**IFT 598: IFT 458: Middleware Prog & Database Sec (2022 Fall)**

**Assignment-2: Database Access**

**Name: ABHISHEK MATHUR**

**Professor: Dinesh Sthapit**

**This is our project folder structure:**

**Graphical user interface, text, application

Description automatically generated**

**SQL Structure**

**Graphical user interface, text, application

Description automatically generated**

**Step 1: Next, we initialize the Node.js App with a package.json file:**

**A picture containing graphical user interface

Description automatically generated**

**Step 2: We need to install necessary modules: express, dotenv, mssql, morgan and cors**

**A picture containing shape

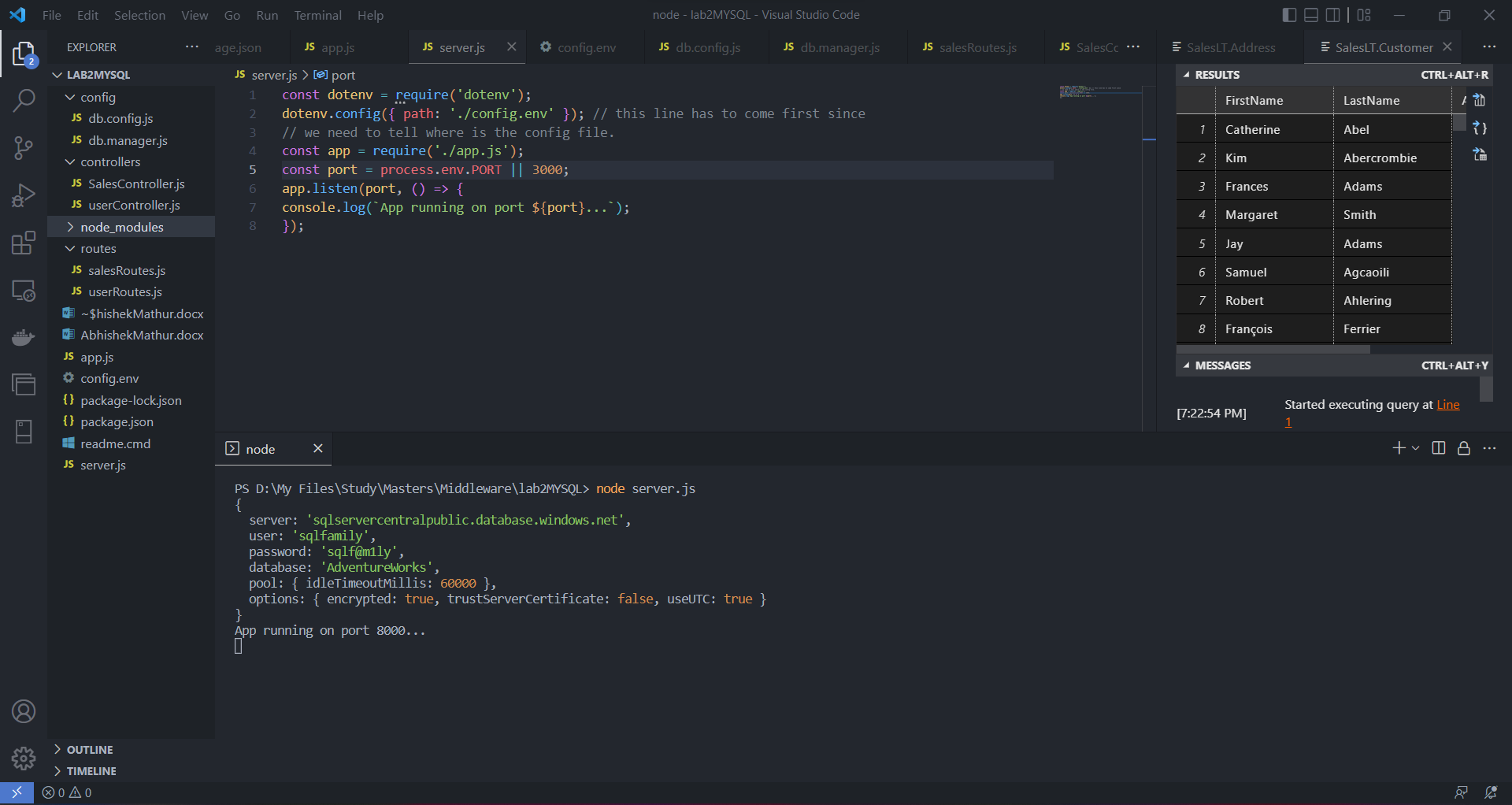
Description automatically generated**

**Setup 3: Setup Express web server**

**A screenshot of a computer

Description automatically generated with medium confidence**

**Step 4: let’s create a new server.js file**

****

**Step 5: Yeah, the first step is done. We’re going work with configuration in the next section.**

**Graphical user interface, text, application

Description automatically generated**

**Step 6: Connect to SQL Server using Node.js we need to install mssql packagenpm**

**Shape

Description automatically generated with medium confidence**

**Graphical user interface, text

Description automatically generated**

**Step 7: Create another file in the config folder and name it db.manager.js And type the following code:**

**Graphical user interface, text, application

Description automatically generated**

**Step 8a: Create a file in routs folder and name it salesRouter. And type the following code**

**Graphical user interface, text

Description automatically generated**

**Step 8b: Create a file under controllers folder and name it salesController.js and type the following**

**Text

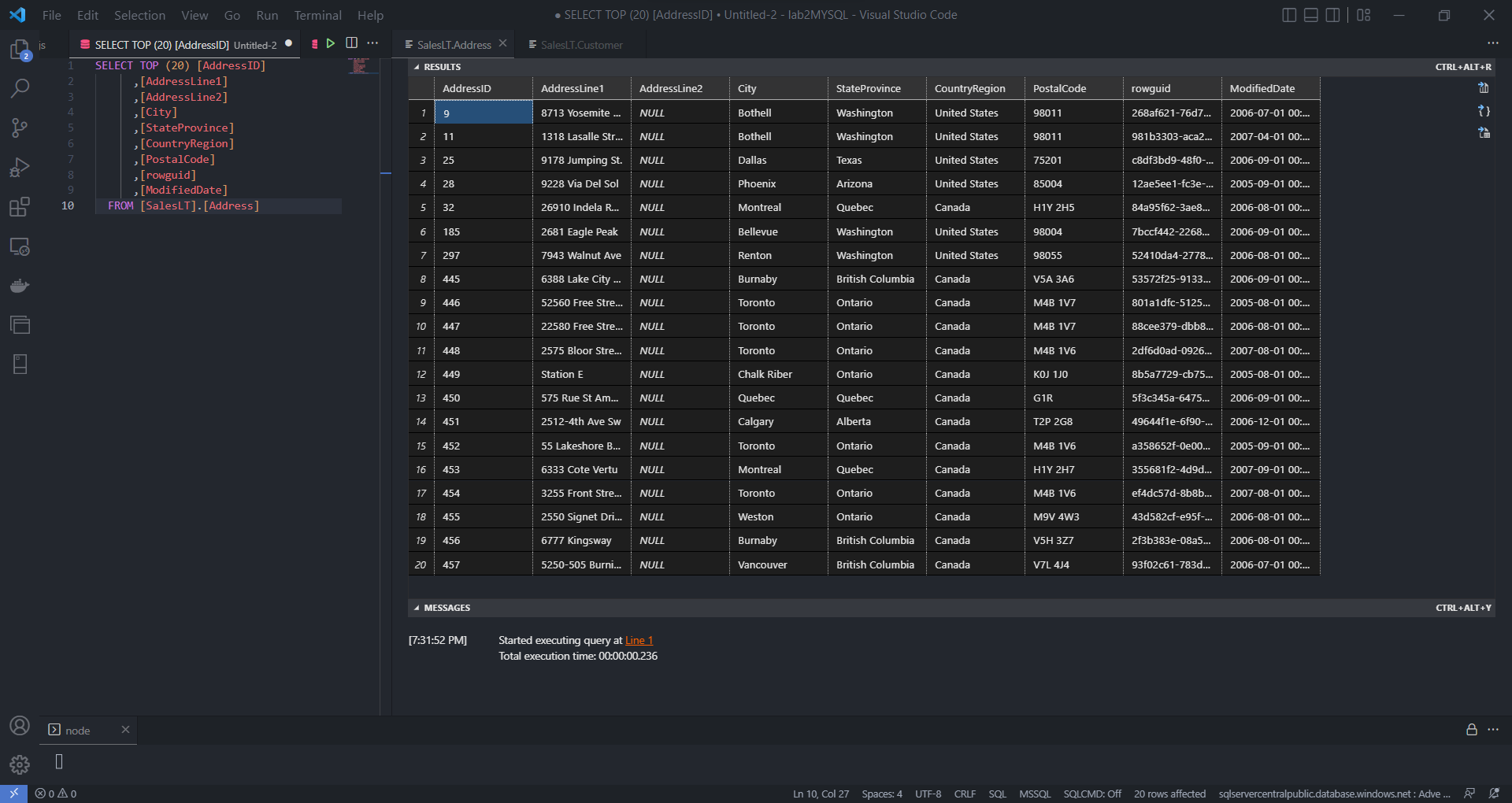
Description automatically generated**

**Test the APIs**

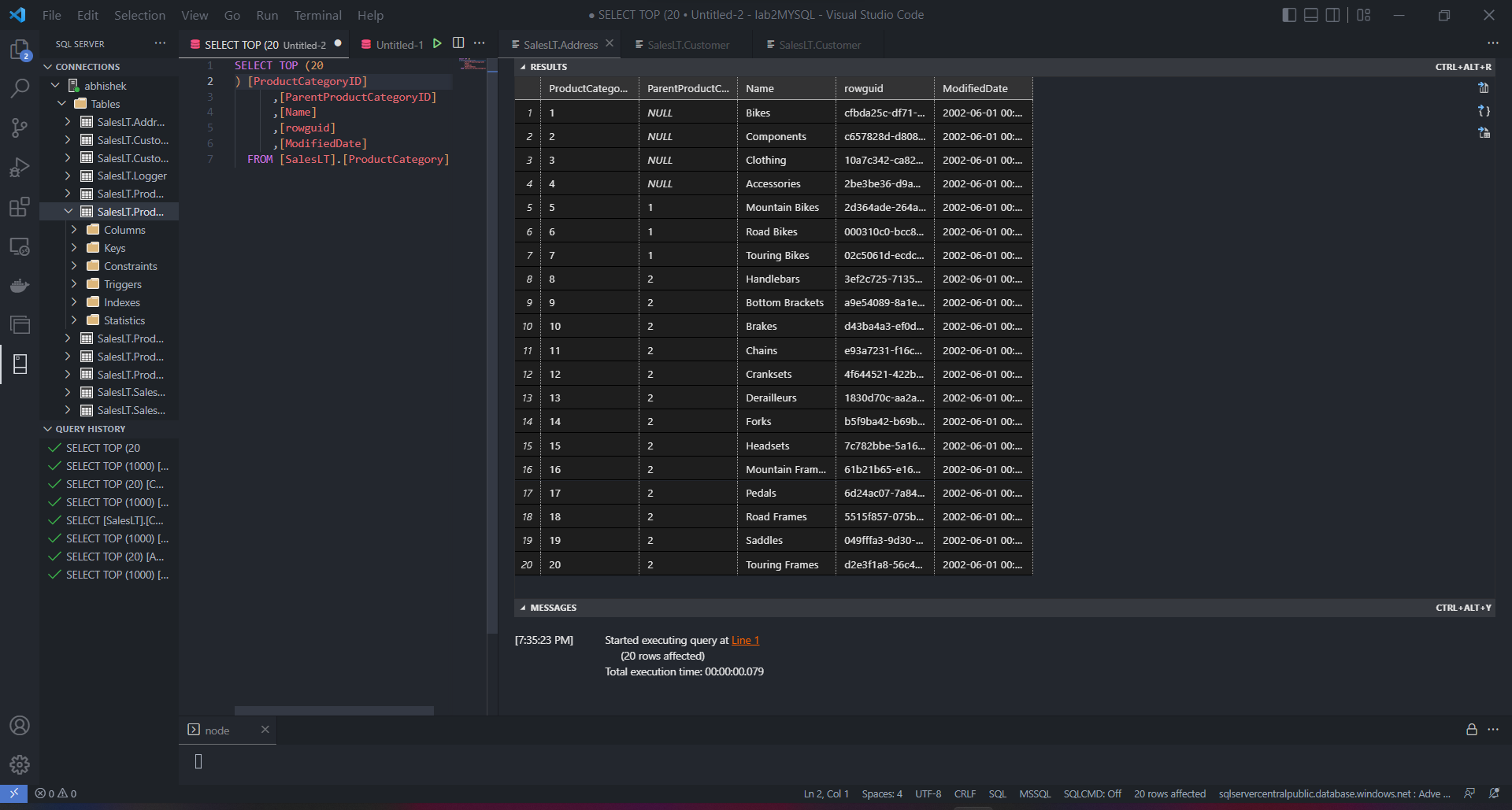
**Graphical user interface, text, application, email

Description automatically generated**

**ScreenShot Contains the Table Views**

****

**This criterion is linked to a Learning OutcomeCreated Correct Individual SQL SELECT Statement to Query Data in four Tables**

****

SELECT TOP (20) [ProductModelID]

      ,[Name]

      ,[CatalogDescription]

      ,[rowguid]

      ,[ModifiedDate]

  FROM [SalesLT].[ProductModel]

**A screenshot of a computer

Description automatically generated with medium confidence**

SELECT TOP (20) [ProductModelID]

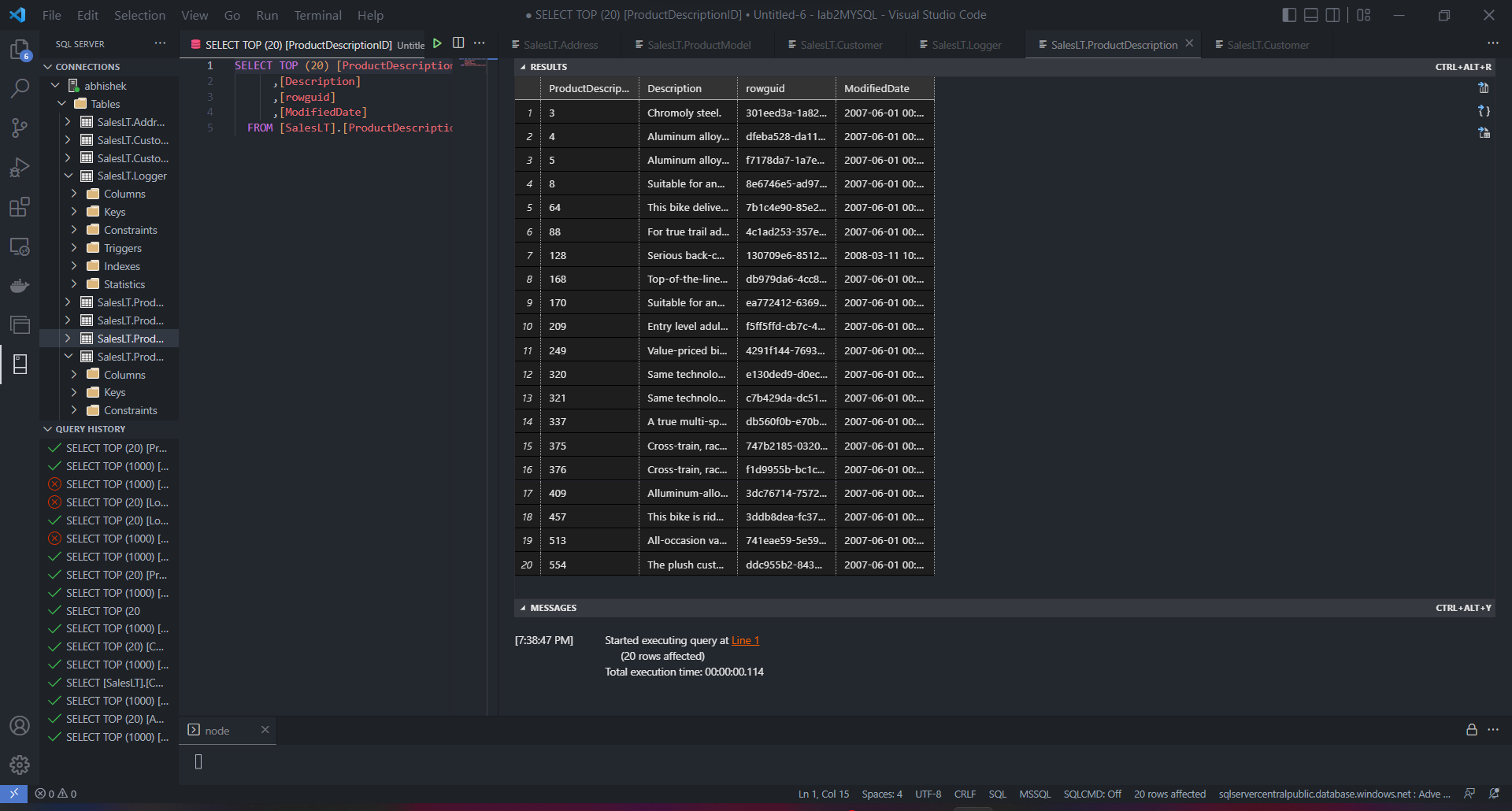
      ,[Name]

      ,[CatalogDescription]

      ,[rowguid]

      ,[ModifiedDate]

  FROM [SalesLT].[ProductModel]

****

SELECT TOP (20) [ProductDescriptionID]

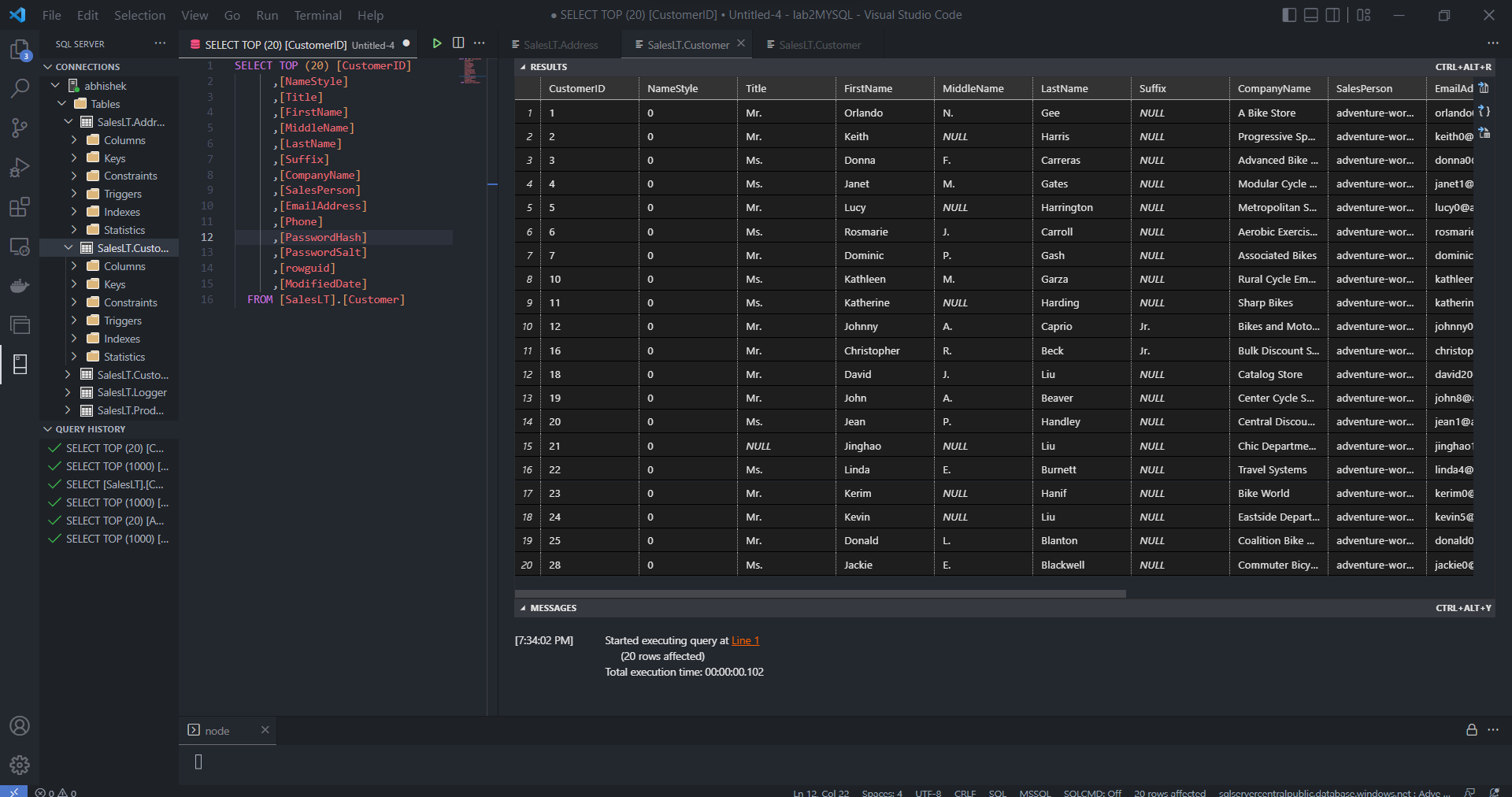
      ,[Description]

      ,[rowguid]

      ,[ModifiedDate]

  FROM [SalesLT].[ProductDescription]

**Provide the SQL Statements that you used to get Customers from the Database.**

****

SELECT TOP (20) [CustomerID]

      ,[NameStyle]

      ,[Title]

      ,[FirstName]

      ,[MiddleName]

      ,[LastName]

      ,[Suffix]

      ,[CompanyName]

      ,[SalesPerson]

      ,[EmailAddress]

      ,[Phone]

      ,[PasswordHash]

      ,[PasswordSalt]

      ,[rowguid]

      ,[ModifiedDate]

  FROM [SalesLT].[Customer]

**This criterion is linked to a Learning OutcomeCreated Correct SQL statements to JOIN Data in more than one table**

**This criterion is linked to a Learning OutcomeCreated Correct SQL statements to JOIN Data in more than one table**

**Show an example of joining two tables using COMPLETE SQL STATEMENTS with related data.**

**Graphical user interface, text

Description automatically generated**

SELECT [SalesLT].[Customer].FirstName, [SalesLT].[Customer].LastName,

    [SalesLT].[CustomerAddress].AddressType

FROM [SalesLT].[Customer]

    INNER JOIN [SalesLT].[CustomerAddress] ON

[SalesLT].[CustomerAddress].CustomerID=[SalesLT].[Customer].CustomerID;

**GIT Repository:**

[**https://github.com/abhishekmathur47/DataAccess**](https://github.com/abhishekmathur47/DataAccess)