# Setup Visual Studio Code for C++ Development

## Overview

Visual Studio Code (VS Code) is a wonderful code editor. VS Code is Very powerful and light weight, but with that, it is not equipped right out of the box for C++ development. No worries though! I will be going over a basic setup for VS code, that allows for building and debugging C++ projects. One unique thing about this tutorial is that I will also be providing different types of details and short cut keys that are very useful for screen reader users, because I myself am completely blind and use a screen reader. Alrighty, now with that out of the way, let’s get started!

## Things to know

You will need to have the following installed on your machine to follow this tutorial:

* Visual Studio Code
* NVDA Screen Reader or JAWS (Some things will be different)
* A C++ compiler

Here is some technical information about the hardware and software that I am using:

* Screen Reader: NVDA Version 2021.2
* Computer: HP Windows 11 OS
* System: Intel(R) Core(TM) i5-1035G1 CPU @ 1.00GHz 1.20 GHz, 64-bit operating system, x64-based processor
* Visual Studio Code: Version 1.74.3 (user setup)
* gcc.exe (Rev2, Built by MSYS2 project) 12.1.0
* g++.exe (Rev2, Built by MSYS2 project) 12.1.0

## Installing the VS Code Extension

The following extension will need to be install to VS Code:

* C/C++ by Microsoft.

To do this:

1. Open VS Code.
2. Press the short cut key stroke Control+shift+x.
3. In the search field, type C/C++.
4. Press tab until you have reached the results list. It should be the first or second in the list.
5. Once you have the extension in focus, press tab and you should hear “install”.
6. Press enter.

Note: Sometimes you will need to reload VS Code after installing a new extension. This can be done with control+r.

Create Folder and Test C++ Program

1. Create a new colder called “HelloWorld”.
2. Go into the folder and create a file called “hello.cpp”.
3. Open the HelloWorld folder in VS Code.
4. Choose “Yes…” for trusting the new folder just created.
5. Open the “hello.cpp” file and write the following:

#include <iostream>

int main()

{

    std::cout << "Hello, World!\n";

    return 0;

}

Make sure to save the file with control+s.

Now, we need to build the file by doing the following:

1. Press f1 to open the command pallet and type “run build task”, excluding the quotations. (You may also use the short cut key stroke control+shift+B)
2. You should see a list of available compilers on your machine. Select C/C++: Build active file with G++.
3. Once the program is done building, you will here a sound that starts low and gets higher as it plays. This mean success! A sound with a high to low change is a failed build.
4. Pressing F6 till you hear “terminal G++” and getting out of forms mode with NVDA modifier key + spacebar or Jaws key + z and up arrowing will show you the file output build results.

Now, it is time to run our program.

Using a git bash terminal or the integrated terminal in VS Code, do the following:

1. Change directories into the HelloWorld directory you created.
2. Run the following:

Gitbash:

$ ./hello

Windows CMD or PowerShell:

PS C:\users\kblac\Documents\HelloWorld> .\hello

If everything is correct, you should get the following output:

Hello, World!