## Getting Started with the Tabular Object Model (TOM)

The purpose of these hands-on lab exercises is to provide campers with experience writing.

We have had .NET Framework and .NET Core which have been seen as two different platforms. The new version of .NET aims to unify these different platforms into a single platform. In this lab we will refer to the new unified runtime as .NET 5 just to make things more clear. Note that Microsoft will begin referring to the new runtime as simply .NET instead of .NET 5..

This lab was inspired by a set of blog posts by Phil Seamark.

### Setup: Install Visual Studio Code and the .NET 5 SDK

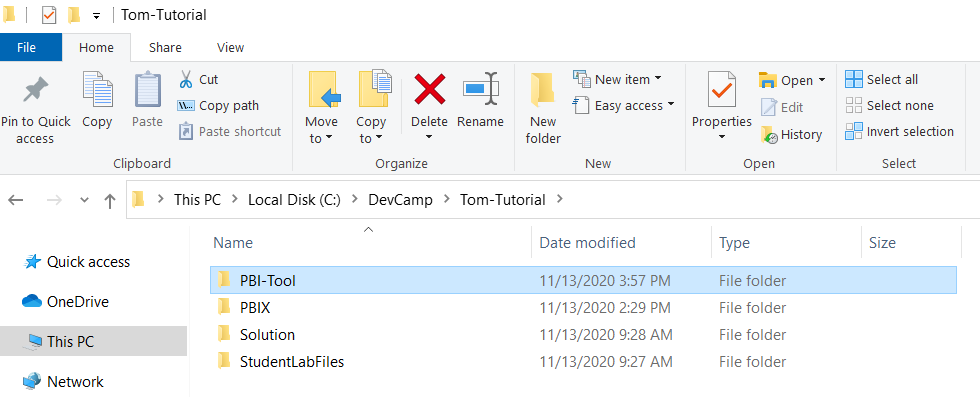
In this exercise, you will download xx.

1. Install the .NET 5 SDK.
   1. <https://dotnet.microsoft.com/download/dotnet/thank-you/sdk-5.0.100-windows-x64-installer>
   2. More info: <https://dotnet.microsoft.com/download>
2. Install Visual Studio Code
   1. Open a browser and navigate to [code.visualstudio.com](https://code.visualstudio.com/)
   2. download and run the installer for the current version for Windows.
   3. Use default settings when prompted during the install.

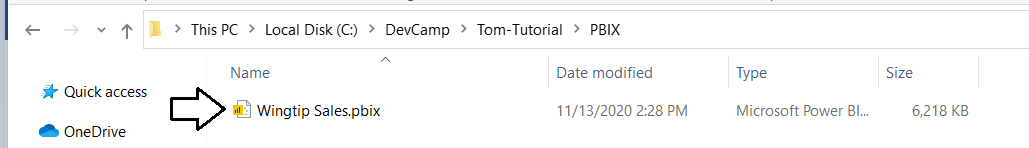
### Exercise 1: xxx

In this exercise, you will download xx.

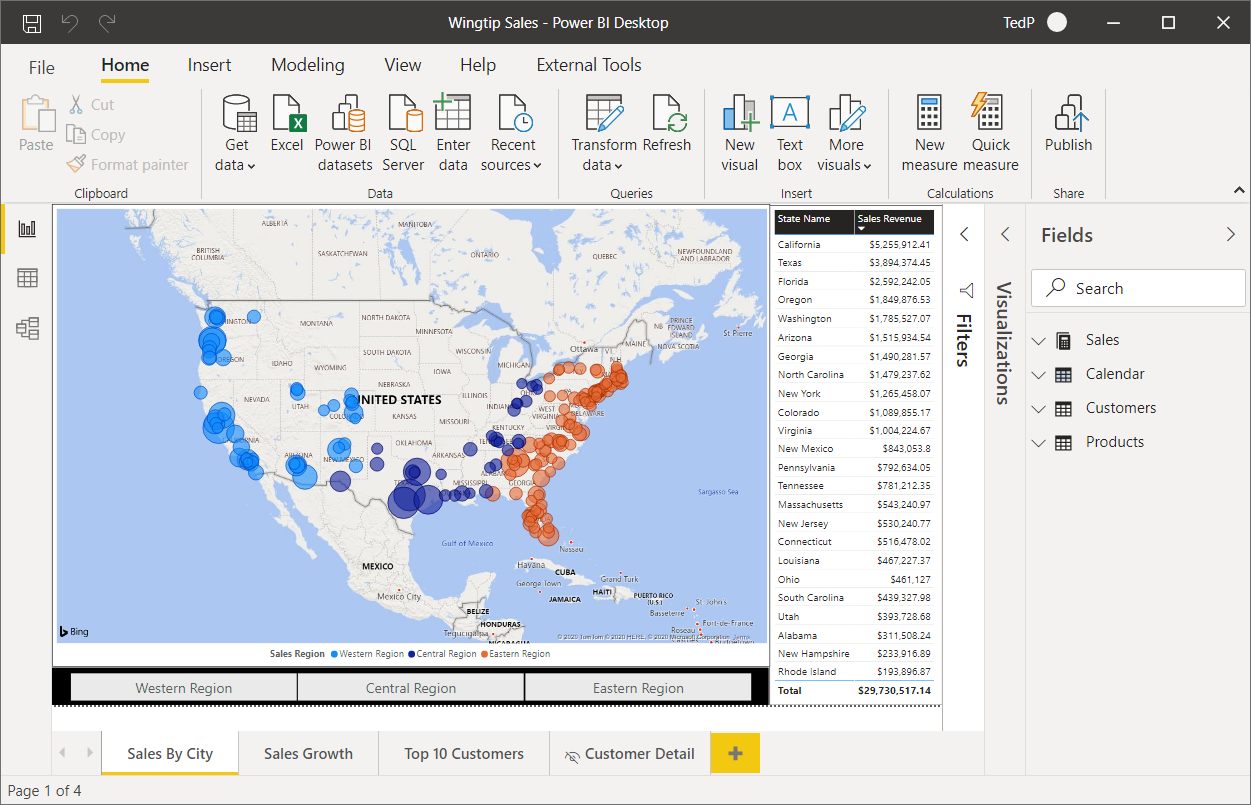
1. Create a new folder for the tutorial.
   1. Create a new folder on your local hard drive named **Tom-Tutorial**.
   2. Create a child inside the **Tom-Tutorial** folder named **PBI-Tool**.

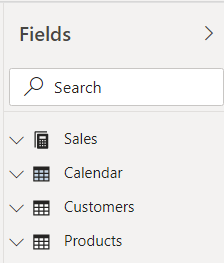


1. Locate Wingtip Sales.pbix and open it in Power BI Desktop.
   1. Aaaa

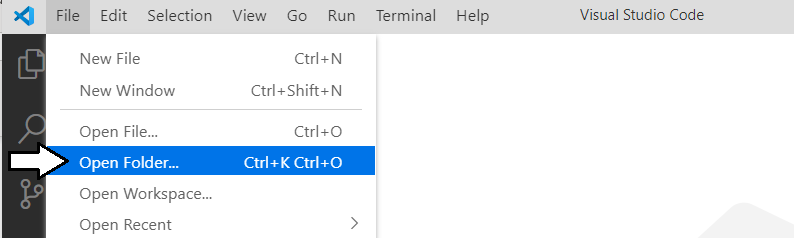


* 1. dddd

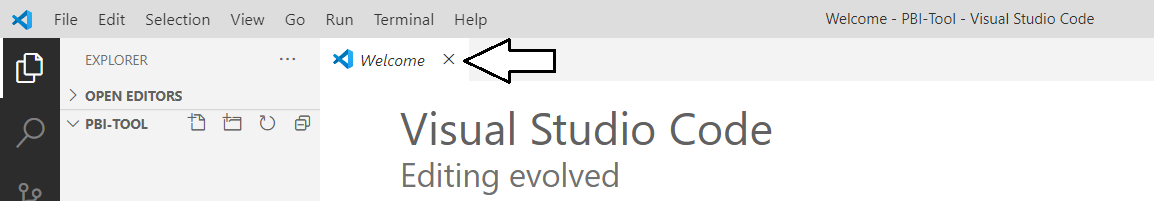




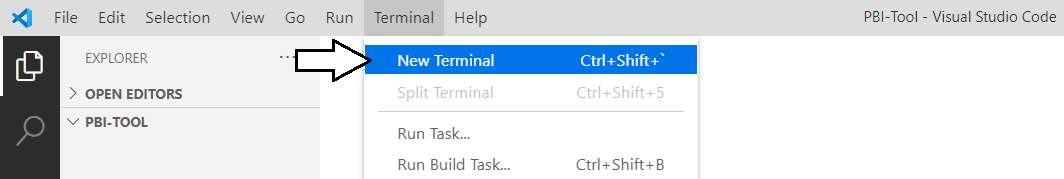
1. Launch Visual Studio Code and open the **PBI-Tool** folder.
   1. Sss



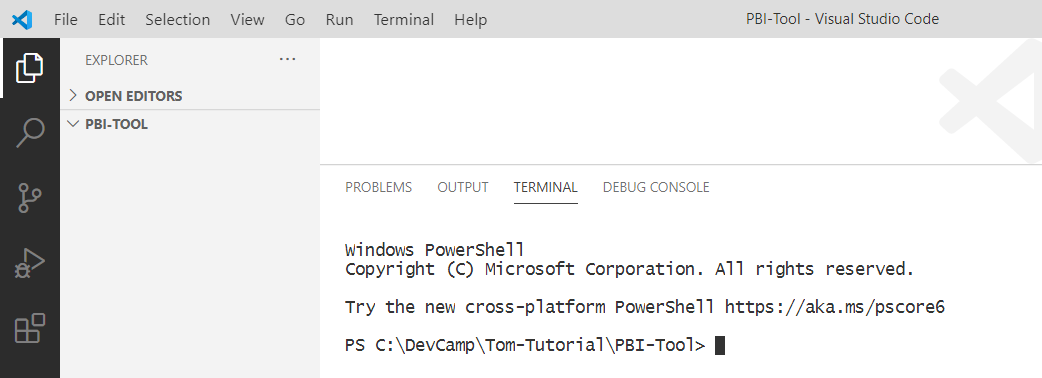
* 1. Close the start page



1. Verify the current version of .NET
   1. Open terminal.



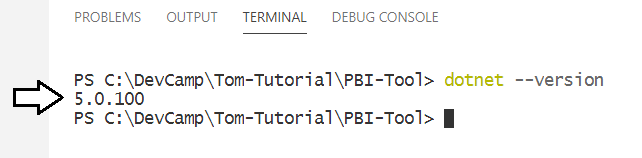
* 1. dddd



* 1. Run this command.

dotnet –version

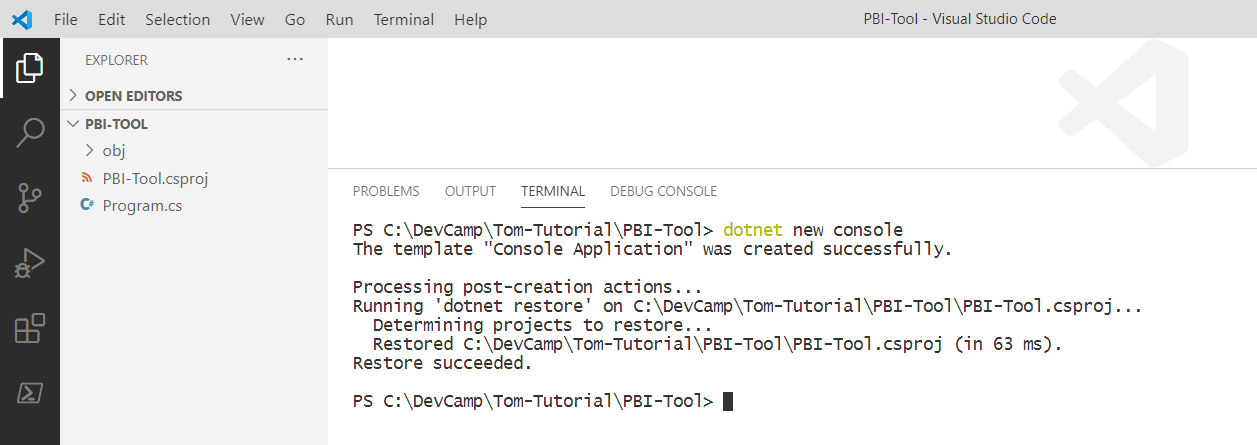
* 1. You should see the version number.



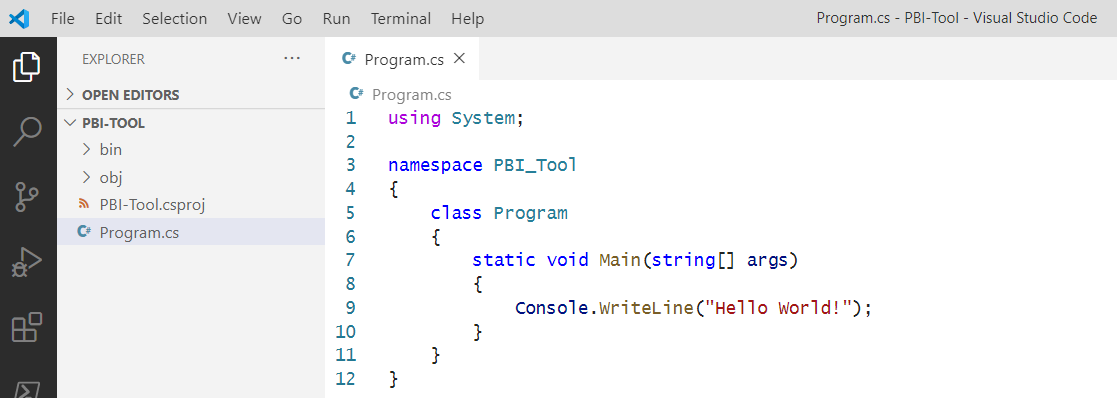
1. Create new folder and navigate to it.
   1. Xx

dotnet new console

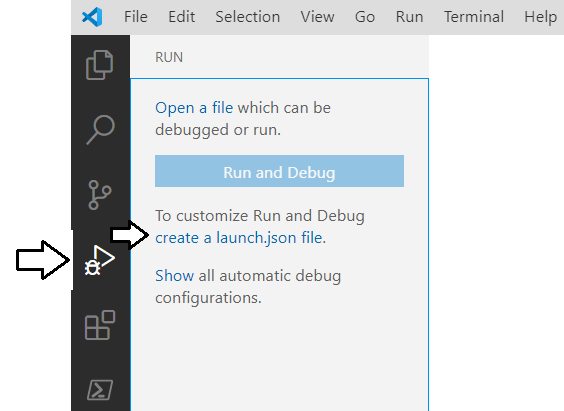
* 1. Dddd



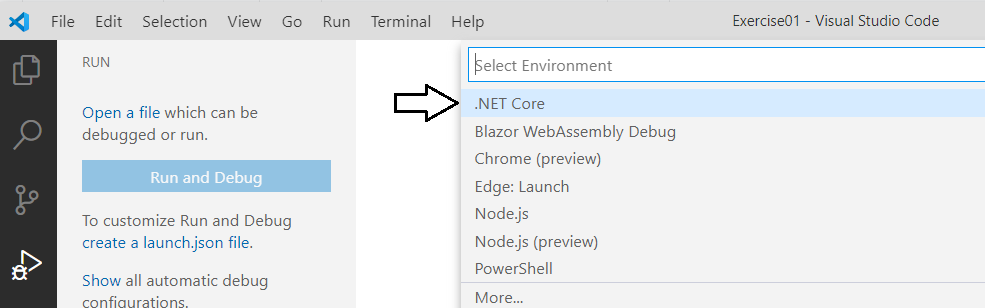
* 1. Xxxxx



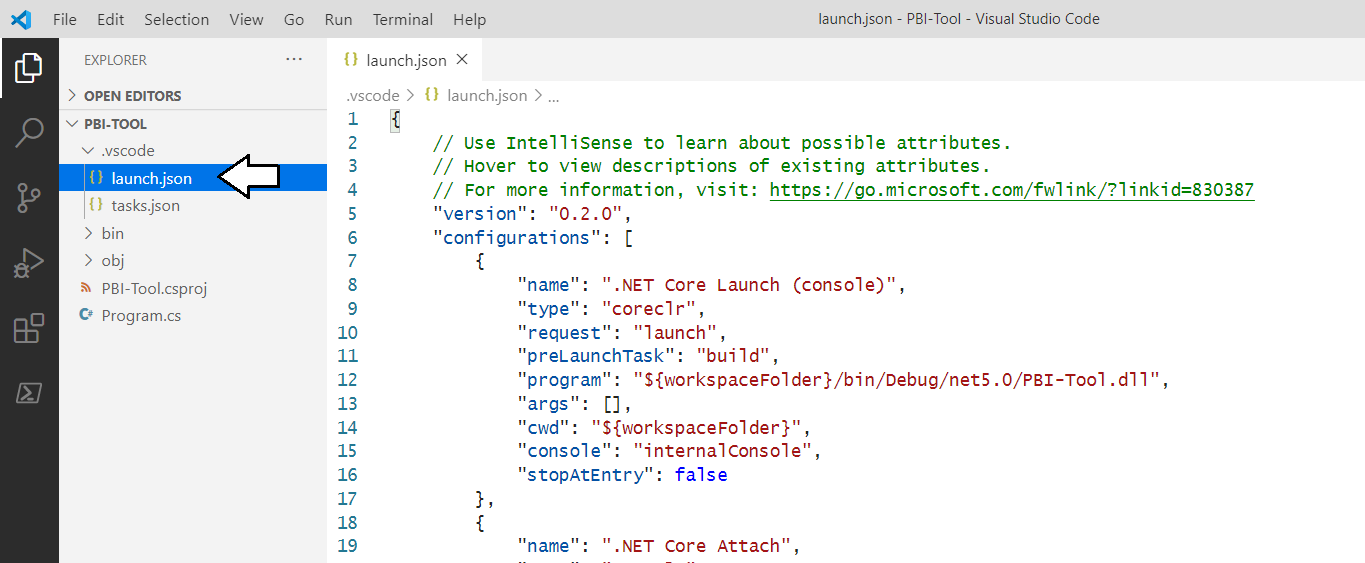
* 1. X



* 1. X



* 1. sssss



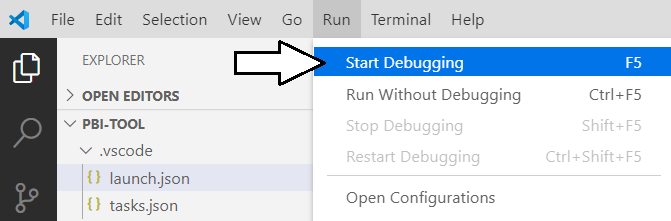
Ssss

"logging": { "moduleLoad": false }

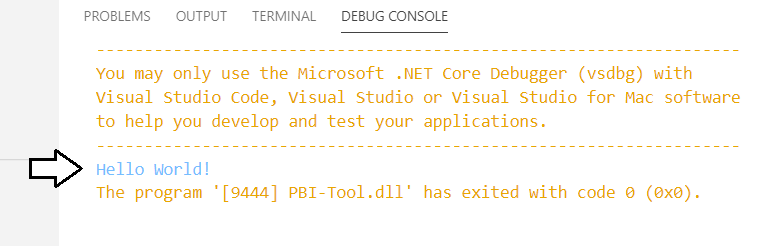
X



* 1. sssss



* 1. dddd



x

* 1. x

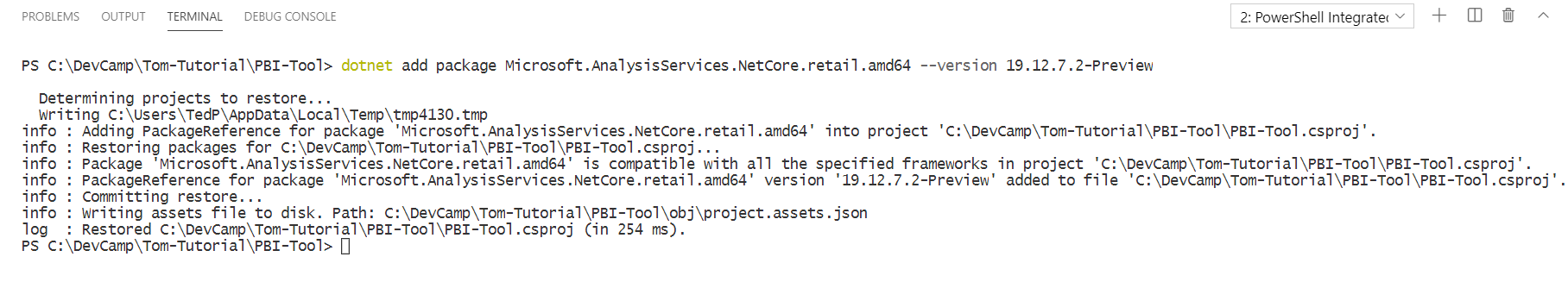
1. Add package
   1. Sss

dotnet add package Microsoft.AnalysisServices.NetCore.retail.amd64 --version 19.12.7.2-Preview

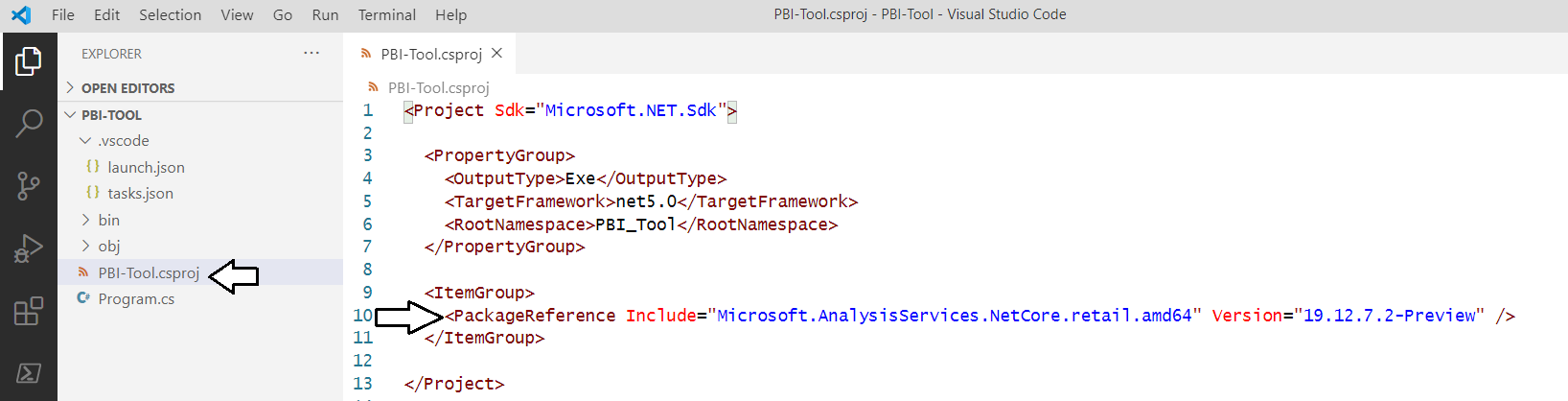
ssssss

<https://www.nuget.org/packages/Microsoft.AnalysisServices.NetCore.retail.amd64>

* 1. Ssss



* 1. X



* 1. Close **PBI-Tool.csproj** without saving any changes.

1. Modify the C# code in the program.
   1. Sss

using System;

using Microsoft.AnalysisServices.Tabular;

class Program {

const string connectString = "localhost:50000"; // update for port number on your machine

static void Main(string[] args) {

Server server = new Server();

server.Connect(connectString);

Model model = server.Databases[0].Model;

foreach(Table table in model.Tables) {

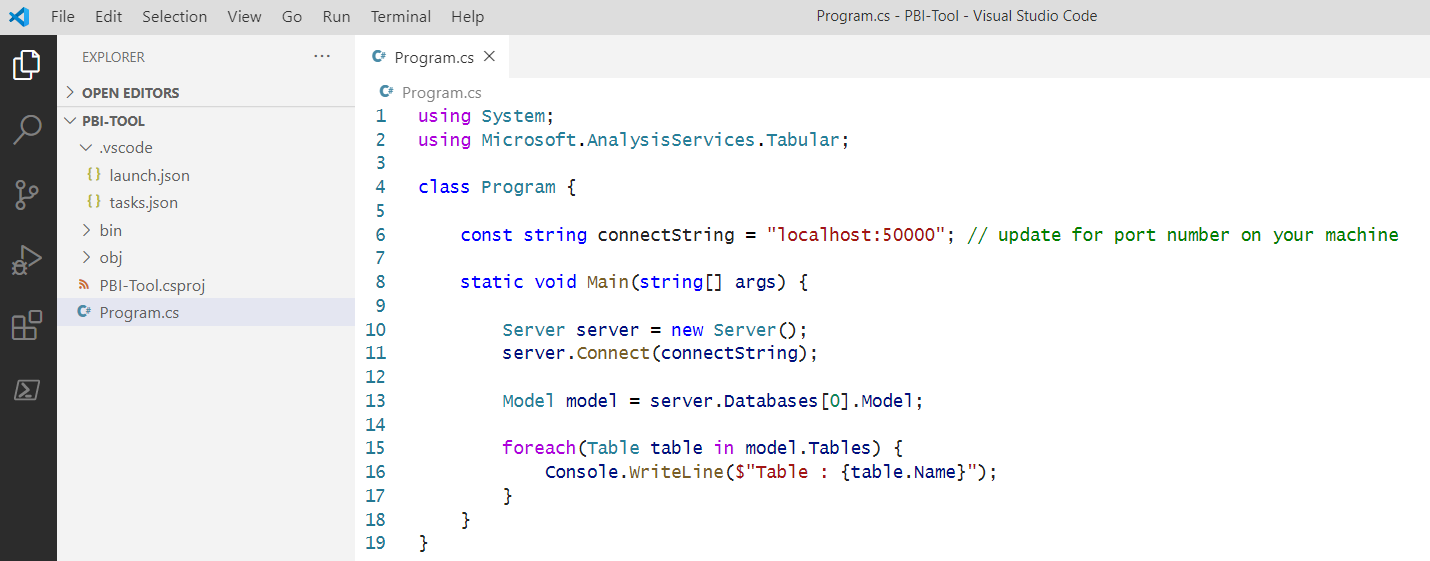
Console.WriteLine($"Table : {table.Name}");

}

}

}

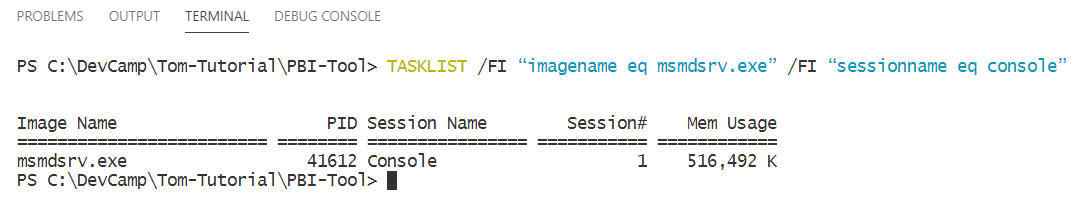
* 1. Aaa

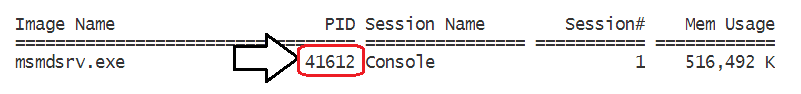


1. Xxxssss
   1. sss

TASKLIST /FI “imagename eq msmdsrv.exe” /FI “sessionname eq console”

* 1. Looks like this

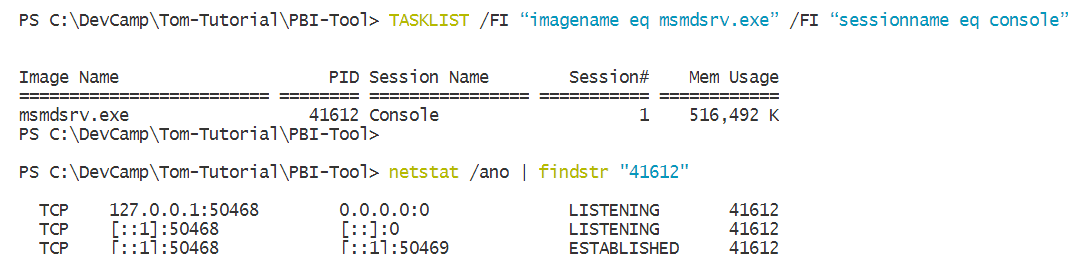




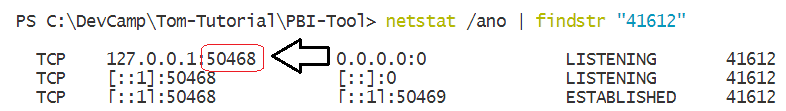
* 1. X

netstat /ano | findstr "41612"

* 1. X



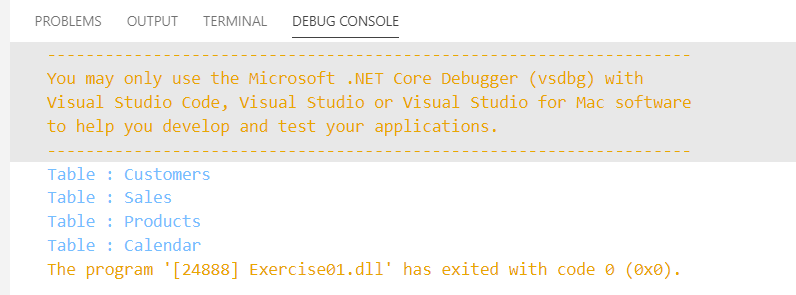
* 1. Z



* 1. x



1. zzzzz
   1. ss



1. x
   1. caddd

Table table = model.Tables["Sales"];

if (table.Measures.ContainsName("VS Code Measure")) {

Measure measure = table.Measures["VS Code Measure"];

measure.Expression = "\"Hello Again World\"";

}

else {

Measure measure = new Measure() {

Name = "VS Code Measure",

Expression = "\"Hello World\""

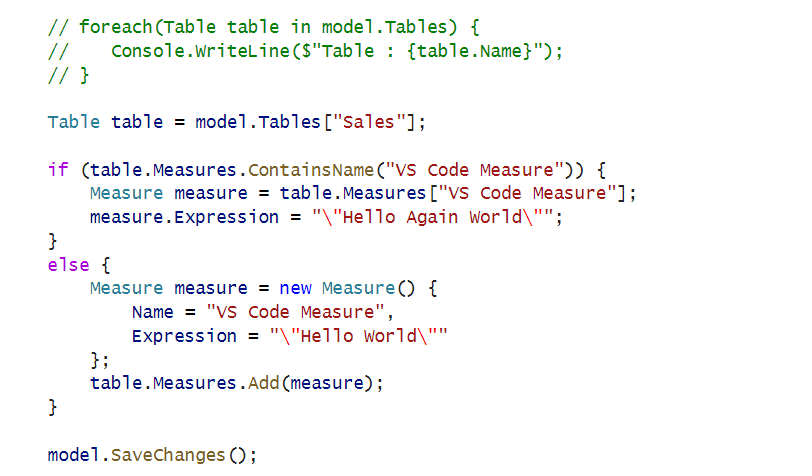
};

table.Measures.Add(measure);

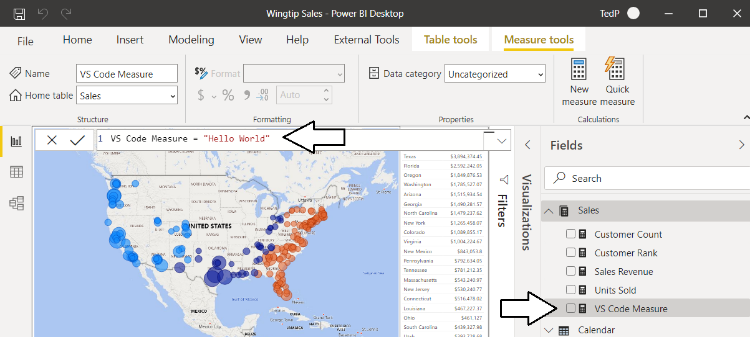
}

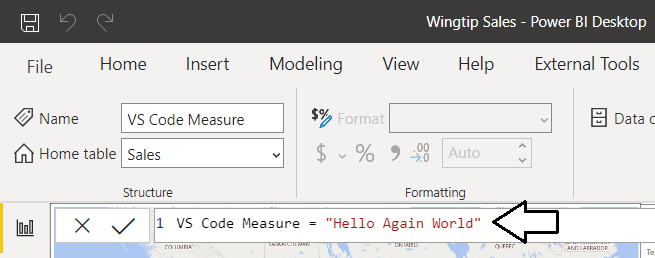
model.SaveChanges();

* 1. x



* 1. x





* 1. x