Visual Studio Configuration

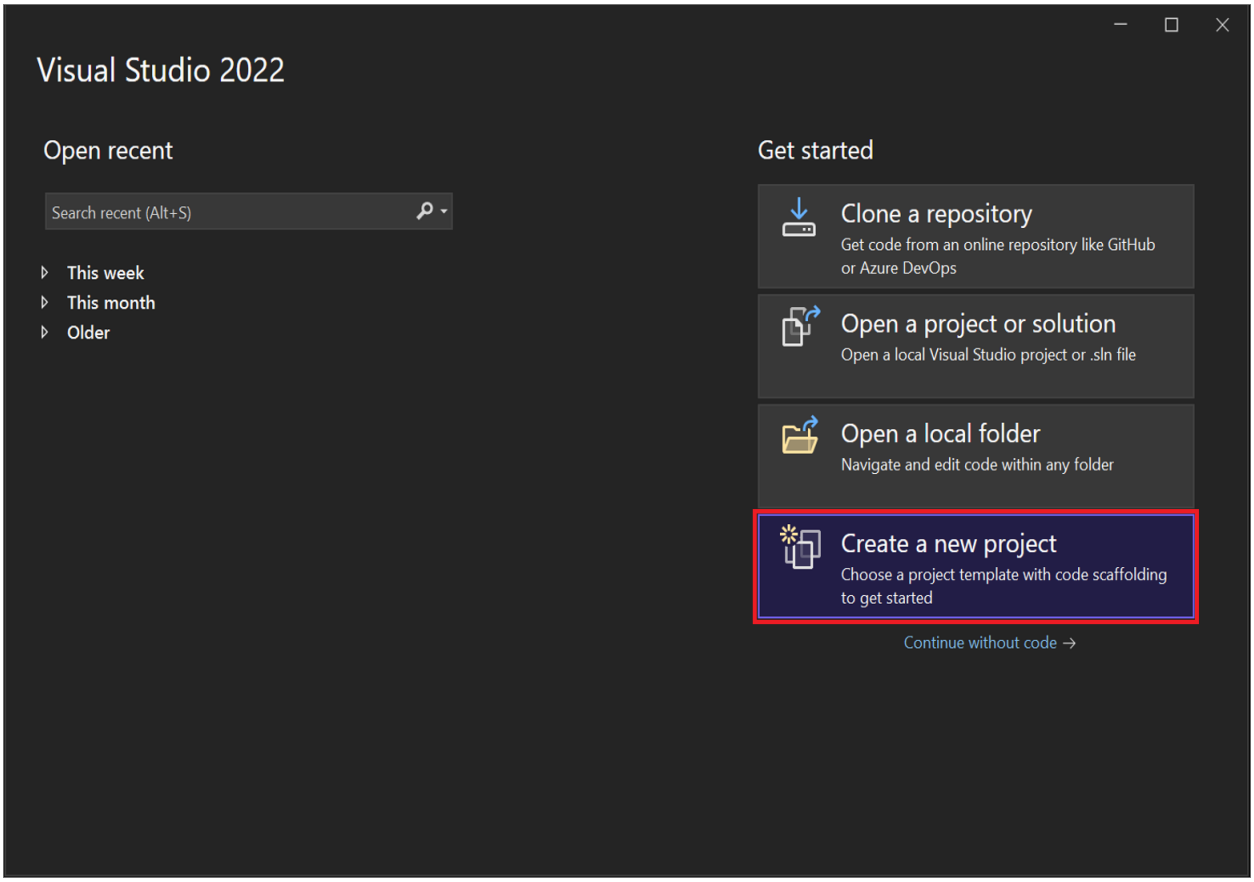
PAVITHRA C

UID: 293947

➢ Create a new project

Open the "Create a new project" dialog

There are multiple ways to create a new project in Visual Studio. When you first open Visual Studio, the start window appears, and from there, you can select **Create a new project**.

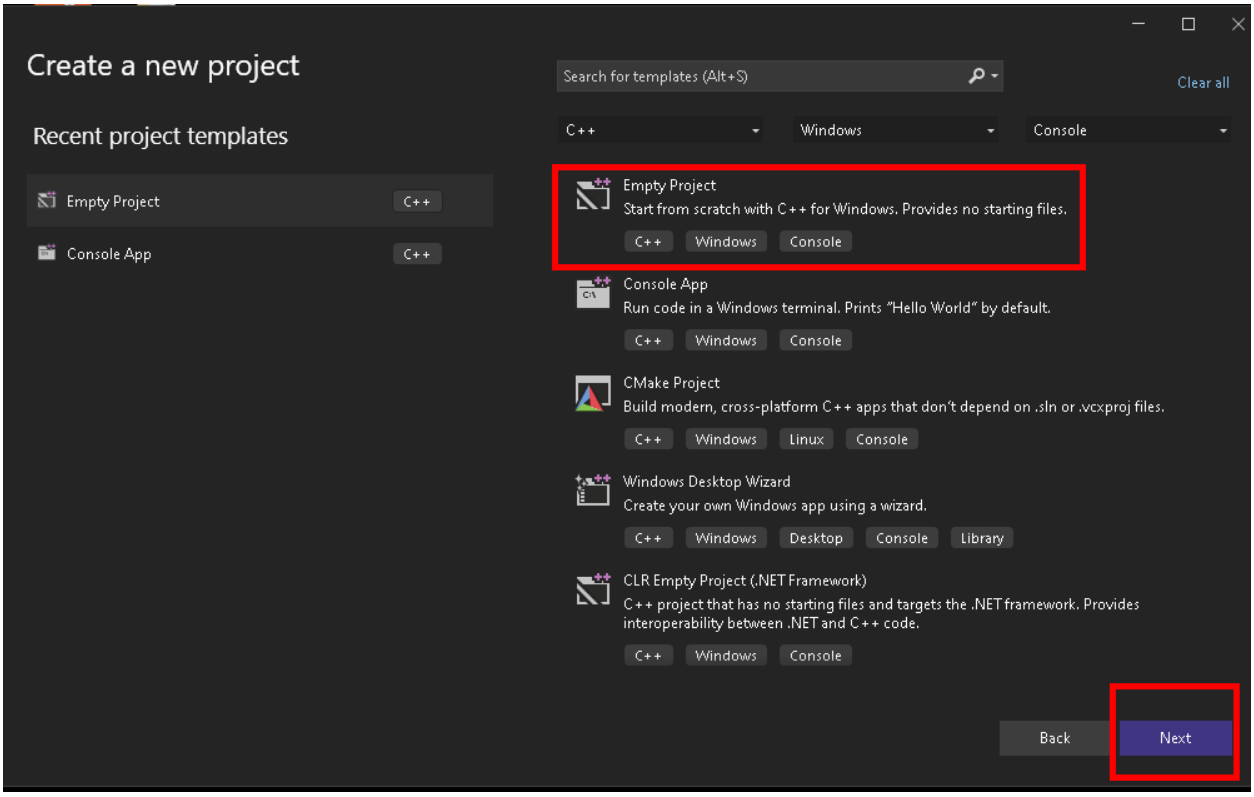


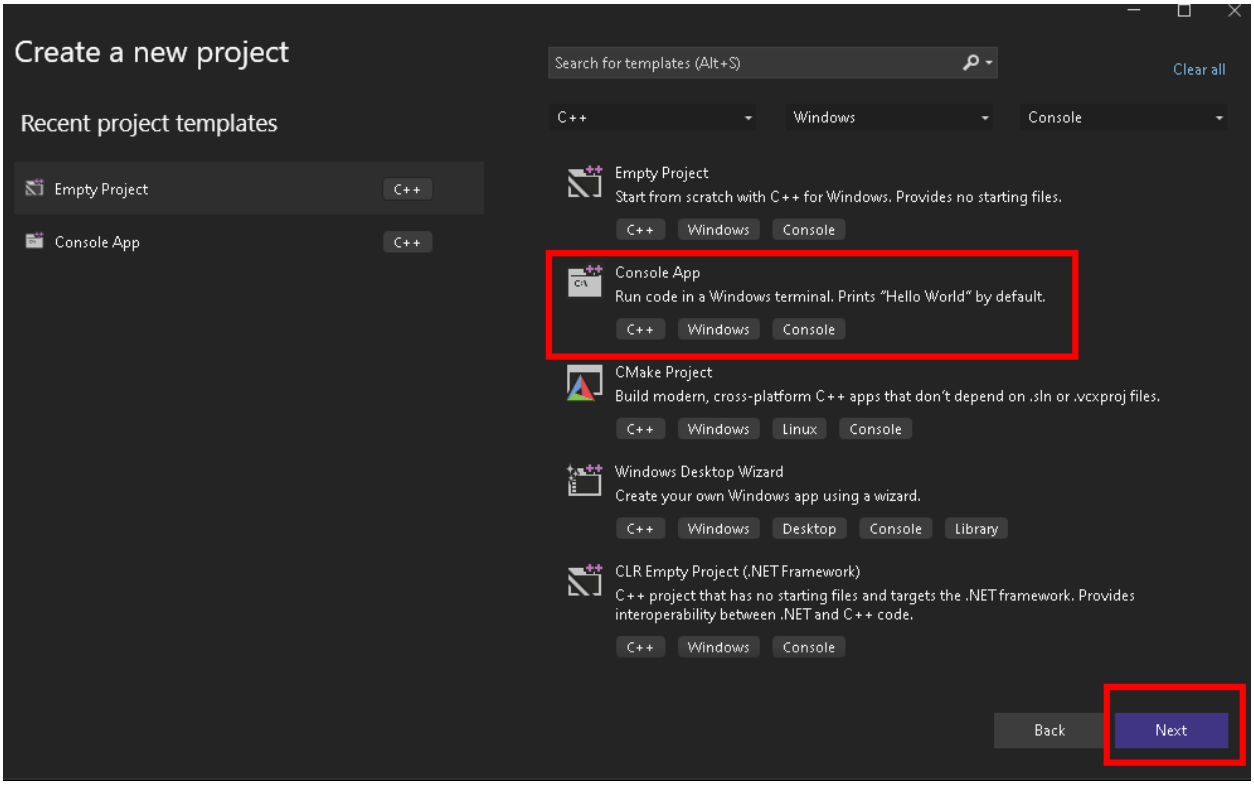
If the Visual Studio development environment is already open, you can create a new project by choosing **File** > **New** > **Project** on the menu bar. You can also select the **New Project** button on the toolbar

➢ Select a template type

On the Create a new project dialog, a list of your recently selected templates appears on the left. The templates are sorted by most recently used.

If you're not selecting from the recently used templates, you can filter all available project templates by Language.



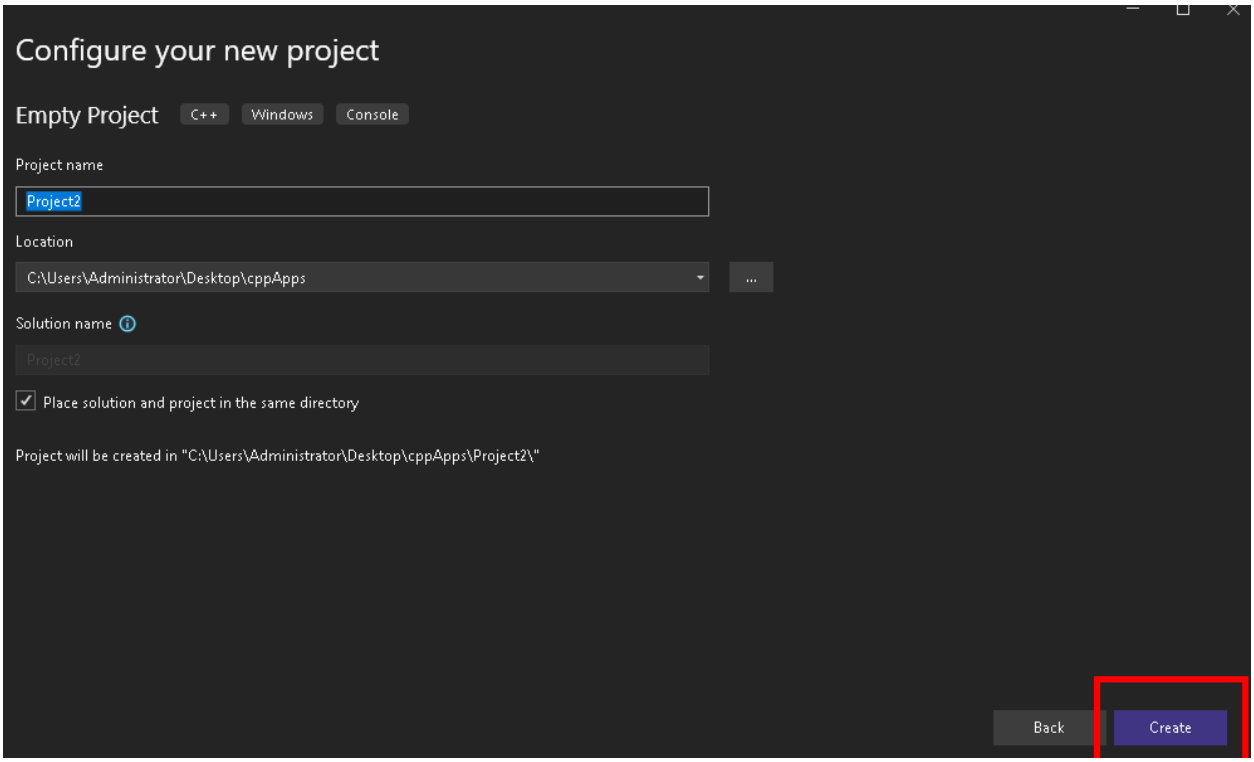


The tags that appear under each template correspond to the three dropdown filters (language, platform, and project type).

Select a template and then select **Next**.

➢ Configure your new project

The Configure your new project dialog has options to name your project (and solution), select a disk location, and more.



Choose place for solution and project I the same directory or different,

Select **Create**.

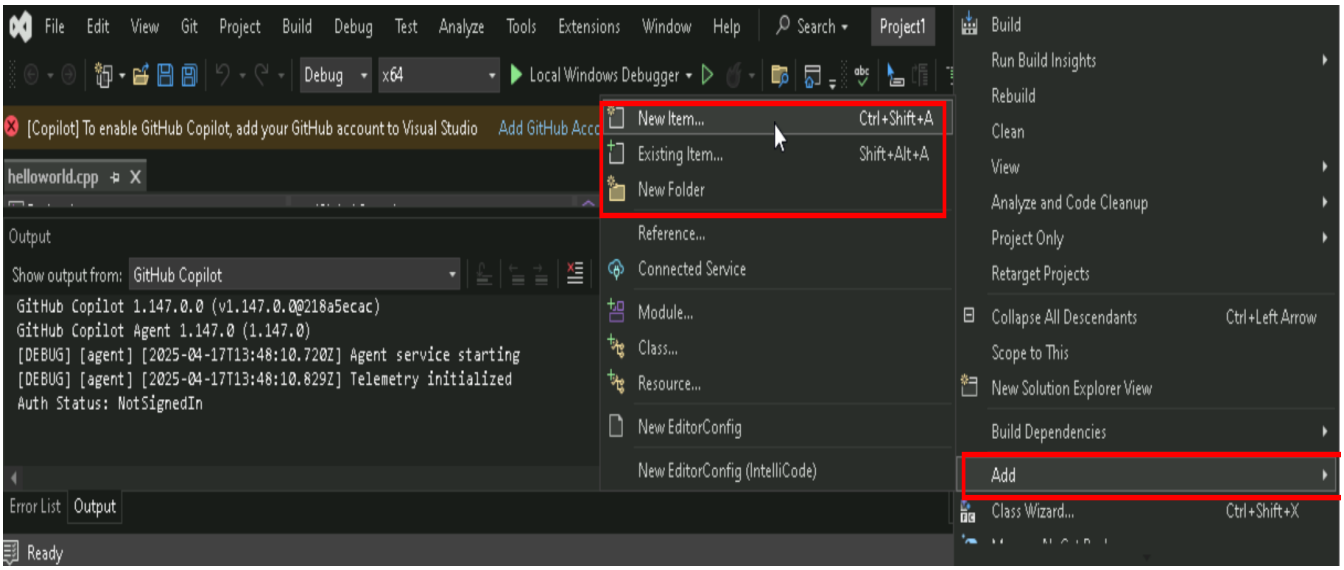
➢ Solution Explorer toolbar

UI element	Action
Back button	Toggle between search results
Forward button	Toggle between search results
Home button	Return to the default view
Switch Views button	Switch between the default Solution view and the optional Folder view
Pending Changes Filter button & drop-down menu	View open files or files with pending changes
Sync with Active Document button	Locate a file from the code editor
Refresh button (conditional)	Appears only when you select a dependency, such as a function or a package
Collapse All button	Collapse the file view in the main window
Show All Files button	View all files, including unloaded projects
Properties button	View and change settings for specific files and components
Preview Selected Items button	View a selected file or component in the code editor

➢ Create a new file , folder

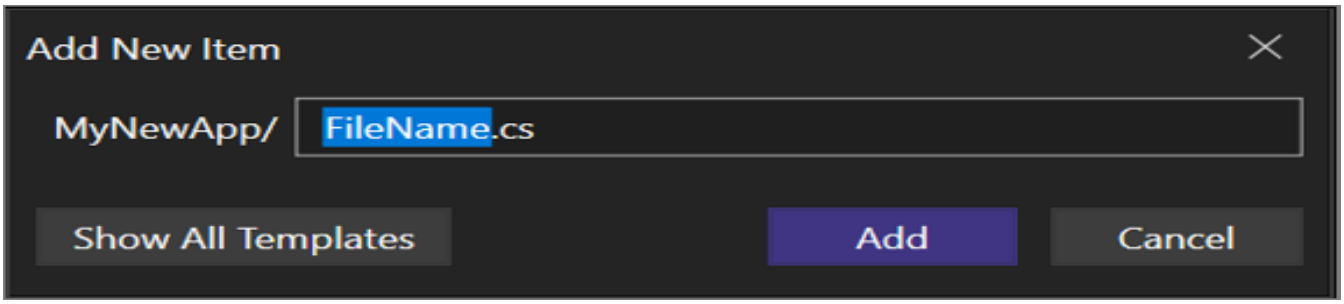
- **The Add menu**

In the Solution Explorer context menu, one of the most useful options is the Add fly-out menu. From it, you can add another project to a solution. You can also add an item to a project, and more.



- **The Quick Add dialog**

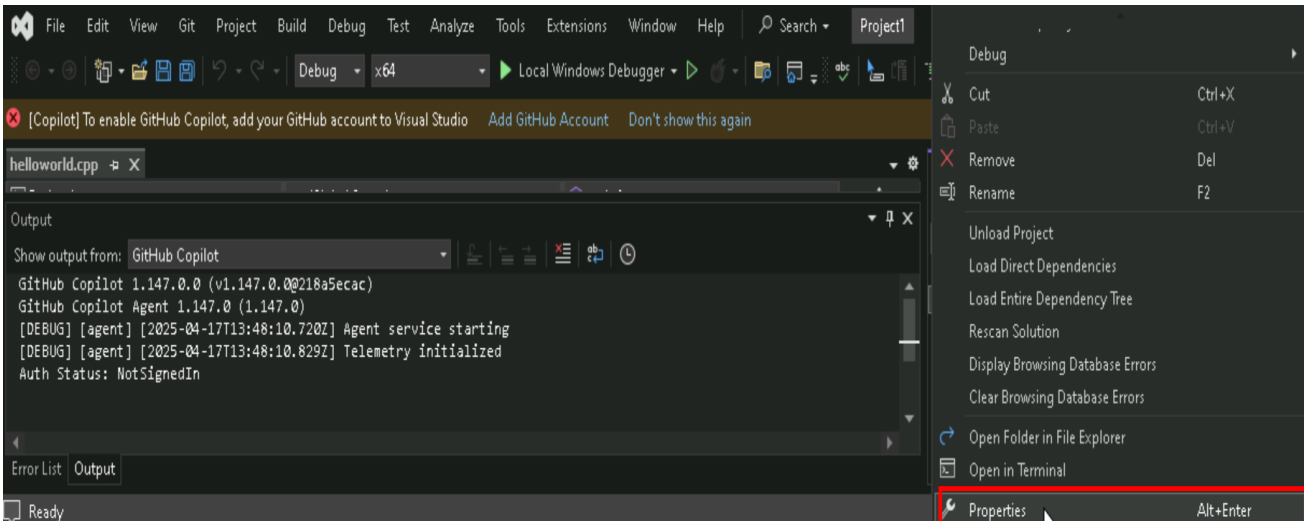
You can use the Quick Add dialog to create an empty file or a simple C# class. In Solution Explorer, right-click on a folder or project and then select **Add > New Item**.



➢ Properties

Solution properties

To access properties on the solution, right-click the solution node in Solution Explorer and select Properties. What you see in the context menu from the Solution node also depends on your project type, programming language, or platform.



After you create a new project, you can use Solution Explorer to view and manage the project and solution and their associated items

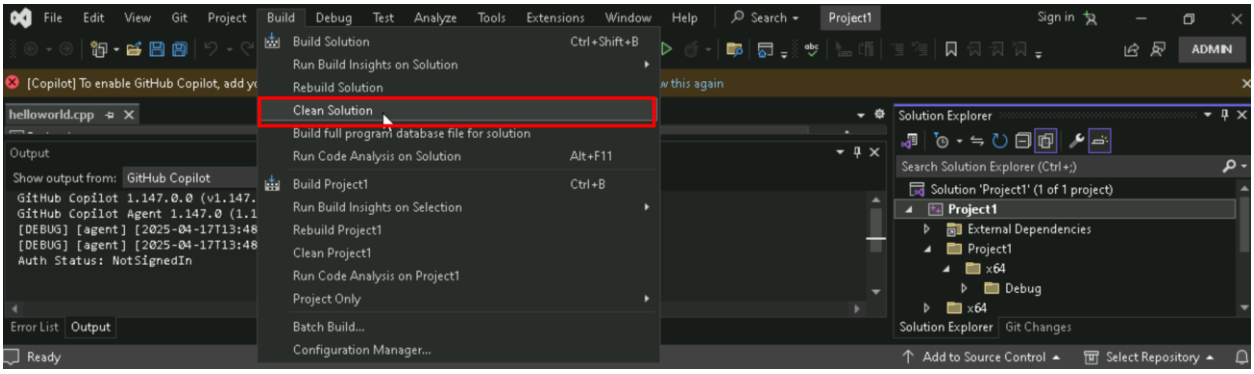
➢ Build Menu

- **Clean the solution:**

On the menu bar, choose Build > Clean Solution

This will remove all compiled assemblies and object files for all projects within the solution.

To delete any intermediate and output files. With only the project and component files left, new instances of the intermediate and output files can then be built.

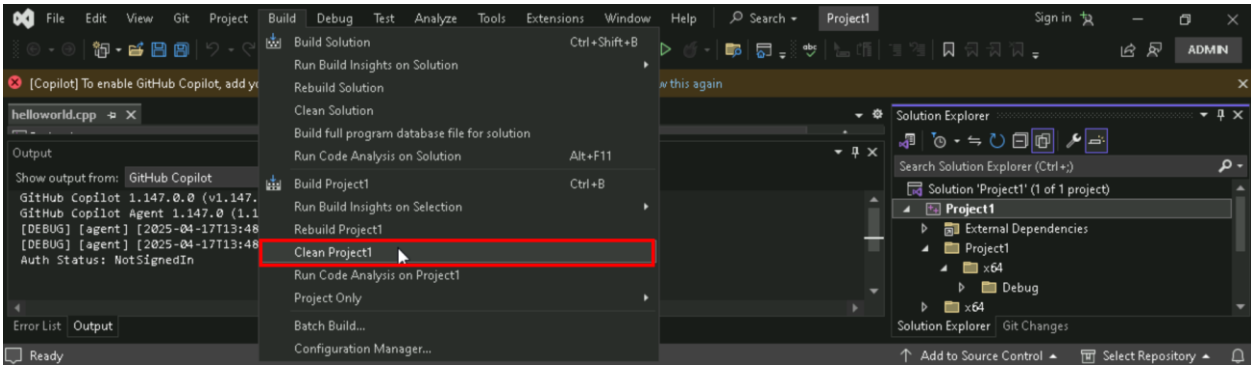


- **Cleaning a Project:**

Open your project in Visual Studio.

Go to Build > Clean ProjectName (where ProjectName is the name of your project).

This will remove compiled assemblies and object files for the specific project you selected.

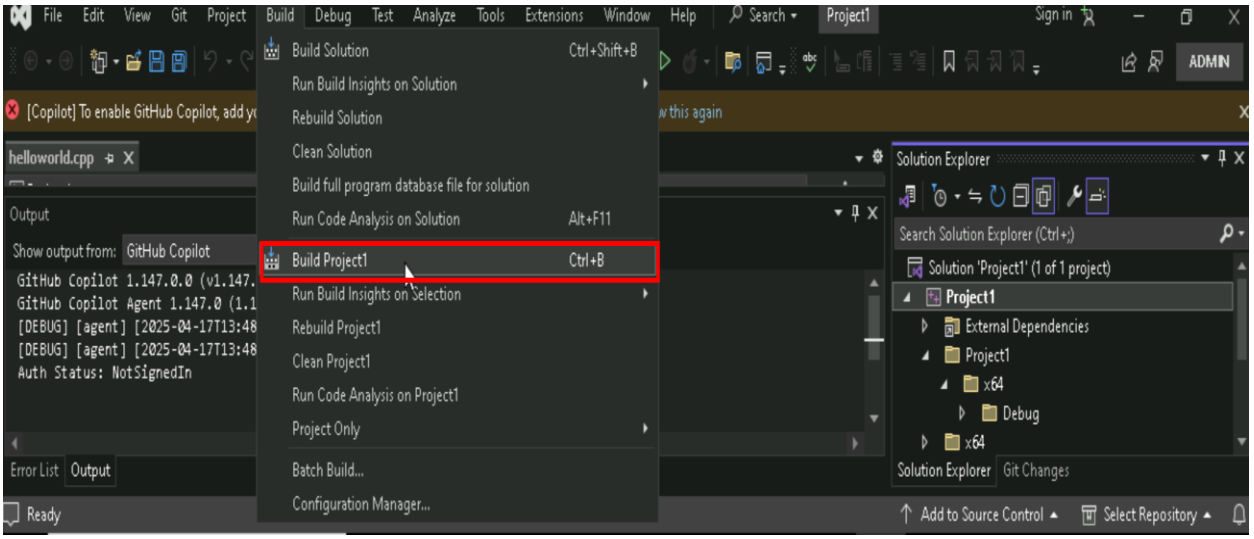


- **Build a Project:**

On the menu bar, choose **Build**, and then choose either **Build** *ProjectName* or **Rebuild** *ProjectName*.

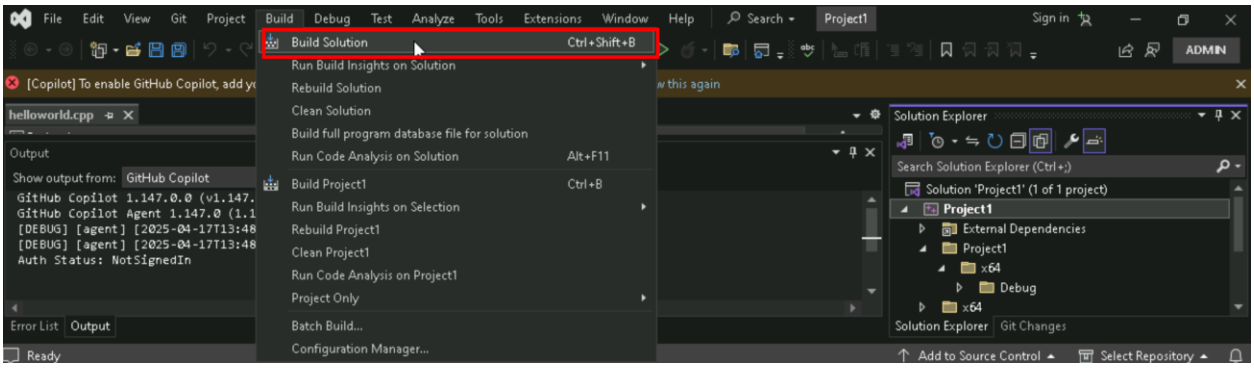
Choose **Build** *ProjectName* to build only those project components that have changed since the most recent build.

Choose **Rebuild** *ProjectName* to "clean" the project and then build the project files and all project components.

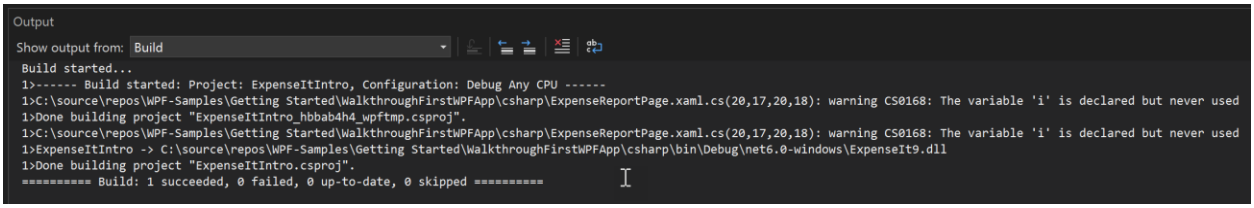


- **Build a Solution :**

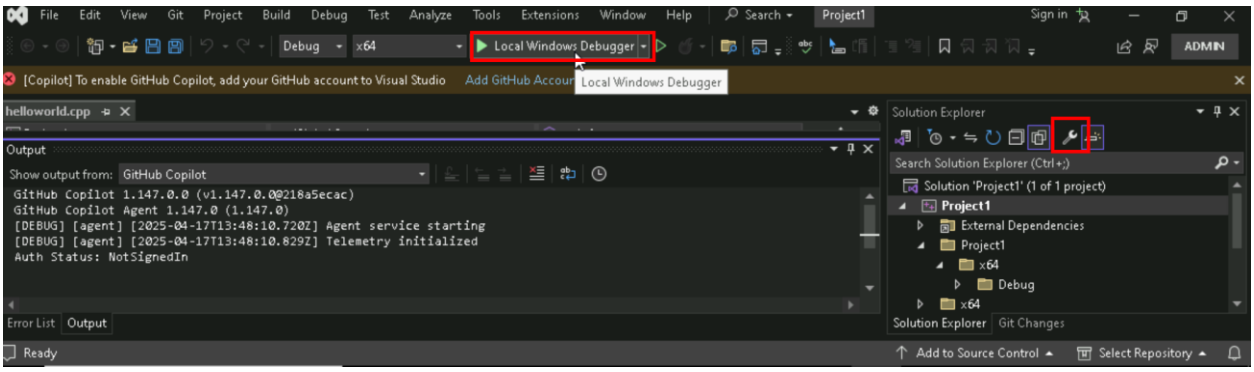
To compile only those project files and components that have changed since the most recent build.



The Output window displays the results of the build. The build succeeded.



➢ Local Windows Debugger



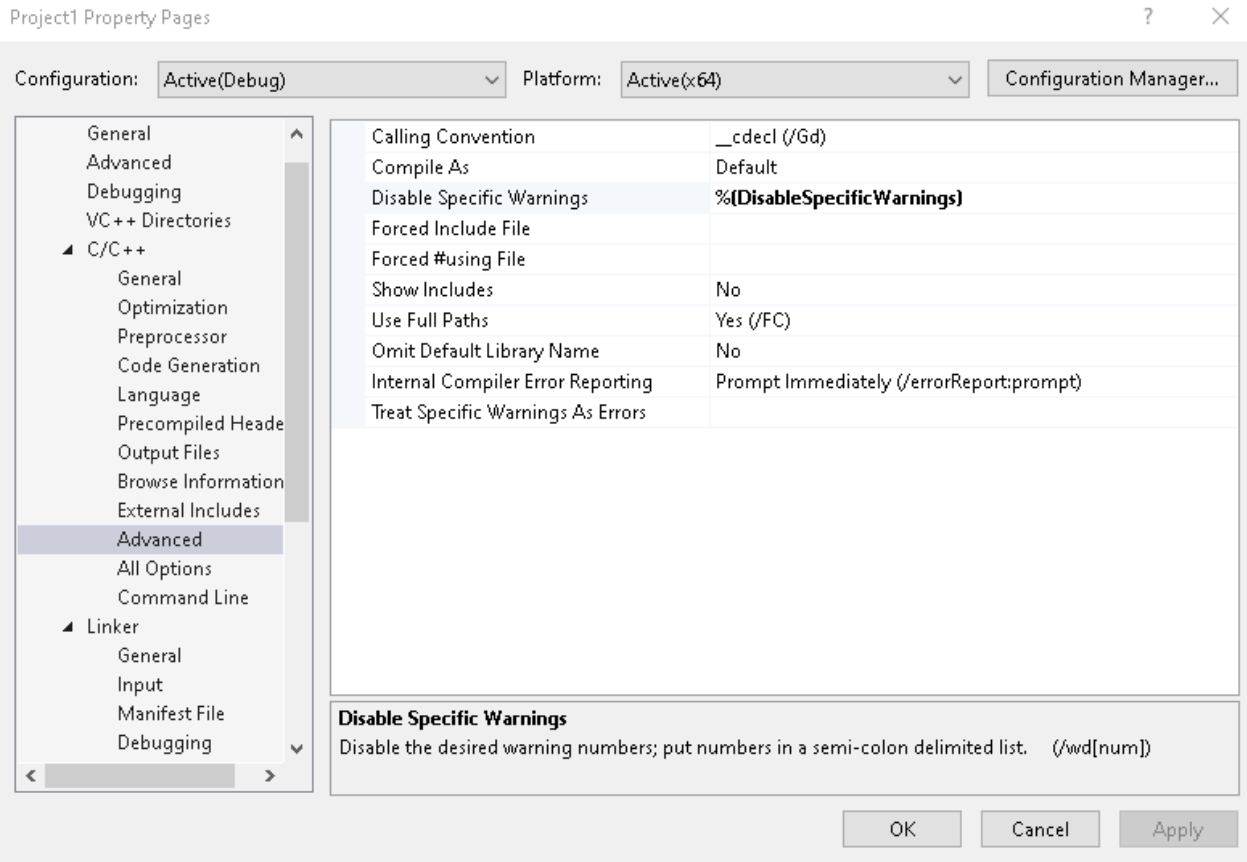
To debug on a local PC, select **Local Windows Debugger** from the debugger dropdown on the Visual Studio toolbar

➢ Disable Specific Warning

- **Method 1 :**

In **Solution Explorer**, choose the project or source file

On the menu bar, choose **Property Pages**.



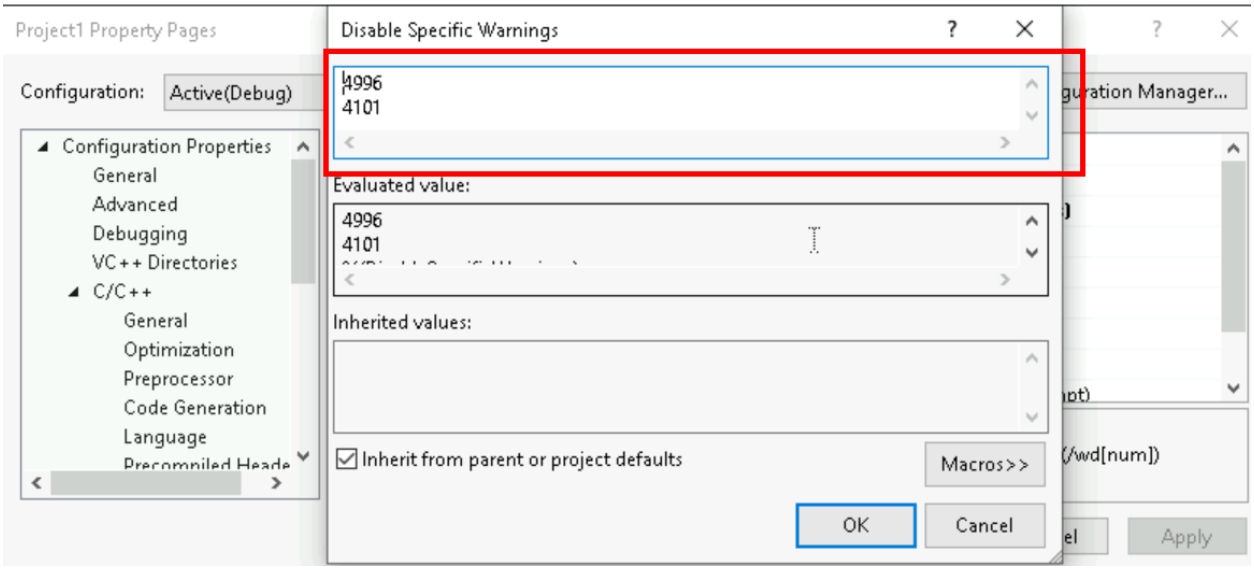
Choose the Configuration Properties category, choose the C/C++ category, and then choose the Advanced page.

Perform the following steps:

- In the Disable Specific Warnings box, specify the error codes of the warnings that you want to suppress, separated by a semicolon.

- In the Disable Specific Warnings box, choose Edit to display more options.

Choose the OK button, and then rebuild the solution.



- **Method 2 :**

In **Solution Explorer**, choose the project or source file

On the menu bar, choose **Property Pages**.

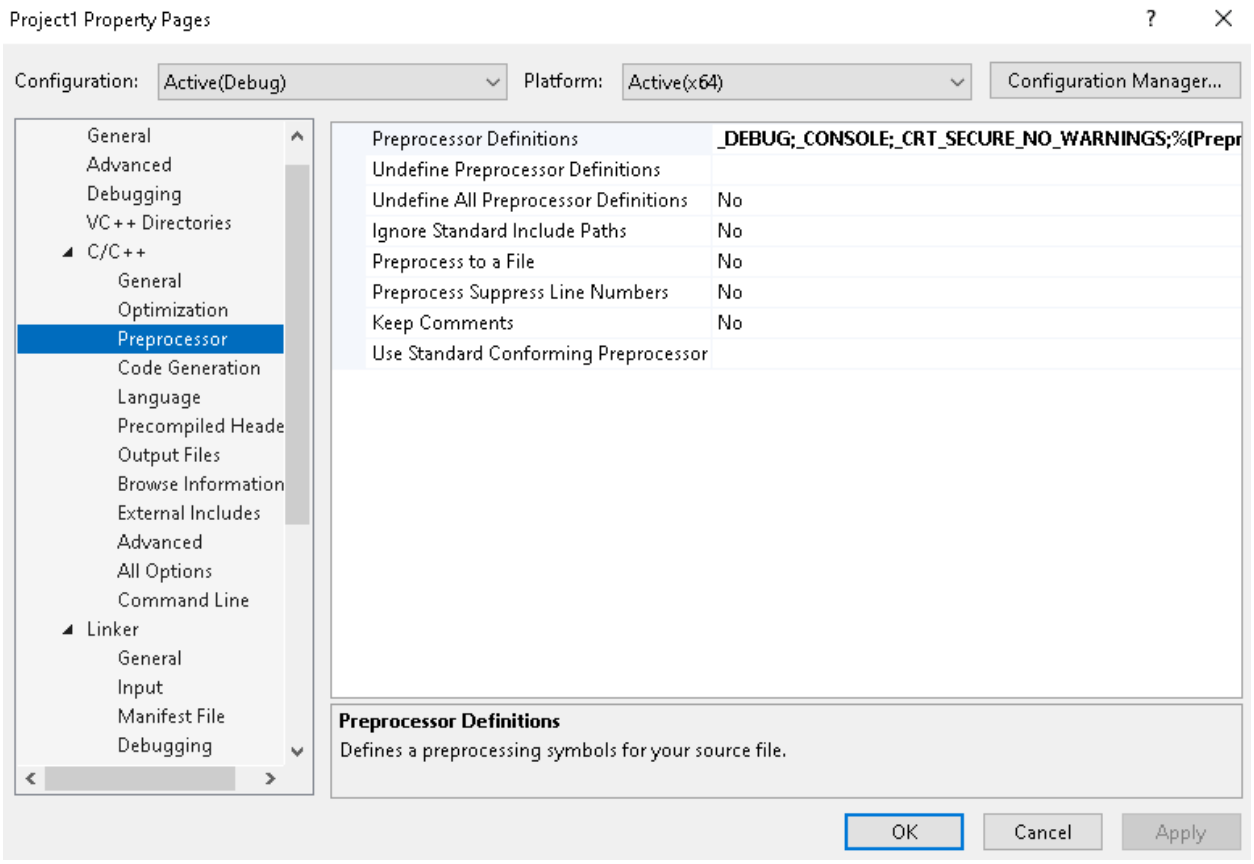
Choose the Configuration Properties category, choose the C/C++ category, and then choose the Preprocessor page.

Perform the following steps:

- In the Preprocessor definitions box, specify the error type of the warnings that you want to suppress, separated by a semicolon.

- In the Preprocessor definitions box, choose Edit to display more options.

Choose the OK button, and then rebuild the solution.

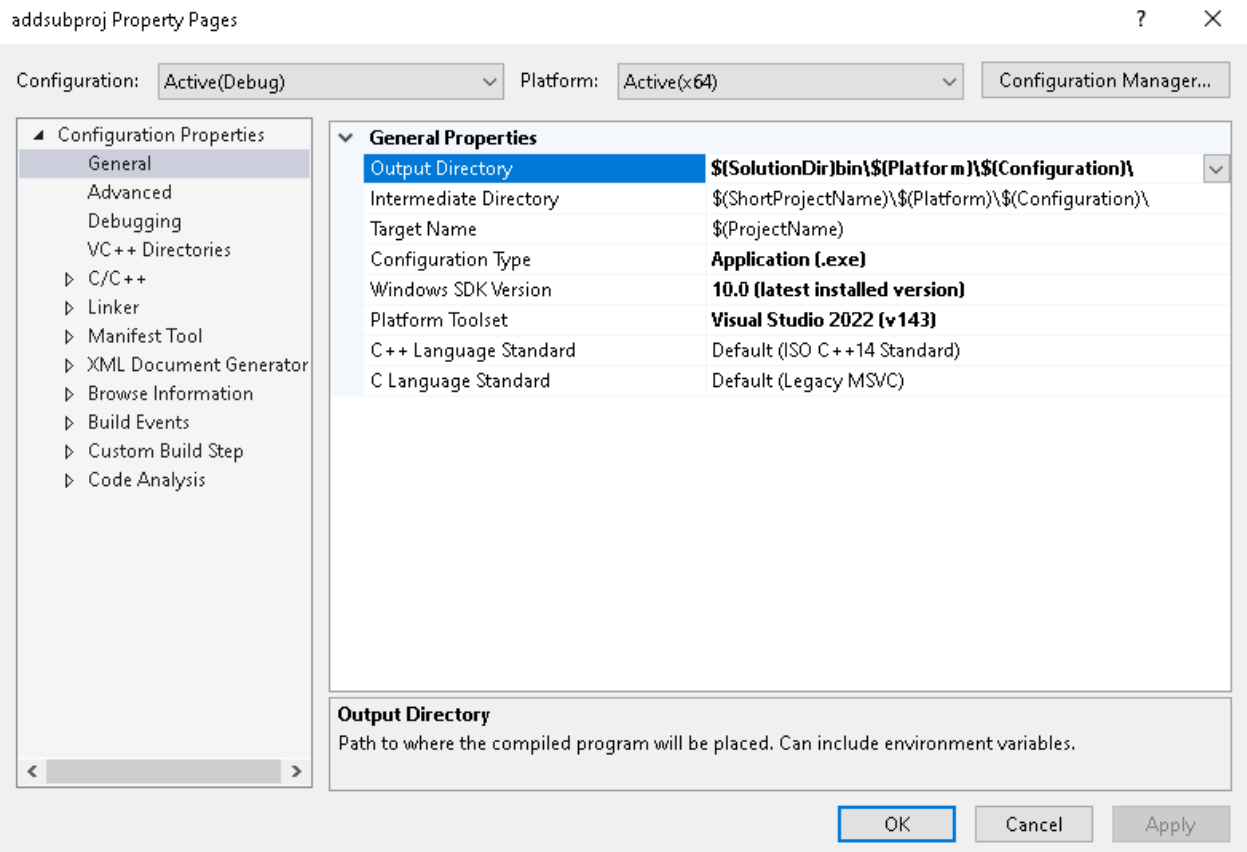


The **Output** window no longer shows the warnings that you specified.

➢ Output Directory

Specifies the location of the output files for the project's configuration. The path is relative; if you enter an absolute path, the absolute path is saved in the project.

The default path is bin\Debug.

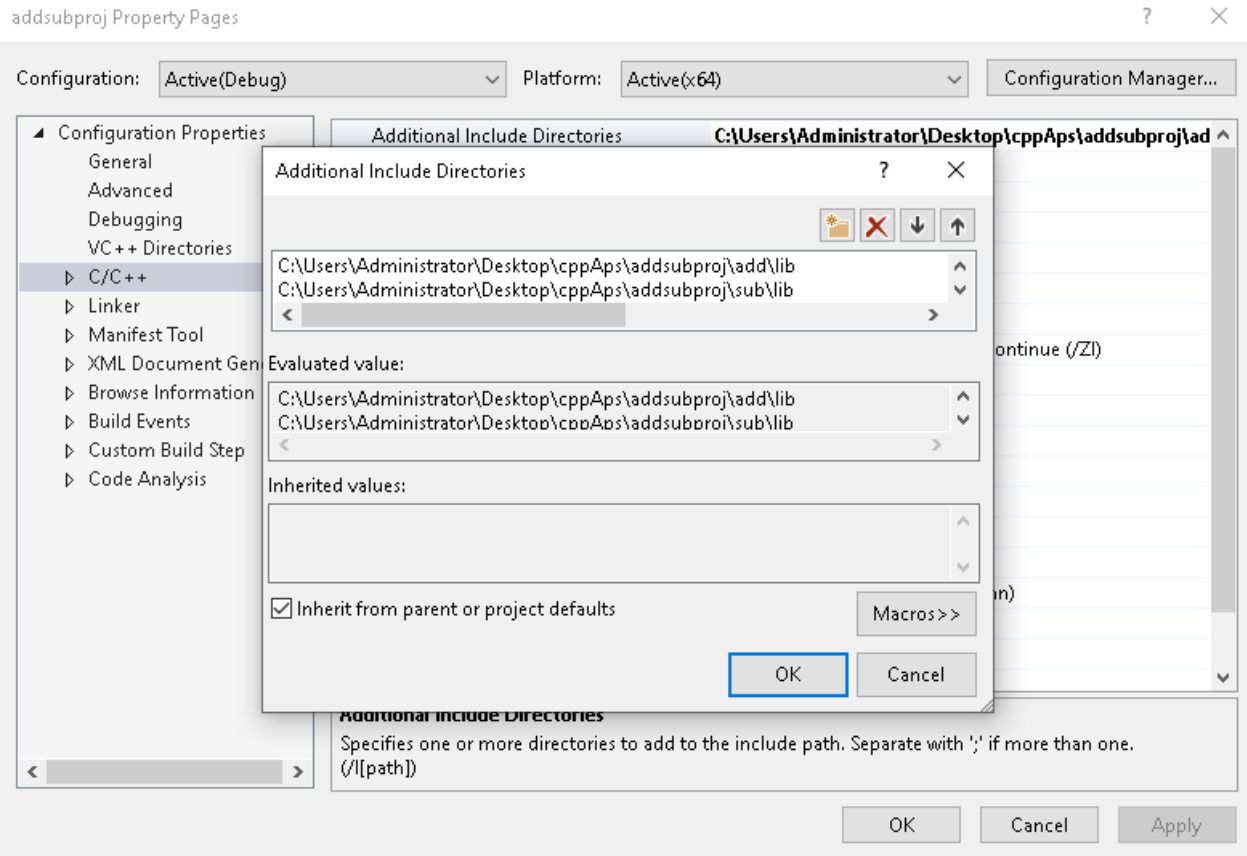


➢ Additional Include Directories

In Solution Explorer, choose the project or source file

On the menu bar, choose Property Pages.

Choose the Configuration Properties category, choose the C/C++ category, and then choose the General page.



Perform the following steps:

- In the Additional Include Directories box, choose Edit to display more options.
- In the Additional Include Directories box, specify the path of the directory of the include file that you want to add.

- Choose add new, to add another include directories.
- Choose the OK button, and then rebuild the solution.

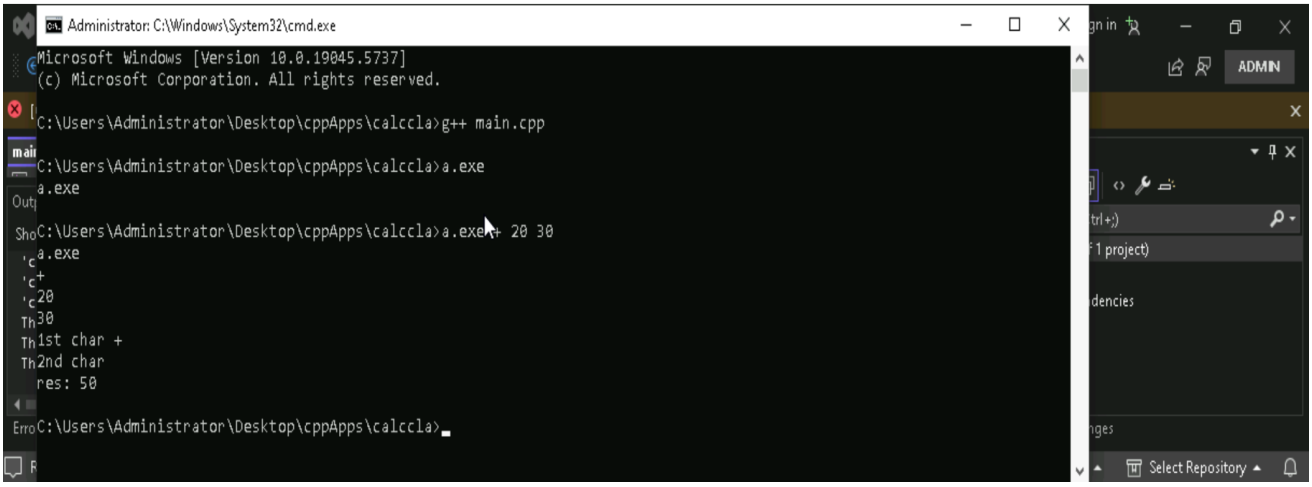
Command Line Arguments

Command-line arguments are arguments that are passed to a program when it is executed from the command line or terminal.There are two ways to execute, they are as follows:-

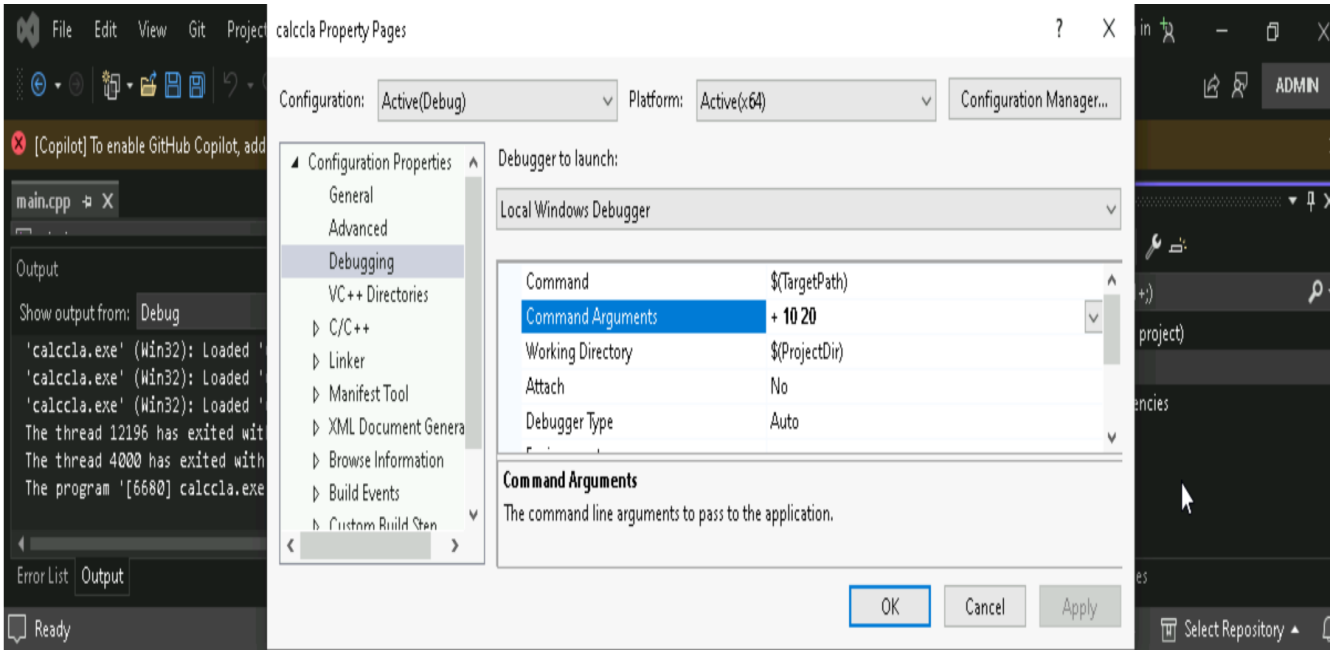
- **Method 1:**
  The arguments are passed to a program when it is executed from the command prompt.

Perform the following steps:

- In the command prompt, compile the process.
- After compilation process, the arguments are passed .
- The application itself becomes an argument.



- **Method 2:**



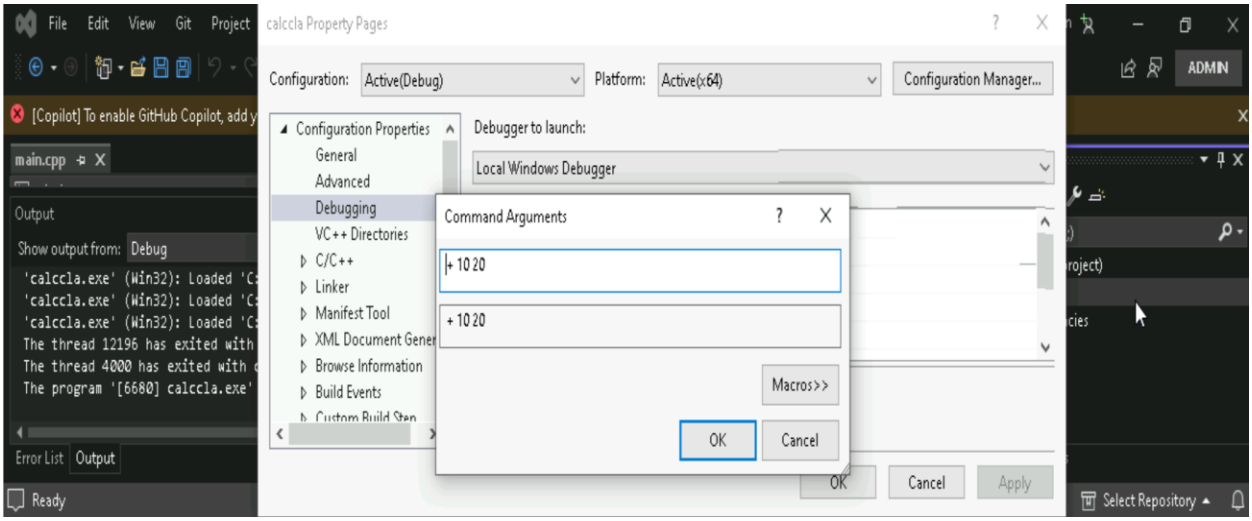
In Solution Explorer, choose the project or source file

On the menu bar, choose Property Pages.

Choose the Configuration Properties category, choose the Debugging category.

Perform the following steps:

- Go to Command Arguments box.
- In the Command Arguments, choose Edit to display more options.
- In the Command Arguments box, the arguments are passed.



- Choose the Apply button.
- Choose the OK button, and then rebuild the solution.