SWINBURNE VIETNAM

HO CHI MINH CAMPUS

Class: COS40006

Deployment Portfolio Task 1

Level Attempt: HD

Instructor: Dr. Thomas Hang

Student names: Le Quang Hai

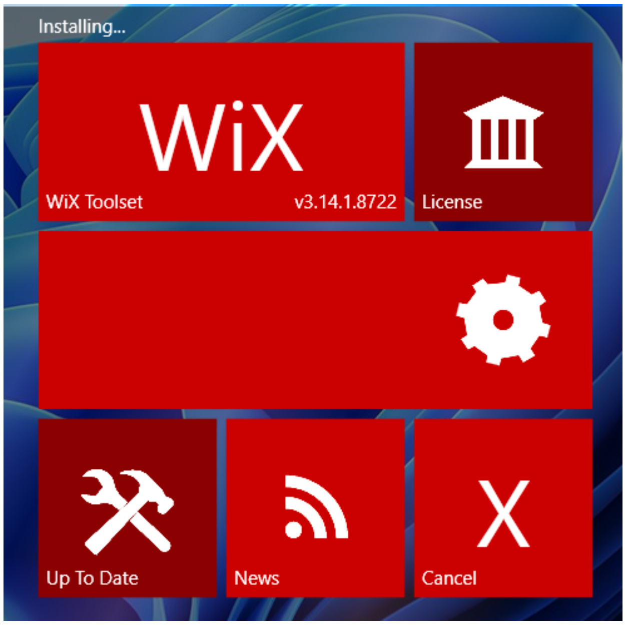
Due date: 19 May 2024

Submission date: 19 May 2024

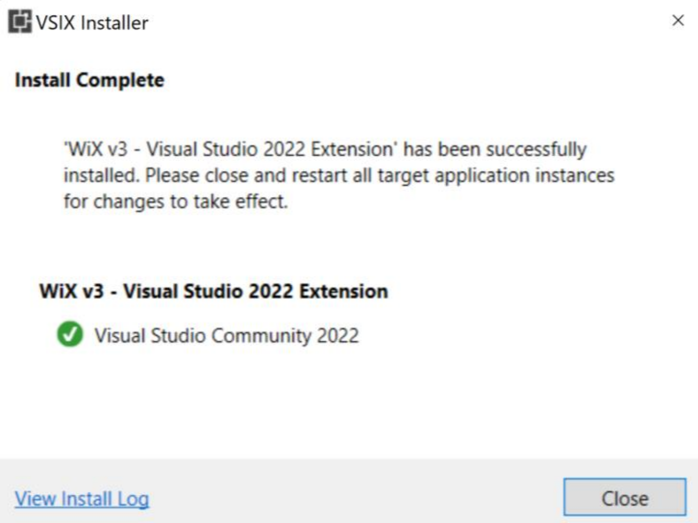
Pass Task: Follow the Wix walkthrough and deploy a sample desktop.

Set up Visual Studio

Install Wix toolset



Install Wix for Visual Studio



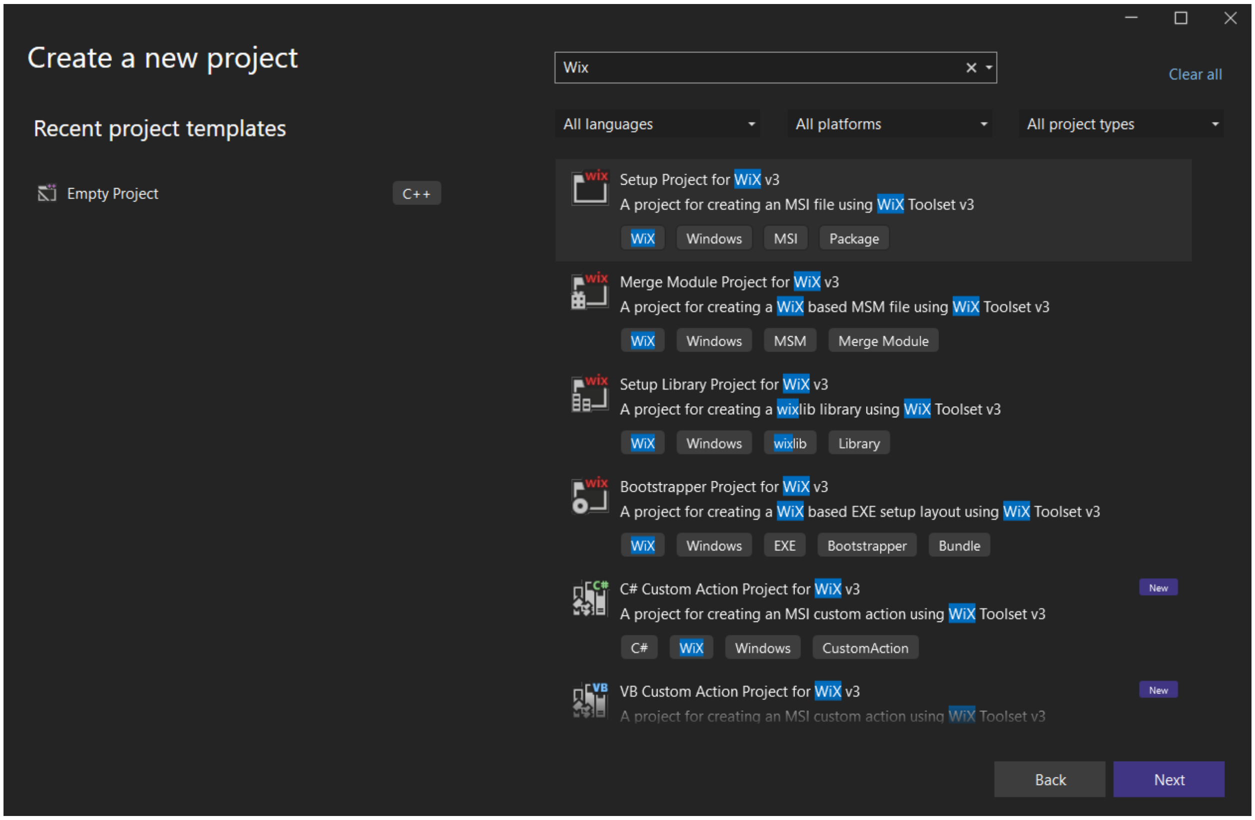
Install Complete

'WiX v3 - Visual Studio 2022 Extension' has been successfully installed. Please close and restart all target application instances for changes to take effect.

WiX v3 - Visual Studio 2022 Extension

✓ Visual Studio Community 2022

Add Wix to Visual Studio



Create a new project

Recent project templates

Empty Project — C++

Search: Wix

Clear all

All languages — All platforms — All project types

Setup Project for WiX v3
A project for creating an MSI file using WiX Toolset v3
WiX · Windows · MSI · Package

Merge Module Project for WiX v3
A project for creating a WiX based MSM file using WiX Toolset v3
WiX · Windows · MSM · Merge Module

Setup Library Project for WiX v3
A project for creating a wixlib library using WiX Toolset v3
WiX · Windows · wixlib · Library

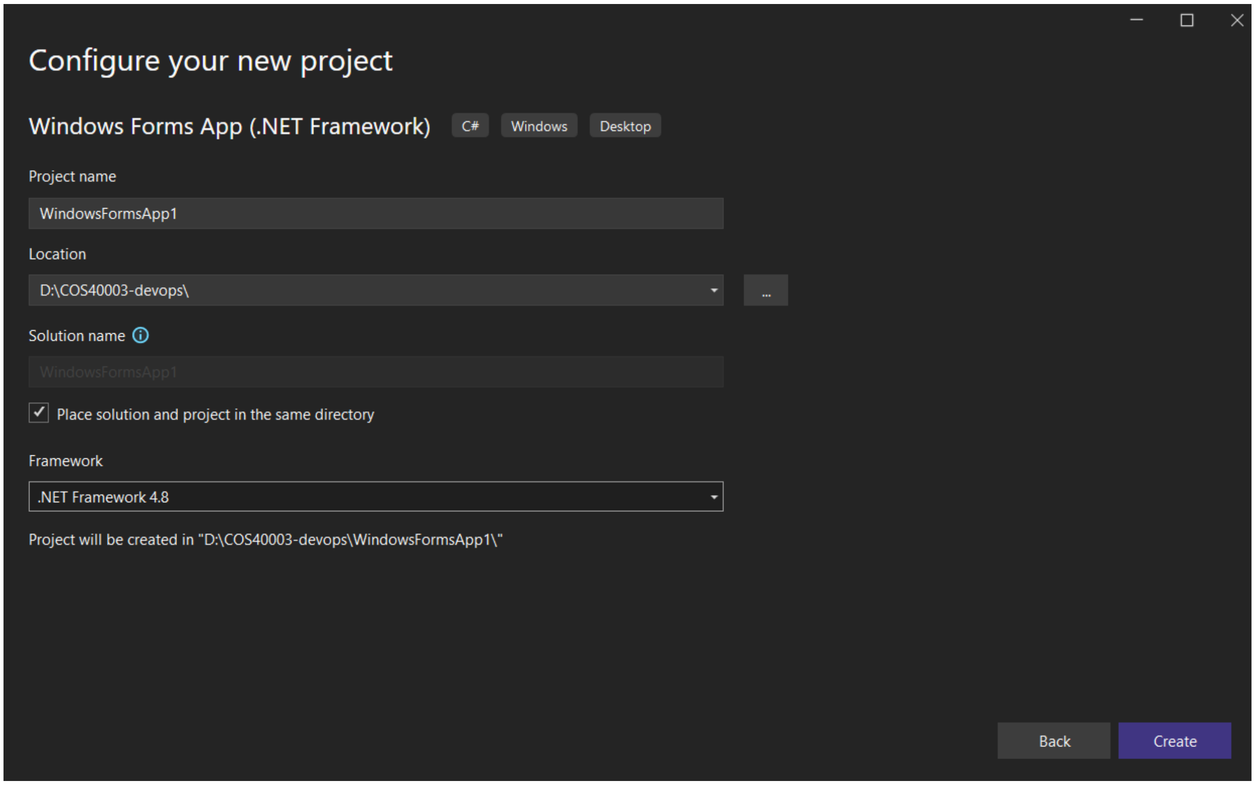
Bootstrapper Project for WiX v3
A project for creating a WiX based EXE setup layout using WiX Toolset v3
WiX · Windows · EXE · Bootstrapper · Bundle

C# Custom Action Project for WiX v3 — New
A project for creating an MSI custom action using WiX Toolset v3
C# · WiX · Windows · CustomAction

VB Custom Action Project for WiX v3 — New
A project for creating an MSI custom action using WiX Toolset v3

Create a project to package.

Create Windows Form App



Configure your new project

Windows Forms App (.NET Framework) C# Windows Desktop

Project name: WindowsFormsApp1

Location: D:\COS40003-devops\

Solution name: WindowsFormsApp1

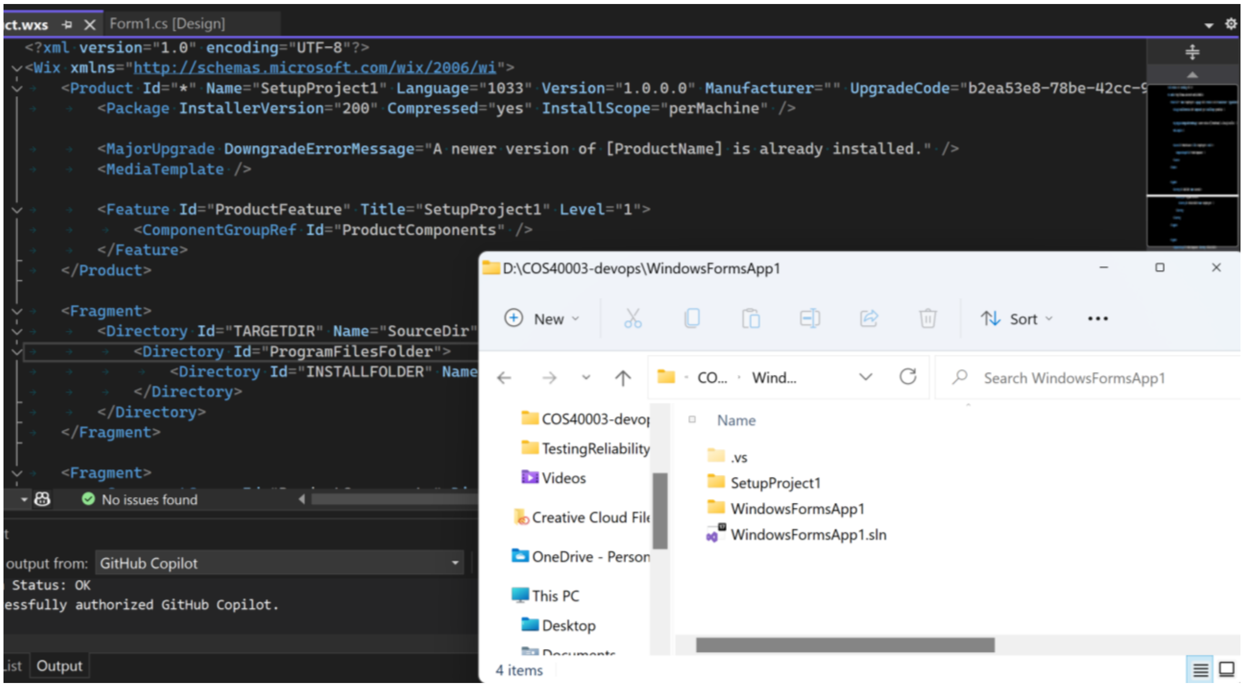
☑ Place solution and project in the same directory

Framework: .NET Framework 4.8

Project will be created in "D:\COS40003-devops\WindowsFormsApp1\"

Wix hello world installer

Create Wix Setup Project



```
<?xml version="1.0" encoding="UTF-8"?>
<Wix xmlns="http://schemas.microsoft.com/wix/2006/wi">
    <Product Id="*" Name="SetupProject1" Language="1033" Version="1.0.0.0" Manufacturer="" UpgradeCode="b2ea53e8-78be-42cc-9
        <Package InstallerVersion="200" Compressed="yes" InstallScope="perMachine" />

        <MajorUpgrade DowngradeErrorMessage="A newer version of [ProductName] is already installed." />
        <MediaTemplate />

        <Feature Id="ProductFeature" Title="SetupProject1" Level="1">
            <ComponentGroupRef Id="ProductComponents" />
        </Feature>
    </Product>

    <Fragment>
        <Directory Id="TARGETDIR" Name="SourceDir"
            <Directory Id="ProgramFilesFolder">
                <Directory Id="INSTALLFOLDER" Name
            </Directory>
        </Directory>
    </Fragment>

    <Fragment>
```

No issues found

output from: GitHub Copilot

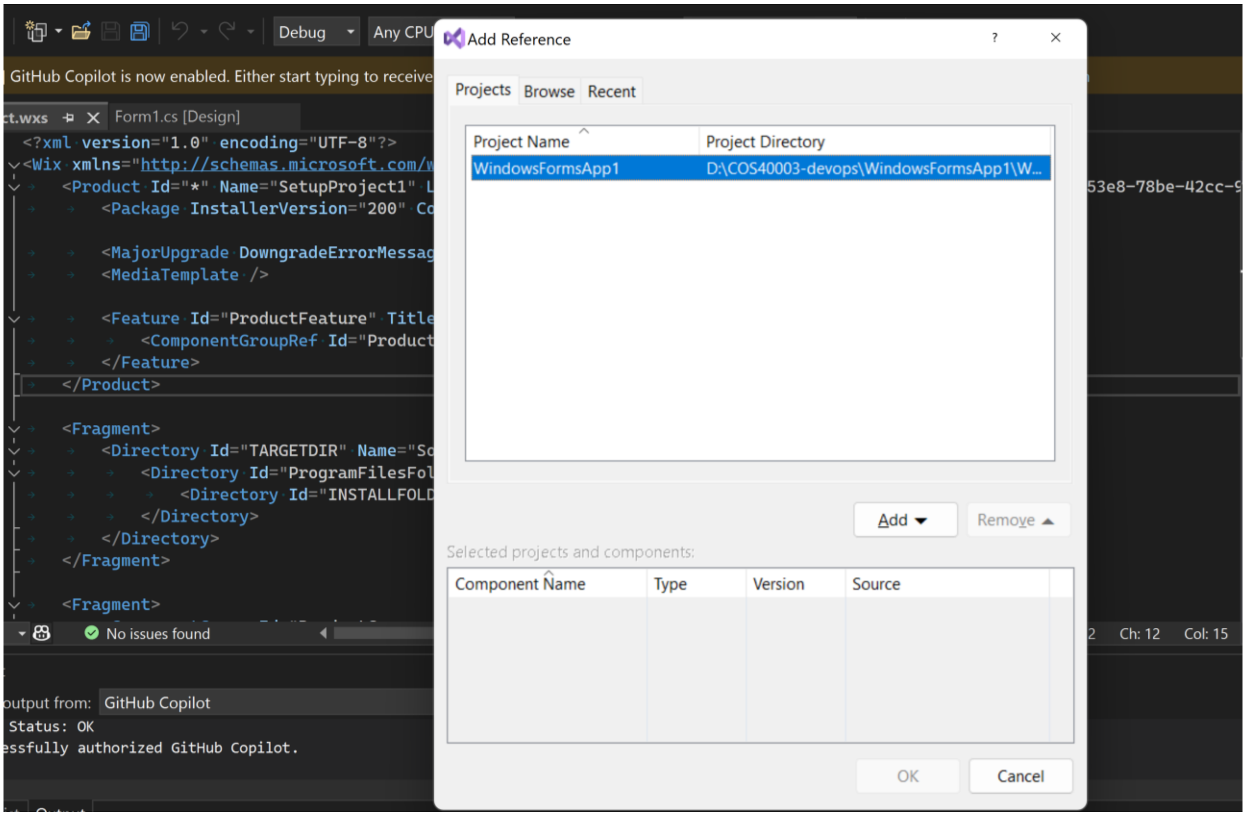
```
 Status: OK
essfully authorized GitHub Copilot.
```

D:\COS40003-devops\WindowsFormsApp1

Name
- .vs
- SetupProject1
- WindowsFormsApp1
- WindowsFormsApp1.sln

4 items

Create a reference between the application and the Wix Project



GitHub Copilot is now enabled. Either start typing to receive

```
<?xml version="1.0" encoding="UTF-8"?>
<Wix xmlns="http://schemas.microsoft.com/w
    <Product Id="*" Name="SetupProject1" L                                                53e8-78be-42cc-9
        <Package InstallerVersion="200" Co

        <MajorUpgrade DowngradeErrorMessag
        <MediaTemplate />

        <Feature Id="ProductFeature" Title
            <ComponentGroupRef Id="Product
        </Feature>
    </Product>

    <Fragment>
        <Directory Id="TARGETDIR" Name="So
            <Directory Id="ProgramFilesFol
                <Directory Id="INSTALLFOLD
            </Directory>
        </Directory>
    </Fragment>

    <Fragment>
```

Project Name	Project Directory
WindowsFormsApp1	D:\COS40003-devops\WindowsFormsApp1\W...

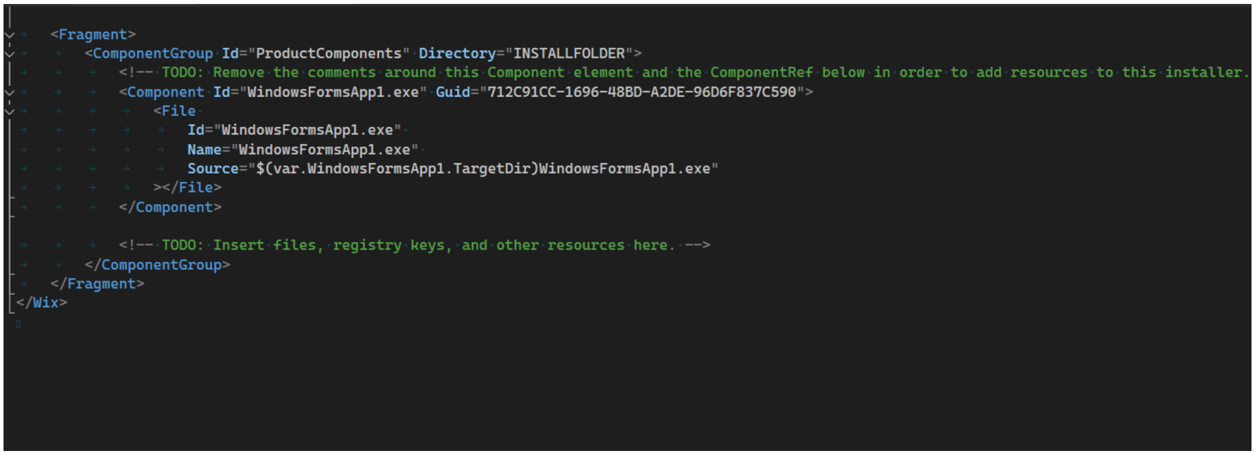
Selected projects and components:

Component Name	Type	Version	Source

Output from: GitHub Copilot

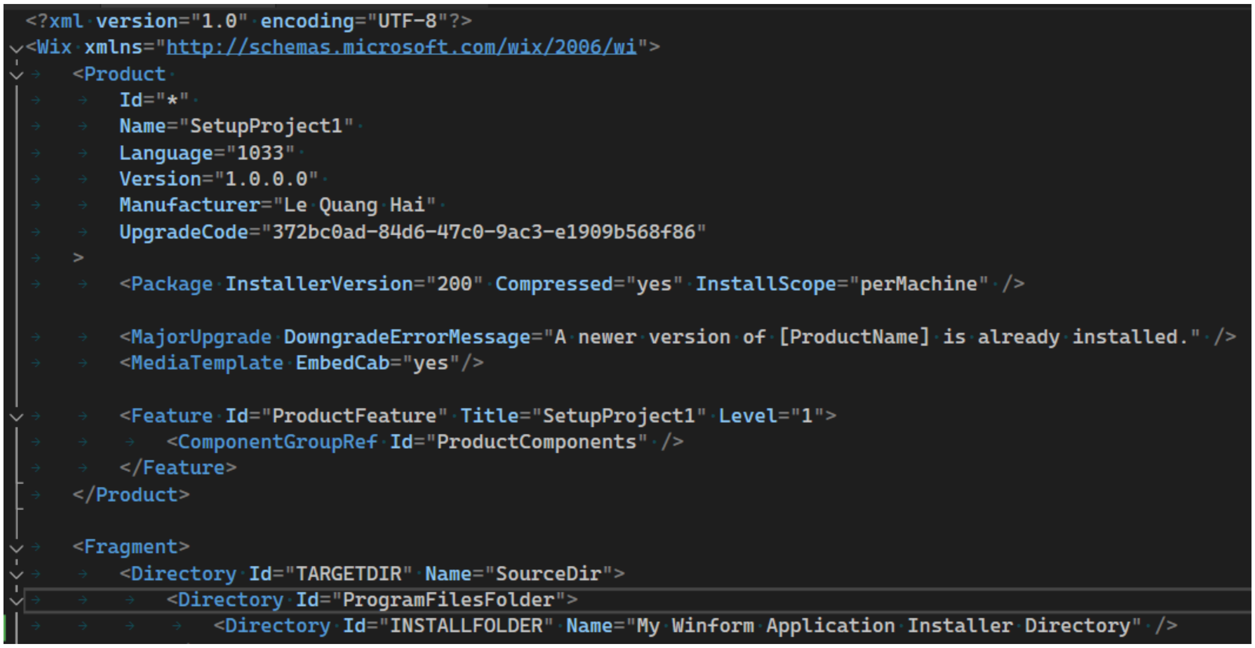
```
Status: OK
essfully authorized GitHub Copilot.
```

Create a Component tag, specify file name, source location and GUID



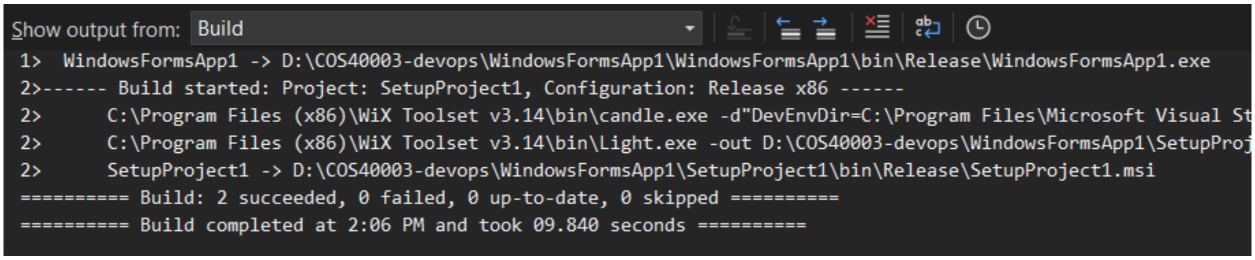
```
    <Fragment>
        <ComponentGroup Id="ProductComponents" Directory="INSTALLFOLDER">
            <!-- TODO: Remove the comments around this Component element and the ComponentRef below in order to add resources to this installer.
            <Component Id="WindowsFormsApp1.exe" Guid="712C91CC-1696-48BD-A2DE-96D6F837C590">
                <File
                    Id="WindowsFormsApp1.exe"
                    Name="WindowsFormsApp1.exe"
                    Source="$(var.WindowsFormsApp1.TargetDir)WindowsFormsApp1.exe"
                ></File>
            </Component>

            <!-- TODO: Insert files, registry keys, and other resources here. -->
        </ComponentGroup>
    </Fragment>
</Wix>
```

Modify the Product, Media, Directory tags



```
<?xml version="1.0" encoding="UTF-8"?>
<Wix xmlns="http://schemas.microsoft.com/wix/2006/wi">
    <Product
        Id="*"
        Name="SetupProject1"
        Language="1033"
        Version="1.0.0.0"
        Manufacturer="Le Quang Hai"
        UpgradeCode="372bc0ad-84d6-47c0-9ac3-e1909b568f86"
    >
        <Package InstallerVersion="200" Compressed="yes" InstallScope="perMachine" />

        <MajorUpgrade DowngradeErrorMessage="A newer version of [ProductName] is already installed." />
        <MediaTemplate EmbedCab="yes"/>

        <Feature Id="ProductFeature" Title="SetupProject1" Level="1">
            <ComponentGroupRef Id="ProductComponents" />
        </Feature>
    </Product>

    <Fragment>
        <Directory Id="TARGETDIR" Name="SourceDir">
            <Directory Id="ProgramFilesFolder">
                <Directory Id="INSTALLFOLDER" Name="My Winform Application Installer Directory" />
```

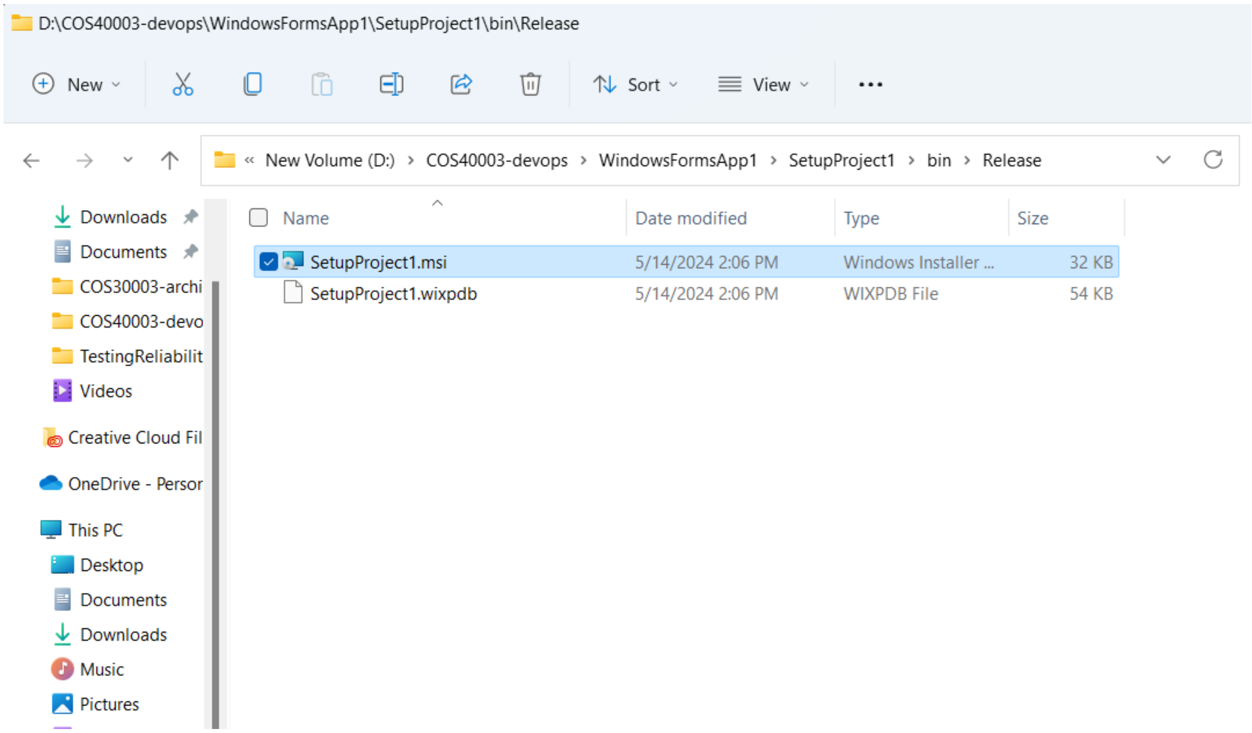
Build succeeded



Show output from: Build

```
1>  WindowsFormsApp1 -> D:\COS40003-devops\WindowsFormsApp1\WindowsFormsApp1\bin\Release\WindowsFormsApp1.exe
2>------ Build started: Project: SetupProject1, Configuration: Release x86 ------
2>      C:\Program Files (x86)\WiX Toolset v3.14\bin\candle.exe -d"DevEnvDir=C:\Program Files\Microsoft Visual St
2>      C:\Program Files (x86)\WiX Toolset v3.14\bin\Light.exe -out D:\COS40003-devops\WindowsFormsApp1\SetupProj
2>      SetupProject1 -> D:\COS40003-devops\WindowsFormsApp1\SetupProject1\bin\Release\SetupProject1.msi
========== Build: 2 succeeded, 0 failed, 0 up-to-date, 0 skipped ==========
========== Build completed at 2:06 PM and took 09.840 seconds ==========
```

SetupProject1.msi file created in /SetupProject1/bin/Release/

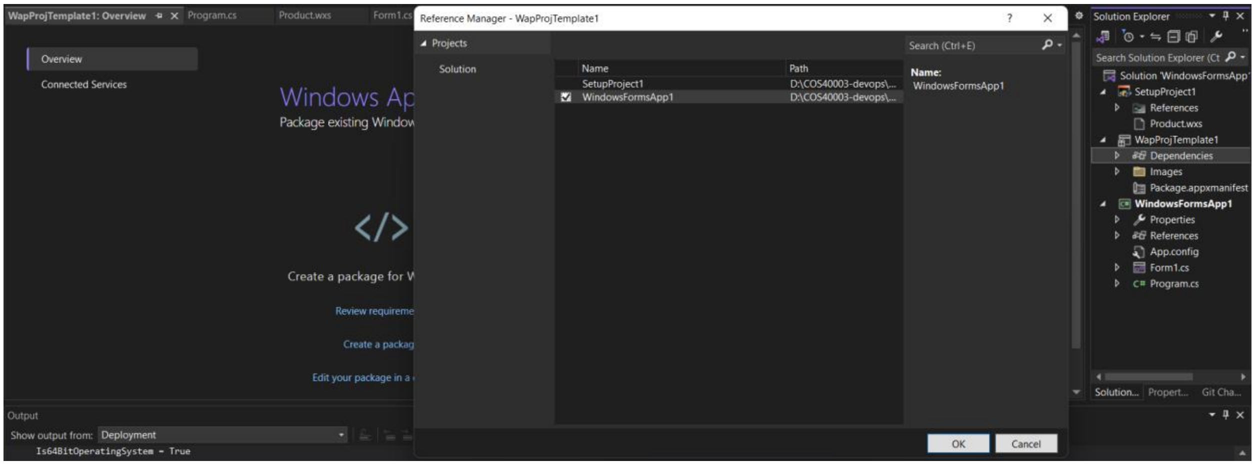


D:\COS40003-devops\WindowsFormsApp1\SetupProject1\bin\Release

Name	Date modified	Type	Size
☑ SetupProject1.msi	5/14/2024 2:06 PM	Windows Installer ...	32 KB
SetupProject1.wixpdb	5/14/2024 2:06 PM	WIXPDB File	54 KB

Universal Windows App Packaging

Create Windows Application Packaging Project and add reference to WindowsFormsApp



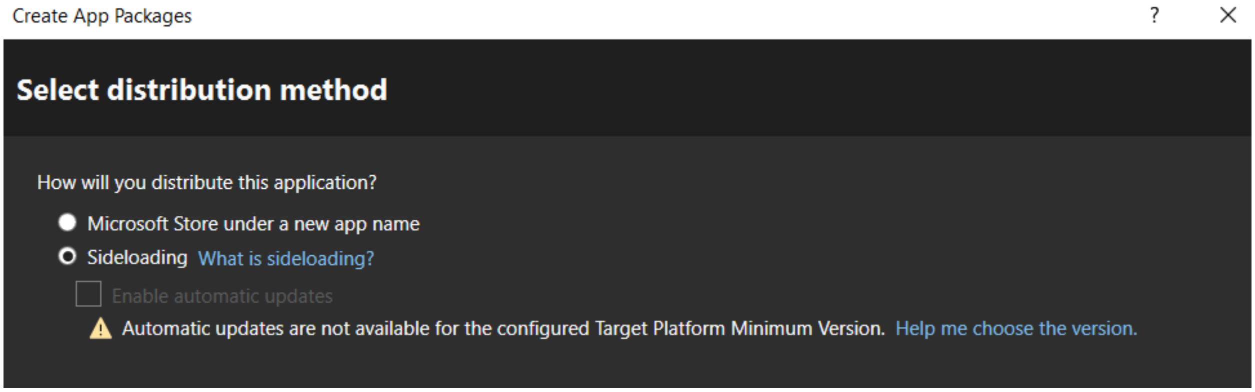
```
Is64BitOperatingSystem - True
```

Reference Manager - WapProjTemplate1

	Name	Path
	SetupProject1	D:\COS40003-devops\...
☑	WindowsFormsApp1	D:\COS40003-devops\...

Name:
WindowsFormsApp1

Create app package



Create App Packages ? ×

Select distribution method

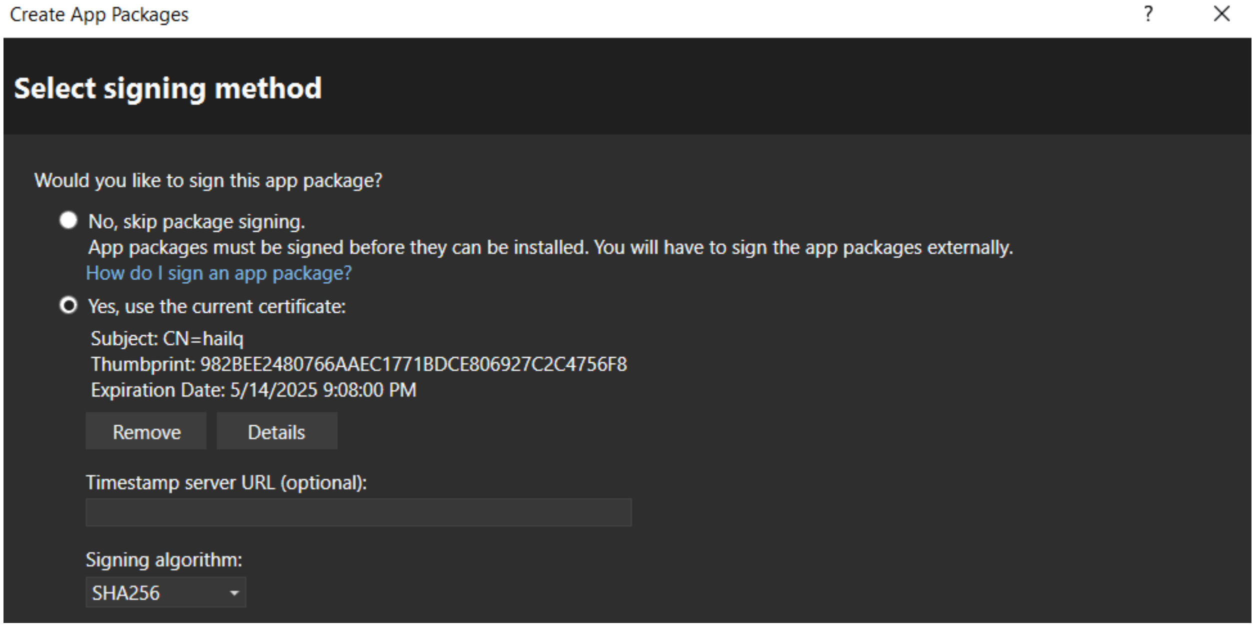
How will you distribute this application?

○ Microsoft Store under a new app name

● Sideloading What is sideloading?

□ Enable automatic updates

⚠ Automatic updates are not available for the configured Target Platform Minimum Version. Help me choose the version.



Create App Packages ? ×

Select signing method

Would you like to sign this app package?

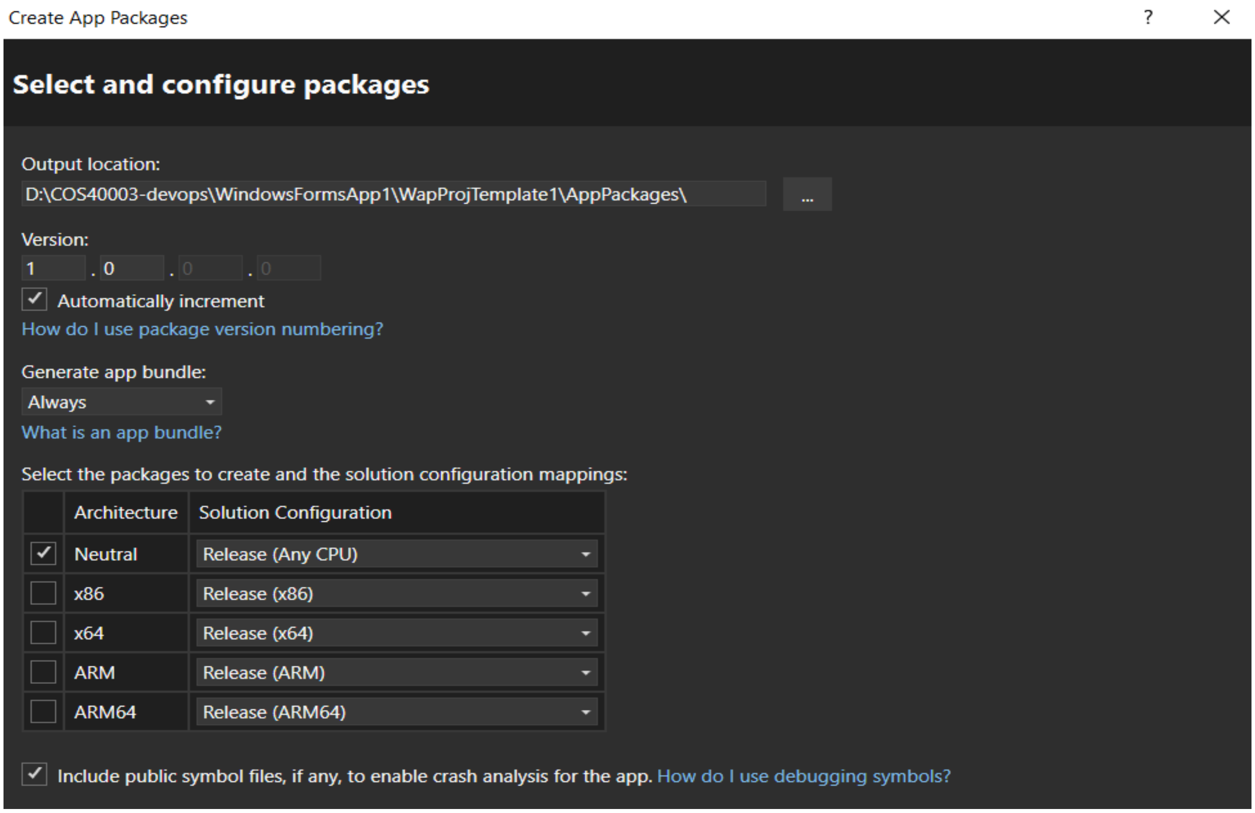
○ No, skip package signing.
App packages must be signed before they can be installed. You will have to sign the app packages externally.
How do I sign an app package?

● Yes, use the current certificate:

Subject: CN=hailq
Thumbprint: 982BEE2480766AAEC1771BDCE806927C2C4756F8
Expiration Date: 5/14/2025 9:08:00 PM

Timestamp server URL (optional):

Signing algorithm:
SHA256



Create App Packages

Select and configure packages

Output location:
D:\COS40003-devops\WindowsFormsApp1\WapProjTemplate1\AppPackages\

Version:
1 . 0 . 0 . 0

☑ Automatically increment

How do I use package version numbering?

Generate app bundle:
Always

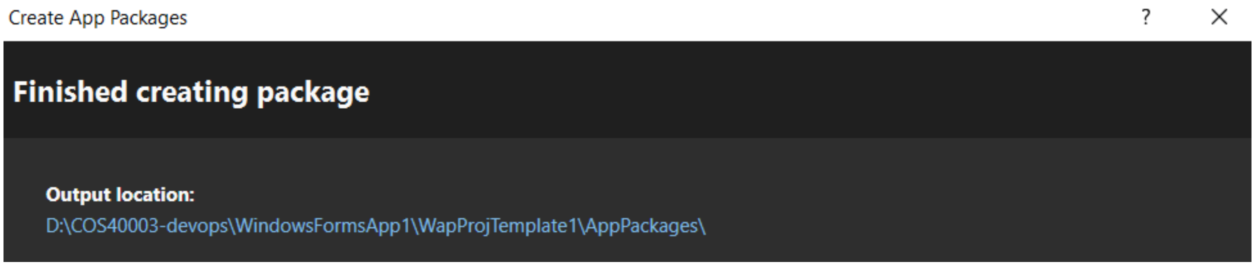
What is an app bundle?

Select the packages to create and the solution configuration mappings:

	Architecture	Solution Configuration
☑	Neutral	Release (Any CPU)
□	x86	Release (x86)
□	x64	Release (x64)
□	ARM	Release (ARM)
□	ARM64	Release (ARM64)

☑ Include public symbol files, if any, to enable crash analysis for the app. How do I use debugging symbols?

Package Creation Completed

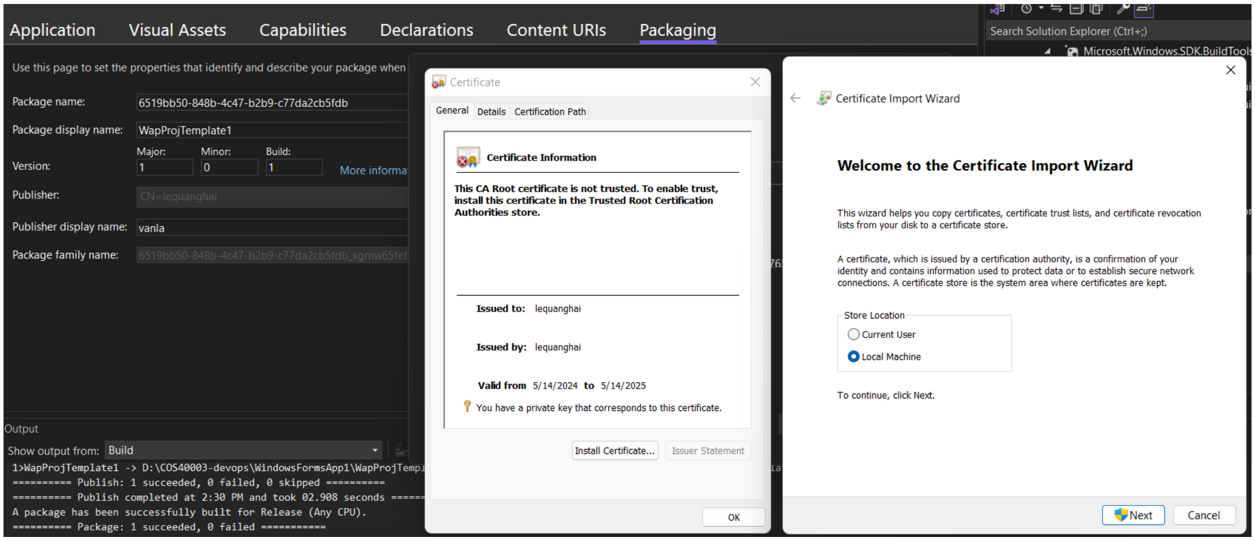


Create App Packages

Finished creating package

Output location:
D:\COS40003-devops\WindowsFormsApp1\WapProjTemplate1\AppPackages\

Install certificate to local machine



Use this page to set the properties that identify and describe your package when

Package name: 6519bb50-848b-4c47-b2b9-c77da2cb5fdb

Package display name: WapProjTemplate1

Version: Major: 1 Minor: 0 Build: 1 More informa

Publisher: CN=lequanghai

Publisher display name: vanla

Package family name: 6519bb50-848b-4c47-b2b9-c77da2cb5fdb_xgmw65fef

Output

Show output from: Build

```
1>WapProjTemplate1 -> D:\COS40003-devops\WindowsFormsApp1\WapProjTempl
========== Publish: 1 succeeded, 0 failed, 0 skipped ==========
========== Publish completed at 2:30 PM and took 02.908 seconds ======
A package has been successfully built for Release (Any CPU).
========== Package: 1 succeeded, 0 failed ===========
```

Certificate Information

This CA Root certificate is not trusted. To enable trust, install this certificate in the Trusted Root Certification Authorities store.

Issued to: lequanghai

Issued by: lequanghai

Valid from 5/14/2024 **to** 5/14/2025

You have a private key that corresponds to this certificate.

Welcome to the Certificate Import Wizard

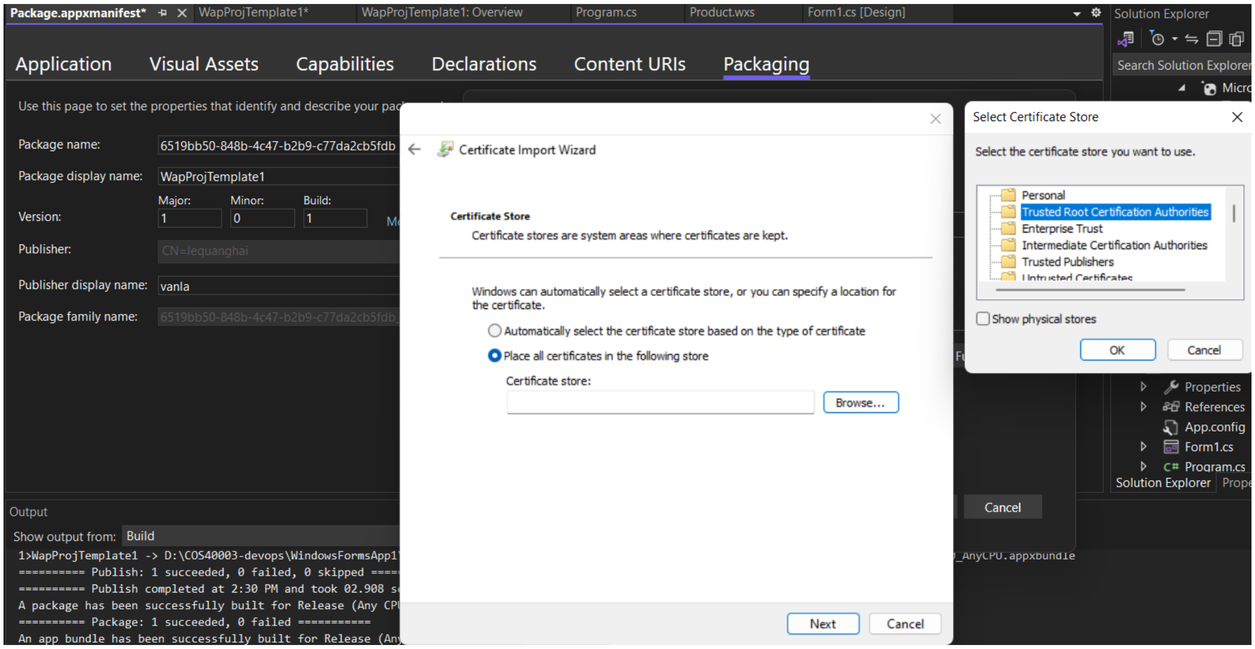
This wizard helps you copy certificates, certificate trust lists, and certificate revocation lists from your disk to a certificate store.

A certificate, which is issued by a certification authority, is a confirmation of your identity and contains information used to protect data or to establish secure network connections. A certificate store is the system area where certificates are kept.

Store Location
○ Current User
● Local Machine

To continue, click Next.

Configure certificate store



Use this page to set the properties that identify and describe your pac

Package name: 6519bb50-848b-4c47-b2b9-c77da2cb5fdb

Package display name: WapProjTemplate1

Version: Major: 1 Minor: 0 Build: 1

Publisher: CN=lequanghai

Publisher display name: vanla

Package family name: 6519bb50-848b-4c47-b2b9-c77da2cb5fdb_

Output

Show output from: Build

```
1>WapProjTemplate1 -> D:\COS40003-devops\WindowsFormsApp1\
========== Publish: 1 succeeded, 0 failed, 0 skipped ====
========== Publish completed at 2:30 PM and took 02.908 s
A package has been successfully built for Release (Any CPU
========== Package: 1 succeeded, 0 failed ===========
An app bundle has been successfully built for Release (Any
_AnyCPU.appxbundle
```

Certificate Store

Certificate stores are system areas where certificates are kept.

Windows can automatically select a certificate store, or you can specify a location for the certificate.

○ Automatically select the certificate store based on the type of certificate
● Place all certificates in the following store

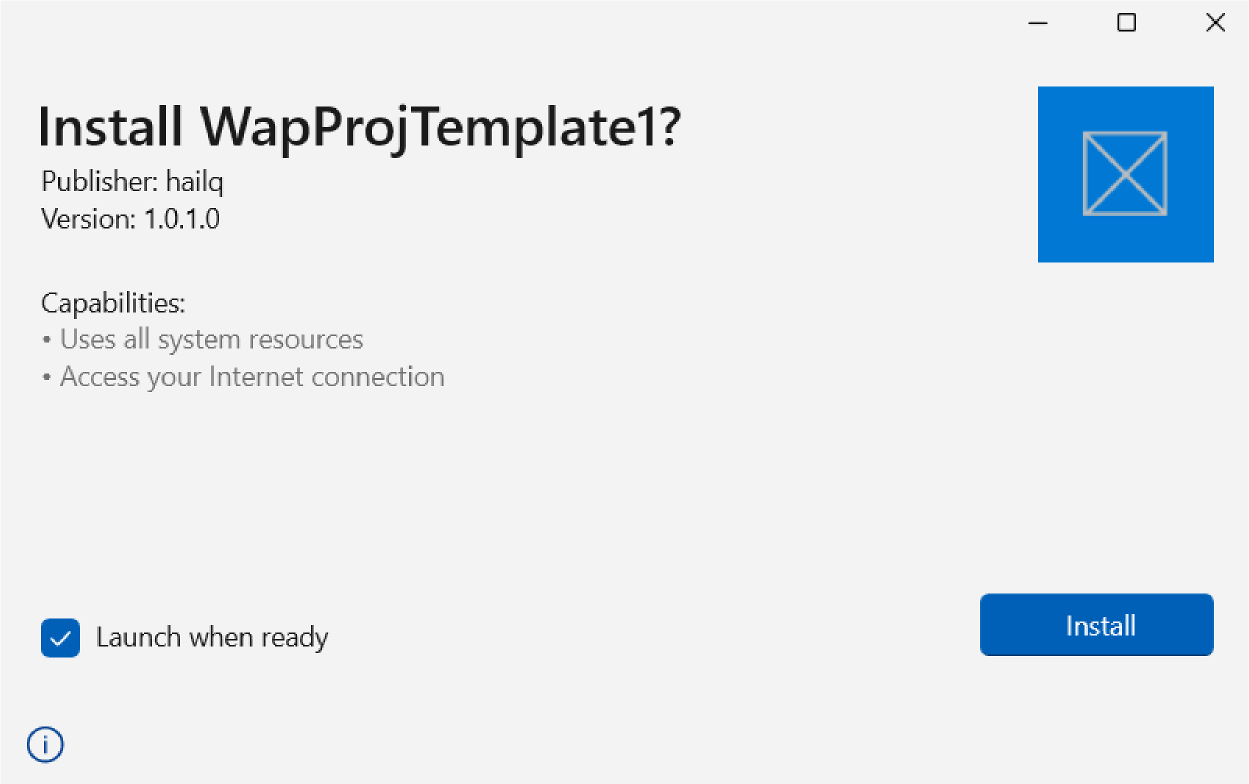
Certificate store:

Select the certificate store you want to use.

- Personal
- Trusted Root Certification Authorities
- Enterprise Trust
- Intermediate Certification Authorities
- Trusted Publishers
- Untrusted Certificates

□ Show physical stores

Now the application can be installed



Install WapProjTemplate1?

Publisher: hailq

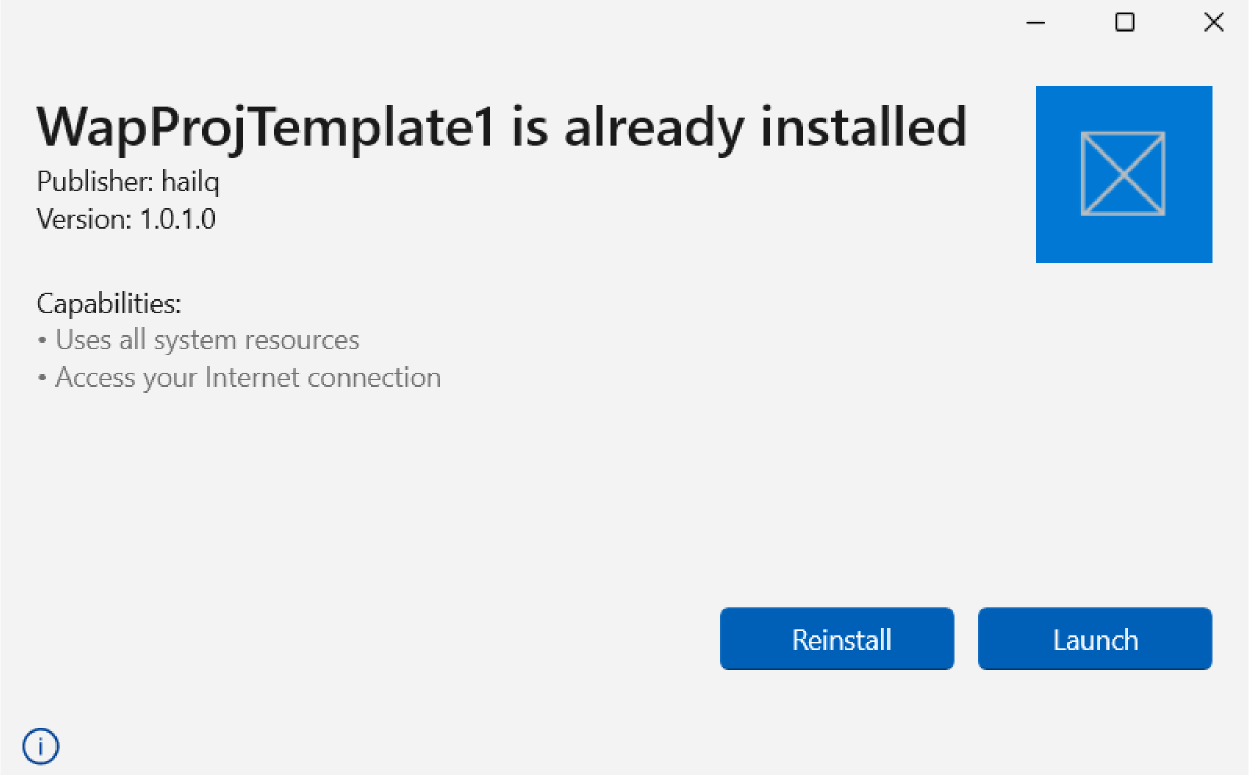
Version: 1.0.1.0

Capabilities:

- Uses all system resources
- Access your Internet connection

☑ Launch when ready

Install successfully



WapProjTemplate1 is already installed

Publisher: hailq

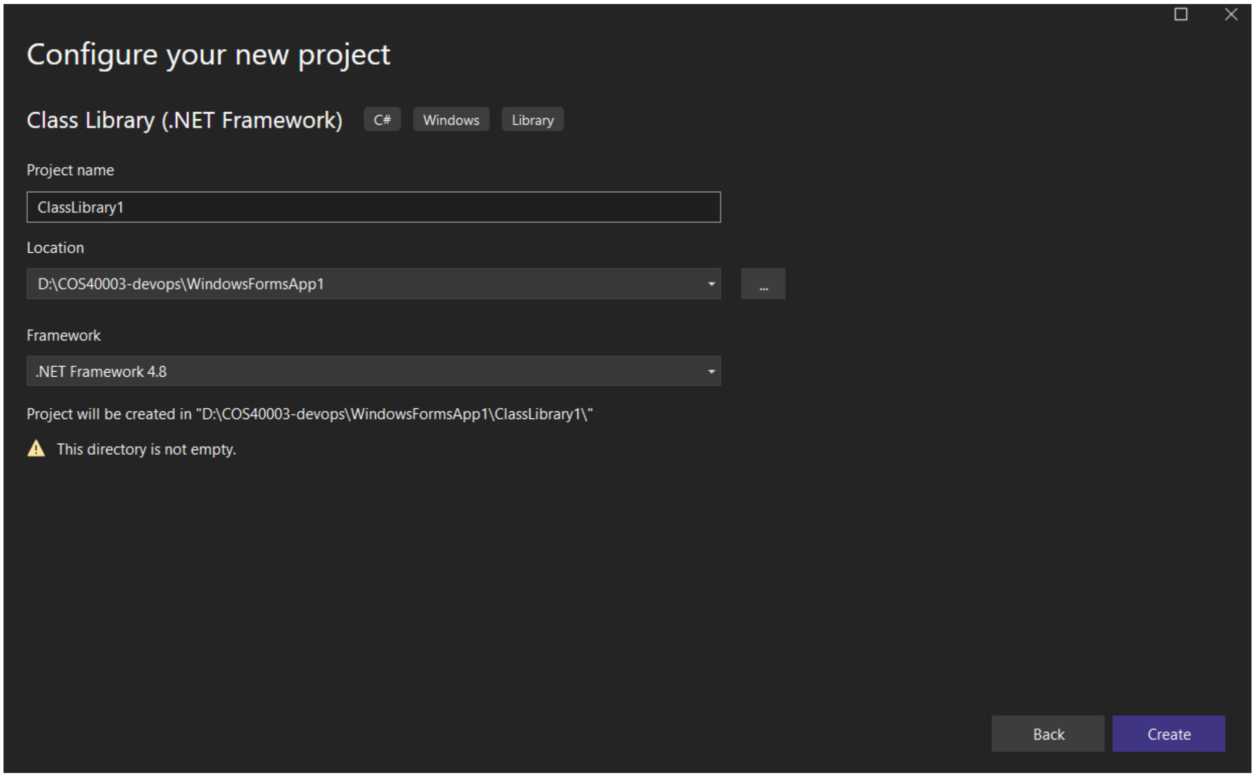
Version: 1.0.1.0

Capabilities:

- Uses all system resources
- Access your Internet connection

C, D: Follow the Wix walkthrough and deploy your own C, C++, or C# with multiple DLLs or dependencies.

Create a new Class Library Project in the solution



Configure your new project

Class Library (.NET Framework) C# Windows Library

Project name: ClassLibrary1

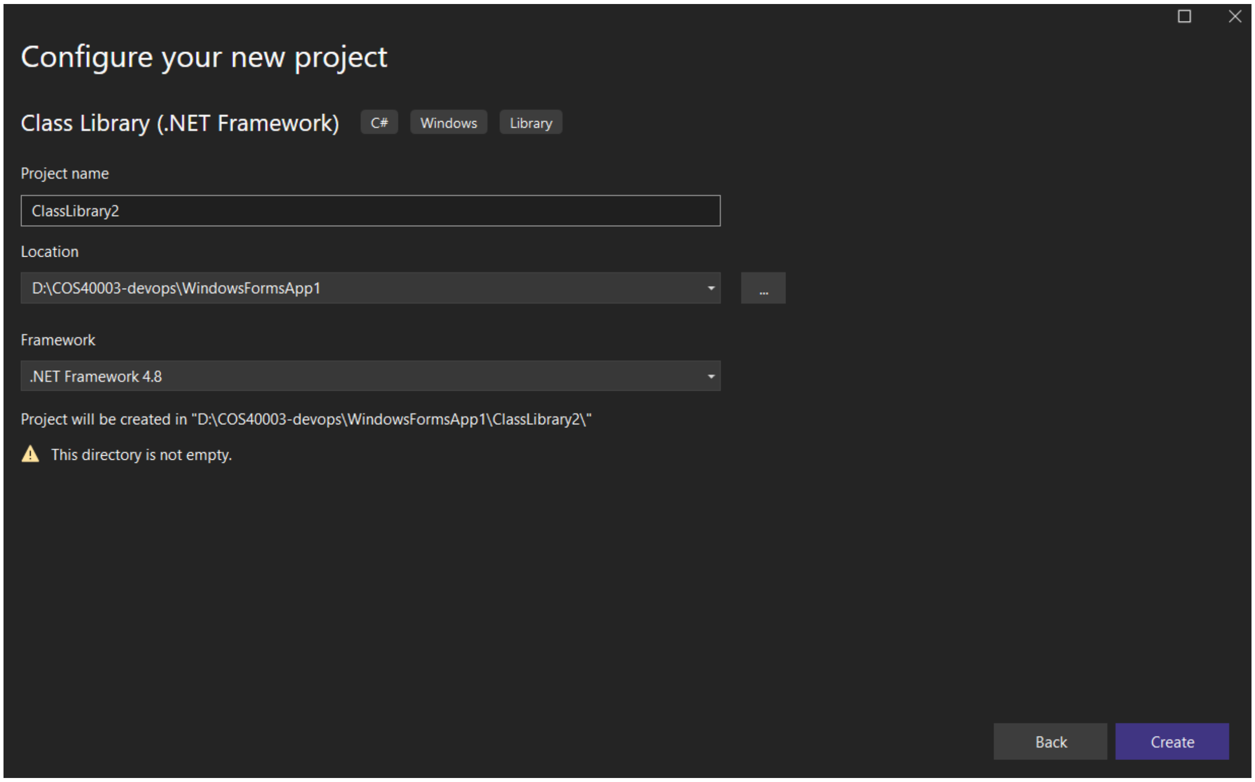
Location: D:\COS40003-devops\WindowsFormsApp1

Framework: .NET Framework 4.8

Project will be created in "D:\COS40003-devops\WindowsFormsApp1\ClassLibrary1\"

⚠ This directory is not empty.

Create another new Class Library Project in the solution



Configure your new project

Class Library (.NET Framework) C# Windows Library

Project name

ClassLibrary2

Location

D:\COS40003-devops\WindowsFormsApp1

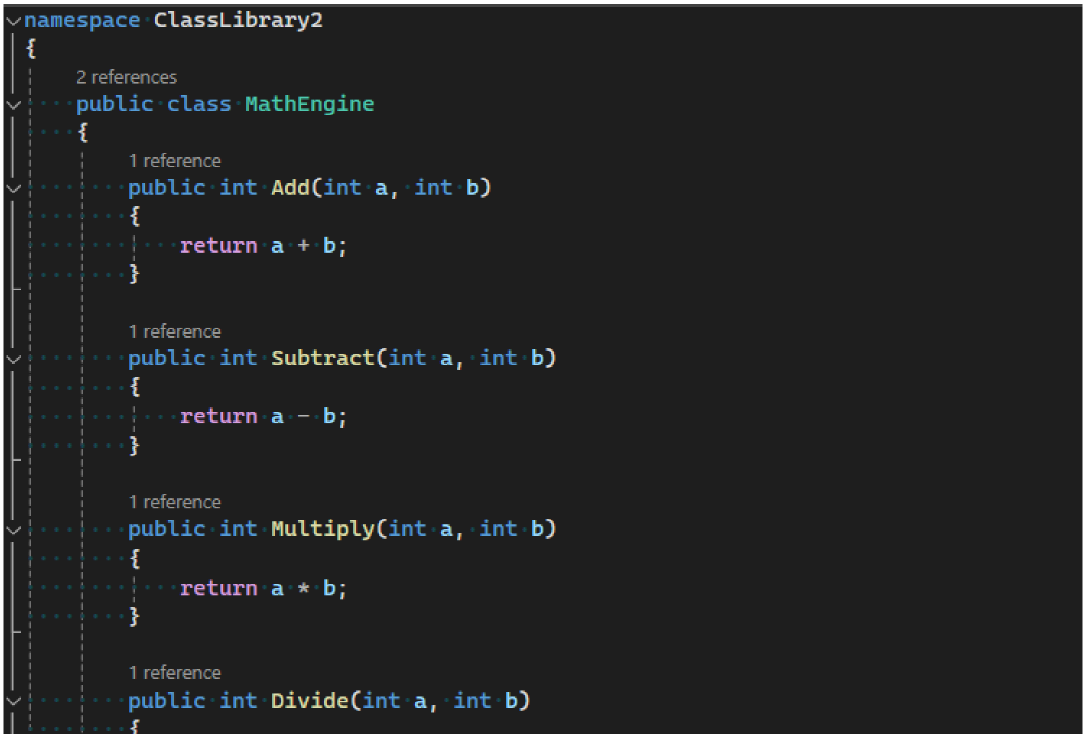
Framework

.NET Framework 4.8

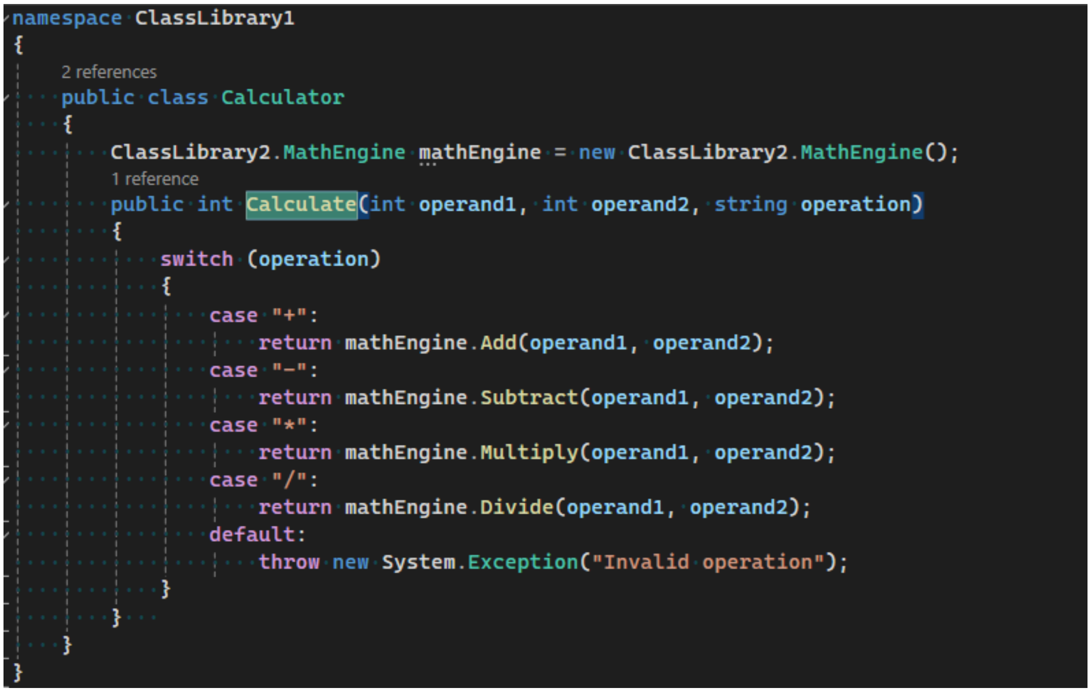
Project will be created in "D:\COS40003-devops\WindowsFormsApp1\ClassLibrary2\"

⚠ This directory is not empty.

In library project 2 create a class MathEngine. In library project 1 create a class Calculator.

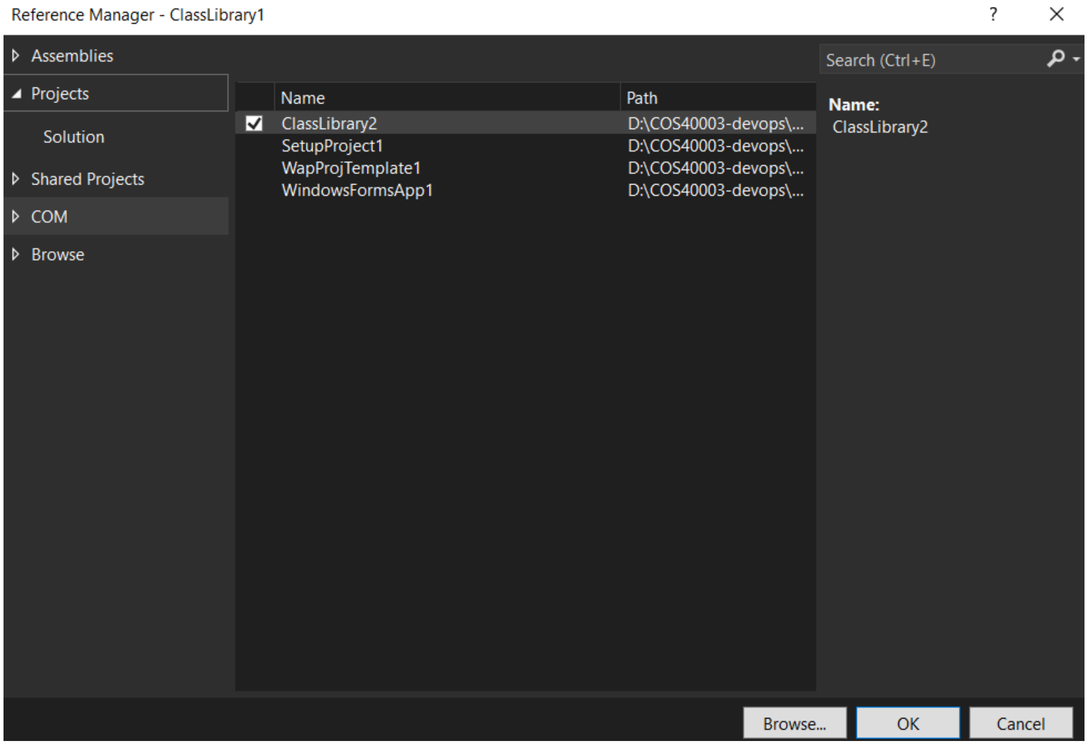


```
namespace ClassLibrary2
{
    2 references
    public class MathEngine
    {
        1 reference
        public int Add(int a, int b)
        {
            return a + b;
        }

        1 reference
        public int Subtract(int a, int b)
        {
            return a - b;
        }

        1 reference
        public int Multiply(int a, int b)
        {
            return a * b;
        }

        1 reference
        public int Divide(int a, int b)
        {
```



```
namespace ClassLibrary1
{
    2 references
    public class Calculator
    {
        ClassLibrary2.MathEngine mathEngine = new ClassLibrary2.MathEngine();
        1 reference
        public int Calculate(int operand1, int operand2, string operation)
        {
            switch (operation)
            {
                case "+":
                    return mathEngine.Add(operand1, operand2);
                case "-":
                    return mathEngine.Subtract(operand1, operand2);
                case "*":
                    return mathEngine.Multiply(operand1, operand2);
                case "/":
                    return mathEngine.Divide(operand1, operand2);
                default:
                    throw new System.Exception("Invalid operation");
            }
        }
    }
}
```

The class from project contains an instance of a class in project 2. Therefore, add reference to project 2 for project 1.

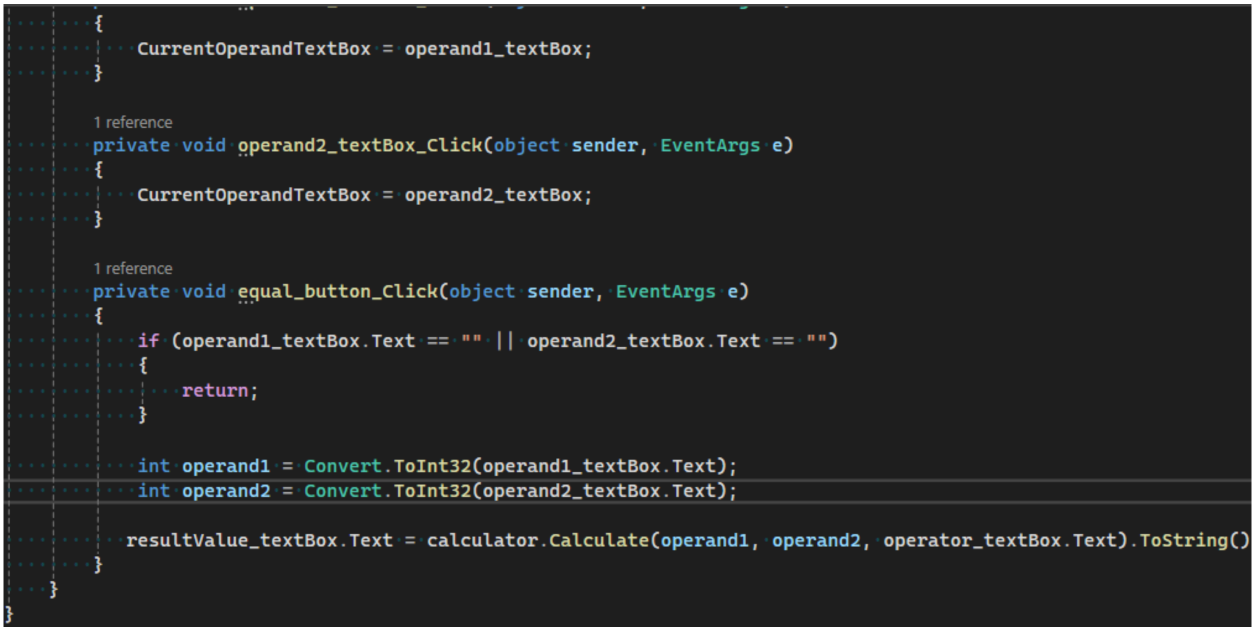


Reference Manager - ClassLibrary1

	Name	Path
☑	ClassLibrary2	D:\COS40003-devops\...
	SetupProject1	D:\COS40003-devops\...
	WapProjTemplate1	D:\COS40003-devops\...
	WindowsFormsApp1	D:\COS40003-devops\...

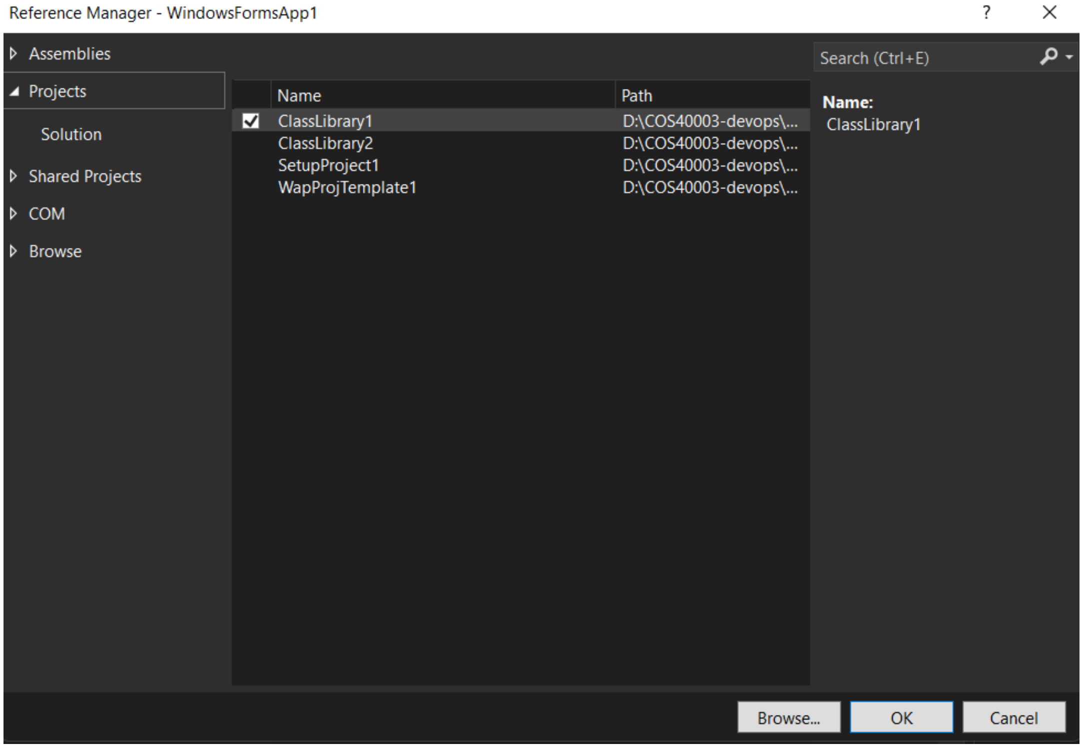
Name:
ClassLibrary2

In the Windows Forms App created earlier, write a simple program to add/subtract/multiple/divide 2 integers the is put into the form



```
        {
            CurrentOperandTextBox = operand1_textBox;
        }

        1 reference
        private void operand2_textBox_Click(object sender, EventArgs e)
        {
            CurrentOperandTextBox = operand2_textBox;
        }

        1 reference
        private void equal_button_Click(object sender, EventArgs e)
        {
            if (operand1_textBox.Text == "" || operand2_textBox.Text == "")
            {
                return;
            }

            int operand1 = Convert.ToInt32(operand1_textBox.Text);
            int operand2 = Convert.ToInt32(operand2_textBox.Text);

           resultValue_textBox.Text = calculator.Calculate(operand1, operand2, operator_textBox.Text).ToString()
        }
    }
}
```

Since the WinFormApp contains a Calculator instance, add reference to the Class Library Project 1 for the WinFormApp

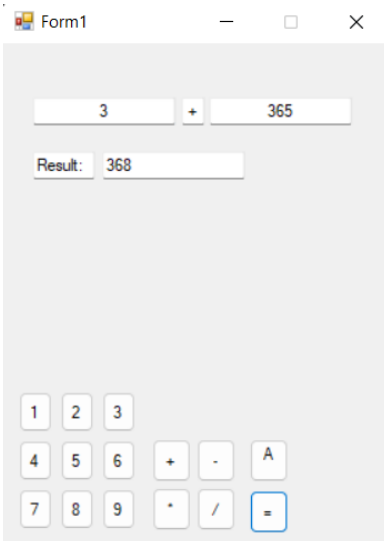


Reference Manager - WindowsFormsApp1

	Name	Path
☑	ClassLibrary1	D:\COS40003-devops\...
	ClassLibrary2	D:\COS40003-devops\...
	SetupProject1	D:\COS40003-devops\...
	WapProjTemplate1	D:\COS40003-devops\...

Name:
ClassLibrary1

Try run the WinFormApp. The app contains 2 text boxes for operand, 1 for result display, 1 for operator display and can be easily modified via the buttons from the bottom.

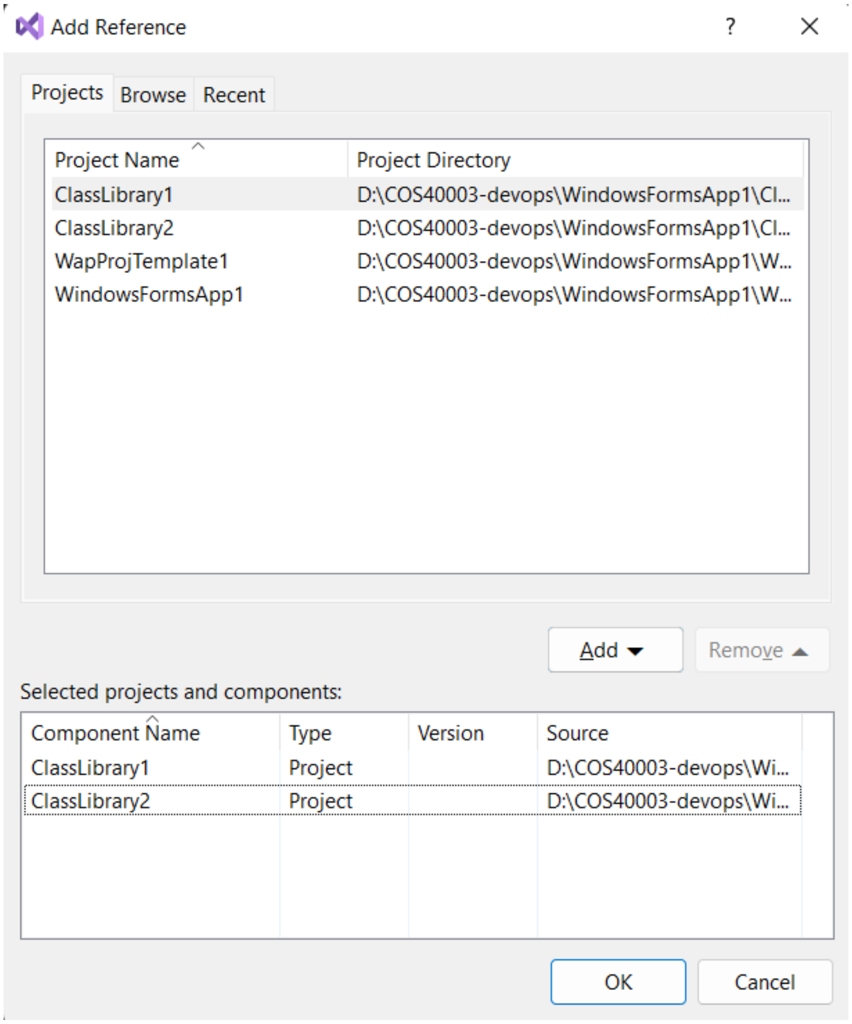


Form1

3 + 365

Result: 368

Now since the Windows Forms App depends on the Library Project 1, which depends on Library Project 2, we need to add both Library Projects and the main Windows Forms App as references in the Wix Set Up Project.



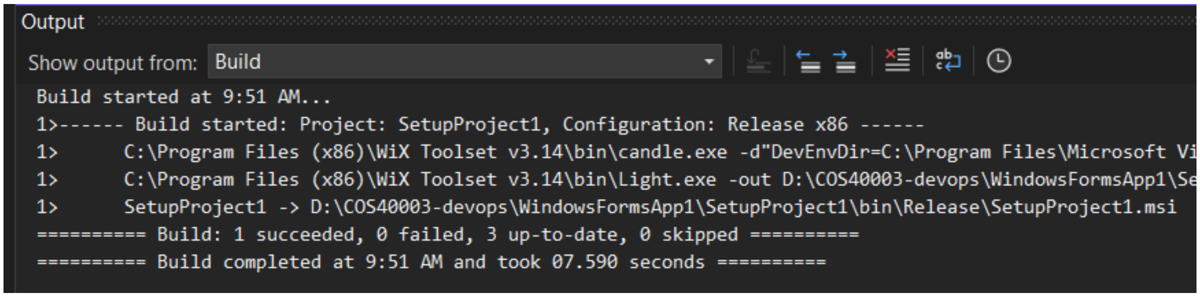
Add Reference

Project Name	Project Directory
ClassLibrary1	D:\COS40003-devops\WindowsFormsApp1\Cl...
ClassLibrary2	D:\COS40003-devops\WindowsFormsApp1\Cl...
WapProjTemplate1	D:\COS40003-devops\WindowsFormsApp1\W...
WindowsFormsApp1	D:\COS40003-devops\WindowsFormsApp1\W...

Selected projects and components:

Component Name	Type	Version	Source
ClassLibrary1	Project		D:\COS40003-devops\Wi...
ClassLibrary2	Project		D:\COS40003-devops\Wi...

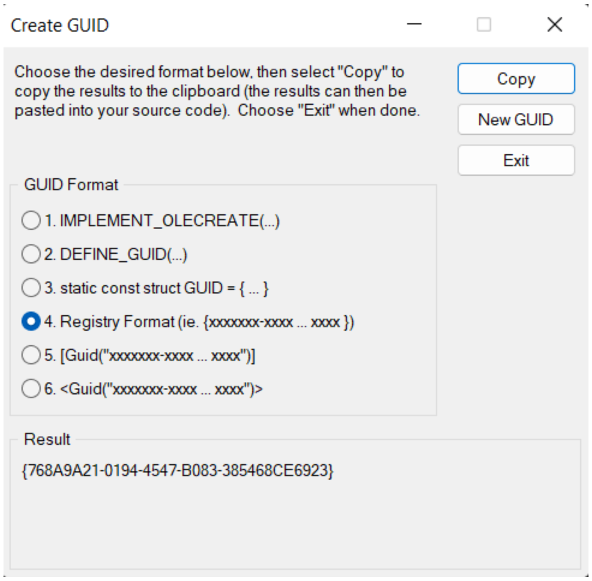
Build the Wix Set Up Project



Show output from: Build

```
Build started at 9:51 AM...
1>------ Build started: Project: SetupProject1, Configuration: Release x86 ------
1>      C:\Program Files (x86)\WiX Toolset v3.14\bin\candle.exe -d"DevEnvDir=C:\Program Files\Microsoft Vi
1>      C:\Program Files (x86)\WiX Toolset v3.14\bin\Light.exe -out D:\COS40003-devops\WindowsFormsApp1\Se
1>      SetupProject1 -> D:\COS40003-devops\WindowsFormsApp1\SetupProject1\bin\Release\SetupProject1.msi
========== Build: 1 succeeded, 0 failed, 3 up-to-date, 0 skipped ==========
========== Build completed at 9:51 AM and took 07.590 seconds ==========
```

Create a GUID to be used for each project

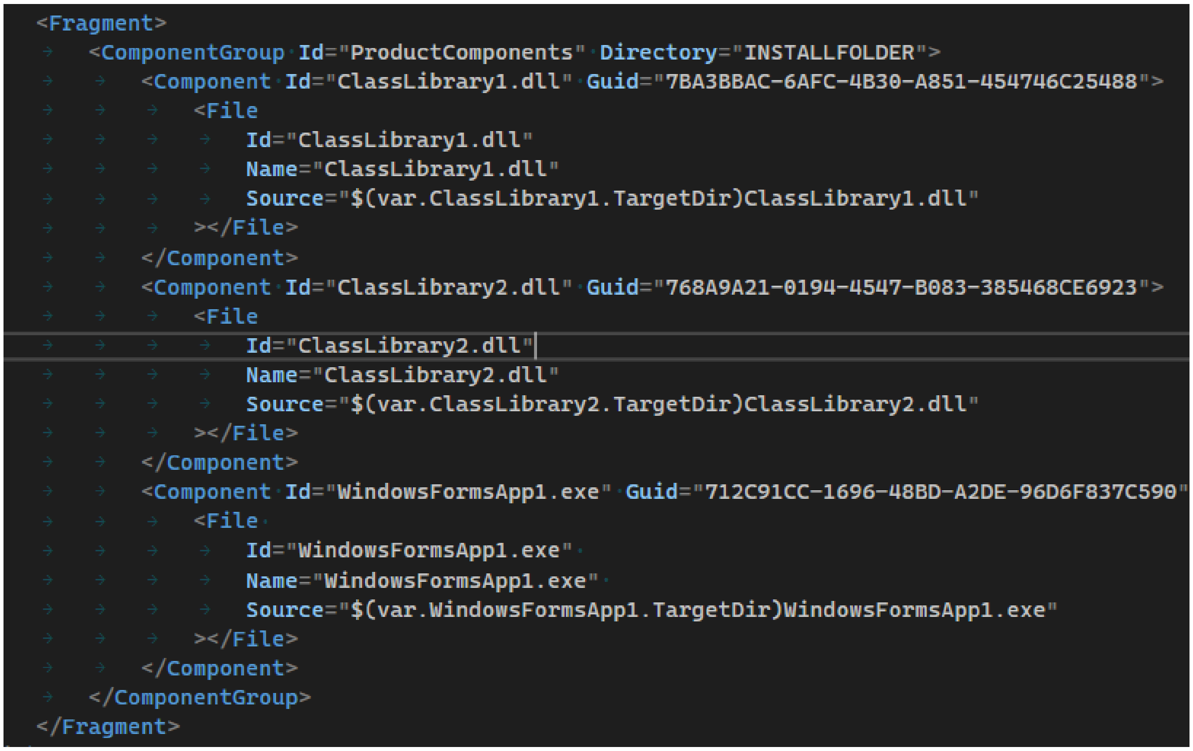


Choose the desired format below, then select "Copy" to copy the results to the clipboard (the results can then be pasted into your source code). Choose "Exit" when done.

○ 1. IMPLEMENT_OLECREATE(...)
○ 2. DEFINE_GUID(...)
○ 3. static const struct GUID = { ... }
● 4. Registry Format (ie. {xxxxxxxx-xxxx ... xxxx })
○ 5. [Guid("xxxxxxxx-xxxx ... xxxx")]
○ 6. <Guid("xxxxxxxx-xxxx ... xxxx")>

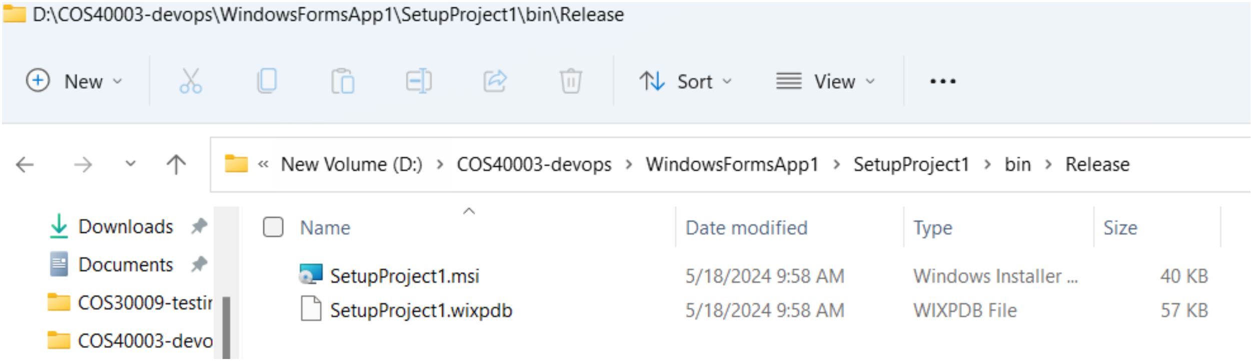
{768A9A21-0194-4547-B083-385468CE6923}

Edit the Product.wxs file in the Wix Set Up Project. If the project reference is added but Product.wxs Components is not modified, the built program will result is error or unexpected behavior.



```
<Fragment>
    <ComponentGroup Id="ProductComponents" Directory="INSTALLFOLDER">
        <Component Id="ClassLibrary1.dll" Guid="7BA3BBAC-6AFC-4B30-A851-454746C25488">
            <File
                Id="ClassLibrary1.dll"
                Name="ClassLibrary1.dll"
                Source="$(var.ClassLibrary1.TargetDir)ClassLibrary1.dll"
            ></File>
        </Component>
        <Component Id="ClassLibrary2.dll" Guid="768A9A21-0194-4547-B083-385468CE6923">
            <File
                Id="ClassLibrary2.dll"
                Name="ClassLibrary2.dll"
                Source="$(var.ClassLibrary2.TargetDir)ClassLibrary2.dll"
            ></File>
        </Component>
        <Component Id="WindowsFormsApp1.exe" Guid="712C91CC-1696-48BD-A2DE-96D6F837C590"
            <File
                Id="WindowsFormsApp1.exe"
                Name="WindowsFormsApp1.exe"
                Source="$(var.WindowsFormsApp1.TargetDir)WindowsFormsApp1.exe"
            ></File>
        </Component>
    </ComponentGroup>
</Fragment>
```

Build the Wix Set Up Project, then the below files are created

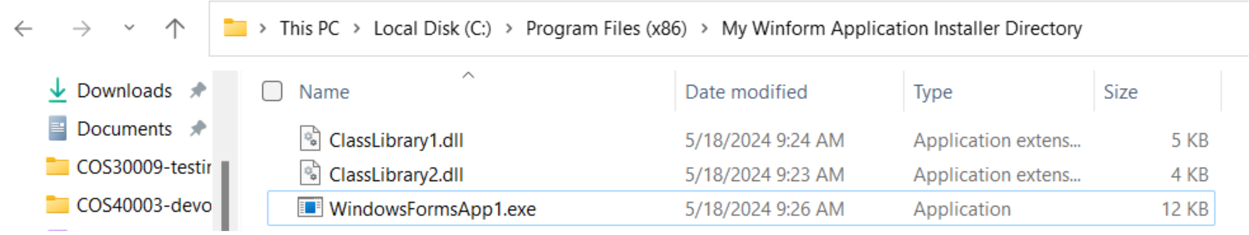


D:\COS40003-devops\WindowsFormsApp1\SetupProject1\bin\Release

New
Sort
View
New Volume (D:) › COS40003-devops › WindowsFormsApp1 › SetupProject1 › bin › Release
Downloads
Documents
COS30009-testir
COS40003-devo

Name	Date modified	Type	Size
SetupProject1.msi	5/18/2024 9:58 AM	Windows Installer ...	40 KB
SetupProject1.wixpdb	5/18/2024 9:58 AM	WIXPDB File	57 KB

Execute the .msi file



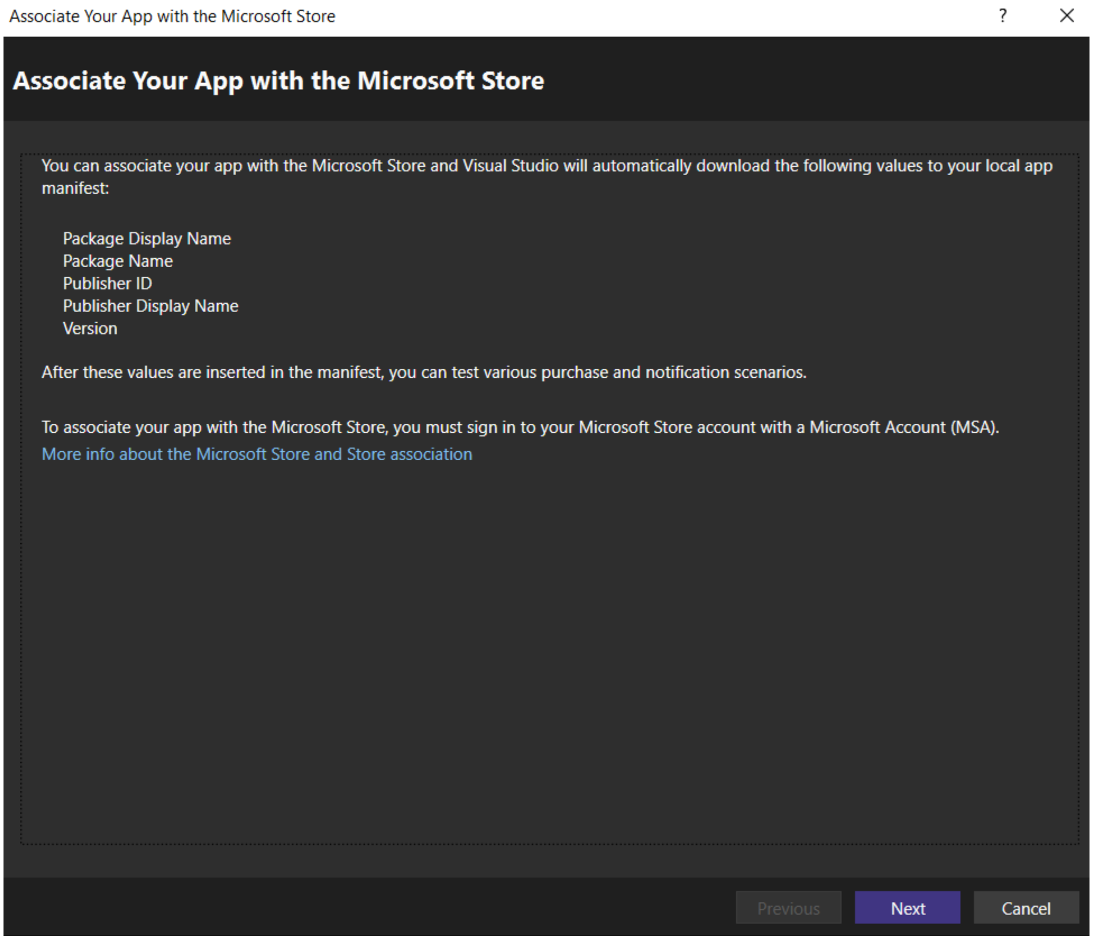
This PC › Local Disk (C:) › Program Files (x86) › My Winform Application Installer Directory
Downloads
Documents
COS30009-testir
COS40003-devo

Name	Date modified	Type	Size
ClassLibrary1.dll	5/18/2024 9:24 AM	Application extens...	5 KB
ClassLibrary2.dll	5/18/2024 9:23 AM	Application extens...	4 KB
WindowsFormsApp1.exe	5/18/2024 9:26 AM	Application	12 KB

- ➔ **We have done deploying our Windows Forms App with 2 DLLs dependency.**

HD Task: Deploy your application to the Microsoft store for public access and downloads.

Use the Windows Application Packaging Project from previous steps. Go to "publish" → "associate your app with Microsoft Store"



Associate Your App with the Microsoft Store

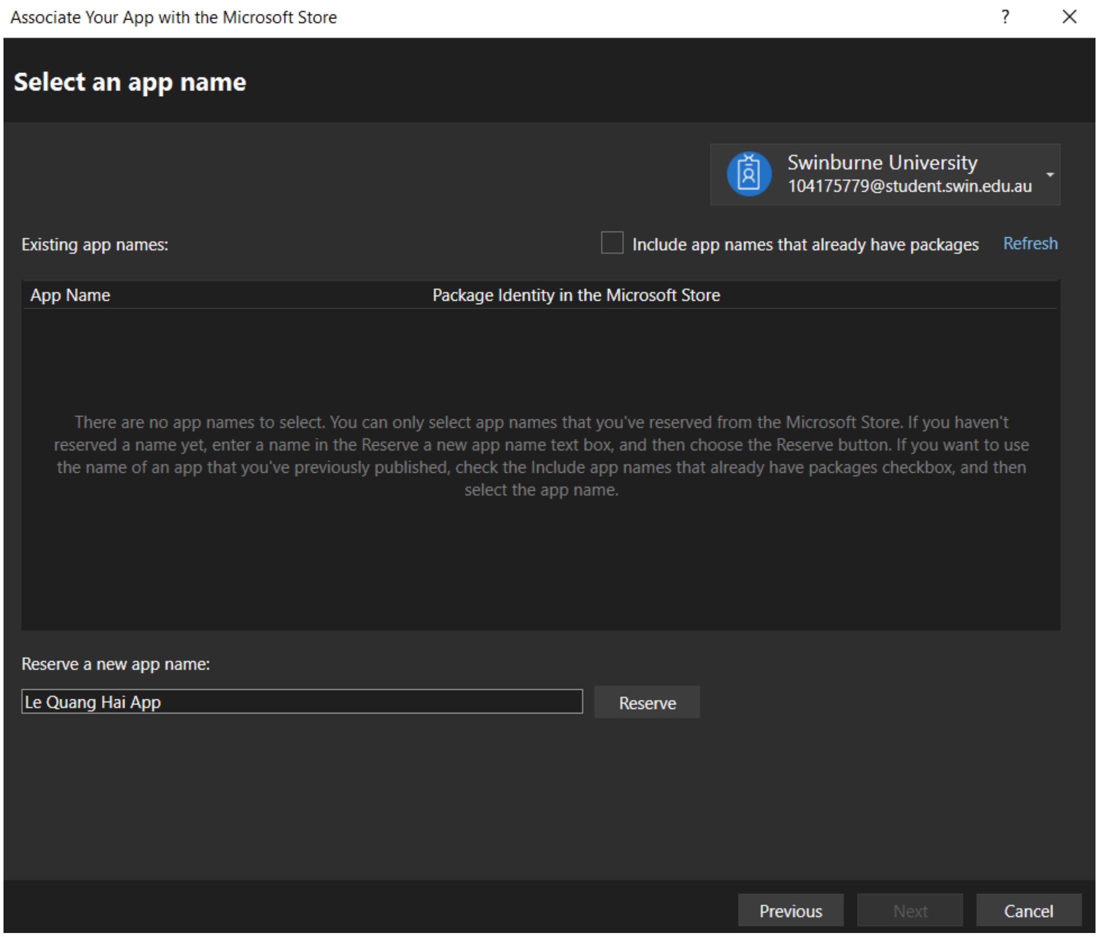
You can associate your app with the Microsoft Store and Visual Studio will automatically download the following values to your local app manifest:

Package Display Name
Package Name
Publisher ID
Publisher Display Name
Version

After these values are inserted in the manifest, you can test various purchase and notification scenarios.

To associate your app with the Microsoft Store, you must sign in to your Microsoft Store account with a Microsoft Account (MSA).
More info about the Microsoft Store and Store association

Select an app name to reserve for the app



Associate Your App with the Microsoft Store

Select an app name

Swinburne University
104175779@student.swin.edu.au

Existing app names: □ Include app names that already have packages Refresh

App Name	Package Identity in the Microsoft Store

There are no app names to select. You can only select app names that you've reserved from the Microsoft Store. If you haven't reserved a name yet, enter a name in the Reserve a new app name text box, and then choose the Reserve button. If you want to use the name of an app that you've previously published, check the Include app names that already have packages checkbox, and then select the app name.

Reserve a new app name: Le Quang Hai App