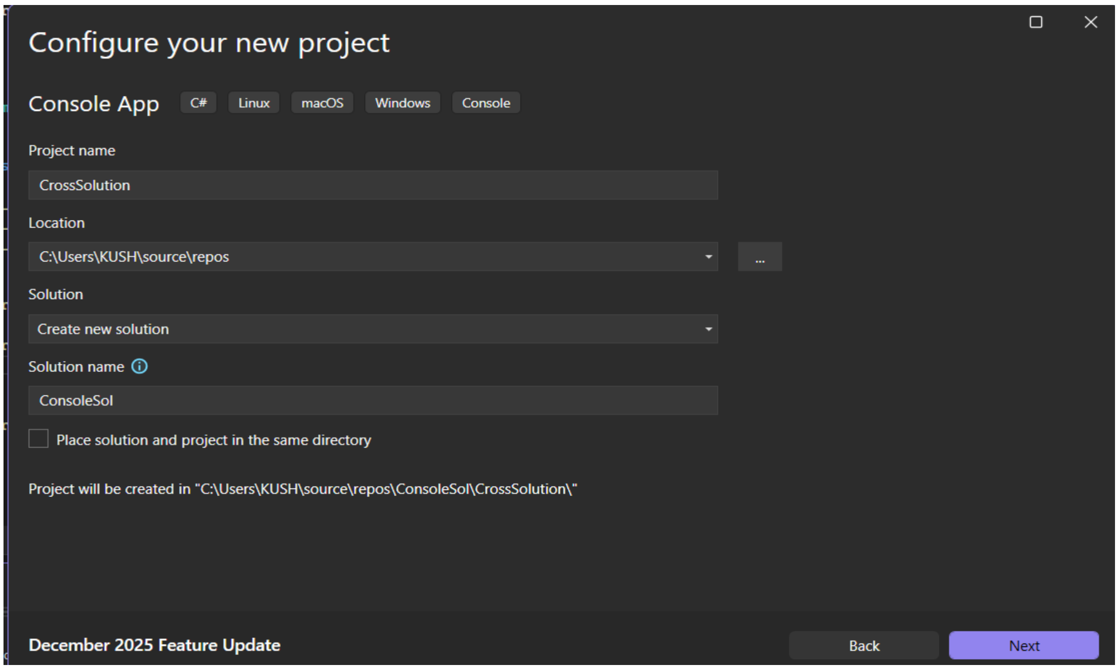
Name-Kushagar **Uid-22BCS10429**

DAY-3: Create a document providing all steps for publishing using the self-contained and dependent modes

Step 1: Create a new Project with the name CrossSolution



Configure your new project

Console App C# Linux macOS Windows Console

Project name: CrossSolution

Location: C:\Users\KUSH\source\repos

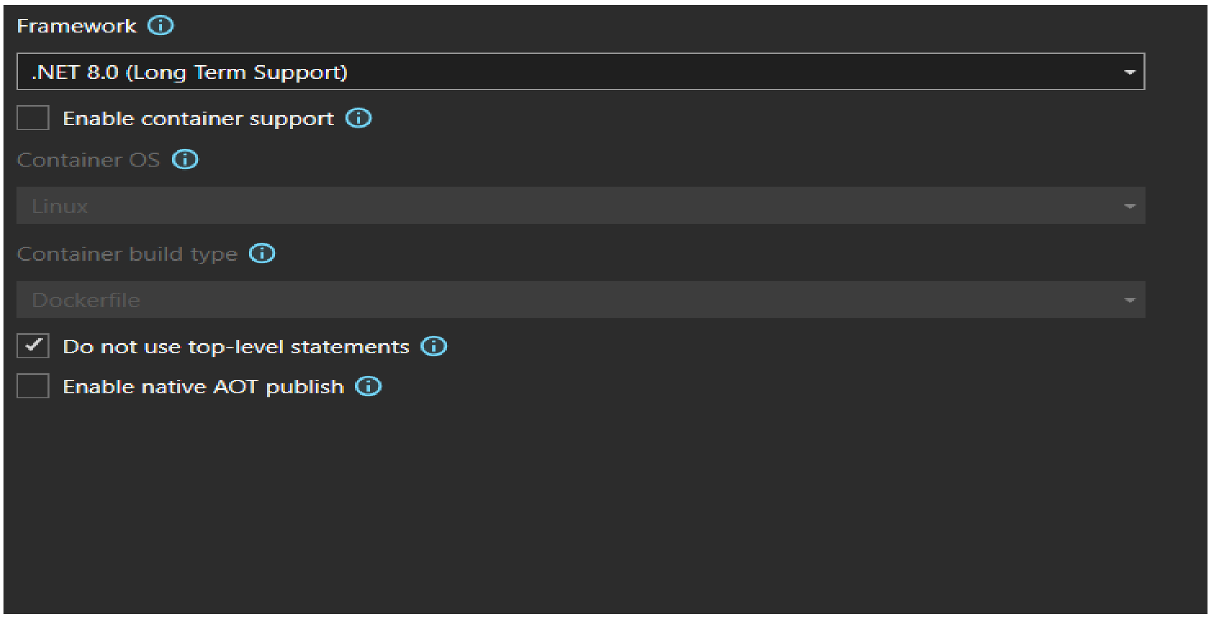
Solution: Create new solution

Solution name: ConsoleSol

☐ Place solution and project in the same directory

Project will be created in "C:\Users\KUSH\source\repos\ConsoleSol\CrossSolution\"

Step 2: Make sure to select the do not add top-level statement and select .NET Framework version 8.0



Framework: .NET 8.0 (Long Term Support)

☐ Enable container support

Container OS: Linux

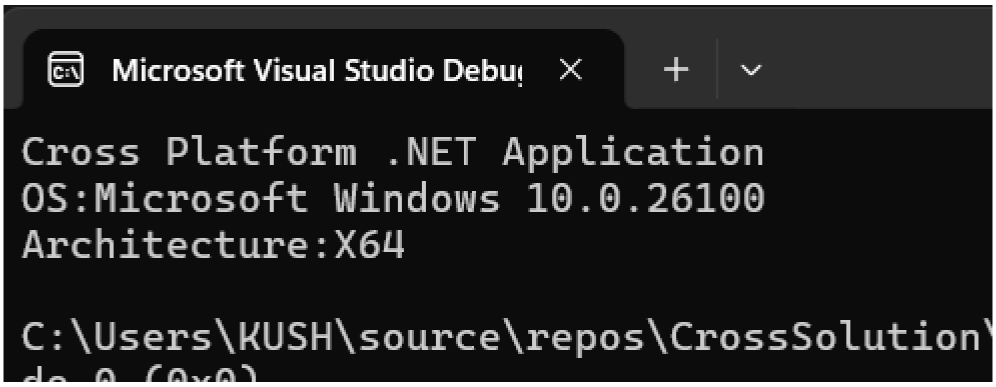
Container build type: Dockerfile

☑ Do not use top-level statements

☐ Enable native AOT publish

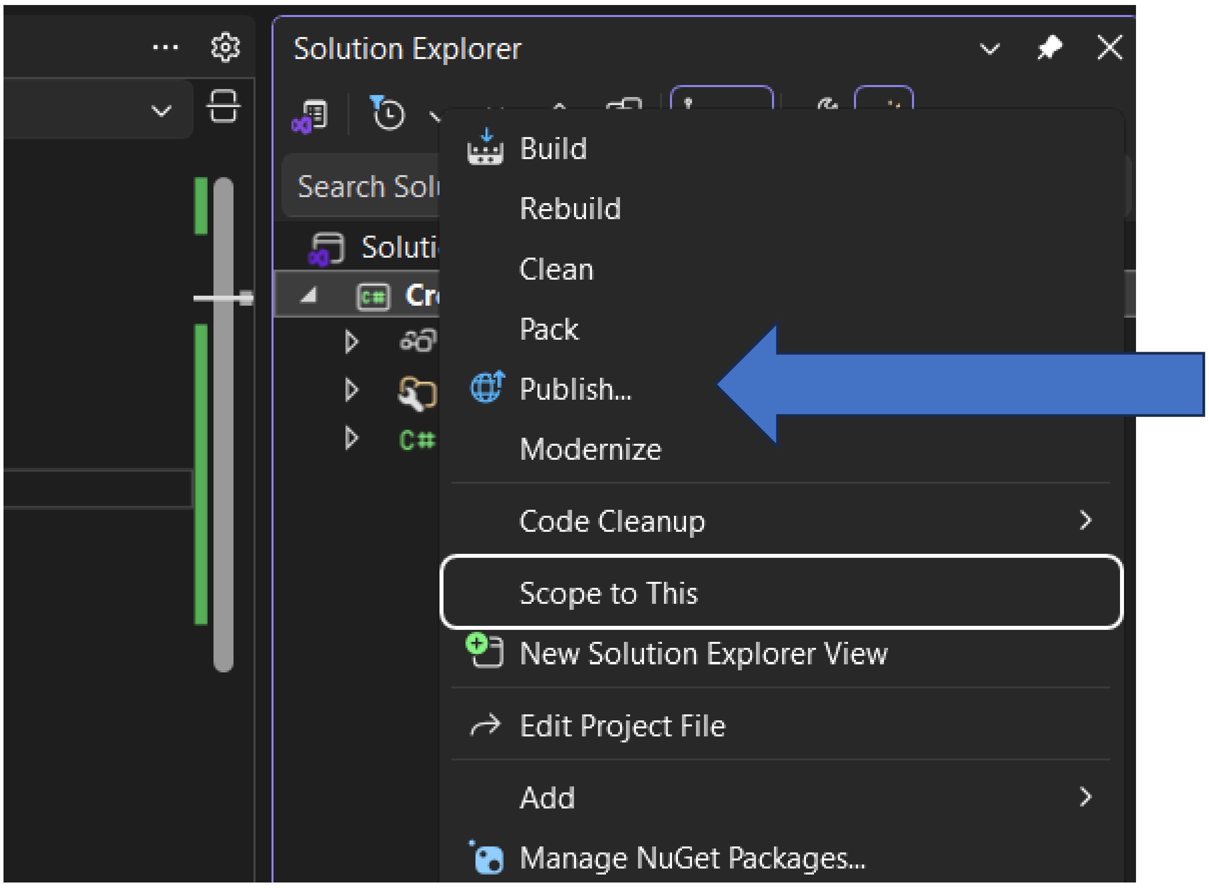
Step 3: In the Main Method, add these three lines of code that will print the Windows version and Architecture of your device

```
Console.WriteLine("Cross Platform .NET Application");
Console.WriteLine($"OS:{RuntimeInformation.OSDescription}");
Console.WriteLine($"Architecture:{RuntimeInformation.ProcessArchitecture}");
```



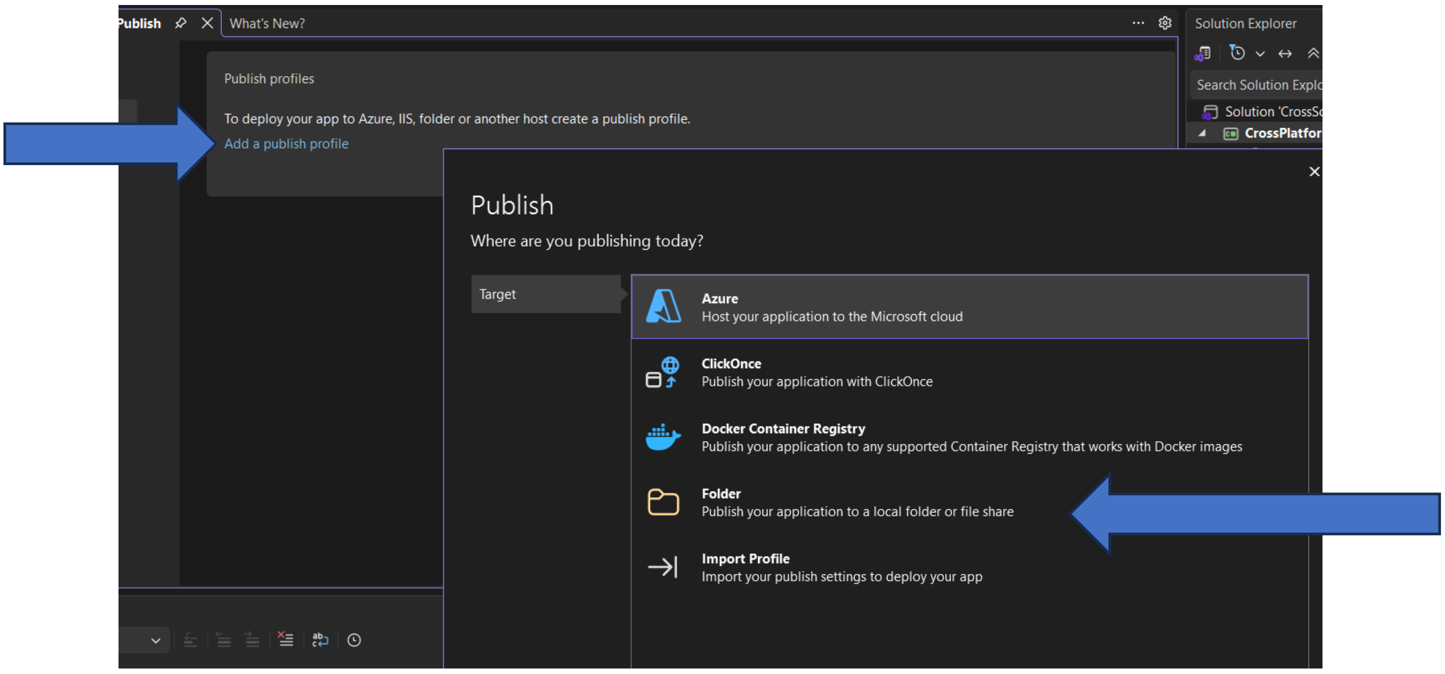
```
Cross Platform .NET Application
OS:Microsoft Windows 10.0.26100
Architecture:X64

C:\Users\KUSH\source\repos\CrossSolution
de 0 (0x0)
```

Step 4: Right-click on your project and click on publish



Step 5: Click on

1. Add a publish Profile
2. Select the folder option in the target



Publish profiles

To deploy your app to Azure, IIS, folder or another host create a publish profile.

Add a publish profile

Publish

Where are you publishing today?

Target

Azure
Host your application to the Microsoft cloud

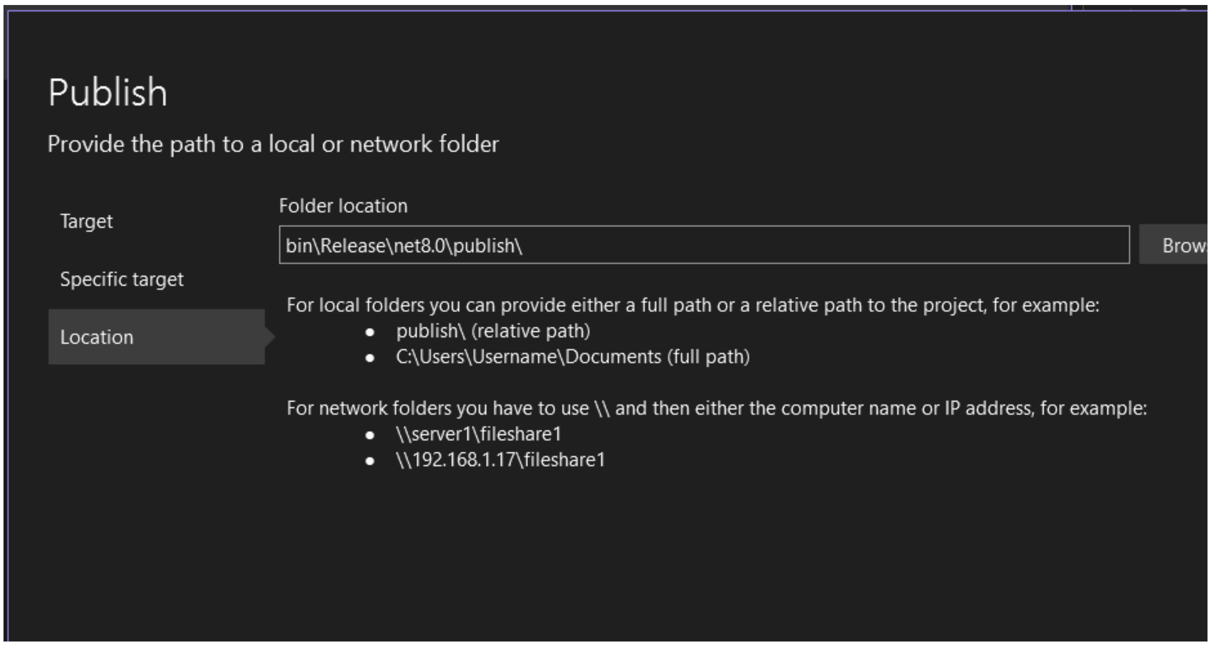
ClickOnce
Publish your application with ClickOnce

Docker Container Registry
Publish your application to any supported Container Registry that works with Docker images

Folder
Publish your application to a local folder or file share

Import Profile
Import your publish settings to deploy your app

Step 6: Again, select the target option in the specific target and then select the location for publishing your project



Publish

Provide the path to a local or network folder

Target

Specific target

Location

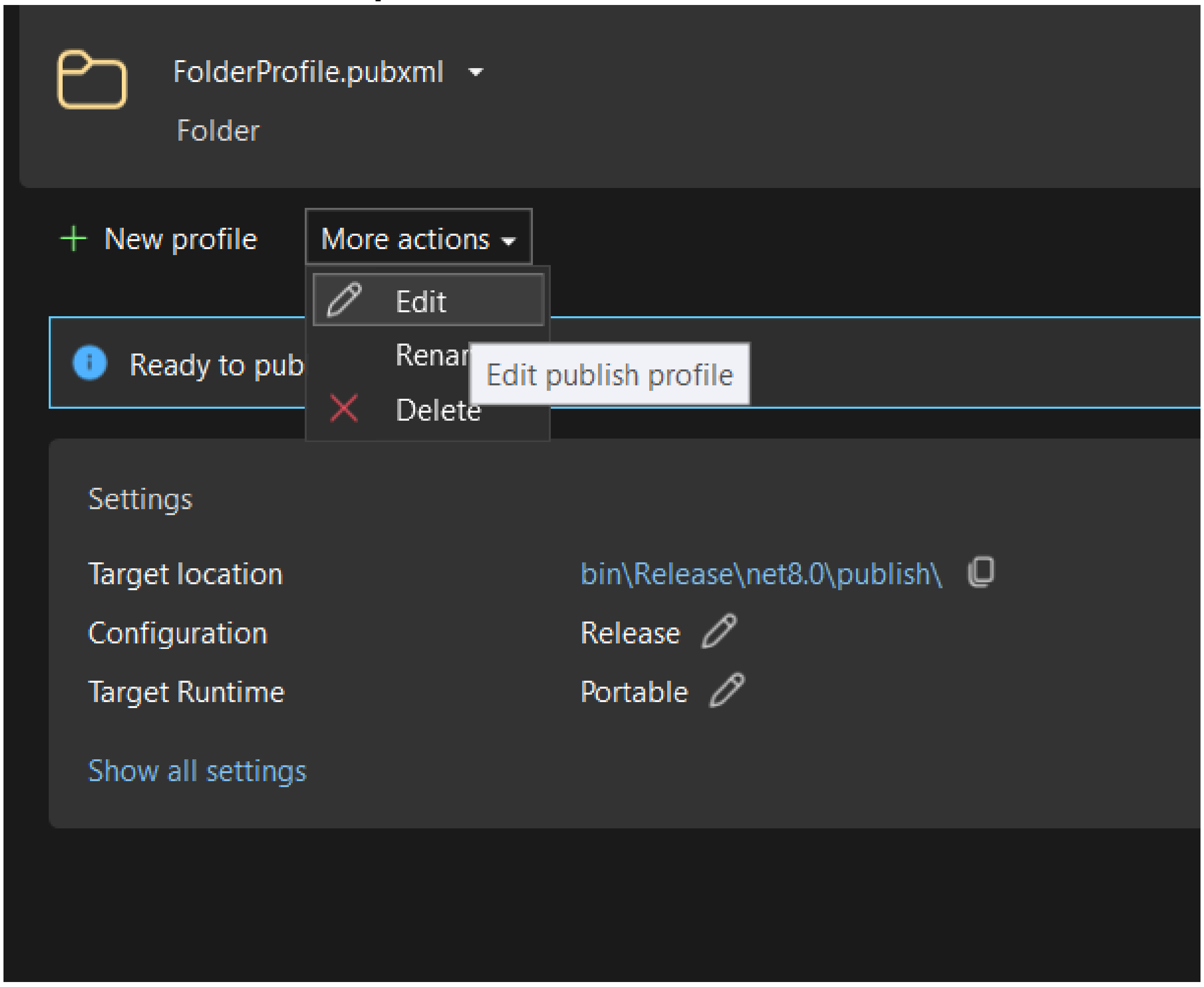
Folder location

bin\Release\net8.0\publish\

For local folders you can provide either a full path or a relative path to the project, for example:
- publish\ (relative path)
- C:\Users\Username\Documents (full path)

For network folders you have to use \\ and then either the computer name or IP address, for example:
- \\server1\fileshare1
- \\192.168.1.17\fileshare1

Step 7: Now tap on more options, and a dropdown menu will open, then select the edit option



Settings

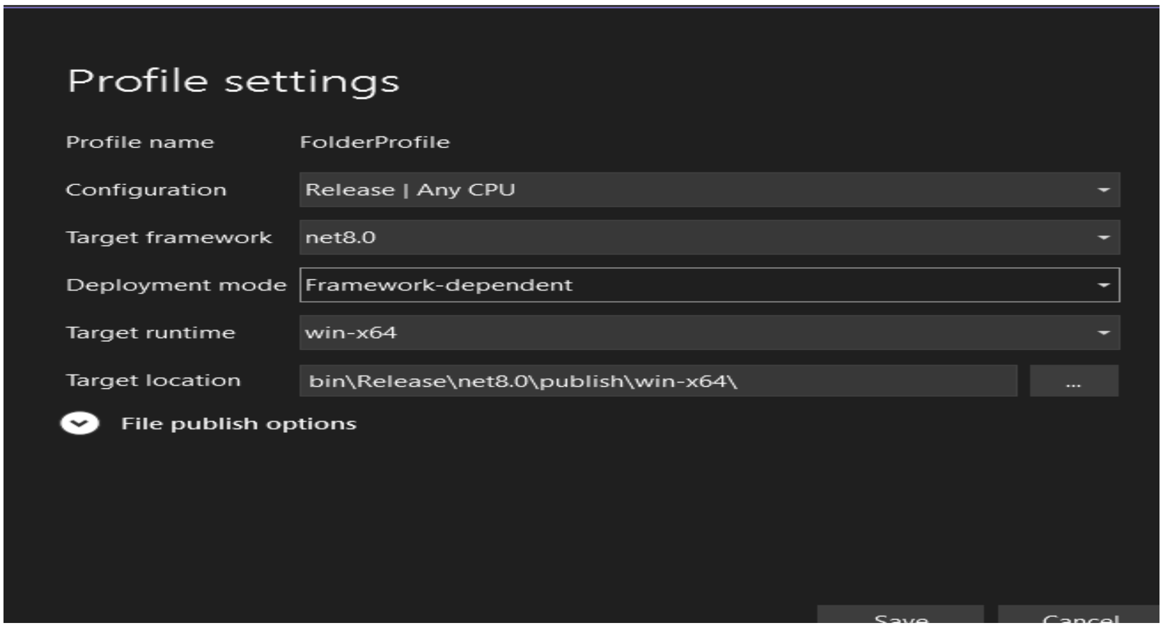
Target location bin\Release\net8.0\publish\

Configuration Release

Target Runtime Portable

Show all settings

Step 8: only change the Target runtime and select win-64 in it



Profile settings

Profile name FolderProfile

Configuration Release | Any CPU

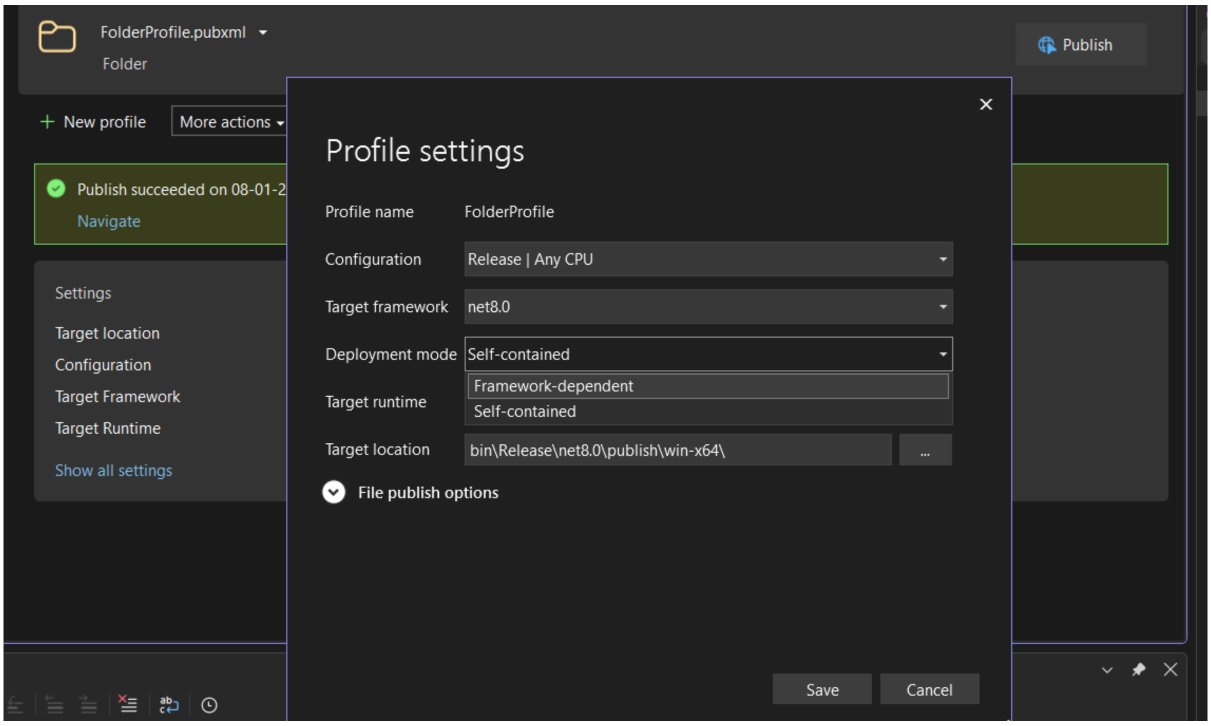
Target framework net8.0

Deployment mode Framework-dependent

Target runtime win-x64

Target location bin\Release\net8.0\publish\win-x64\

Step 9: Now, if you change the deployment mode to self-contained, your project will contain all the necessary things required to run your code



Profile settings

Profile name FolderProfile

Configuration Release | Any CPU

Target framework net8.0

Deployment mode Self-contained

Framework-dependent

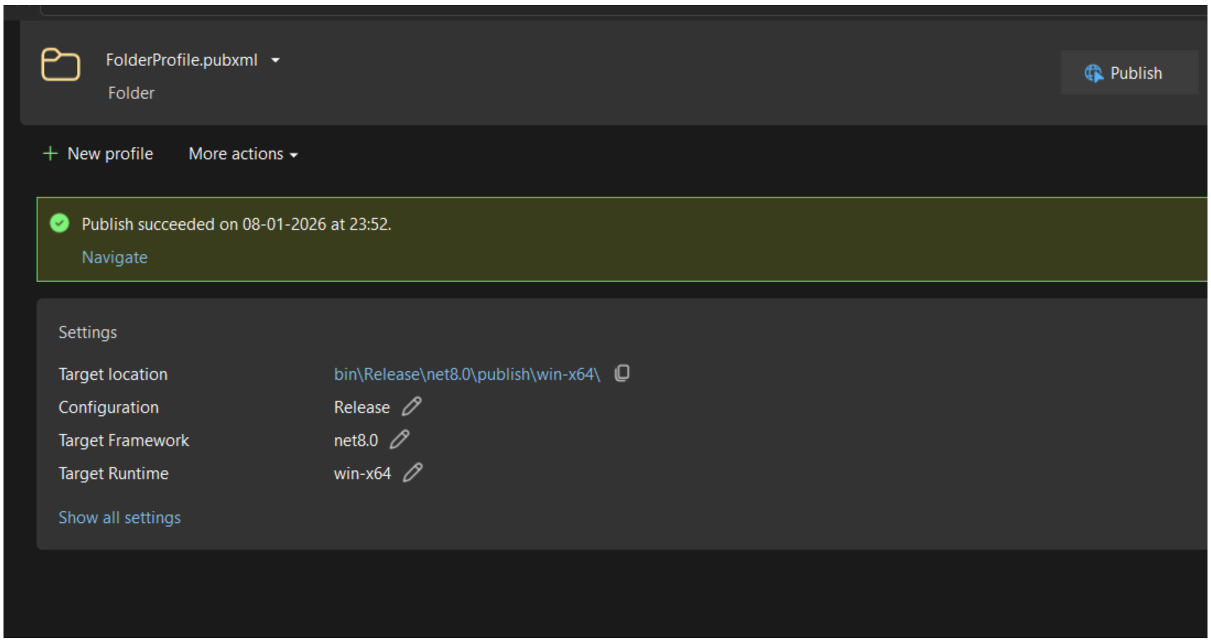
Self-contained

Target runtime

Target location bin\Release\net8.0\publish\win-x64\

File publish options

Step 10: Here you go, now click on publish, and your code is ready to deploy



Publish succeeded on 08-01-2026 at 23:52.

Settings

Target location bin\Release\net8.0\publish\win-x64\

Configuration Release

Target Framework net8.0

Target Runtime win-x64