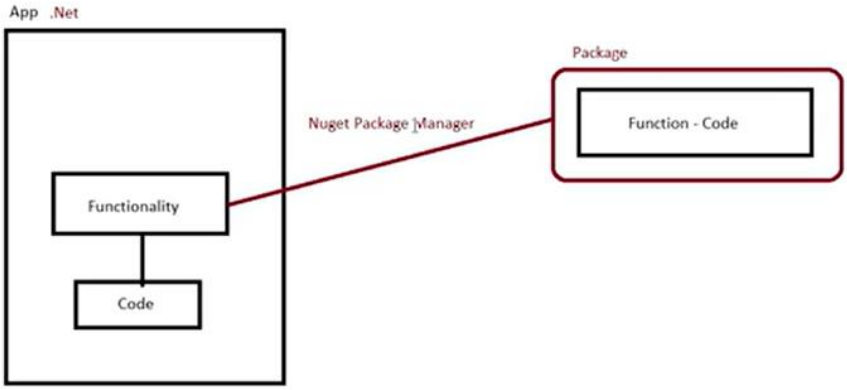
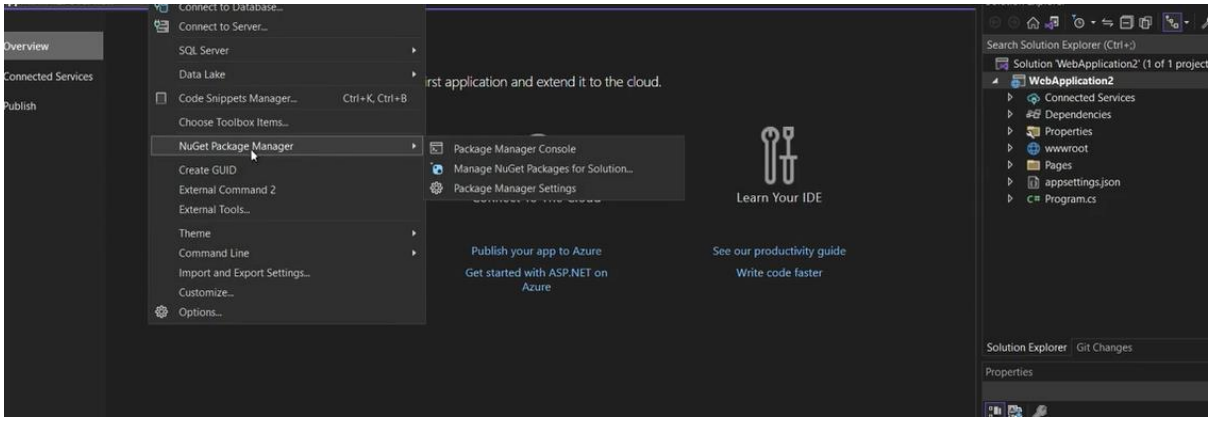
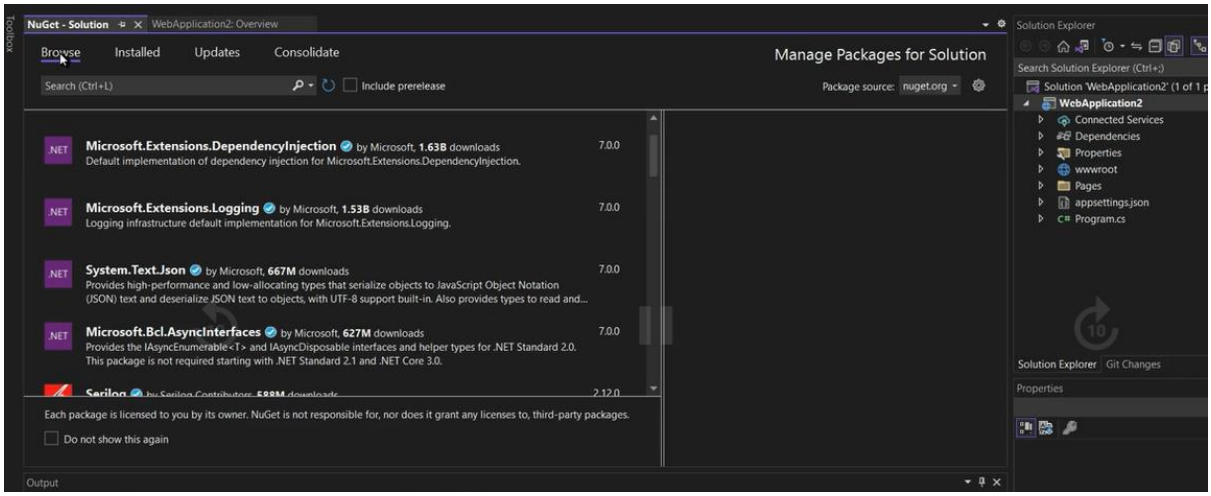
NuGet Package Management

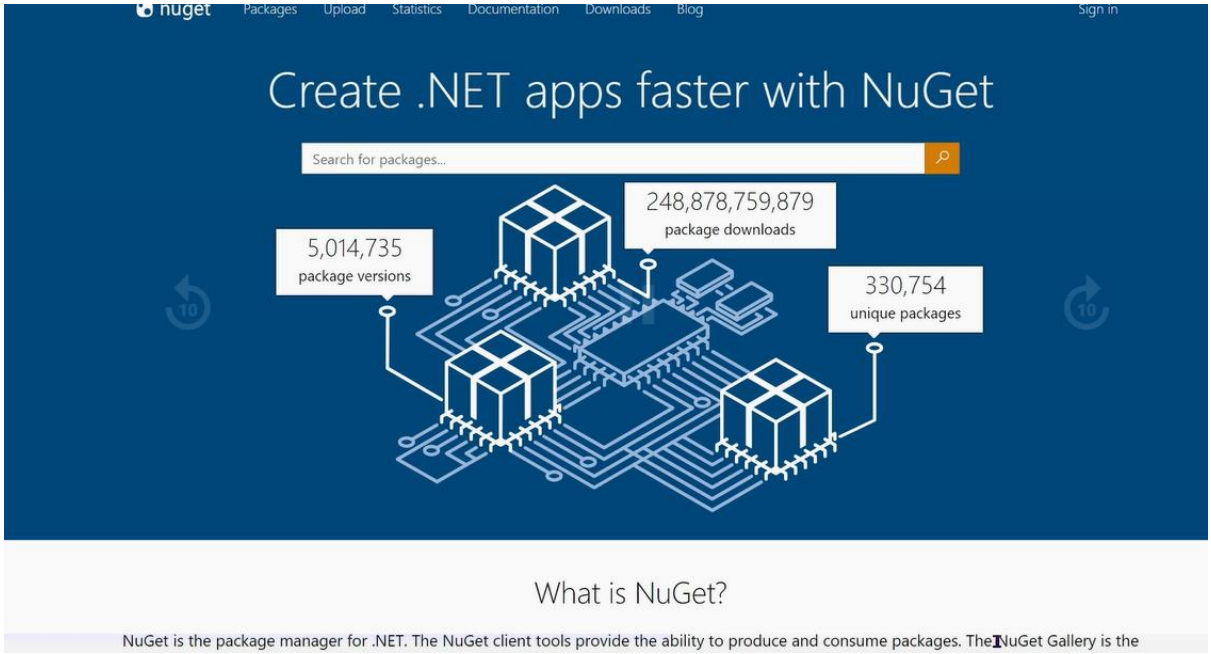


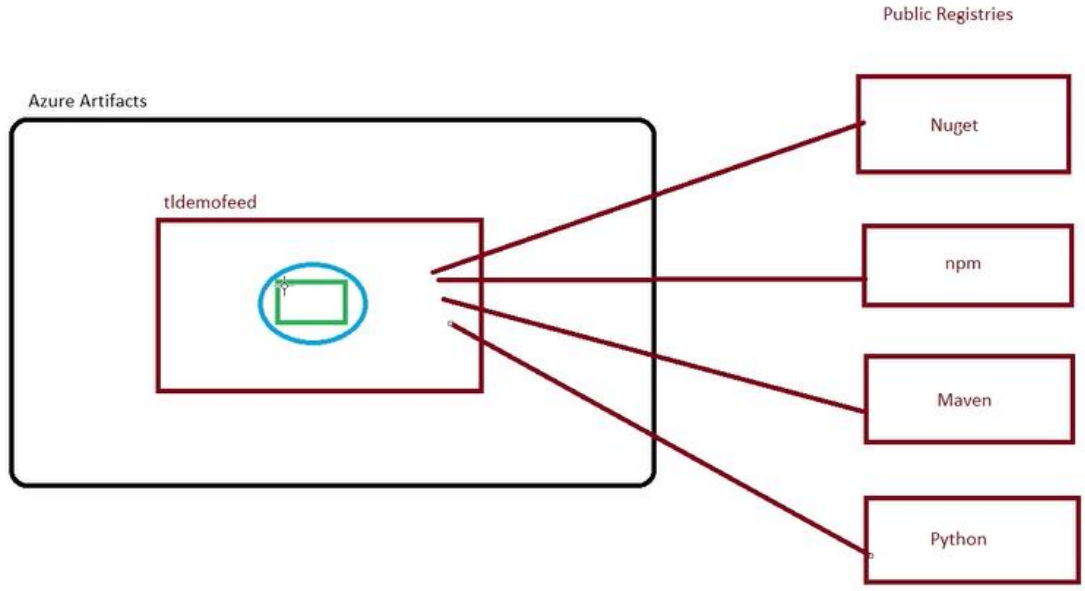
Under Tools You can check for NuGet Package Manager. Click on Manage NuGet Packages for Solution from where you can download NuGet Packages and get installed in your source code.





Visit the official site Nuget.org for recent updates.



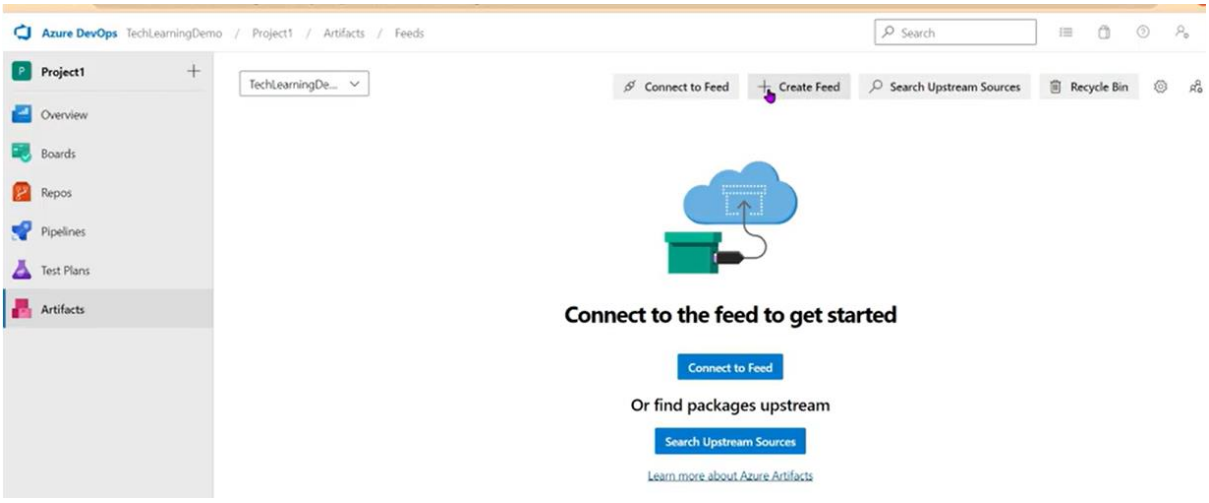


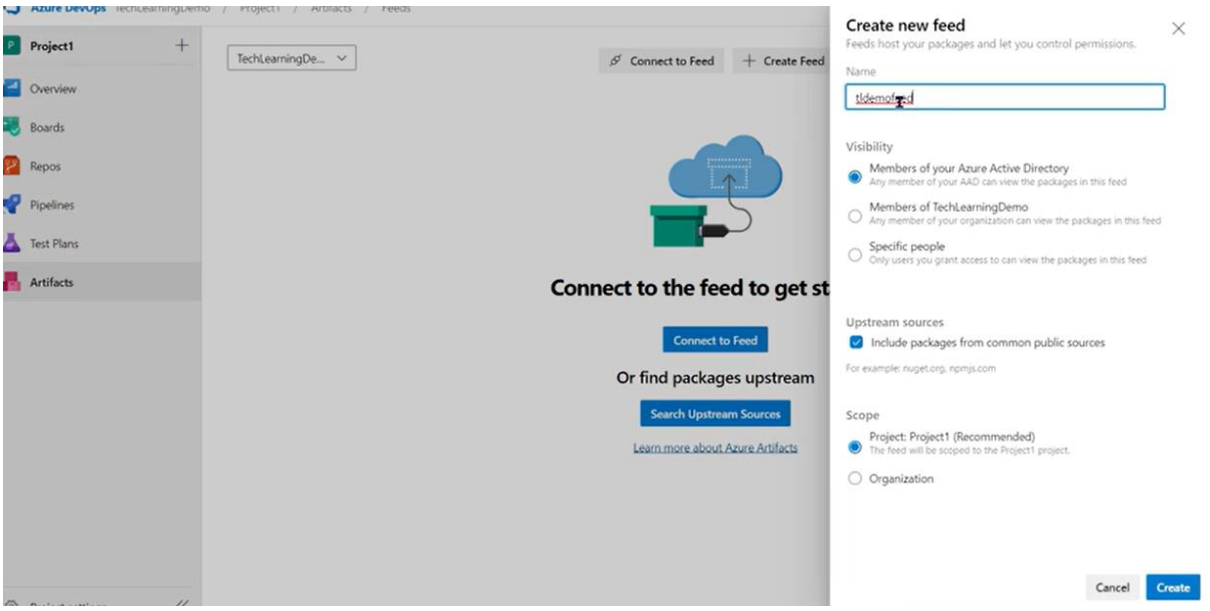
Azure Artifacts

- ✓ Azure Artifacts enables developers to share their code efficiently and manage all their packages from one place.
- ✓ With Azure Artifacts, developers can publish packages to their feeds and share it within the same team, across organizations, and even publicly.
- ✓ Developers can also consume packages from different feeds and public registries such as NuGet.org or npmjs.com.
- ✓ Azure Artifacts supports multiple package types such as NuGet, npm, Python, Maven, and Universal Packages.

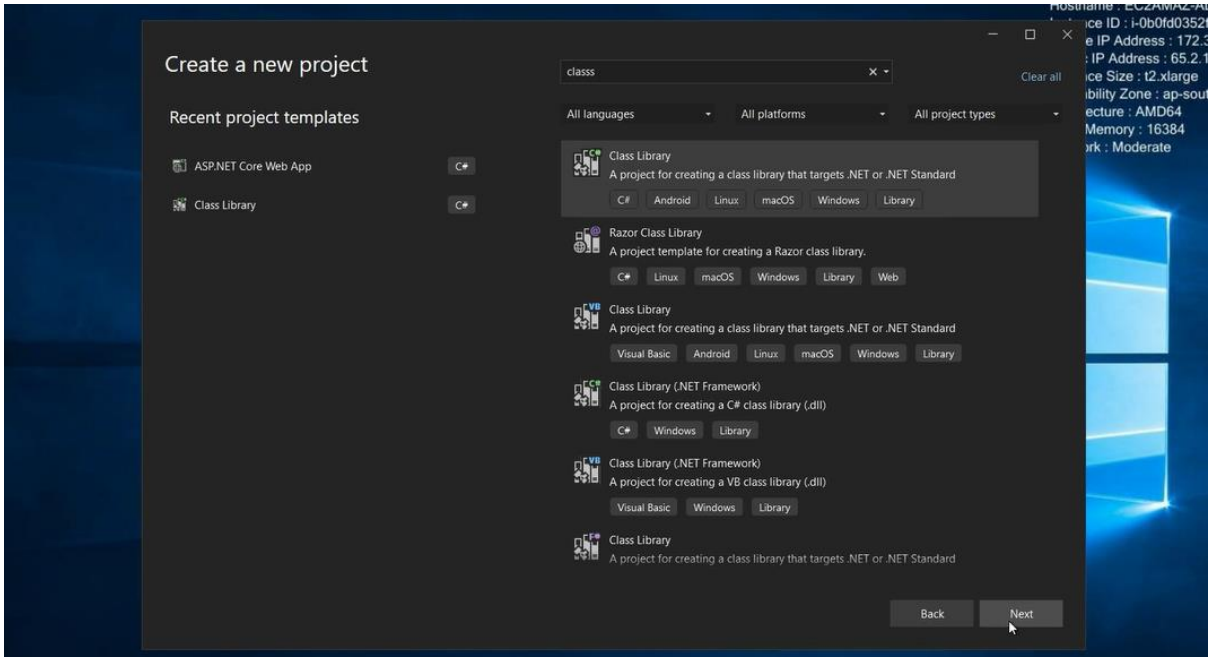
Publishing Packages inside Feeds

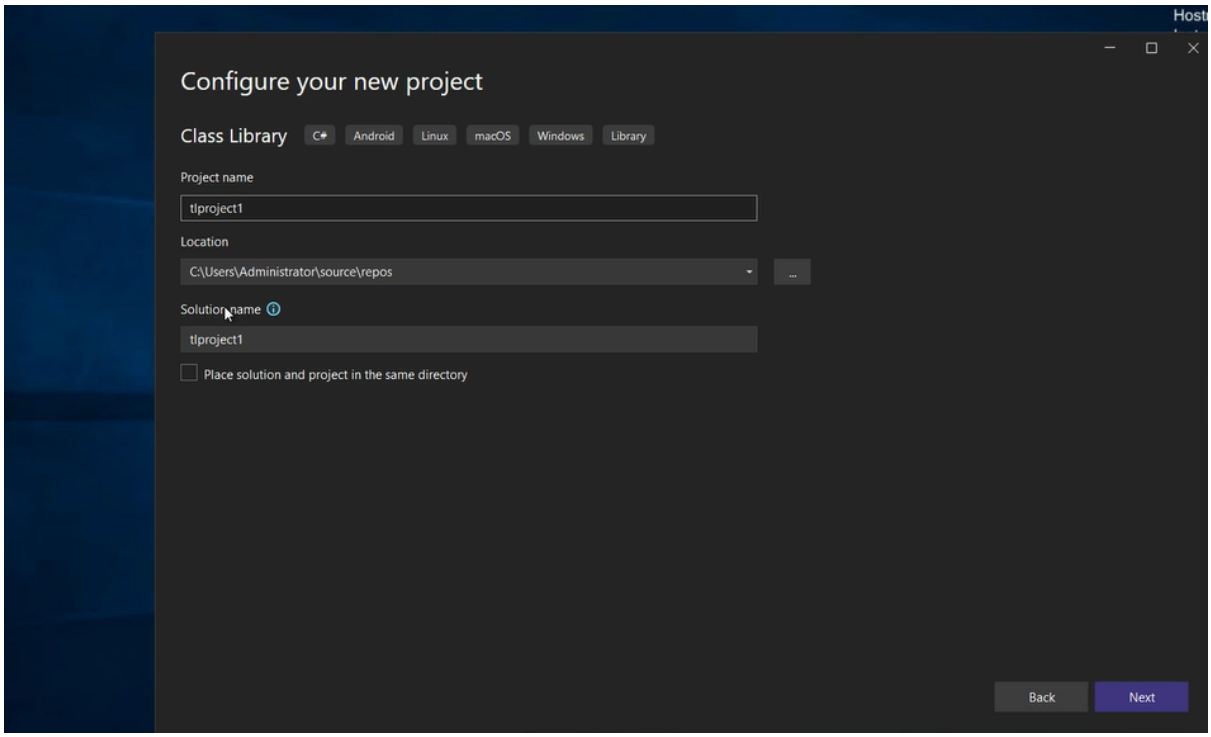
Create a Project in Azure DevOps Portal. Under Artifacts explore the options- create feed and connect to feed.

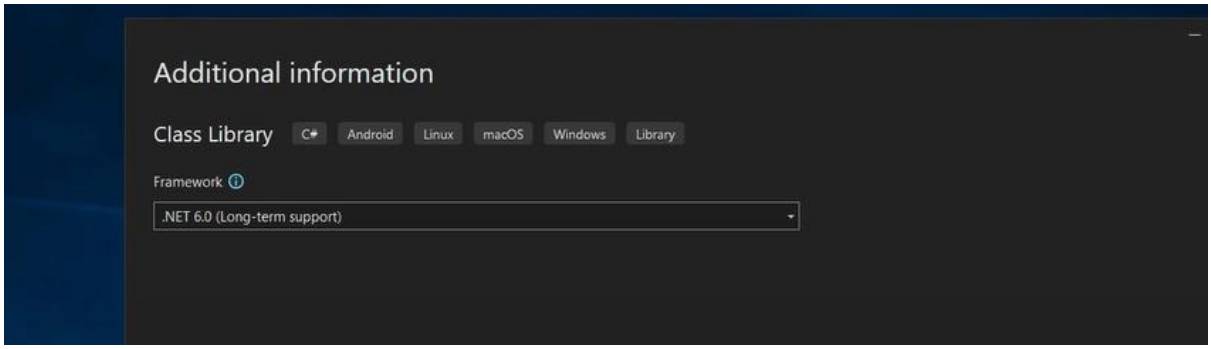


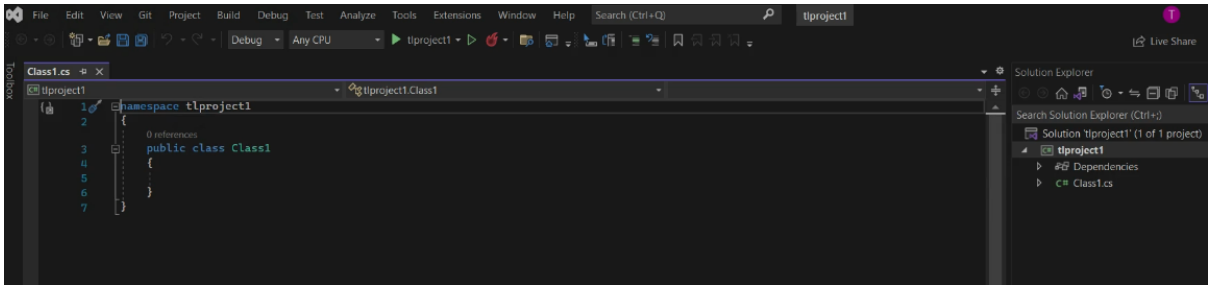


Create a Class Library in Visual Studio.

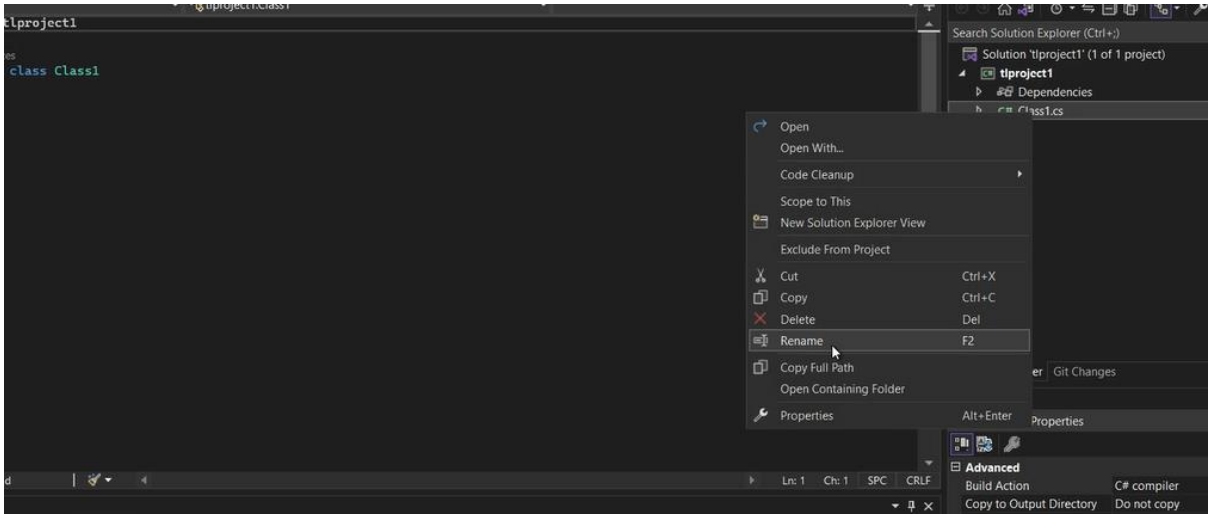


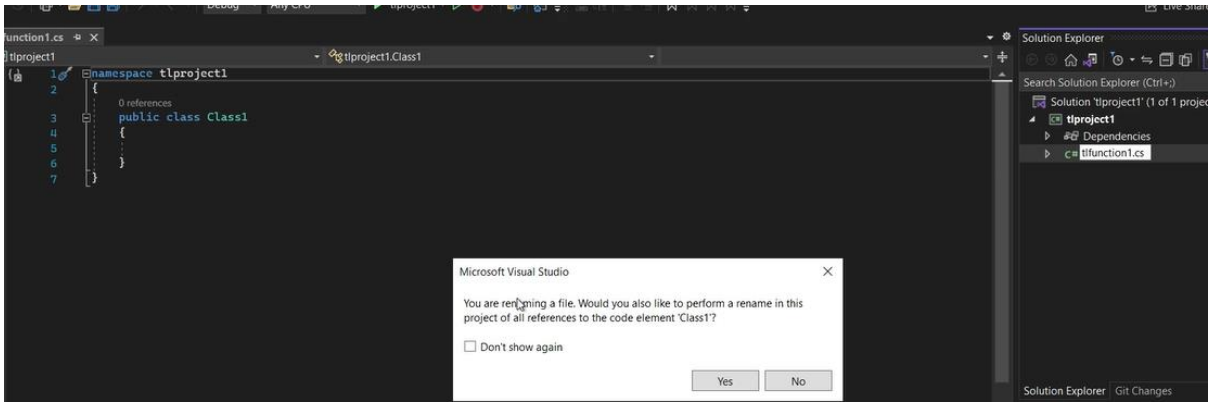




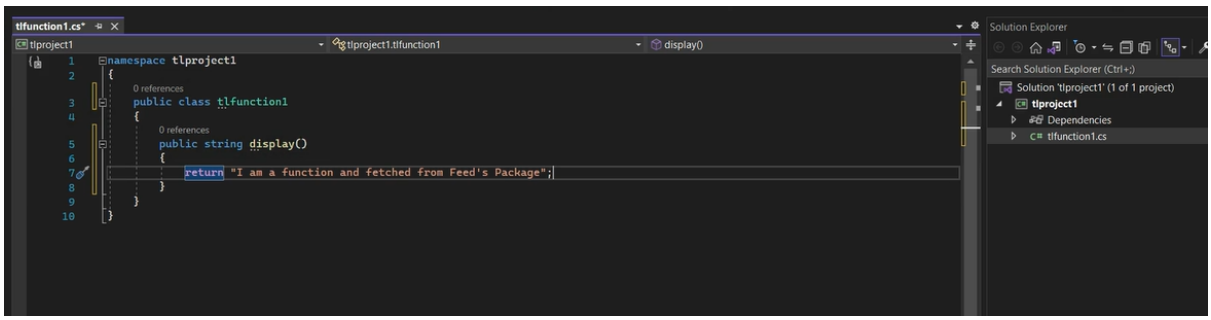


Rename the file Class1.cs to tlfunction1.cs





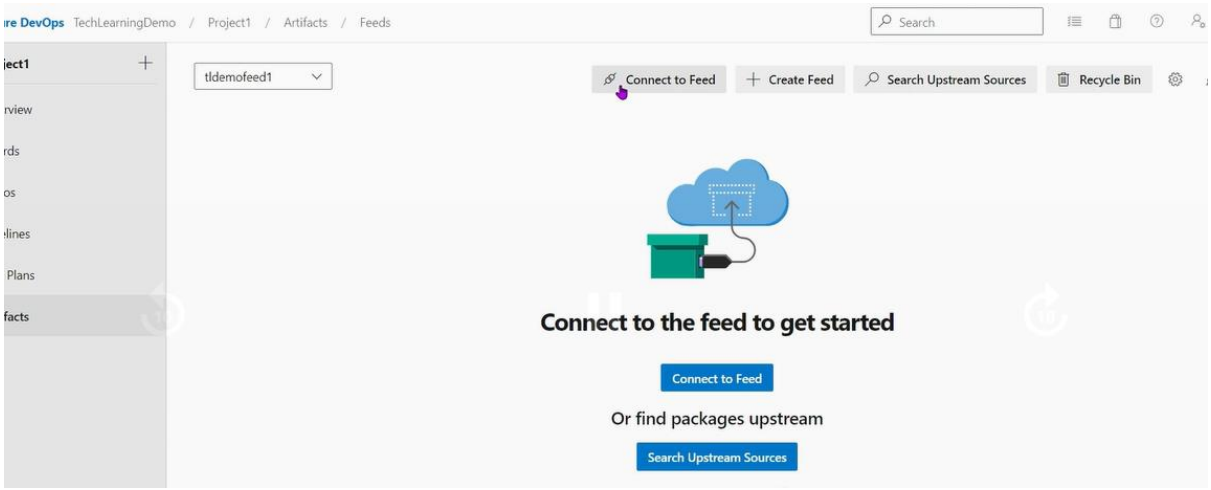
Modify the code as per screenshot and save the file.

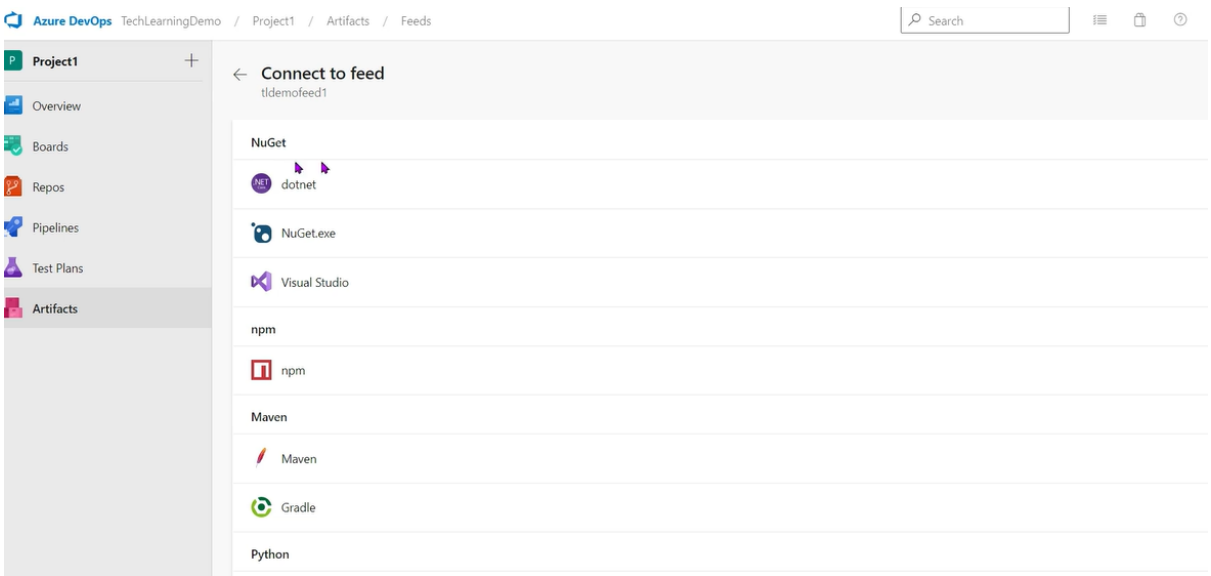


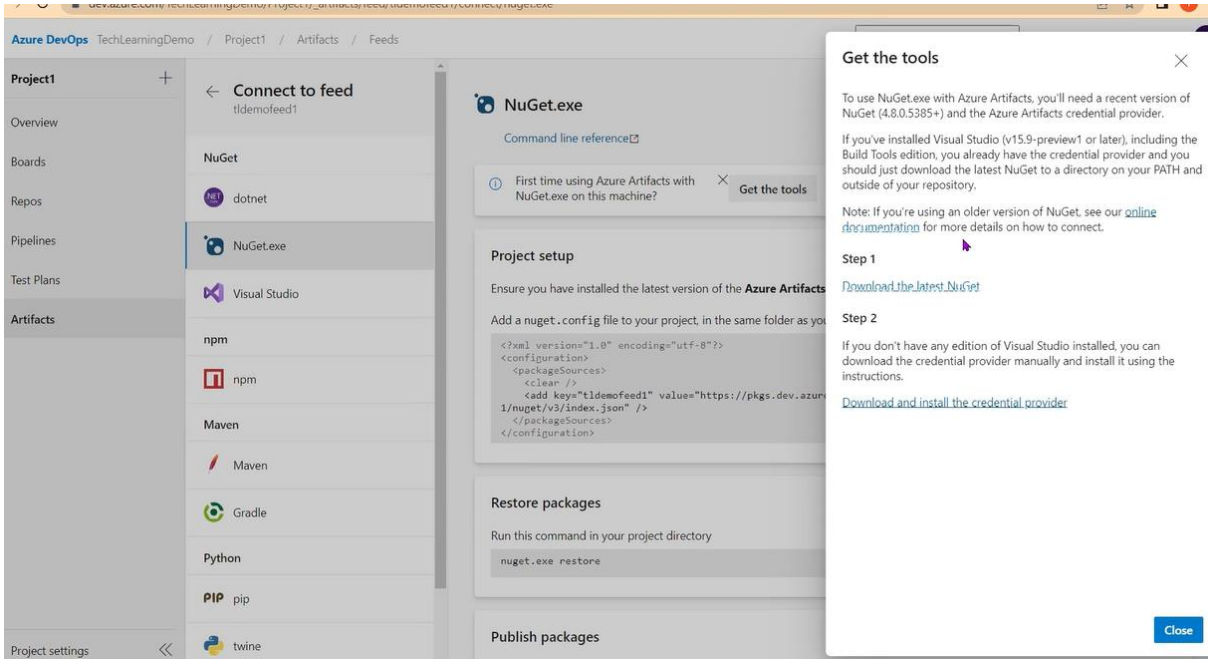
namespace tlproject1

```
{
 public class tlfunction1
 {
  public string display()
   {
    return “I am a function and fetched from Feed’s Package”;
   }
}
}
```

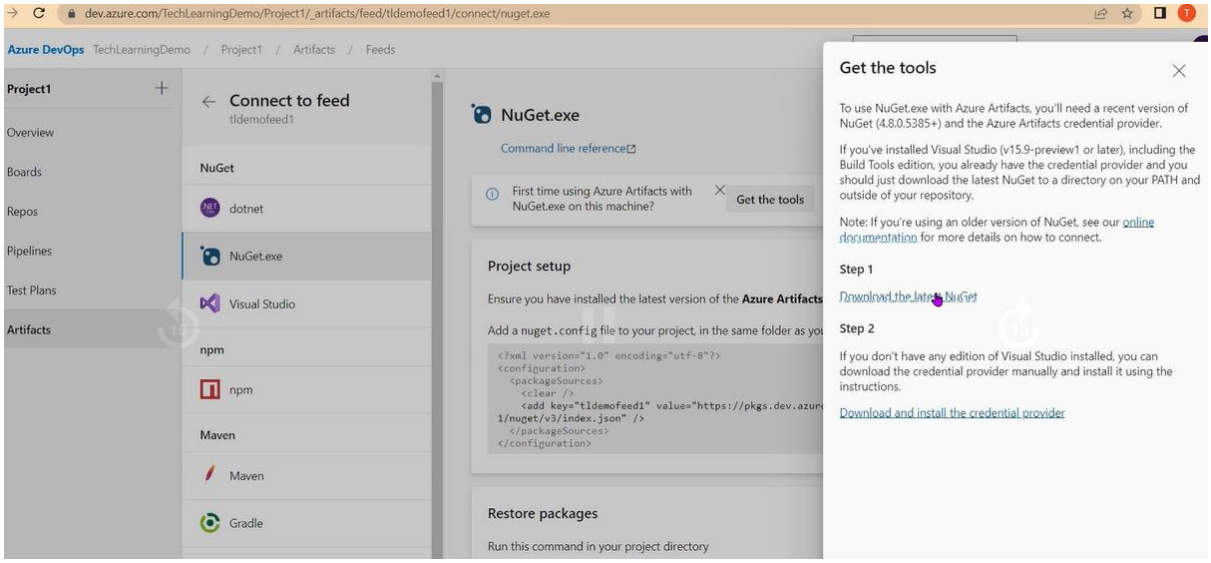
On Azure DevOps Portal connect to Feed, click on Nuget.exe

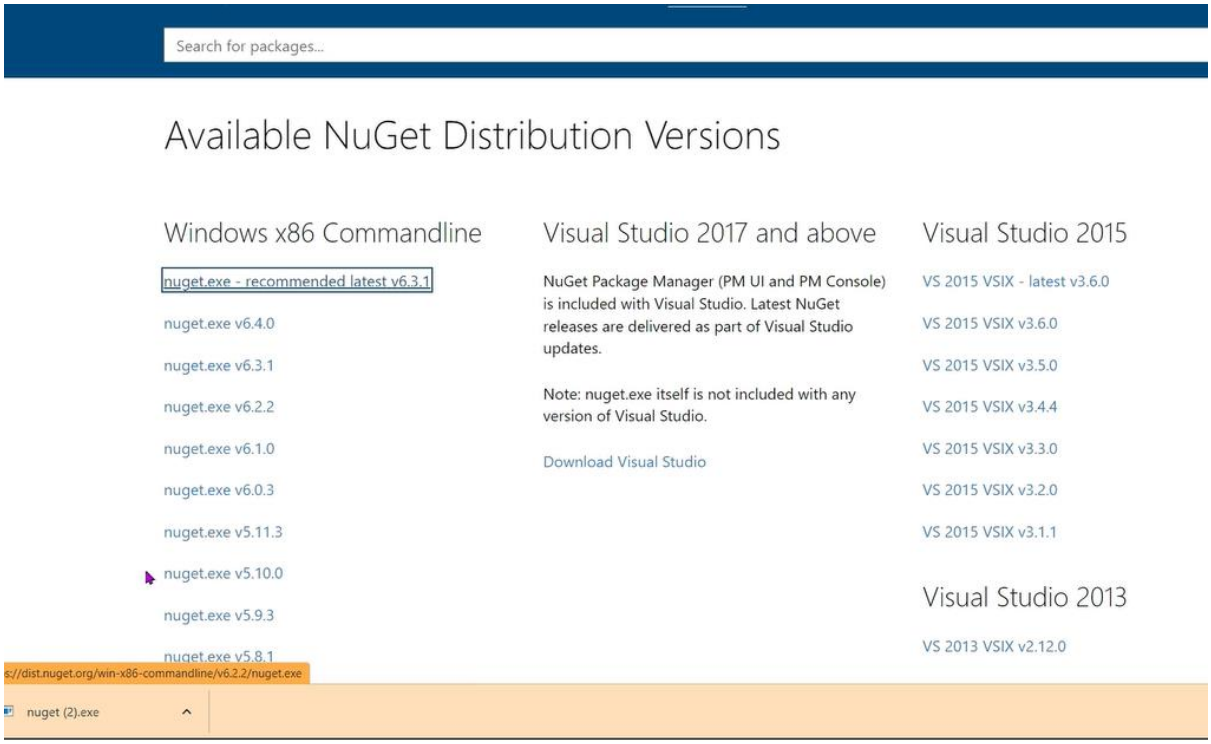




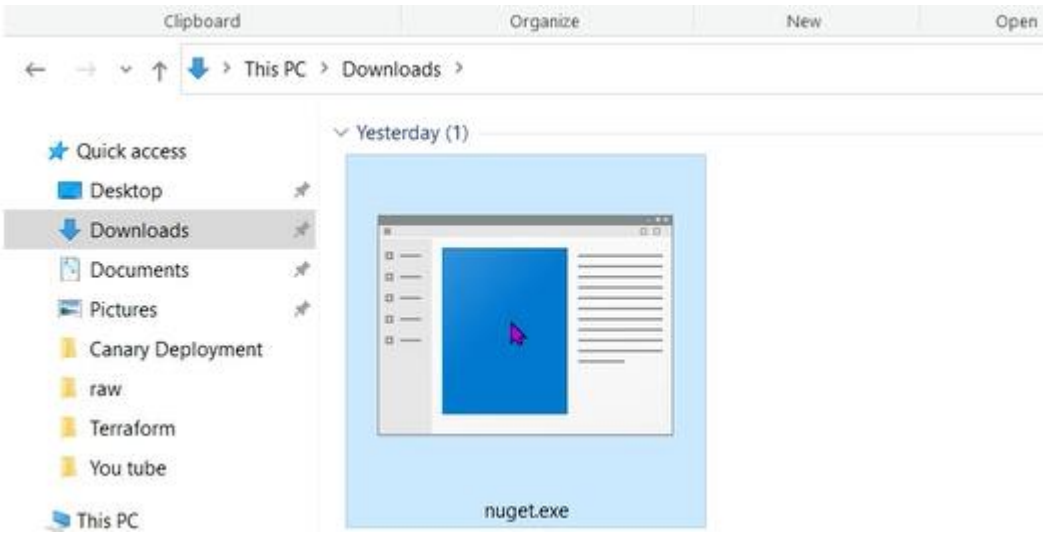


Click on Get the Tools and download the latest version for NuGet.

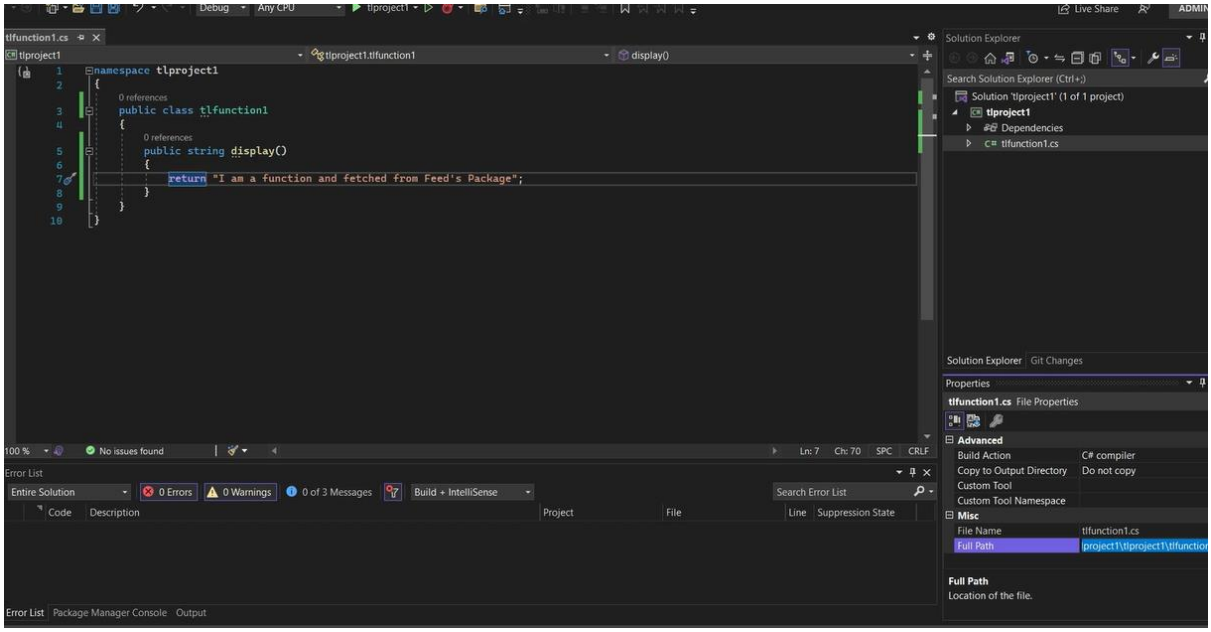


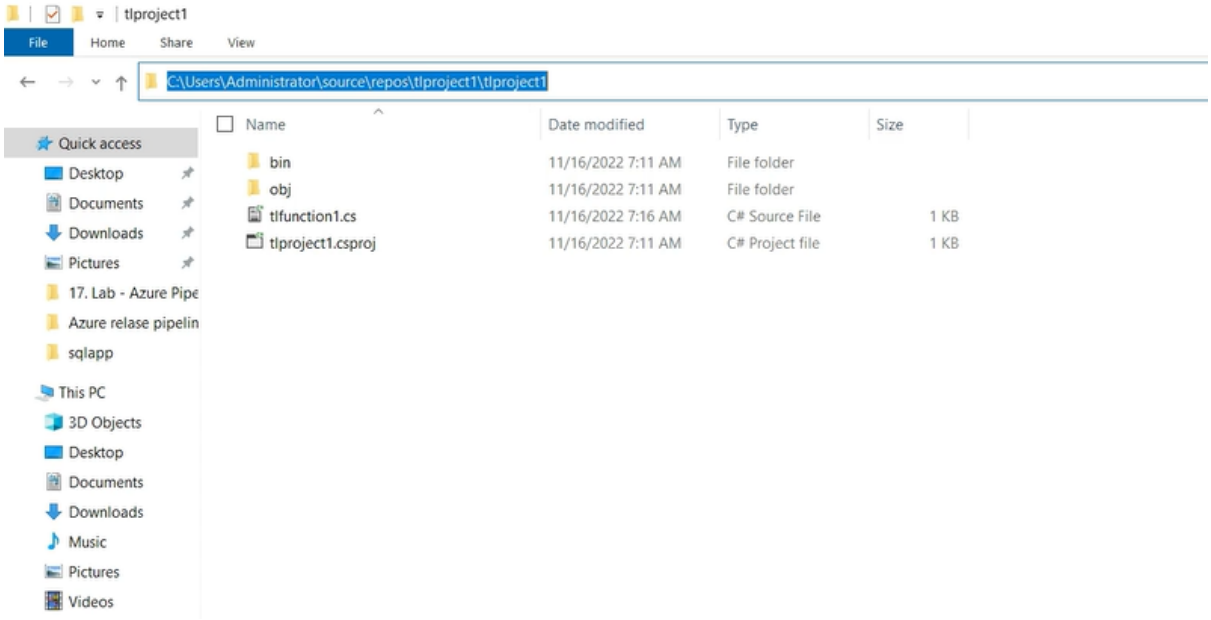


Check in Downloads

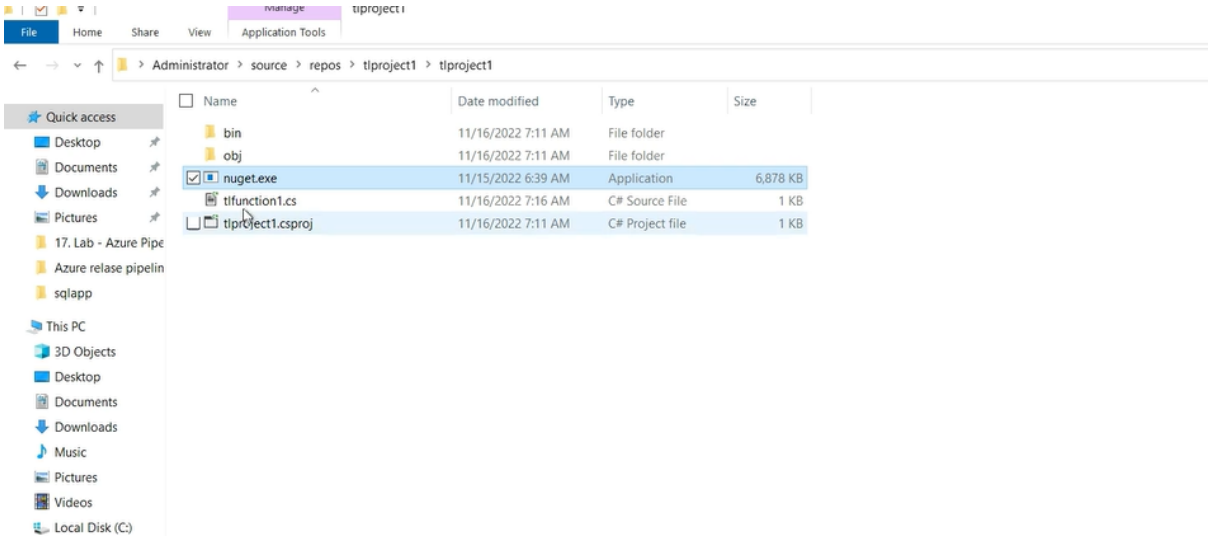


Select Full Path of your project from visual studio and paste in file explorer.

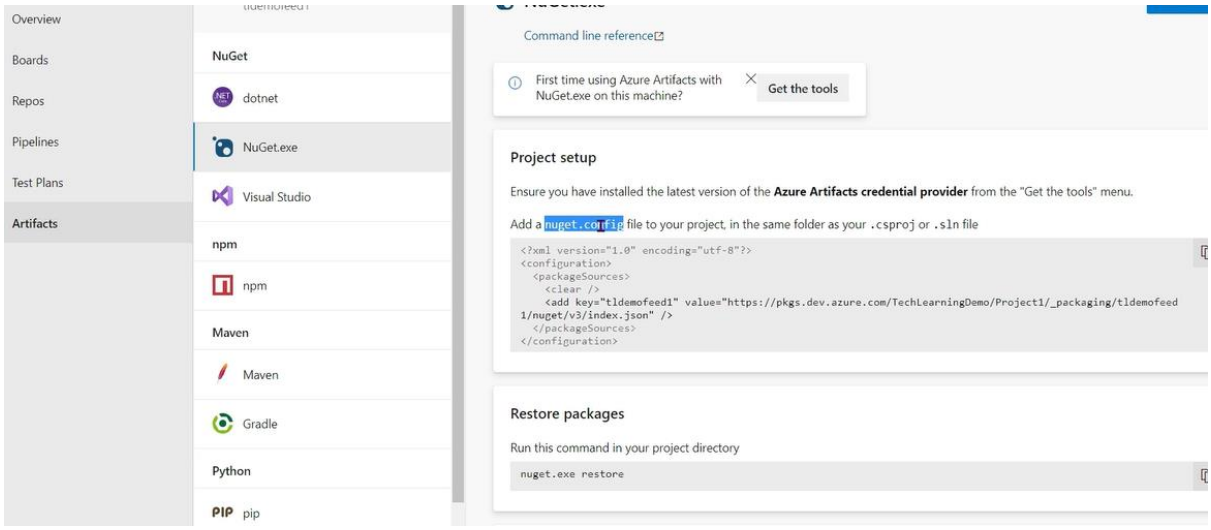


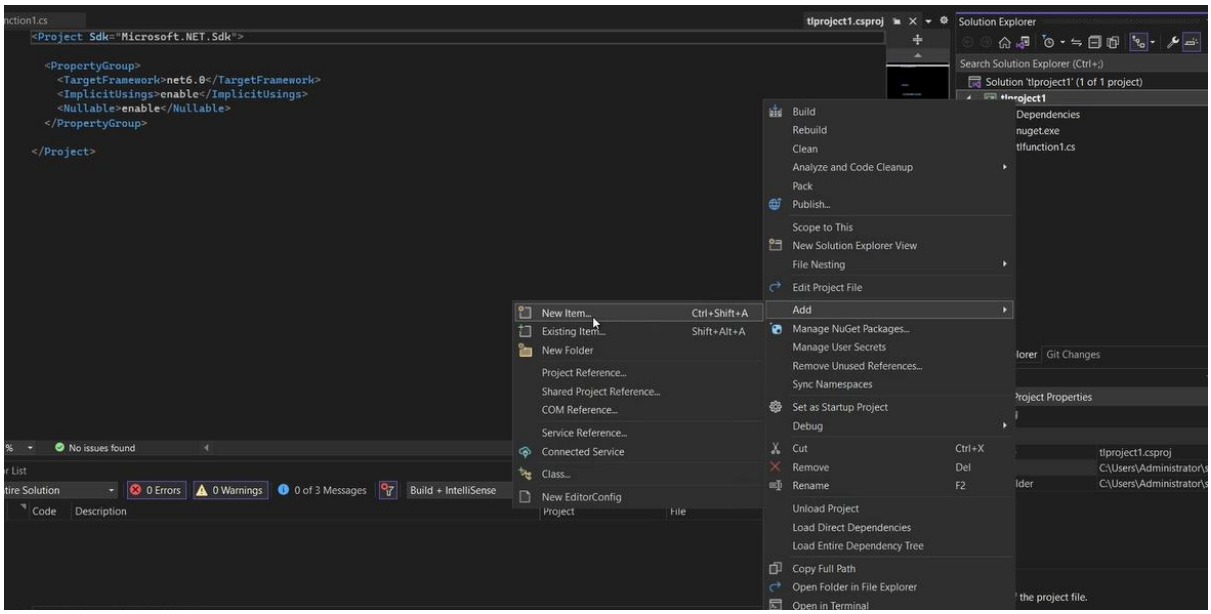


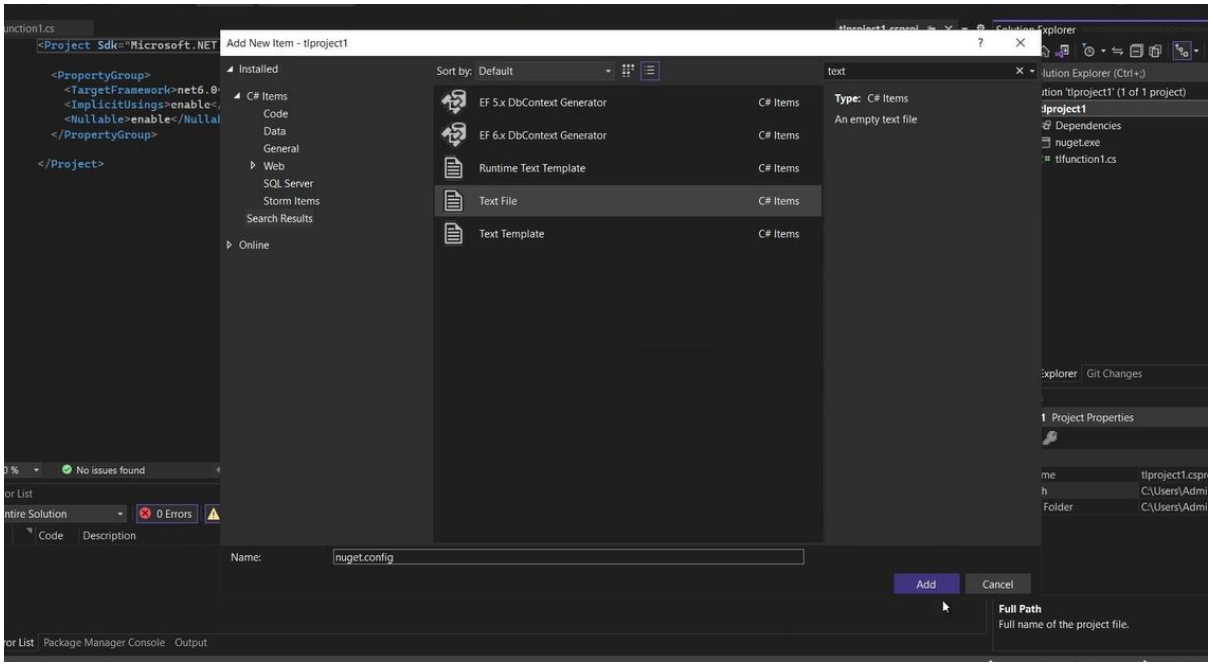
Paste nuget.exe in the same path from Downloads

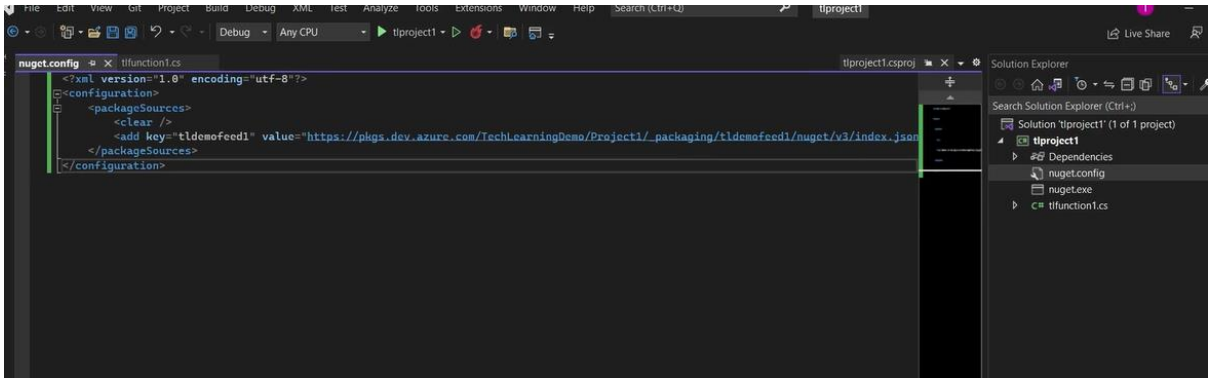


Add Nuget.config file to the Project

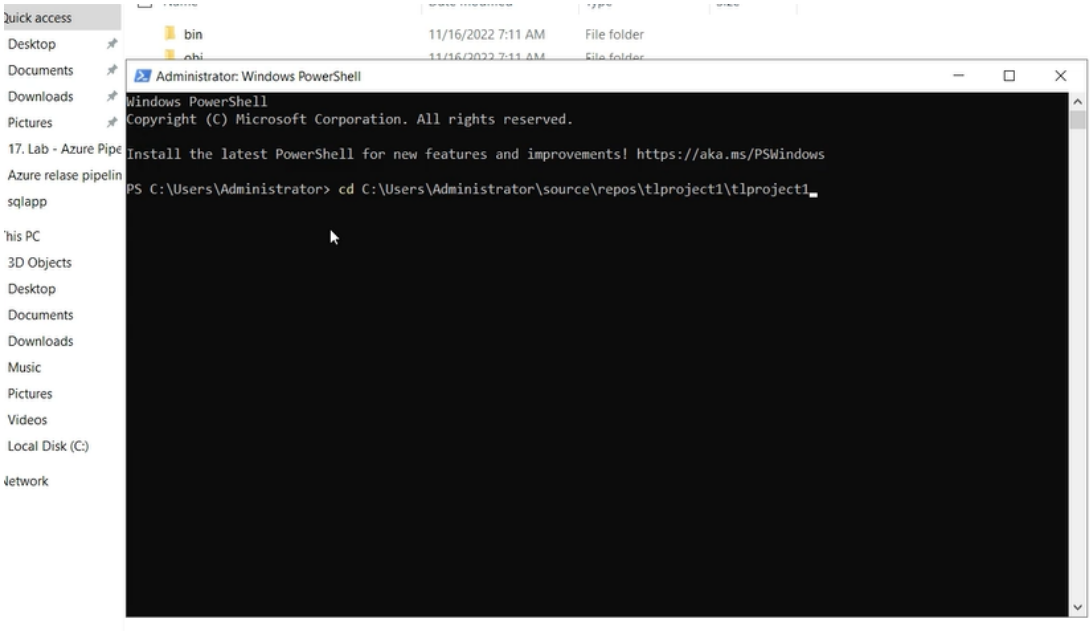


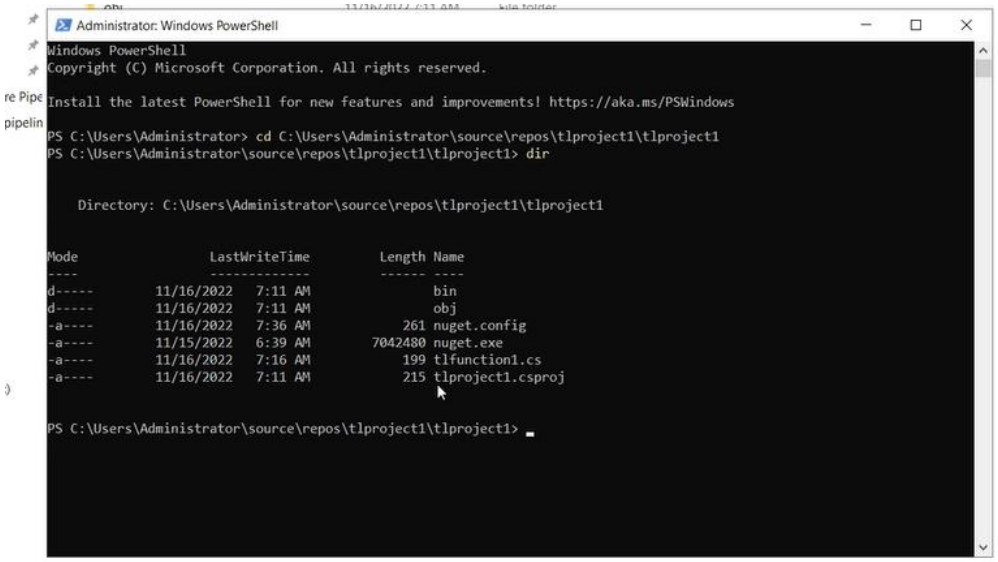






Build your solution locally. Open Windows PowerShell and navigate to the path and list the files and directories.



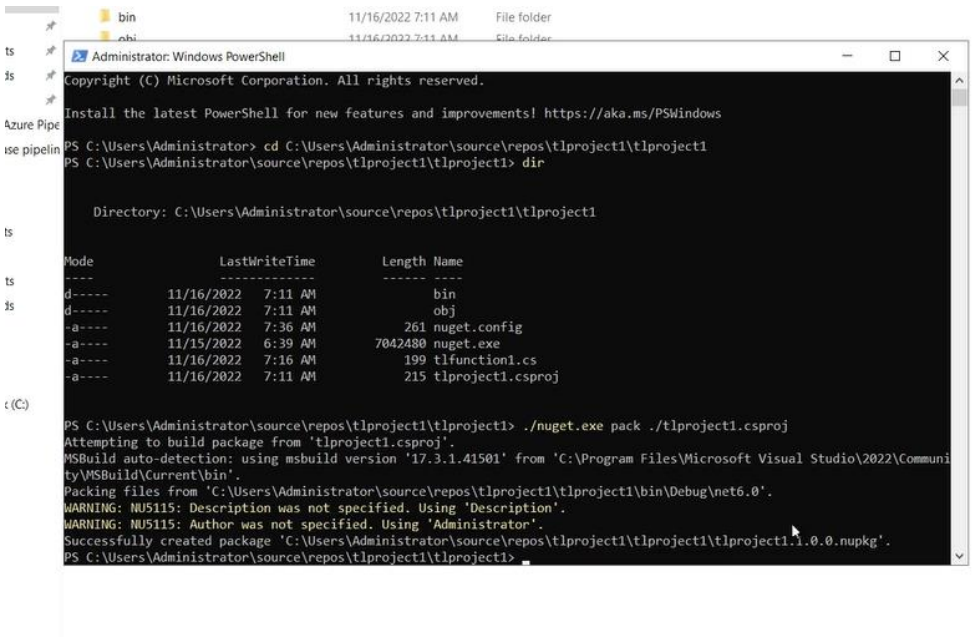


Create Packaging by running the command

./nuget.exe pack ./tlproject1.csproj

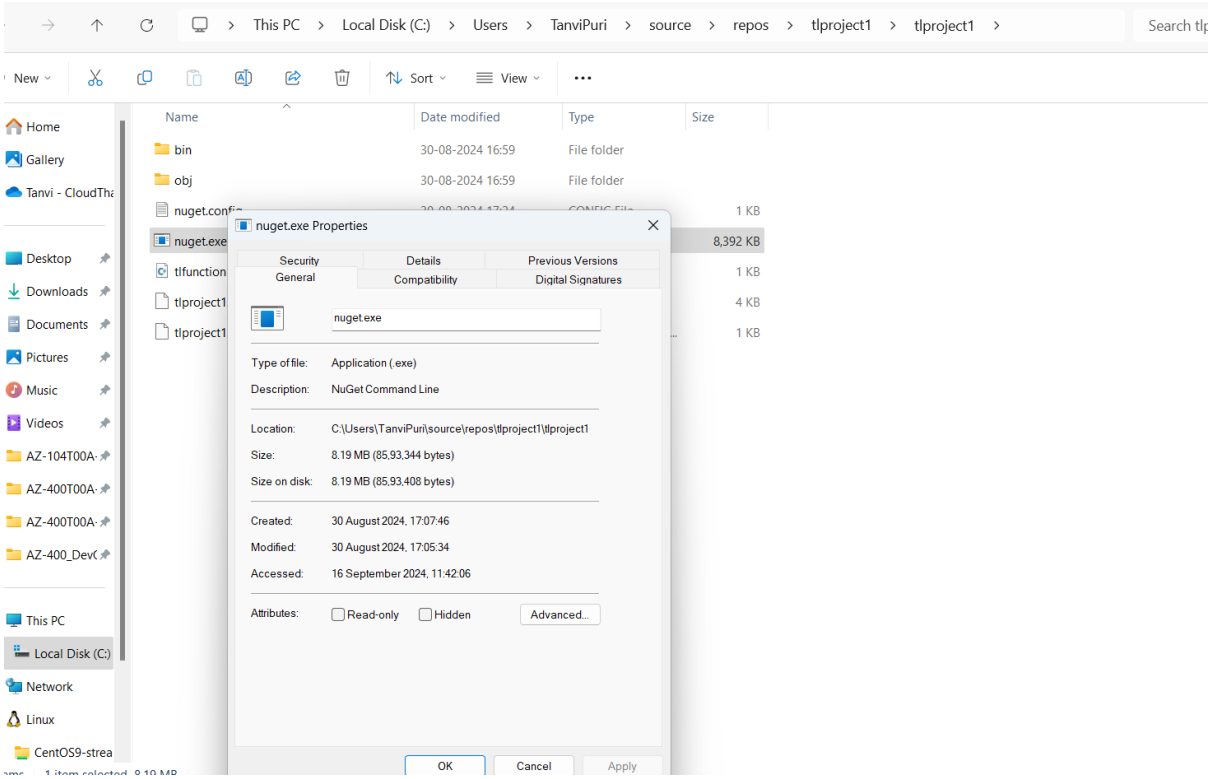
If you face any error regarding environment variables then run the below command and rerun the pack command.

$env:NUGET_ENABLE_LEGACY_CSPROJ_PACK = "true"

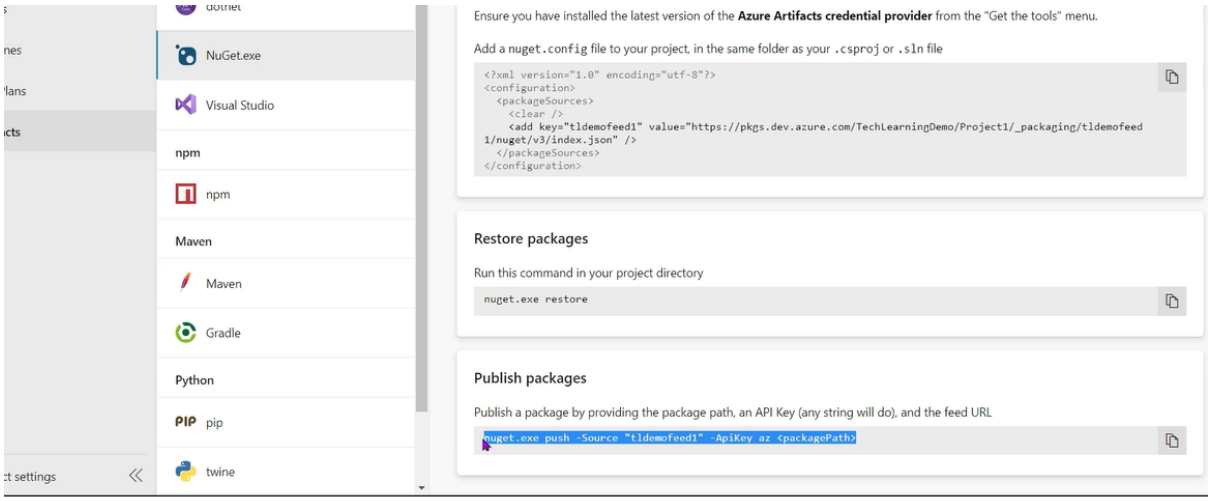


If still facing any error regarding block of nuget. Then navigate to the source project path.

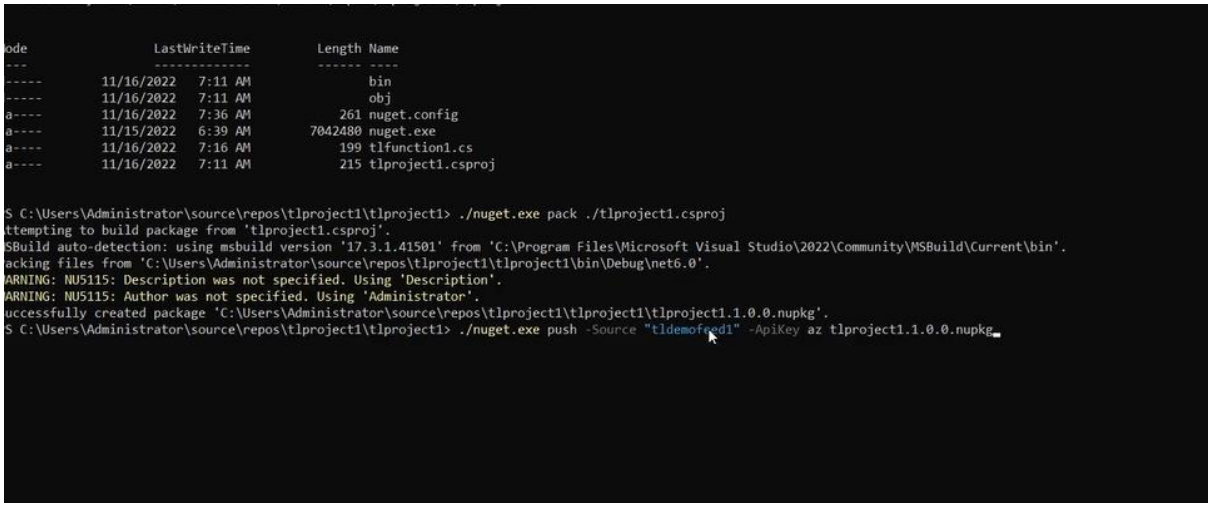
On NuGet right click the properties enable the unblock option.



Select the command to Publish Packages from DevOps Portal.



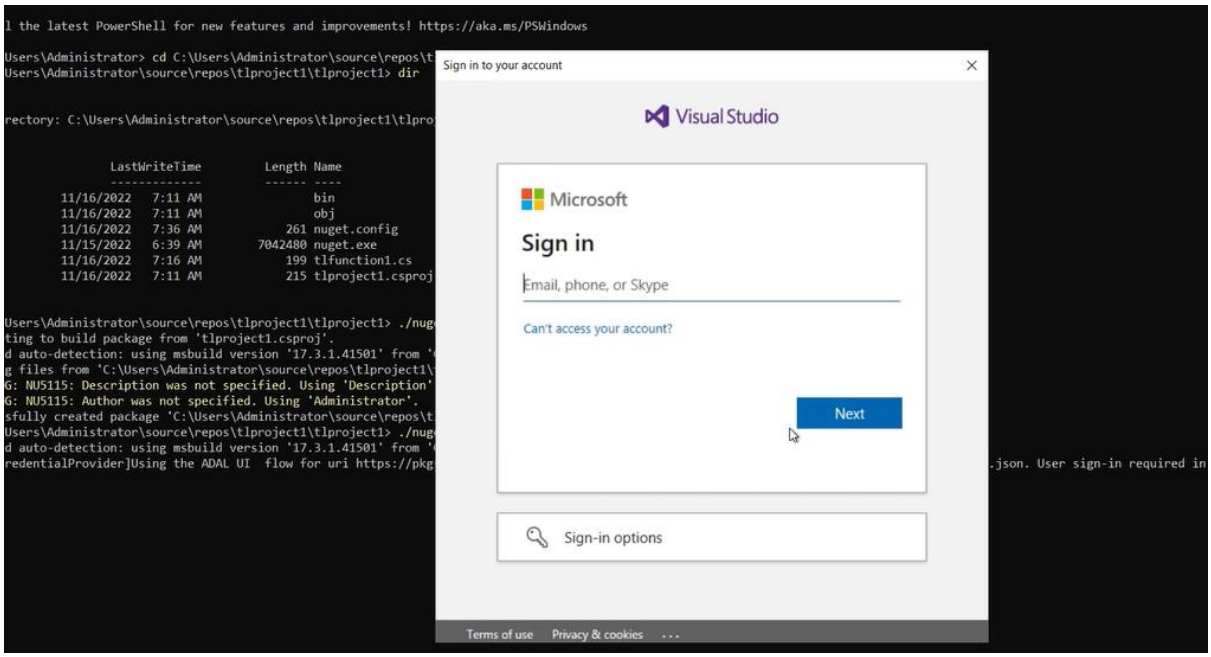
Replace the <packagePath> with your package name just created.



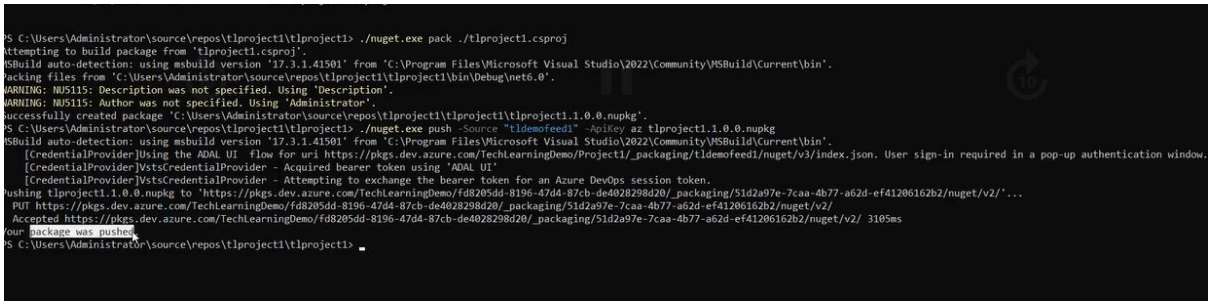
Run the command

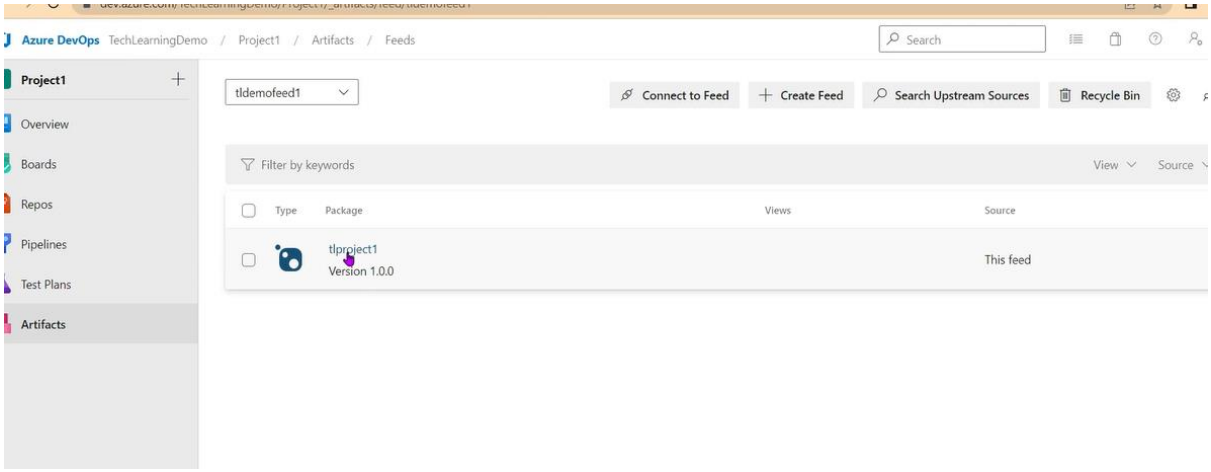
./nuget.exe push -Source “tldemofeed1” -ApiKey az tlproject1.1.0.0.nupkg

Package will be pushed and verify credentials.



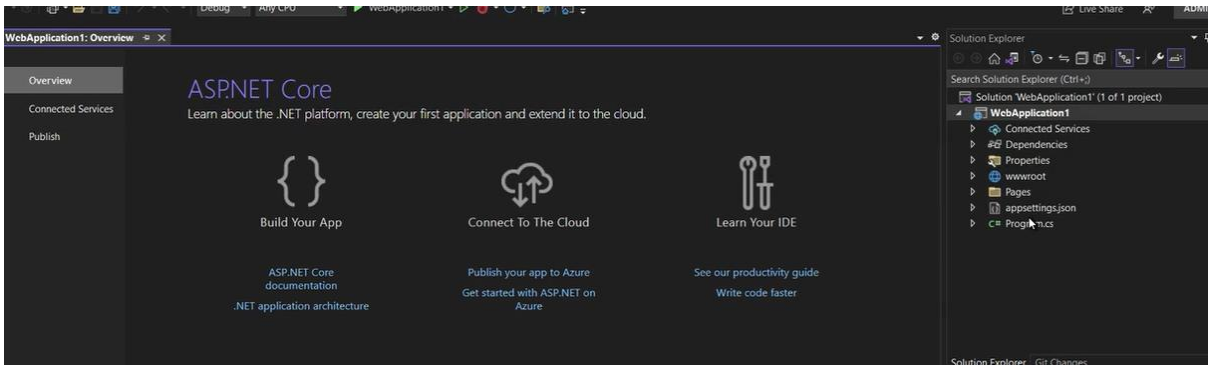
Package was pushed to Azure artifacts feed.

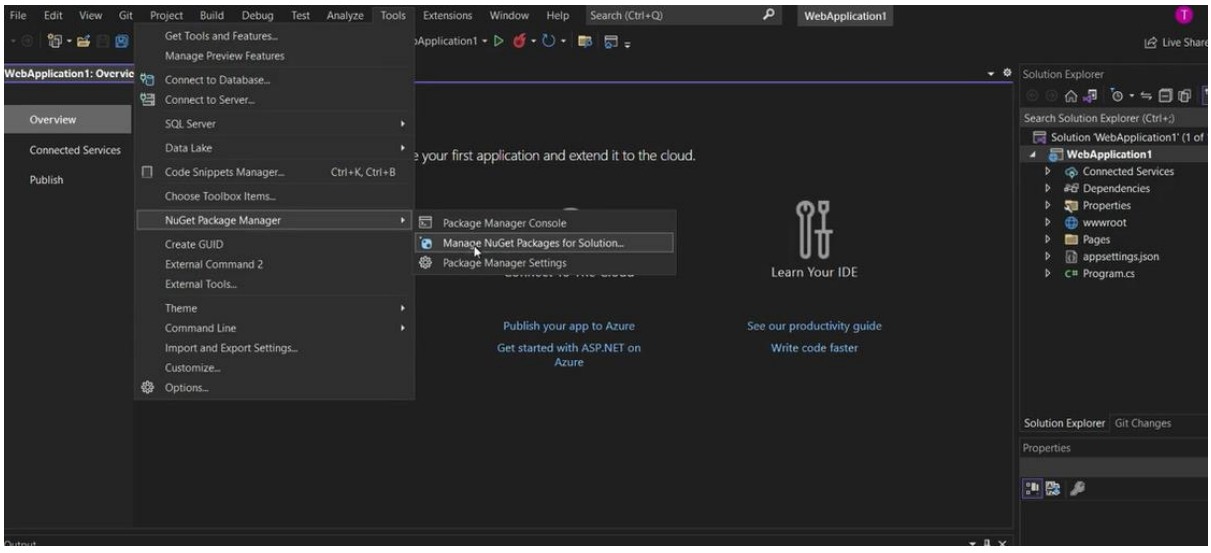




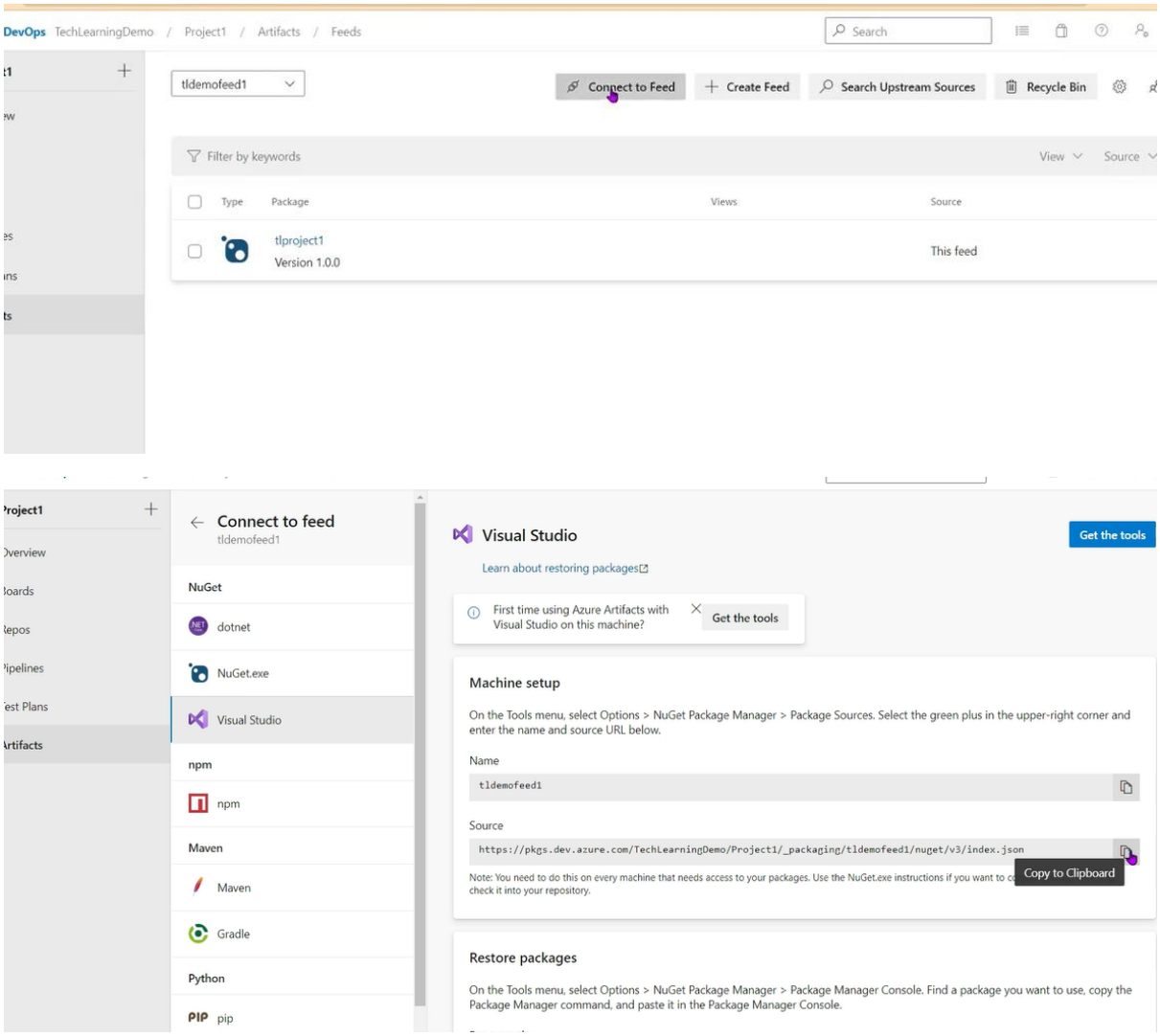
Consume Packages from Feeds

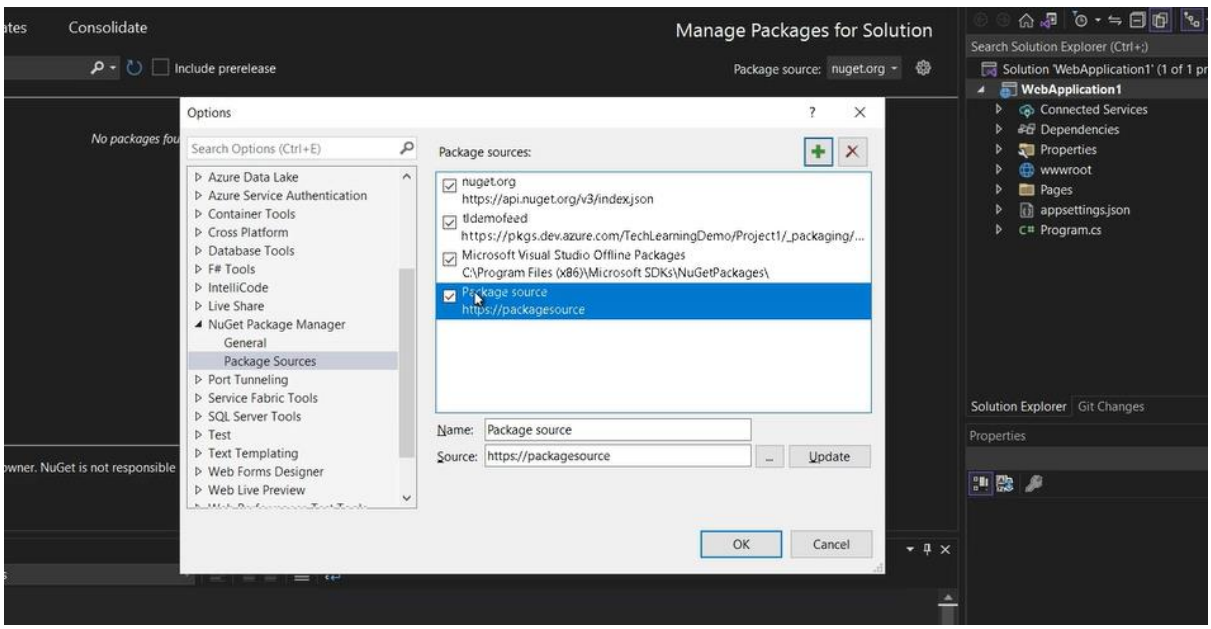
Create ASP.Net Project in Visual Studio



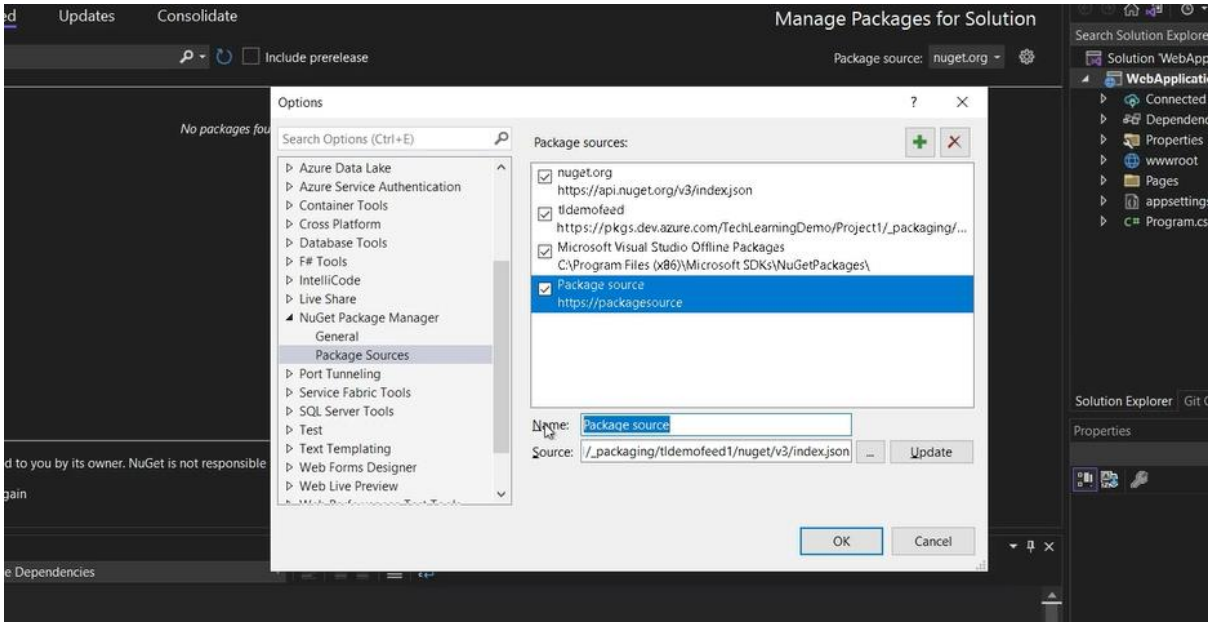


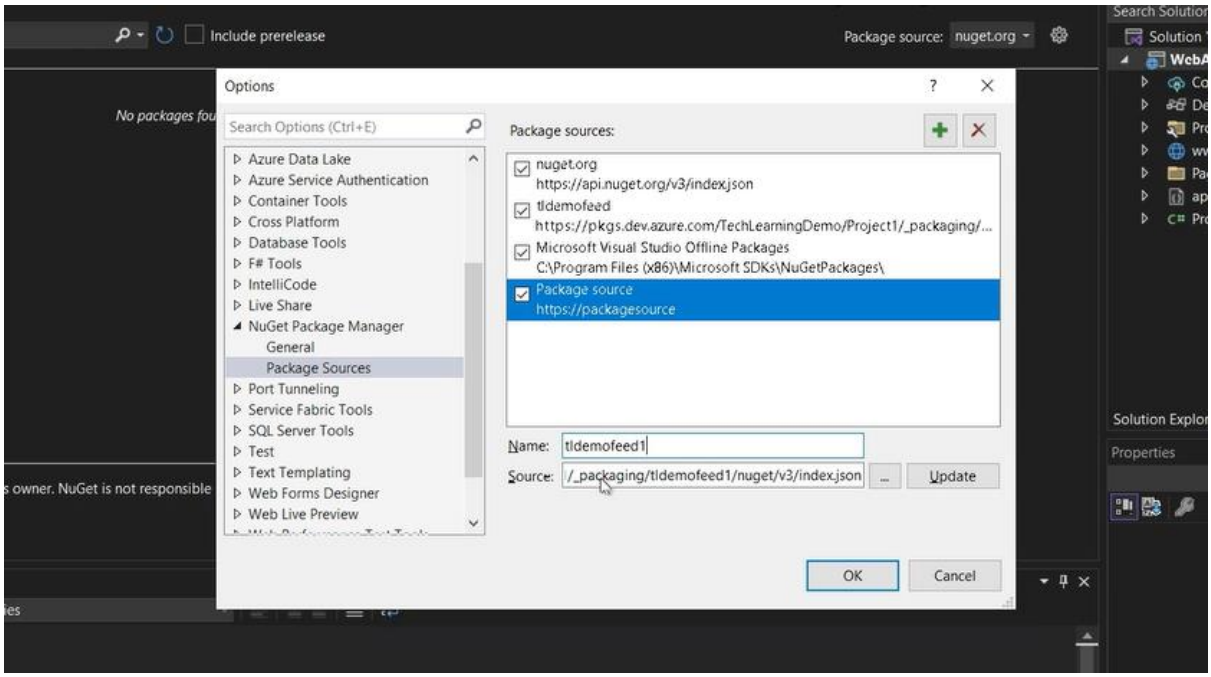
Connect to Feed and click Visual Studio

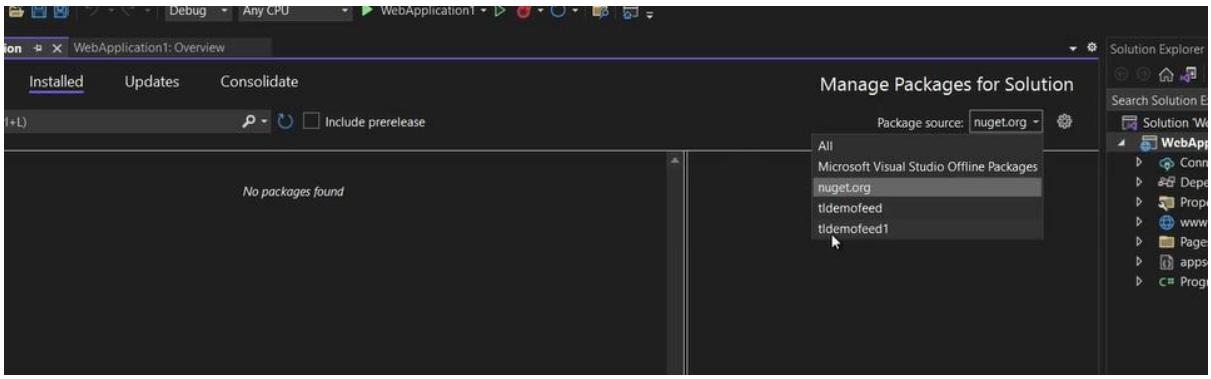




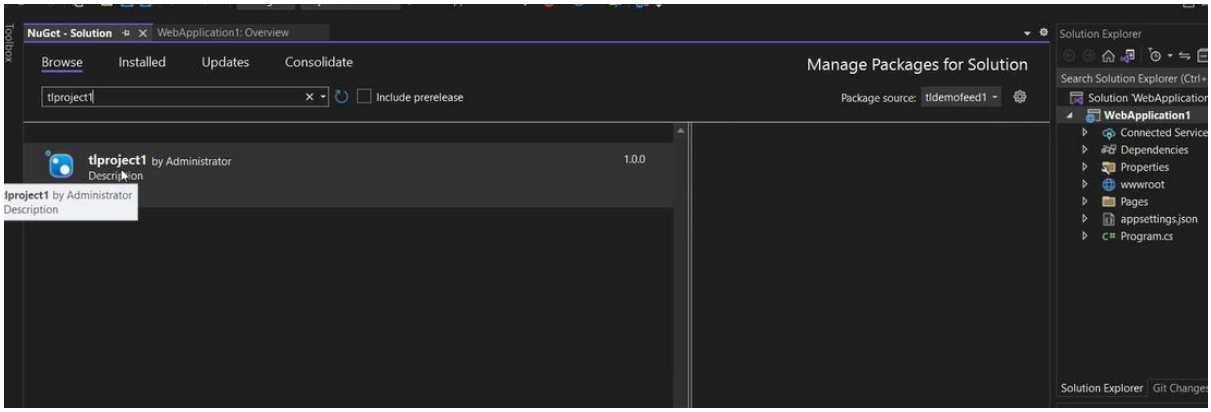
Update the name

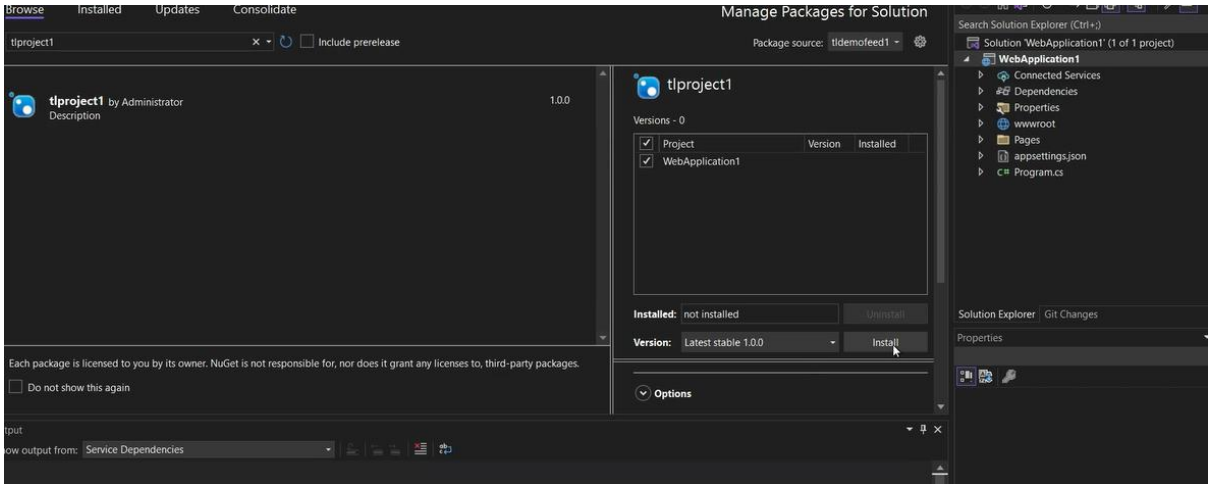


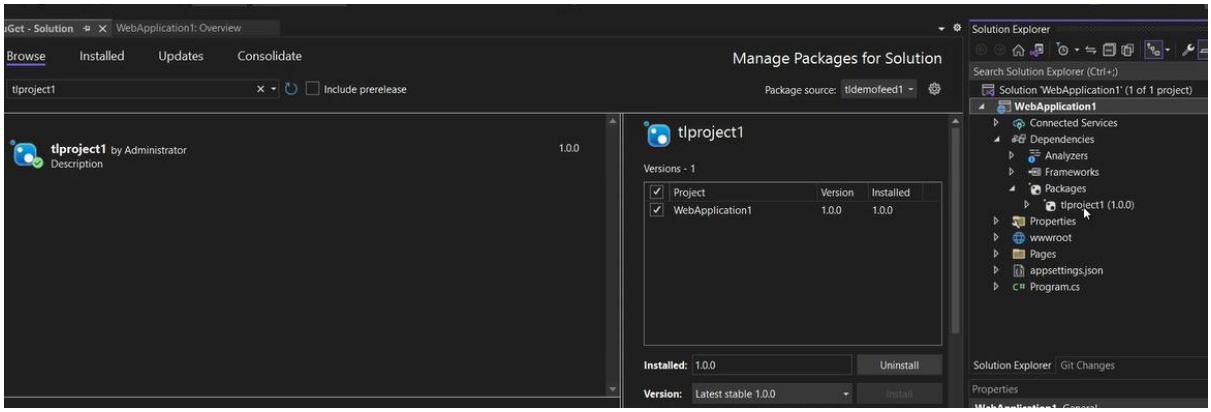




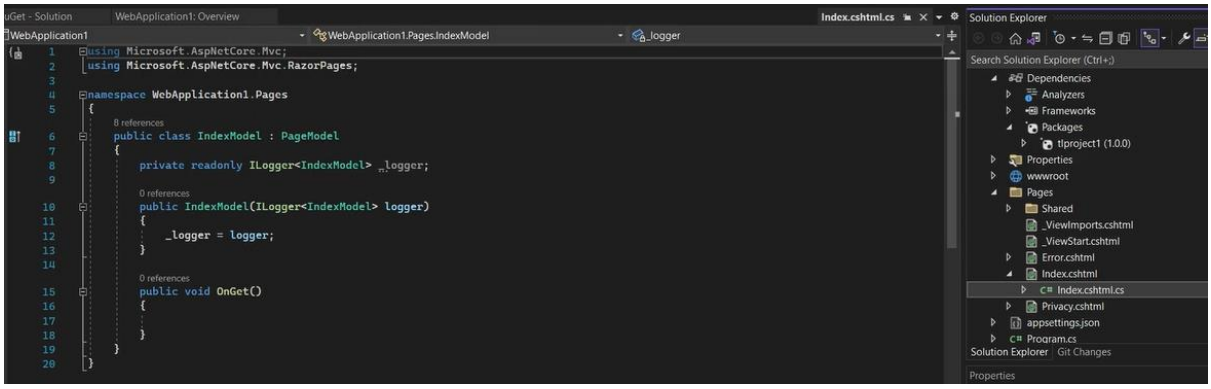
Browse the package and install in your project.

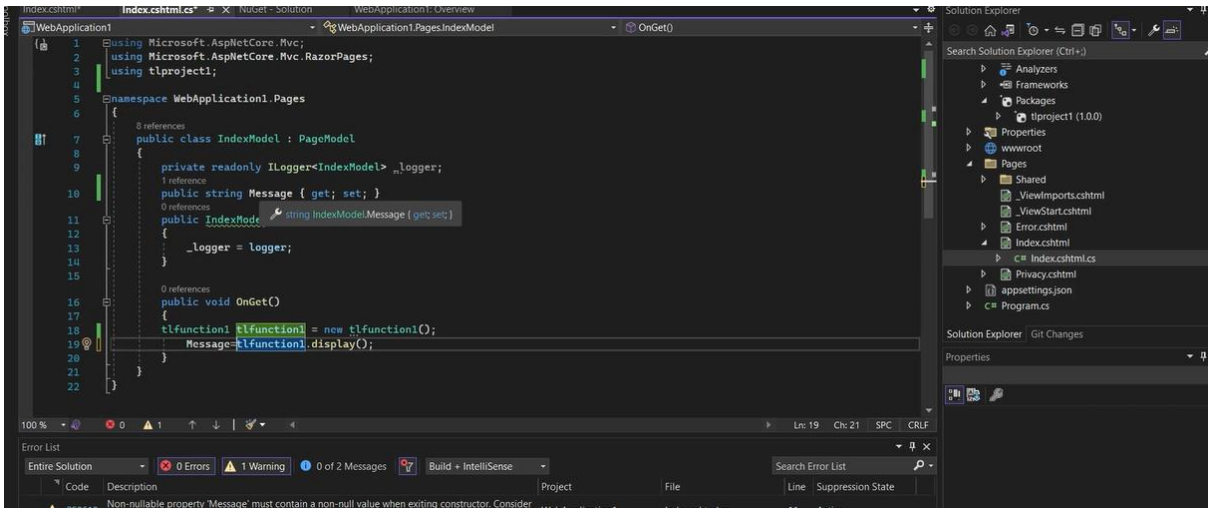




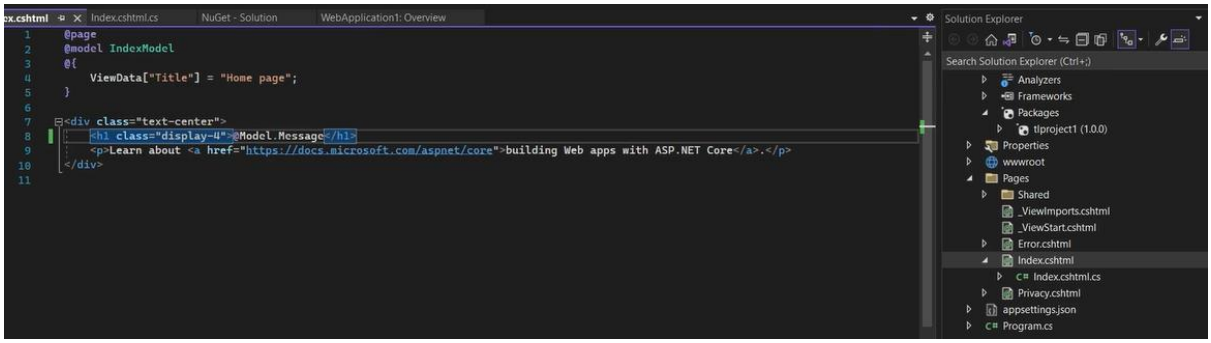


Modify the code as per screenshot

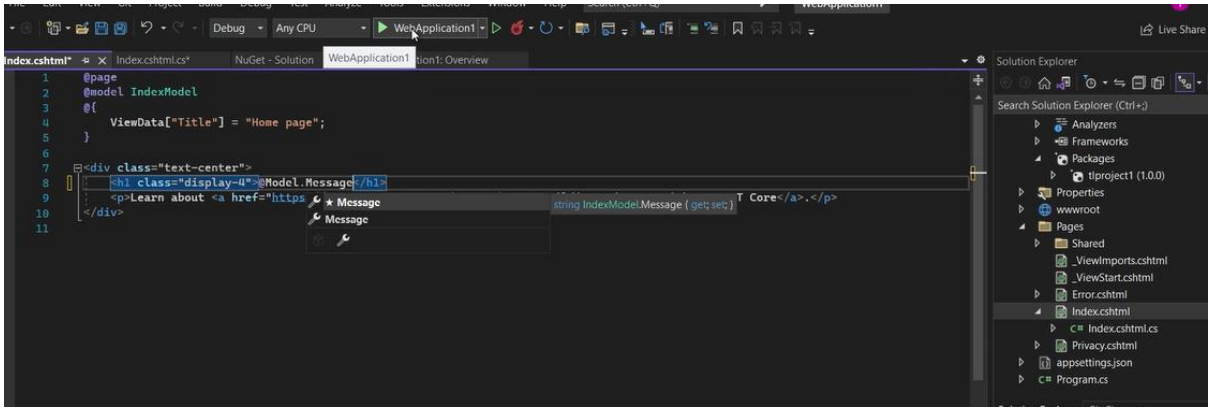


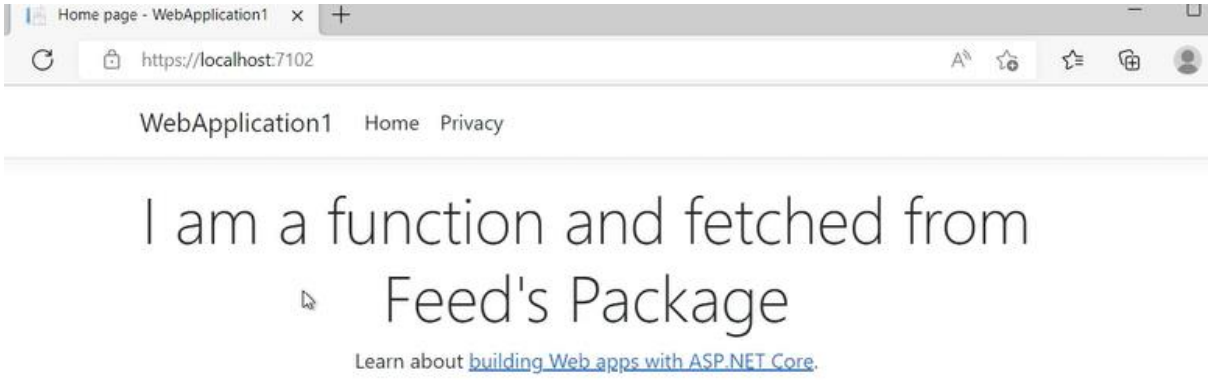


Do changes in default index.cshtml file as per screenshot.



Build your solution




Home page - WebApplication1
https://localhost:7102
WebApplication1
Home
Privacy
I am a function and fetched from Feed's Package
Learn about building Web apps with ASP.NET Core.