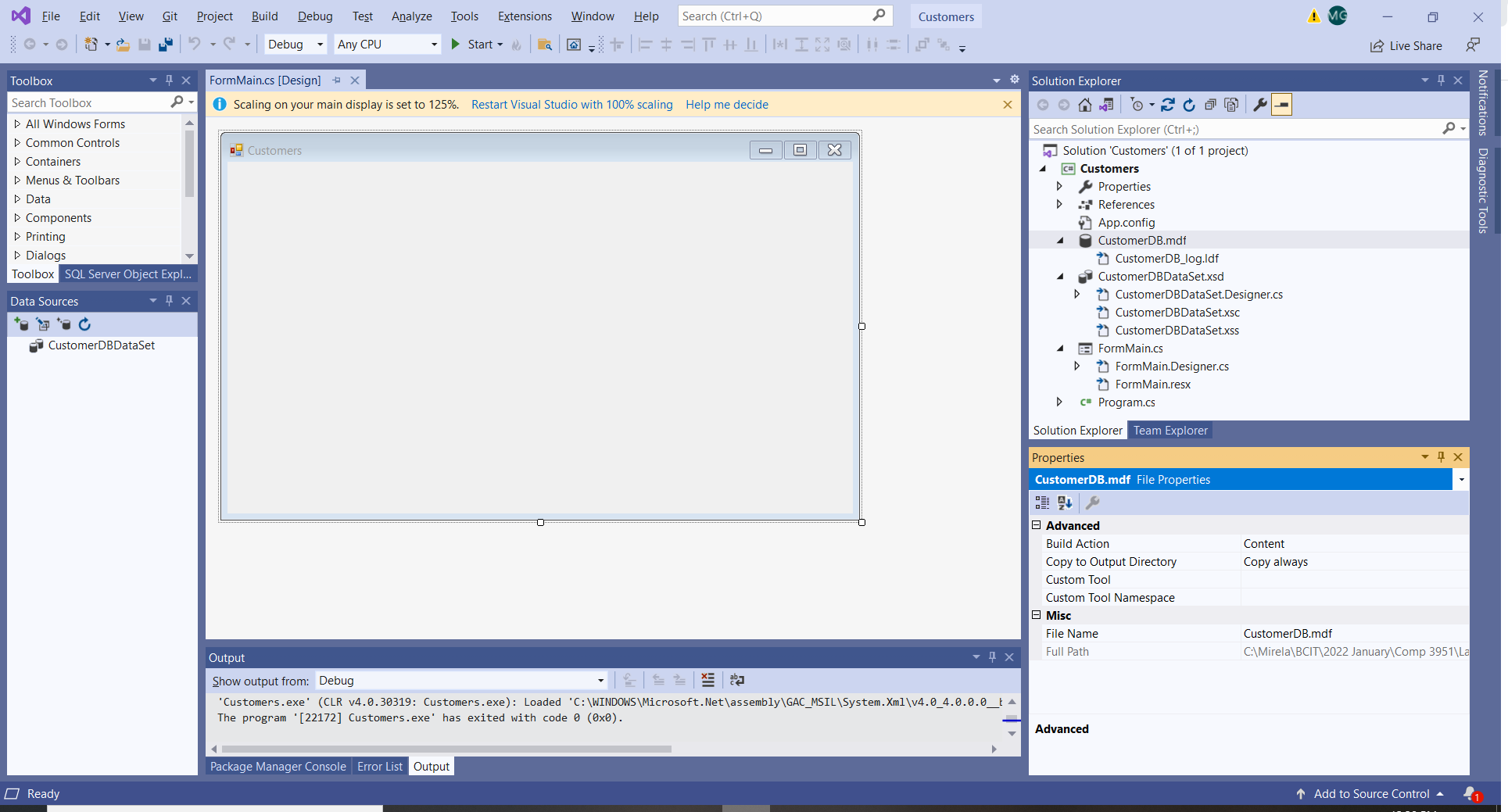




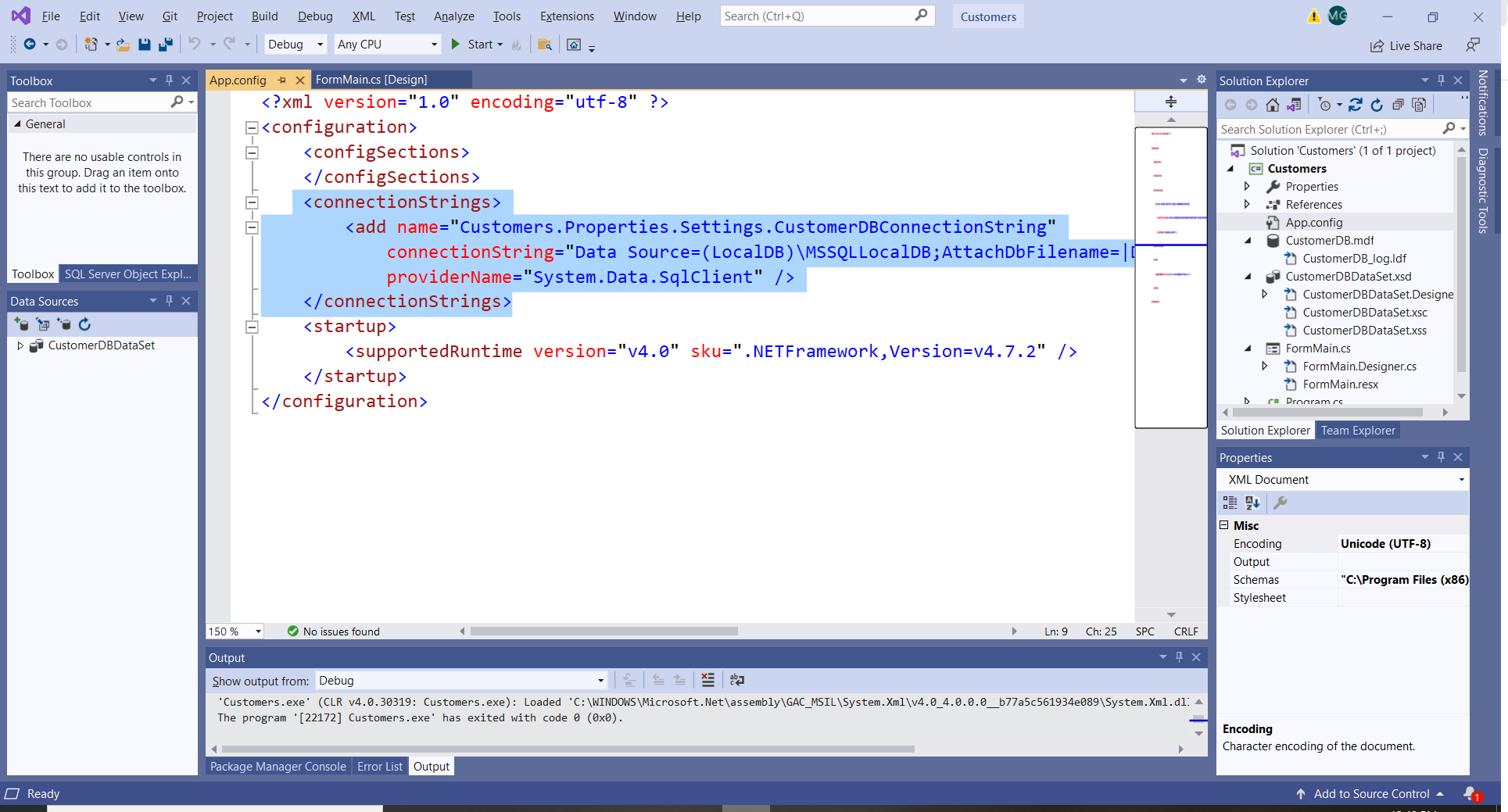
* 1. On the Choose your Database Objects notice that the DB is empty (as expected). Select Finish. (If this would be an existing DB, it would show the DB objects and you would make a selection):



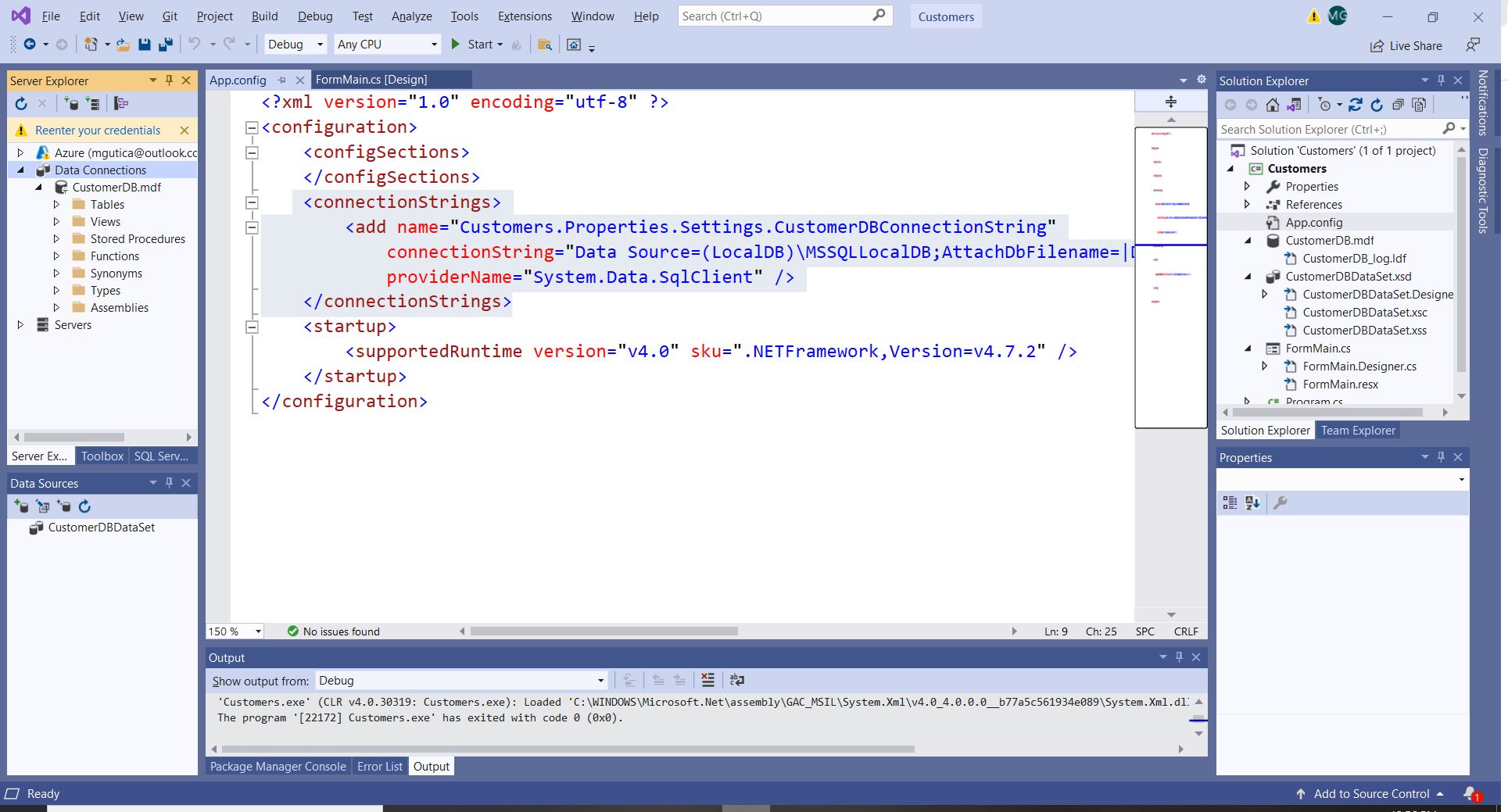
1. View the properties of the data connection. Right click on the DB file (the .mdf file) and select Properties:



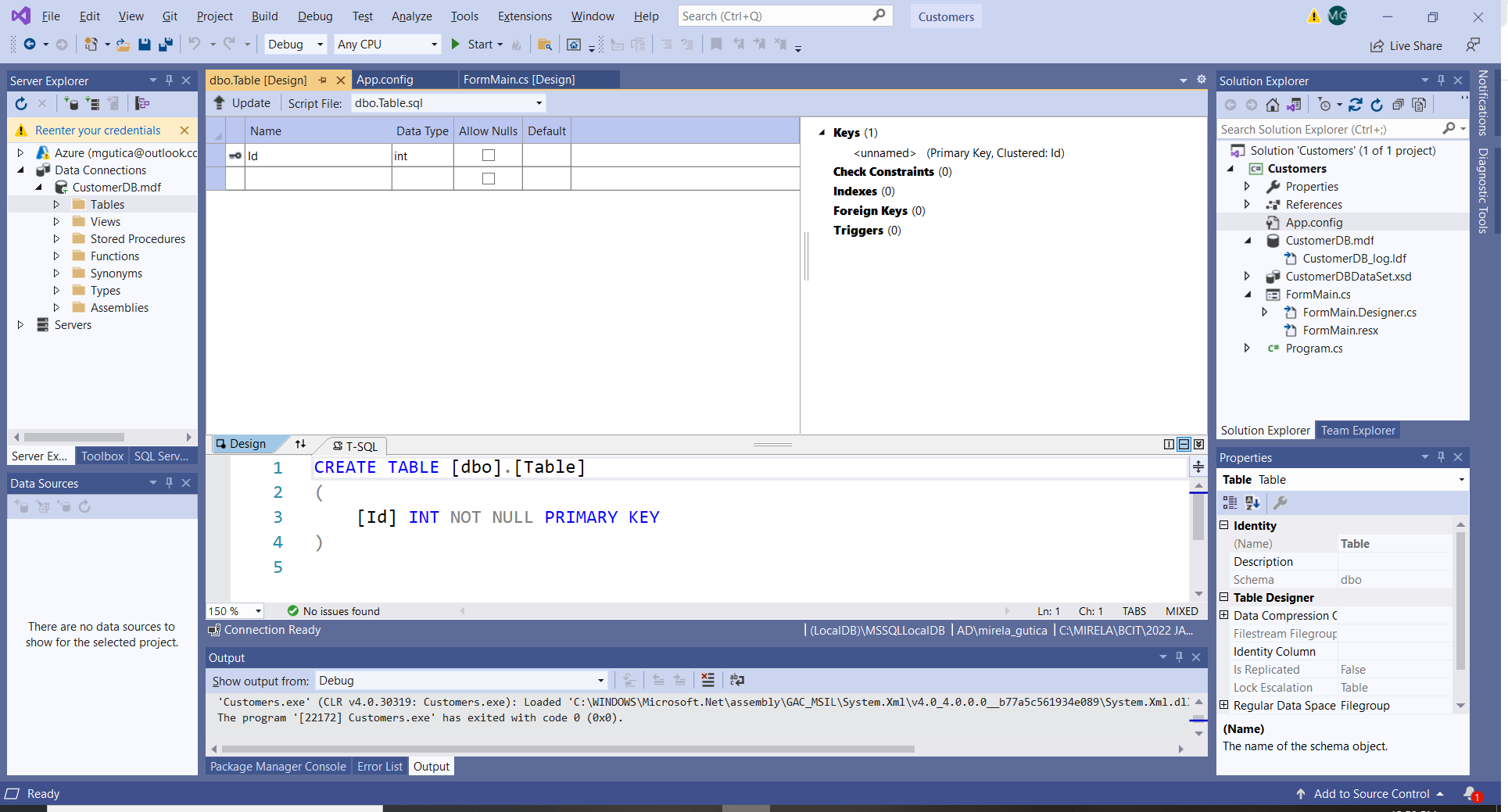
1. The connection string is recorded automatically in the file App.config. Observe the string:

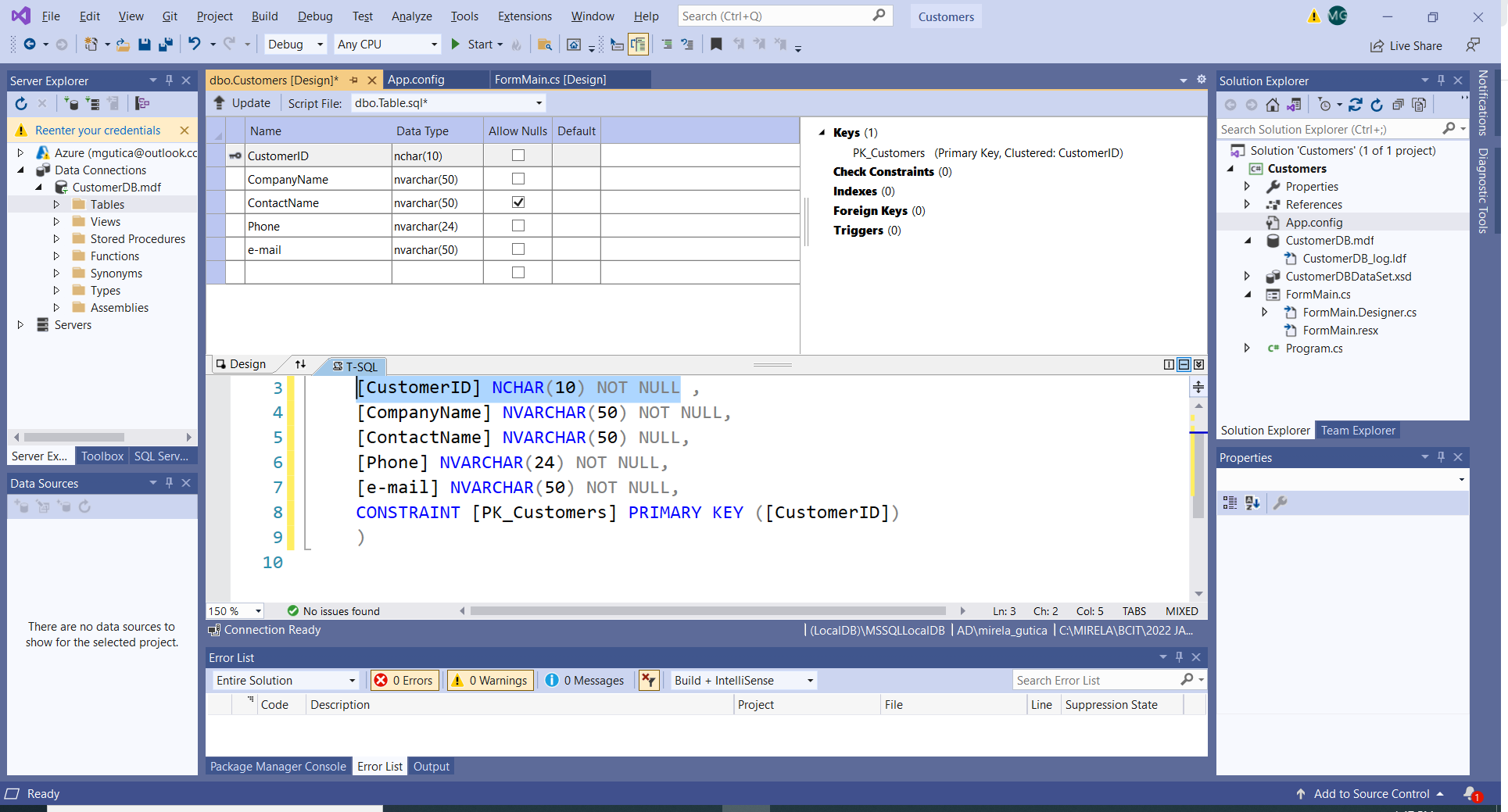


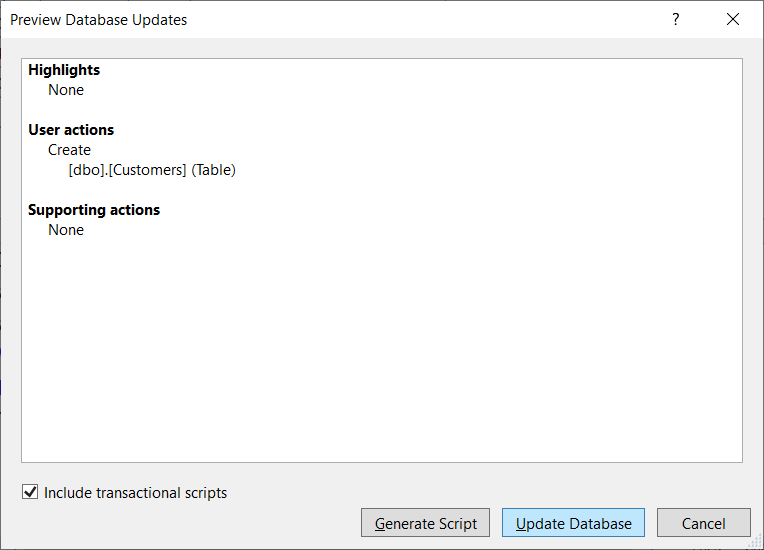
1. Next, you will create tables and keys by using the Table Designer
2. Create the Customers table
   1. On the View menu, open the Server Explorer
   2. Expand the Data Connections node:



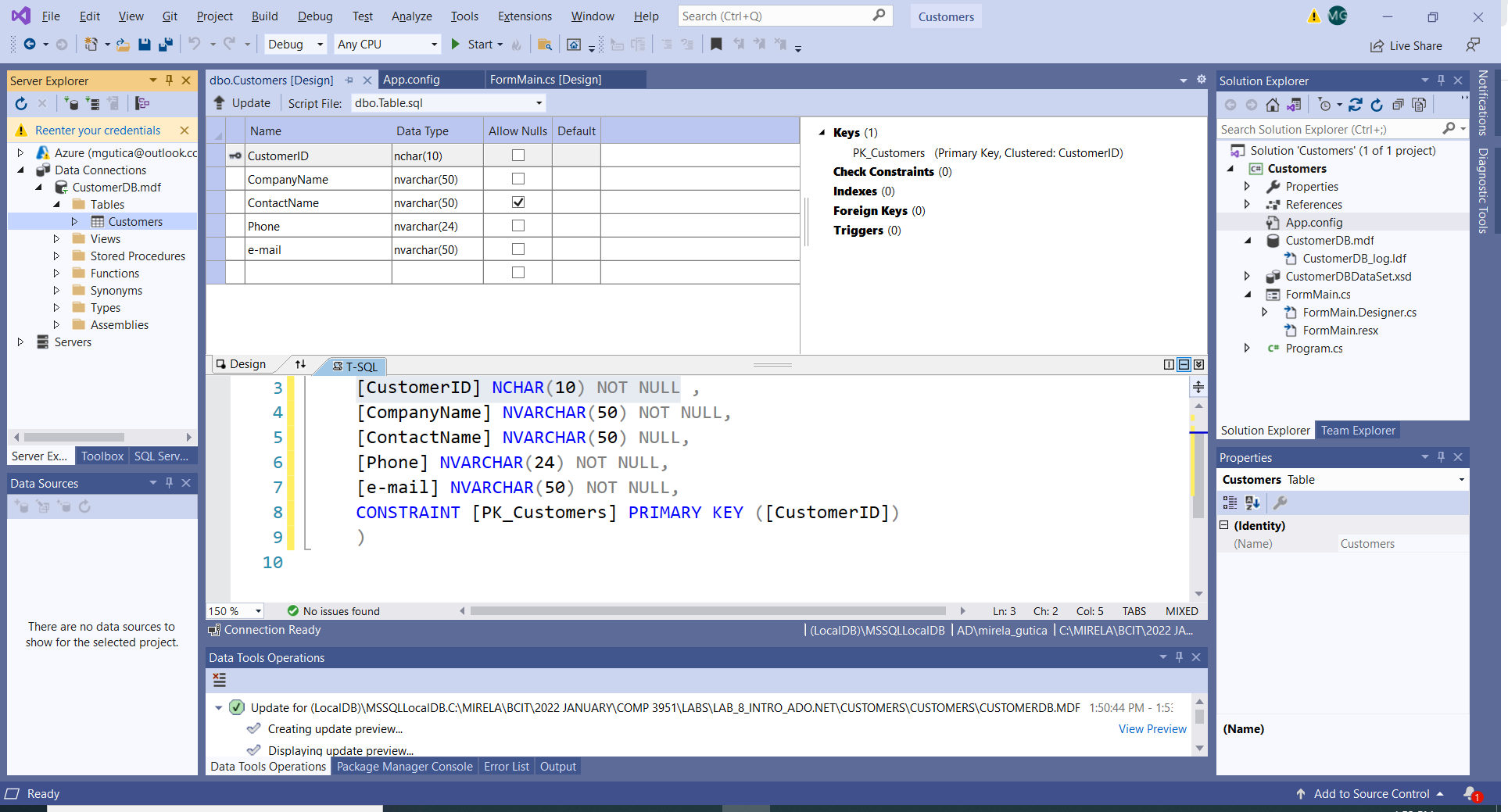
* 1. Right click on Tables and Add a New Table:
  2. The Table Designer opens and shows a grid with one default row, which represents a single column in the table that you're creating. By adding rows to the grid, you'll add columns in the table:



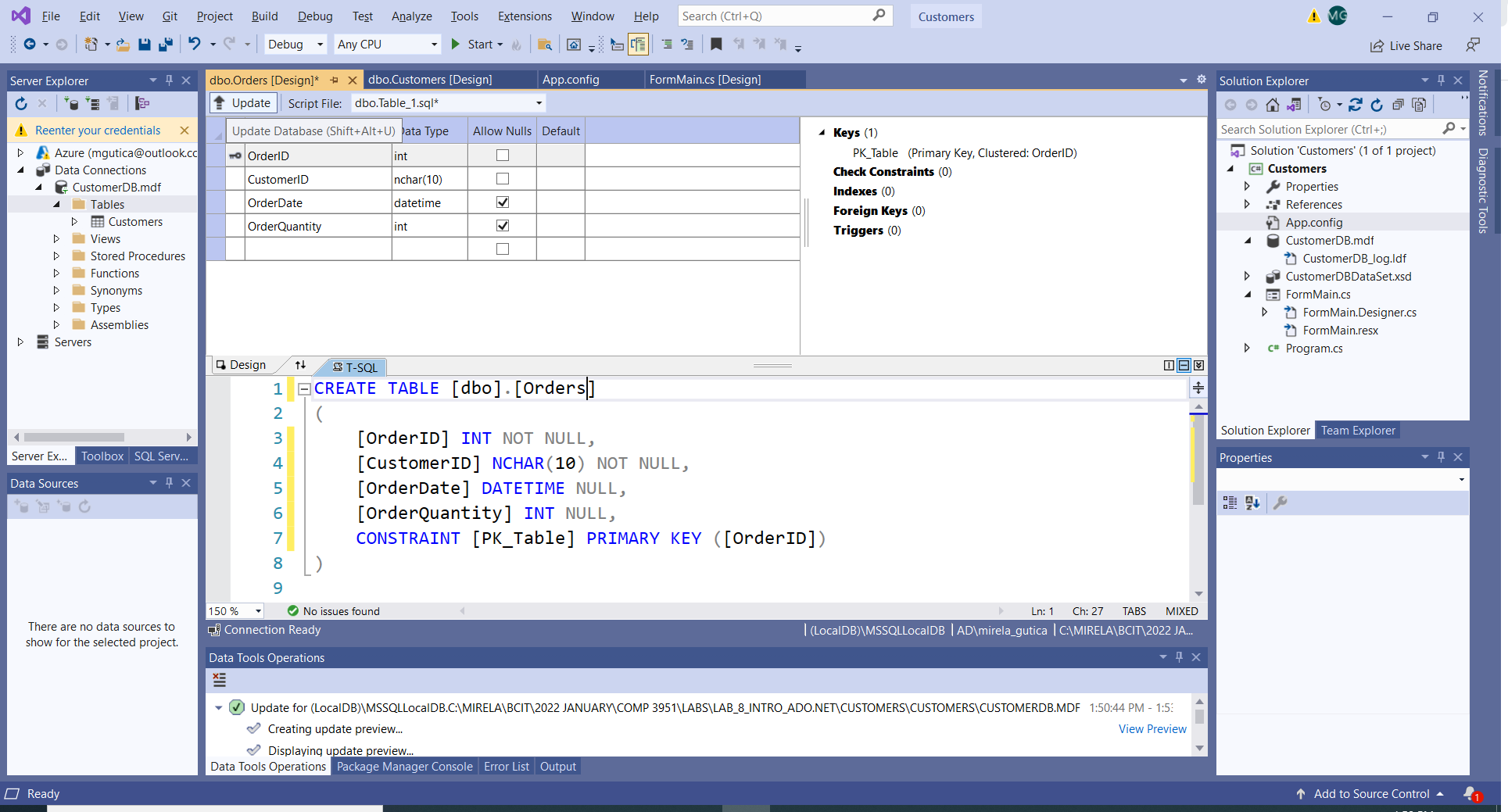
* 1. Delete the Id row. Create the table columns. Right click on the CustomerID item and set it as the Primary Key. Name the table “Customers”. The table should look like this: 
  2. In the upper-left corner of Table Designer, select Update, or press Shift+Alt+U.
  3. In the Preview Database Updates dialog box, select Update Database. The table will be created.



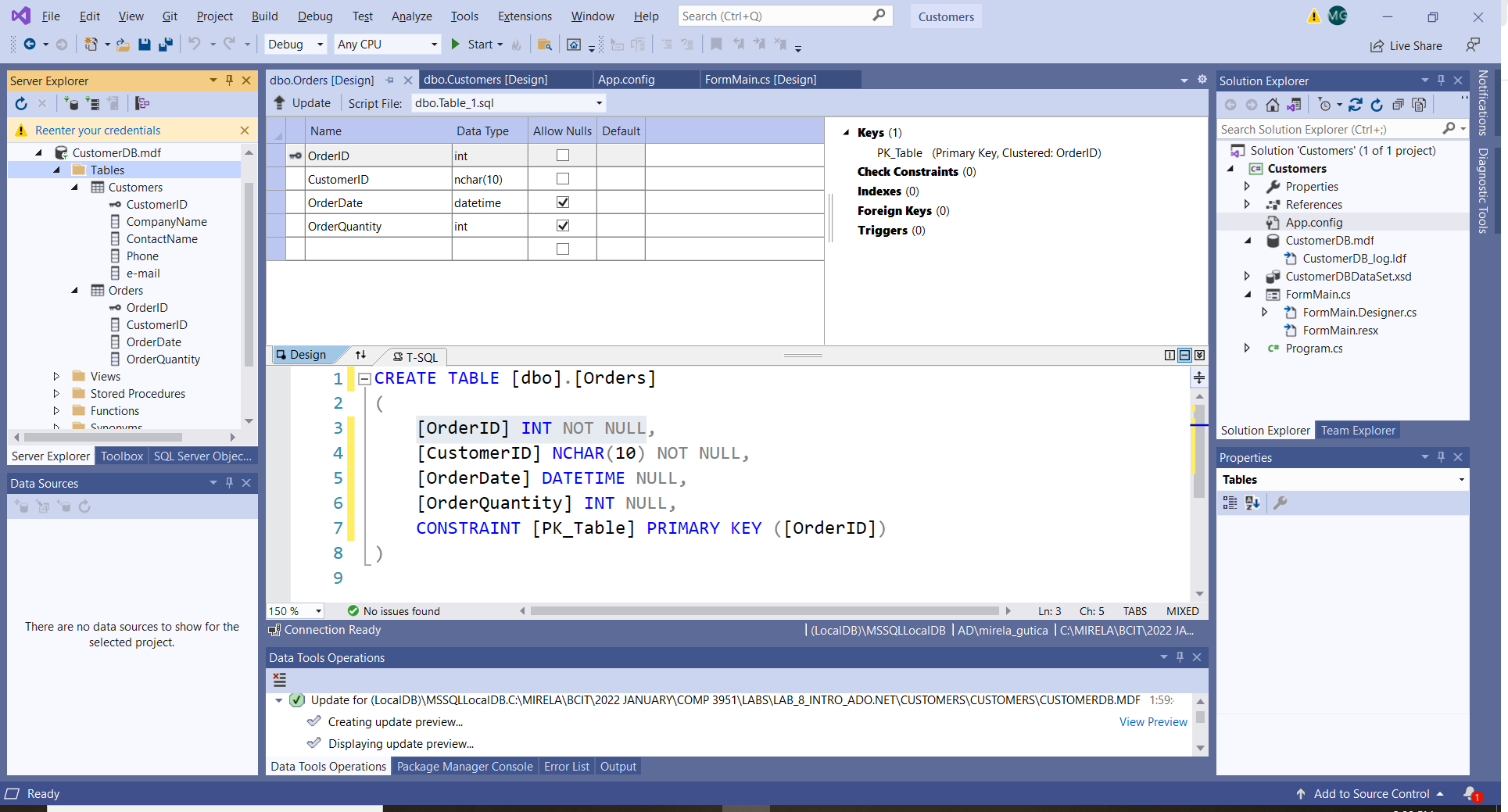
* 1. The table will be created:



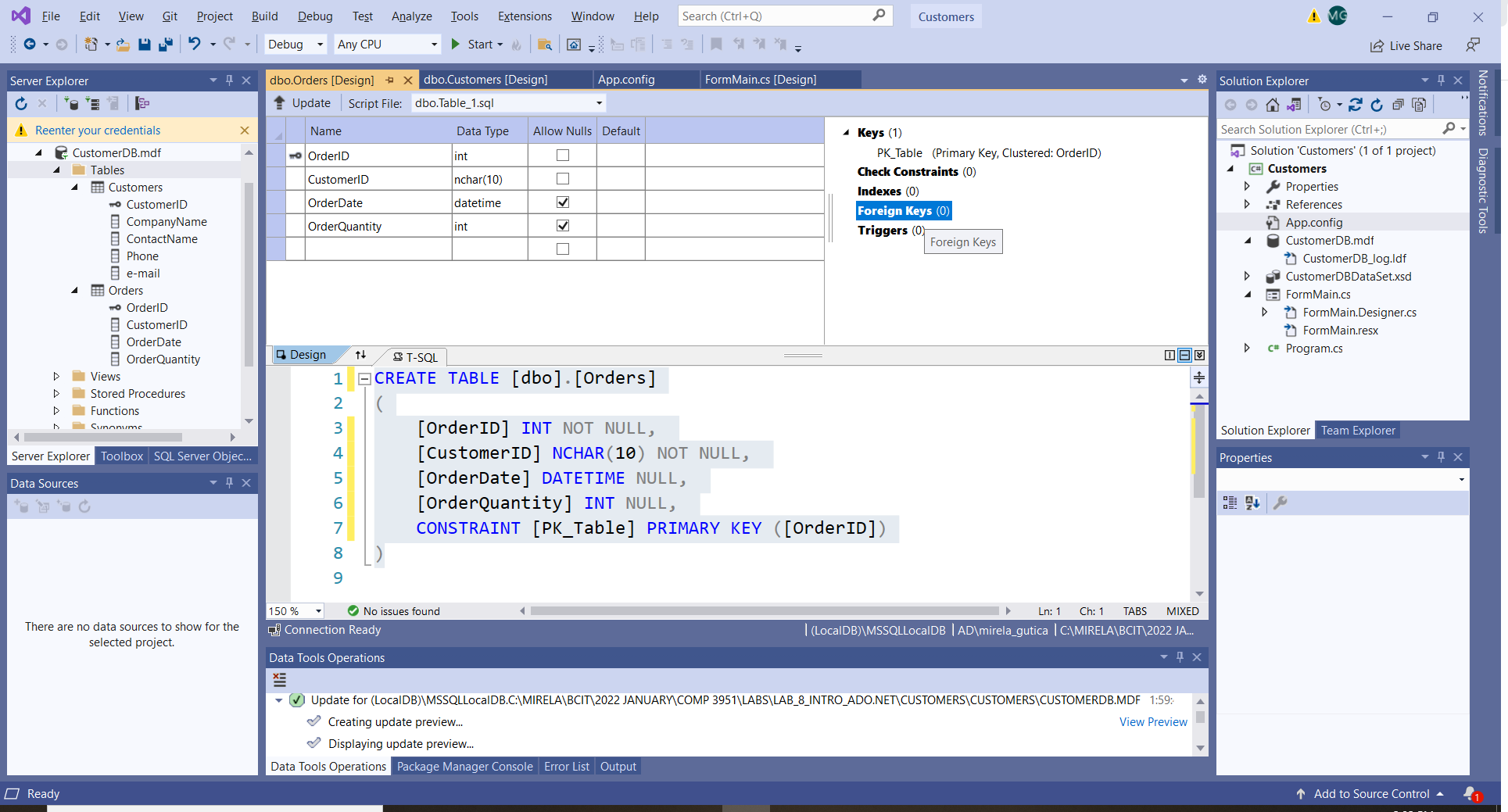
* 1. Using the same steps create another table Orders:



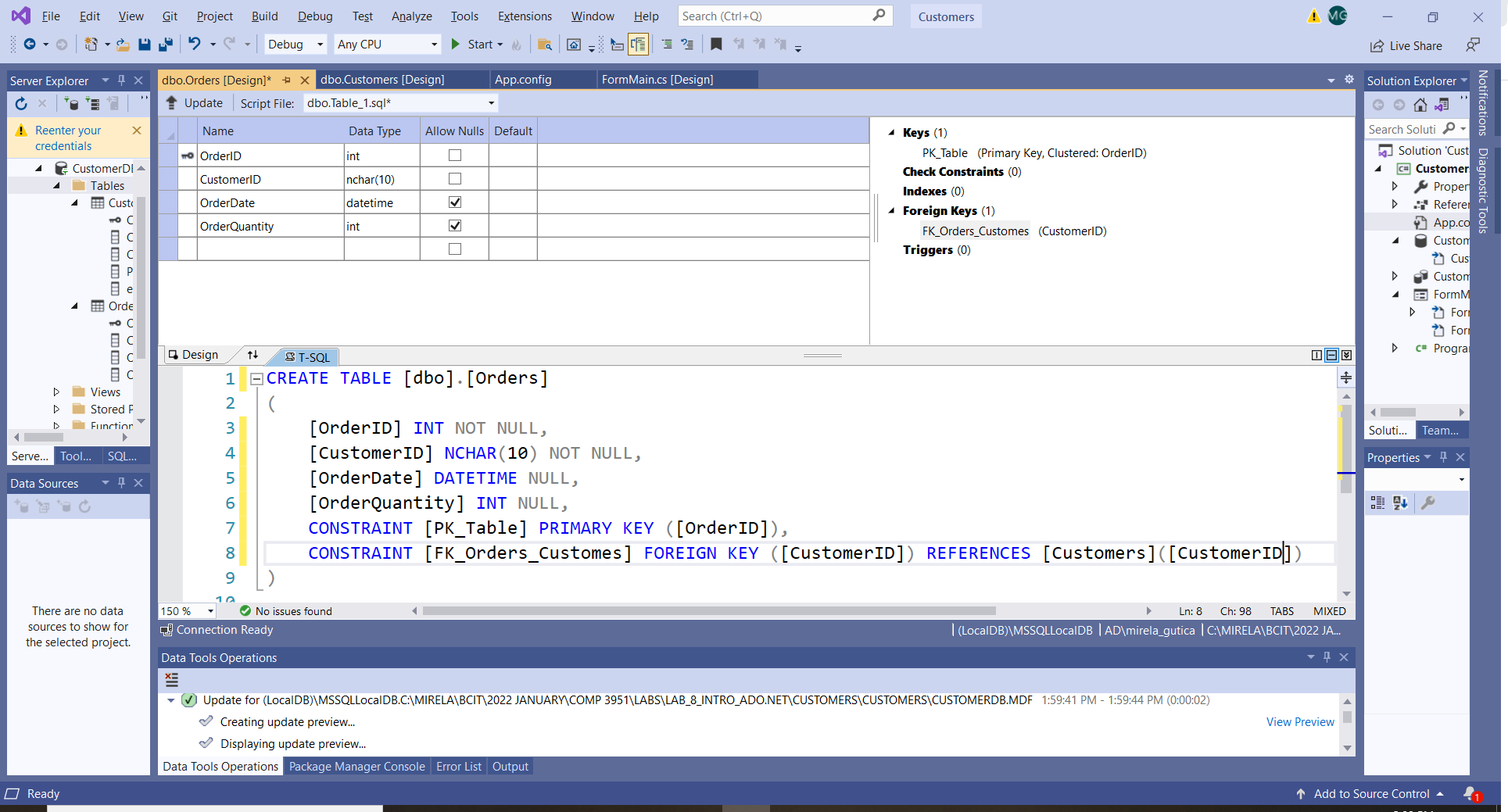
* 1. Update the DB. Now the DB has two tables:



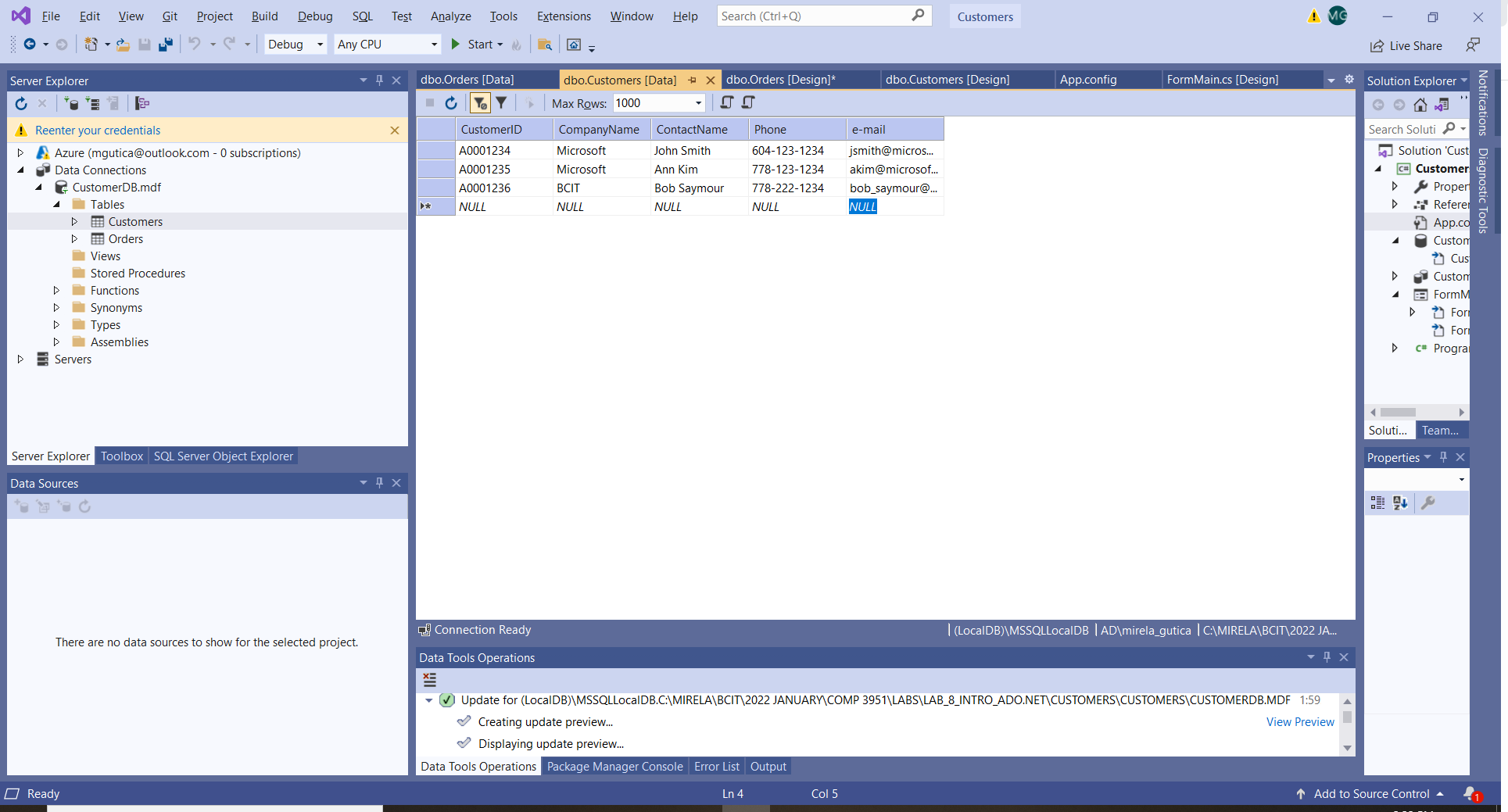
* 1. In the context pane on the right side of the Table Designer grid for the Orders table, right-click on Foreign Keys and select Add New Foreign Key:

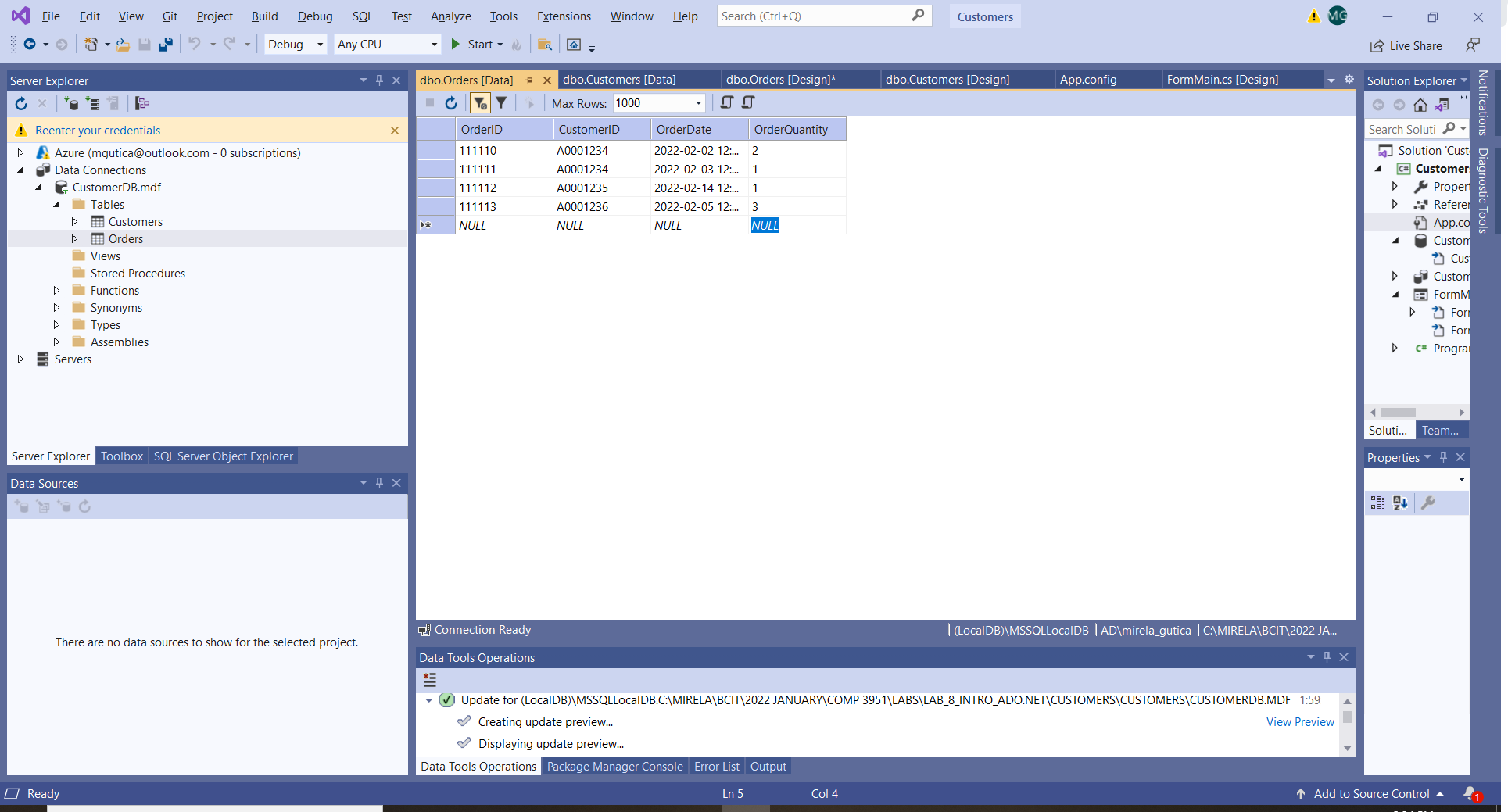


* 1. In the text box that appears, replace the text ToTable with Customers.
  2. In the T-SQL window update the list to indicate CustomerID as the Foreign Key:

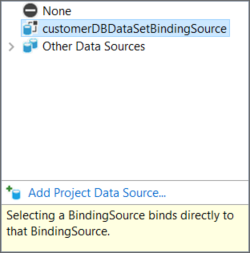


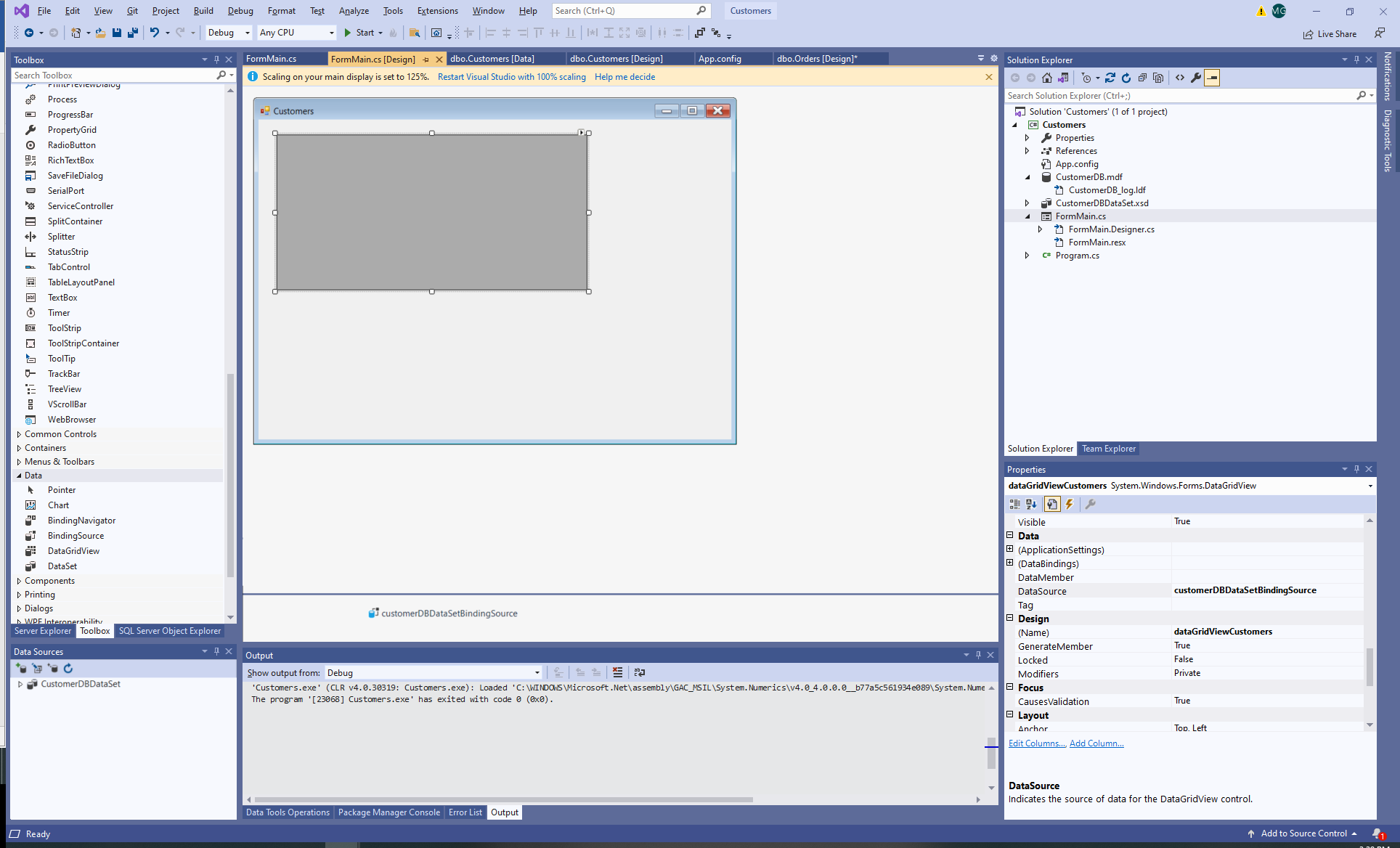
1. Populate the DB with data:
   1. In the Server Explorer:
      1. Refresh
      2. Right click on each table and select Show Table Data
      3. Populate with data of your choice:



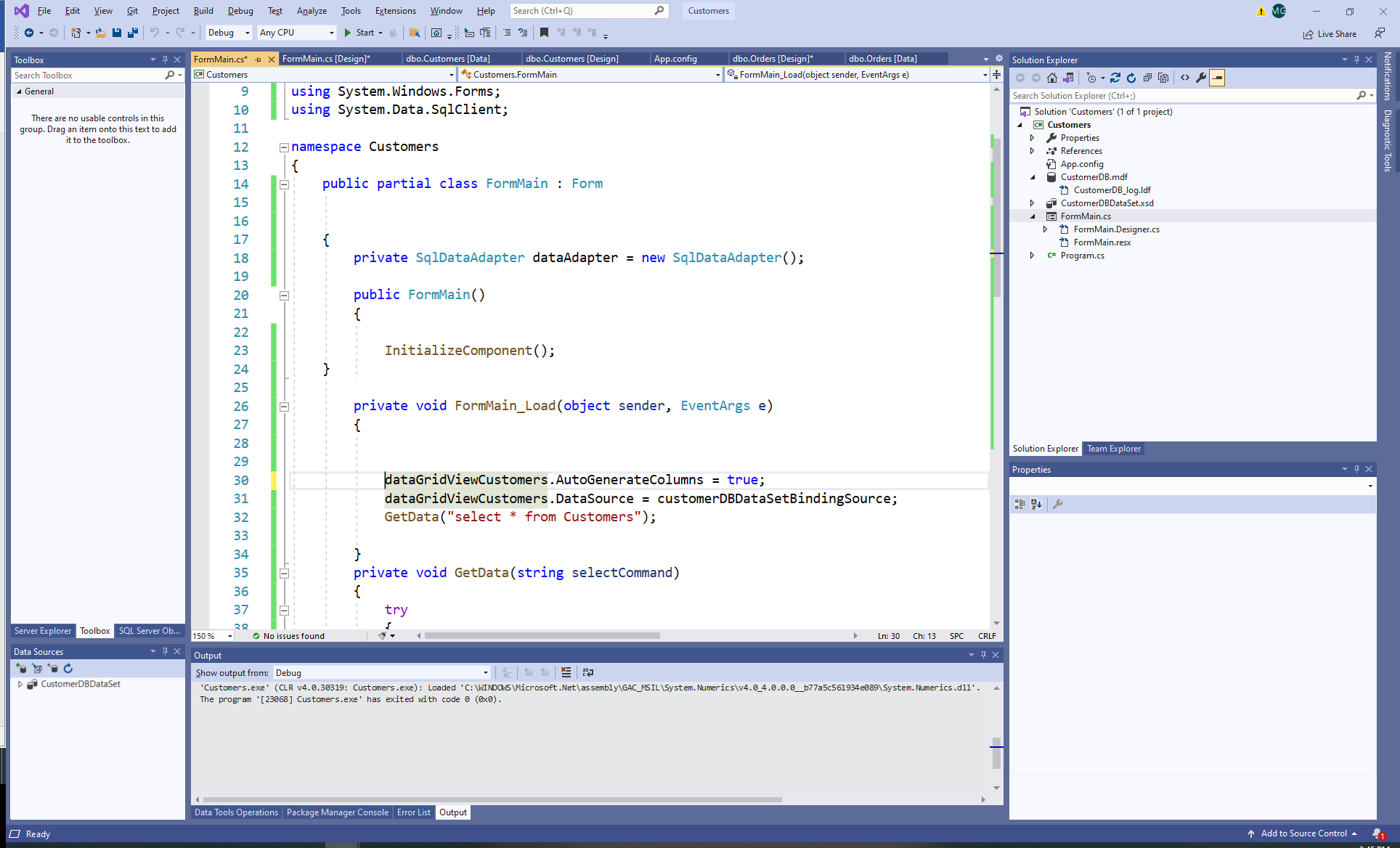


1. Display one of the data tables in the Main Form:
   1. In the ToolBox, identify the Data collection of Controls
   2. dd to the form a DataGridView and a BindingSource.
      1. The DataGridView is connected to the BindingSource
      2. The DataGdriView and BindingSource can be added in the designer or programmatically

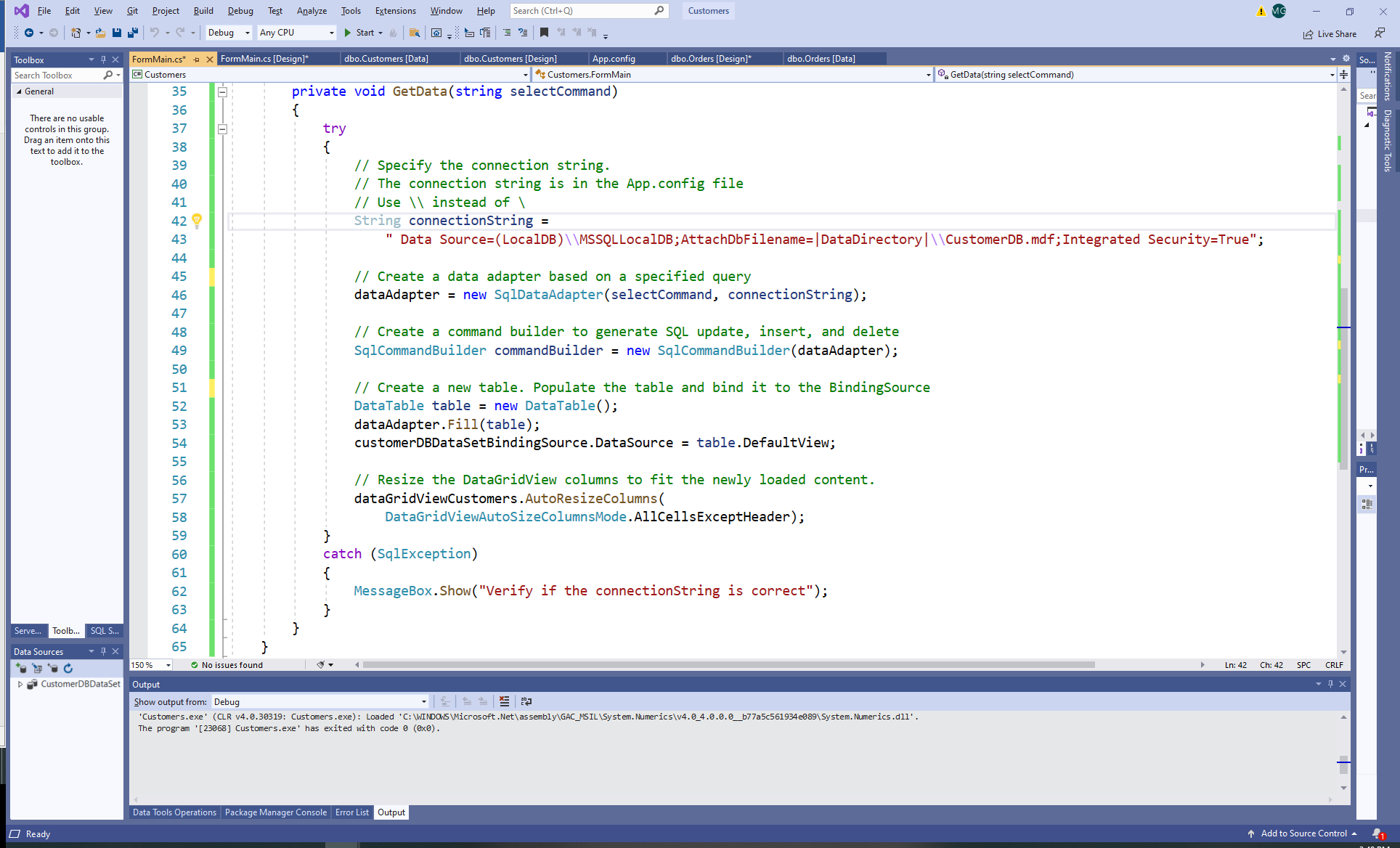




* 1. In the Code Behind add the NameSpace:
     1. using System.Data.SqlClient;
  2. Add an SQL data adapter:
     1. private SqlDataAdapter dataAdapter = new SqlDataAdapter();
  3. On FormLoad() connect the DataGrdidView tio the BindingSource.



* 1. Extract the data from the DB:



* 1. Run the application:

