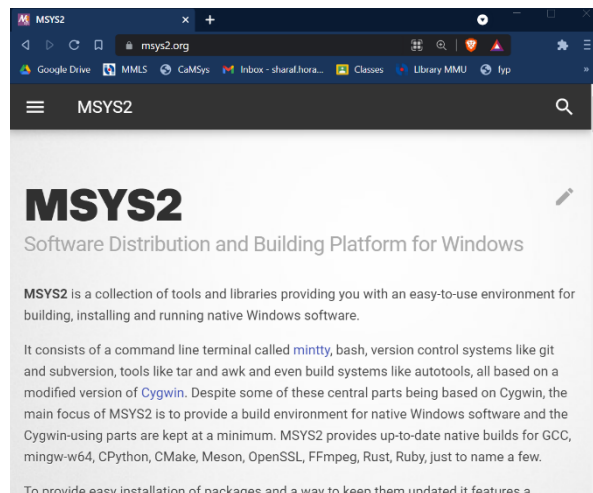


# Basic Guide for Installing Visual Studio Code

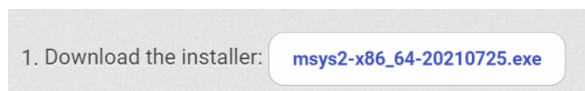
## A. Microsoft Windows

### Installing the MSYS2 system on your computer

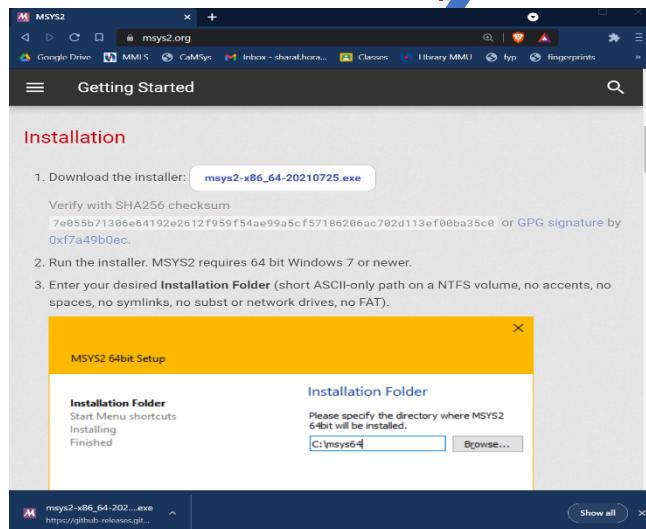
1. Visit the web page (<https://www.msys2.org>)



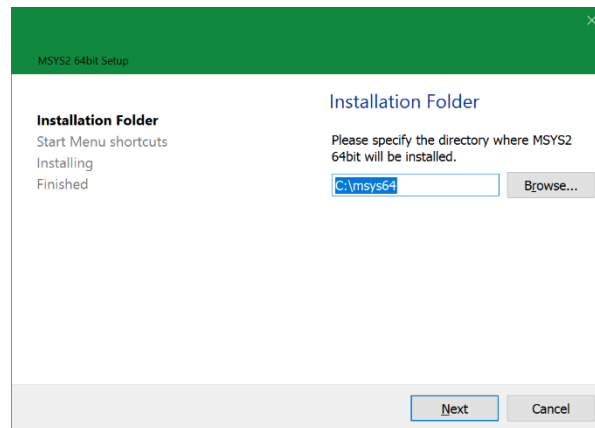
2. Scroll down the page and press on the button to download the file (msys2-x86\_64-20210725.exe). Save the file to your download folder.



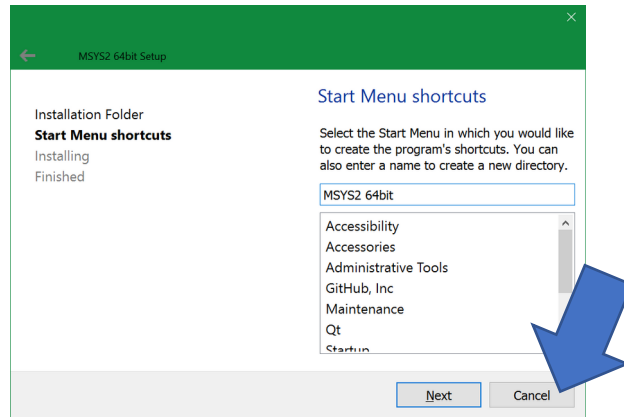
3. When done downloading the file. Click on the filename



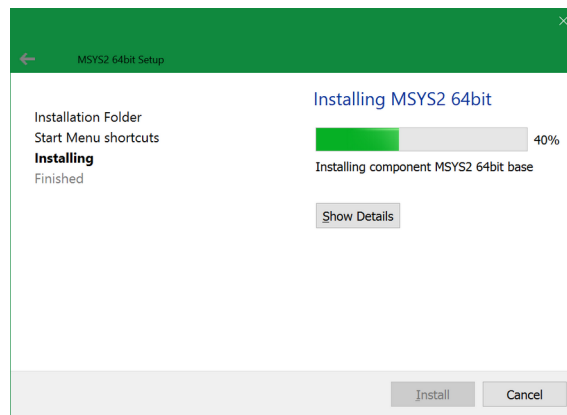
4. The following windows will be displayed. Press the next button.
5. Don't change the name of the folder beside the browser button and press the Next button below.



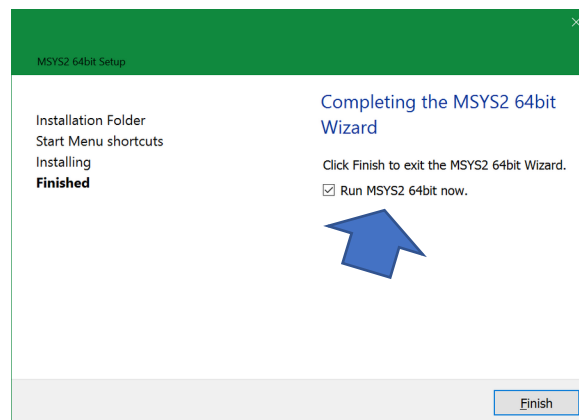
6. Press the Next button. Don't change any of the values displayed in the window.



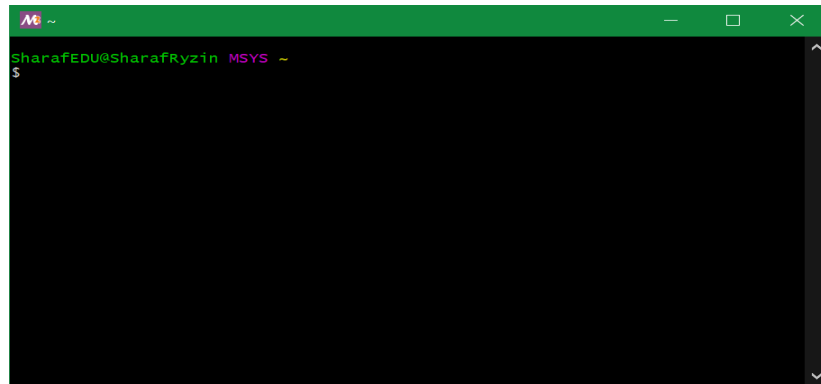
7. Wait until the window finishes the installation of the MSYS2 system for you.



8. When done press the finish button

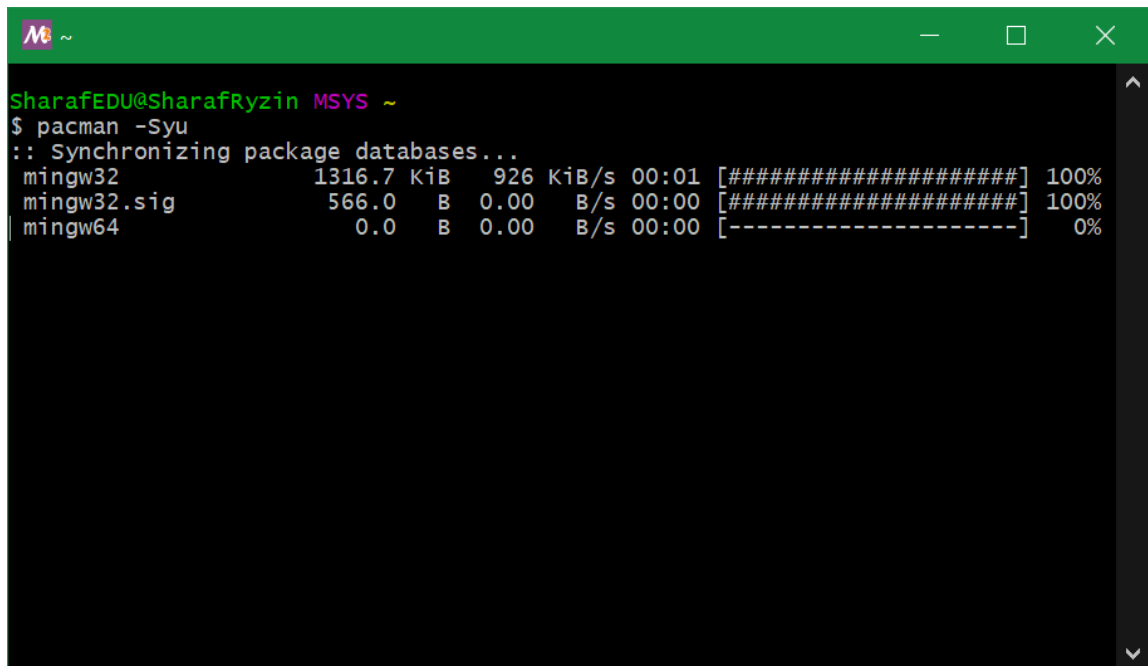


9. When you press the finish button, the following window will appear.

A terminal window with a green title bar. The prompt is 'SharafEDU@SharafRyzin MSYS ~' followed by a '\$' symbol on a new line.

```
SharafEDU@SharafRyzin MSYS ~
$
```

10. Make sure your computer is connected to the internet. Type the following command after the \$ to update the package database and base packages.  
( pacman -Syu )

A terminal window showing the output of the 'pacman -Syu' command. It displays the synchronization of package databases and progress bars for 'mingw32', 'mingw32.sig', and 'mingw64'.

```
SharafEDU@SharafRyzin MSYS ~
$ pacman -Syu
:: Synchronizing package databases...
mingw32           1316.7 KiB   926 KiB/s  00:01 [#####] 100%
mingw32.sig       566.0 B    0.00 B/s  00:00 [#####] 100%
mingw64           0.0 B    0.00 B/s  00:00 [-----] 0%
```

11. If any updates available for your system then you have to respond with "Y" then press the Enter key

```
M6 ~
:: Synchronizing package databases...
mingw32          1316.7 KiB   926 KiB/s 00:01 [#####] 100%
mingw32.sig       566.0 B    0.00 B/s 00:00 [#####] 100%
mingw64          1323.8 KiB   3.07 MiB/s 00:00 [#####] 100%
mingw64.sig       566.0 B    0.00 B/s 00:00 [#####] 100%
ucrt64           1480.0 KiB   3.42 MiB/s 00:00 [#####] 100%
ucrt64.sig        566.0 B    0.00 B/s 00:00 [#####] 100%
clang64          1270.6 KiB   5.67 MiB/s 00:00 [#####] 100%
clang64.sig       566.0 B    0.00 B/s 00:00 [#####] 100%
msys             374.2 KiB    0.00 B/s 00:00 [#####] 100%
msys.sig          566.0 B    0.00 B/s 00:00 [#####] 100%
:: Starting core system upgrade...
warning: terminate other MSYS2 programs before proceeding
resolving dependencies...
looking for conflicting packages...

Packages (4) filesystem-2021.06-1  msys2-runtime-3.2.0-14  pacman-6.0.0-5
                pacman-mirrors-20210706-1

Total Download Size:    8.57 MiB
Total Installed Size:  43.17 MiB
Net Upgrade Size:       0.77 MiB

:: Proceed with installation? [Y/n]
```

12. Press “Y” followed by the “Enter” key to confirm the updates.

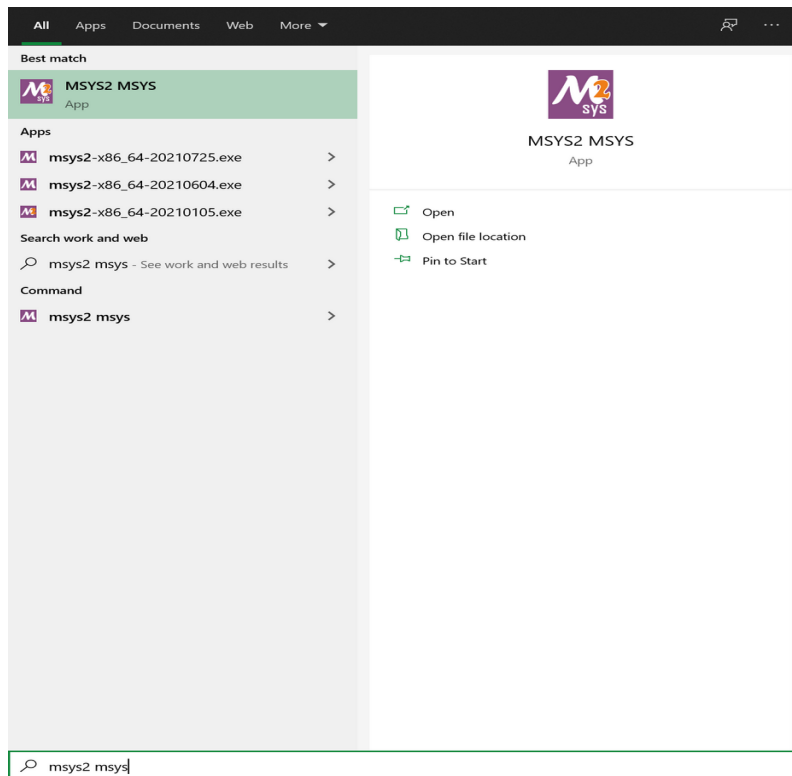
```
M6 ~
pacman-mirrors-20210706-1

Total Download Size:    8.57 MiB
Total Installed Size:  43.17 MiB
Net Upgrade Size:       0.77 MiB

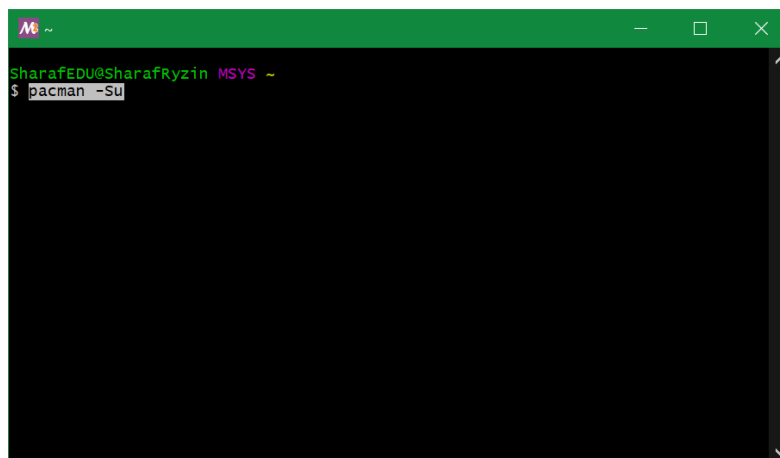
:: Proceed with installation? [Y/n] Y
:: Retrieving packages...
filesystem-2021.... 85.9 KiB   203 KiB/s 00:00 [#####] 100%
msys2-runtime-3.... 3.1 MiB   2.84 MiB/s 00:01 [#####] 100%
pacman-mirrors-2... 3.9 KiB    0.00 B/s 00:00 [#####] 100%
pacman-6.0.0-5-x... 5.4 MiB   1480 KiB/s 00:04 [#####] 100%
(4/4) checking keys in keyring [#####] 100%
(4/4) checking package integrity [#####] 100%
(4/4) loading package files [#####] 100%
(4/4) checking for file conflicts [#####] 100%
(4/4) checking available disk space [#####] 100%
:: Processing package changes...
(1/4) upgrading filesystem [#####] 100%
(2/4) upgrading msys2-runtime [#####] 100%
(3/4) upgrading pacman-mirrors [#####] 100%
(4/4) upgrading pacman [#####] 100%
:: To complete this update all MSYS2 processes including this terminal will be c
losed. Confirm to proceed [Y/n] Y
```

13. You have to wait for the system to finish downloading and installing the packages needed by the MSYS2 to work properly. The window will disappear.

14. Go to the start menu and run the command “MSYS2 MSYS”



15. Type the following command in the terminal after the \$ sign. (pacman -Su)



16. Press "Y" then Enter

```
SharafEDU@SharafRyzin MSYS ~  
$ pacman -Su  
:: Starting core system upgrade...  
there is nothing to do  
:: Starting full system upgrade...  
resolving dependencies...  
looking for conflicting packages...  
  
Packages (26) curl-7.77.0-1 dash-0.5.11.4-1 glib2-2.68.3-1 gnupg-2.2.29-1  
grep-3.6-1 info-6.8-1 less-590-1 libcurl-7.77.0-1  
libedit-20210714_3.1-1 libexpat-2.4.1-1 libgcrypt-1.9.3-1  
libgnutls-3.7.2-1 libgpgme-1.16.0-1 libhogweed-3.7.3-1  
libksba-1.6.0-1 libnettle-3.7.3-1 libp11-kit-0.24.0-1  
libpcre-8.45-1 libpcre2_8-10.37-1 libreadline-8.1.001-1  
libsqlite-3.36.0-1 libxml2-2.9.12-1 nano-5.8-1 nettle-3.7.3-1  
p11-kit-0.24.0-1 pacman-contrib-1.4.0-2  
  
Total Download Size: 15.41 MiB  
Total Installed Size: 48.79 MiB  
Net Upgrade Size: 0.40 MiB  
  
:: Proceed with installation? [Y/n]
```

17. Wait until the system updates the packages

```
SharafEDU@SharafRyzin MSYS ~  
$ pacman -Su  
[14/26] upgrading curl [#####] 100%  
[15/26] upgrading grep [#####] 100%  
[16/26] upgrading dash [#####] 100%  
[17/26] upgrading libhogweed [#####] 100%  
[18/26] upgrading libnettle [#####] 100%  
[19/26] upgrading libgnutls [#####] 100%  
[20/26] upgrading libksba [#####] 100%  
[21/26] upgrading nettle [#####] 100%  
[22/26] upgrading gnupg [#####] 100%  
==> Appending keys from msys2.gpg...  
==> Locally signing trusted keys in keyring...  
==> Importing owner trust values...  
==> Disabling revoked keys in keyring...  
==> Updating trust database...  
gpg: next trustdb check due at 2022-01-23  
[23/26] upgrading libgpgme [#####] 100%  
[24/26] upgrading libpcre2_8 [#####] 100%  
[25/26] upgrading nano [#####] 100%  
[26/26] upgrading pacman-contrib [#####] 100%  
:: Running post-transaction hooks...  
(1/1) Updating the info directory file...  
  
SharafEDU@SharafRyzin MSYS ~  
$
```

18. To install the MinGW C++ compiler and tools you must type the command below:

**pacman -S --needed base-devel mingw-w64-x86\_64-toolchain**

```
SharafEDU@SharafRyzin MSYS ~  
$ pacman -S --needed base-devel mingw-w64-x86_64-toolchain
```

19. Press the **Enter** key twice

```
SharafEDU@SharafRyzin MSYS ~
$ pacman -S --needed base-devel mingw-w64-x86_64-toolchain
warning: file-5.40-2 is up to date -- skipping
warning: gawk-5.1.0-1 is up to date -- skipping
warning: gettext-0.19.8.1-1 is up to date -- skipping
warning: grep-3.6-1 is up to date -- skipping
warning: pacman-6.0.0-5 is up to date -- skipping
warning: perl-5.32.1-1 is up to date -- skipping
warning: sed-4.8-1 is up to date -- skipping
warning: wget-1.21.1-2 is up to date -- skipping
:: There are 47 members in group base-devel:
:: Repository msys
   1) asciidoc   2) autoconf   3) autoconf2.13  4) autogen
   5) automake-wrapper  6) automake1.10  7) automake1.11
   8) automake1.12  9) automake1.13 10) automake1.14
  11) automake1.15 12) automake1.16 13) automake1.6
  14) automake1.7  15) automake1.8 16) automake1.9 17) bison
  18) btyacc       19) diffstat  20) diffutils  21) dos2unix
  22) flex         23) gdb       24) gettext-devel 25) gperf  26) groff
  27) help2man    28) intltool  29) libtool    30) libunrar
  31) libunrar-devel 32) m4       33) make      34) man-db  35) patch
  36) patchutils  37) pkgconf  38) pkgfile   39) quilt
  40) reflex      41) scons    42) swig      43) texinfo  44) texinfo-tex
  45) ttyrec      46) unrar    47) xmlto
```

20. The press “Y” followed by the Enter key

```
mingw-w64-x86_64-gcc-fortran-10.3.0-5
mingw-w64-x86_64-gcc-libgfortran-10.3.0-5
mingw-w64-x86_64-gcc-libs-10.3.0-5
mingw-w64-x86_64-gcc-objc-10.3.0-5
mingw-w64-x86_64-gdb-10.2-2
mingw-w64-x86_64-gdb-multiarch-10.2-2
mingw-w64-x86_64-headers-git-9.0.0.6270.0e0eed623-1
mingw-w64-x86_64-libgccjit-10.3.0-5
mingw-w64-x86_64-libmangle-git-9.0.0.6270.0e0eed623
-1
mingw-w64-x86_64-libwinpthread-git-9.0.0.6270.0e0eed623-1
mingw-w64-x86_64-make-4.3-1
mingw-w64-x86_64-pkgconf-1.7.4-2
mingw-w64-x86_64-tools-git-9.0.0.6270.0e0eed623-1
mingw-w64-x86_64-winpthreads-git-9.0.0.6270.0e0eed623-1
mingw-w64-x86_64-winstorecompat-git-9.0.0.6270.0e0eed623-1
patch-2.7.6-1 patchutils-0.4.2-2 pkgconf-1.7.4-1
pkgfile-21-1 quilt-0.66-2 reflex-20210510-1
scons-3.1.2-6 swig-4.0.2-1 texinfo-6.8-1
texinfo-tex-6.8-1 ttyrec-1.0.8-2 unrar-6.0.7-1
xmlto-0.0.28-2
Total Download Size: 207.90 MiB
Total Installed Size: 1353.29 MiB
:: Proceed with installation? [Y/n]
```

21. The process will take few minutes to complete.

```
(113/124) installing mingw-w64-x86_64-tk [#####] 100%
(114/124) installing mingw-w64-x86_64-xz [#####] 100%
(115/124) installing mingw-w64-x86_64-python [#####] 100%
(116/124) installing mingw-w64-x86_64-xxhash [#####] 100%
(117/124) installing mingw-w64-x86_64-gdb [#####] 100%
(118/124) installing mingw-w64-x86_64-gdb-mult... [#####] 100%
(119/124) installing mingw-w64-x86_64-libgccjit [#####] 100%
(120/124) installing mingw-w64-x86_64-libmangl... [#####] 100%
(121/124) installing mingw-w64-x86_64-make [#####] 100%
(122/124) installing mingw-w64-x86_64-pkgconf [#####] 100%
(123/124) installing mingw-w64-x86_64-tools-git [#####] 100%
(124/124) installing mingw-w64-x86_64-winstore... [#####] 100%
:: Running post-transaction hooks...
(1/1) Updating the info directory file...
SharafEDU@SharafRyzin MSYS ~
$
```



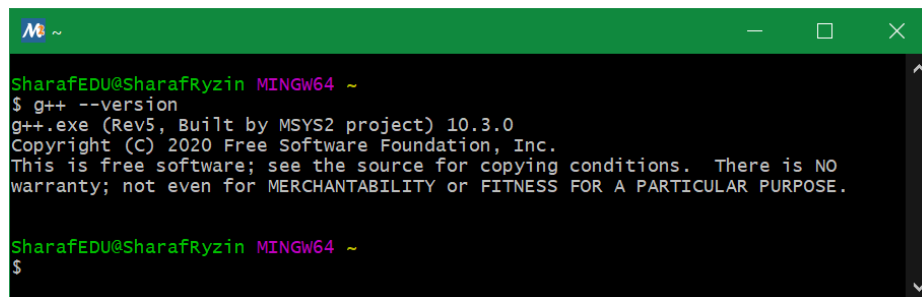
22. Close the window (MSYS2)

23. The system is ready now to use the C++ compiler tools.

24. From the ( start menu → MSYS2 64bit → MSYS2 MinGW 64bit )

25. Type the following command in the terminal after the \$

`g++ --version`

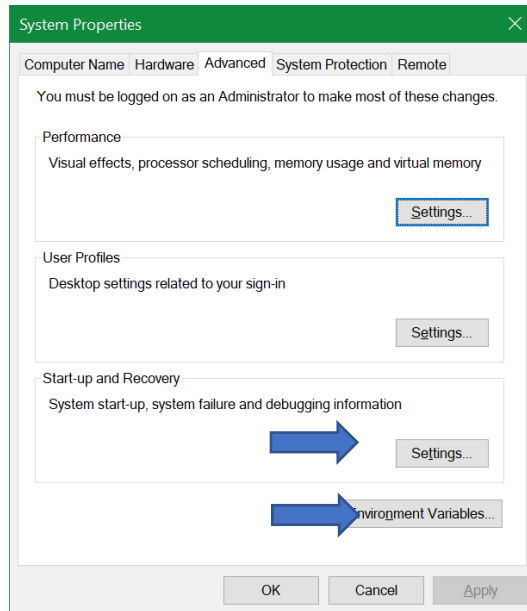
A screenshot of a terminal window with a green title bar. The prompt is 'SharafEDU@SharafRyzin MINGW64 ~'. The command '\$ g++ --version' has been entered, and the output is displayed: 'g++.exe (Rev5, Built by MSYS2 project) 10.3.0', 'Copyright (C) 2020 Free Software Foundation, Inc.', and 'This is free software; see the source for copying conditions. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.'. The prompt '\$' is shown again at the bottom.

```
SharafEDU@SharafRyzin MINGW64 ~  
$ g++ --version  
g++.exe (Rev5, Built by MSYS2 project) 10.3.0  
Copyright (C) 2020 Free Software Foundation, Inc.  
This is free software; see the source for copying conditions. There is NO  
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  
  
SharafEDU@SharafRyzin MINGW64 ~  
$
```

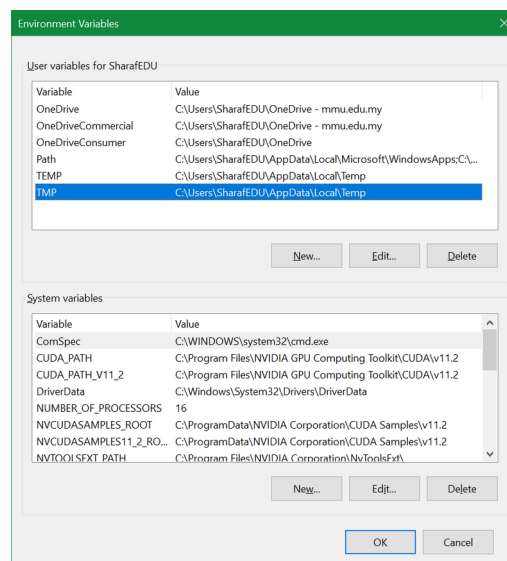
26. If you got the same output as shown in the window, your MSYS2 system and MinGW C++ compiler are installed properly.

## Setting the environment variables

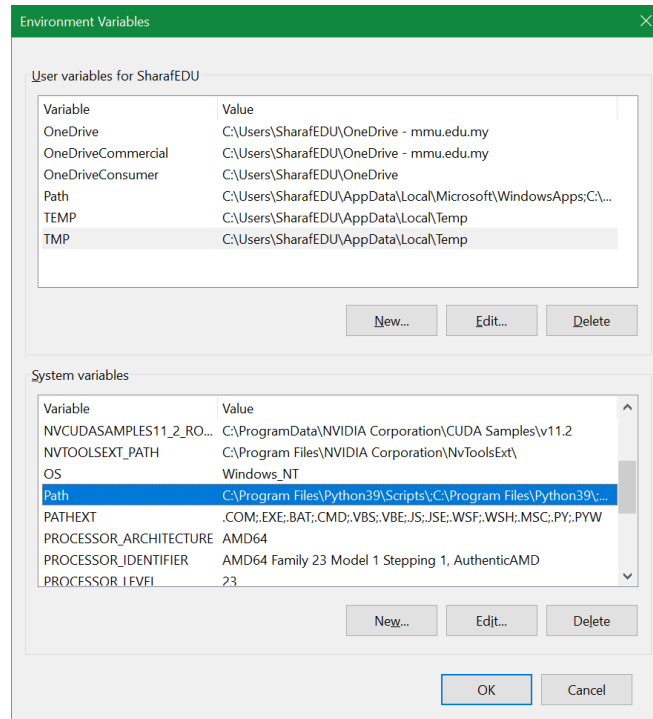
1. Click on the windows key in your keyboard or move the mouse to the start menu and press the left mouse button.
2. Type the word “env” → the system will display a file named “Edit the system environment variables” → click on this file name.
3. The following window will be displayed on the screen



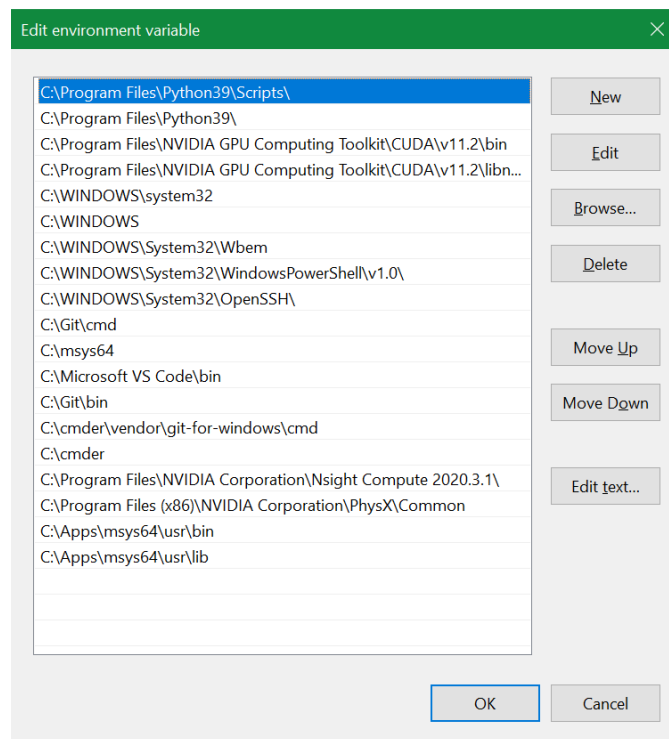
4. Click on the Environment Variables button



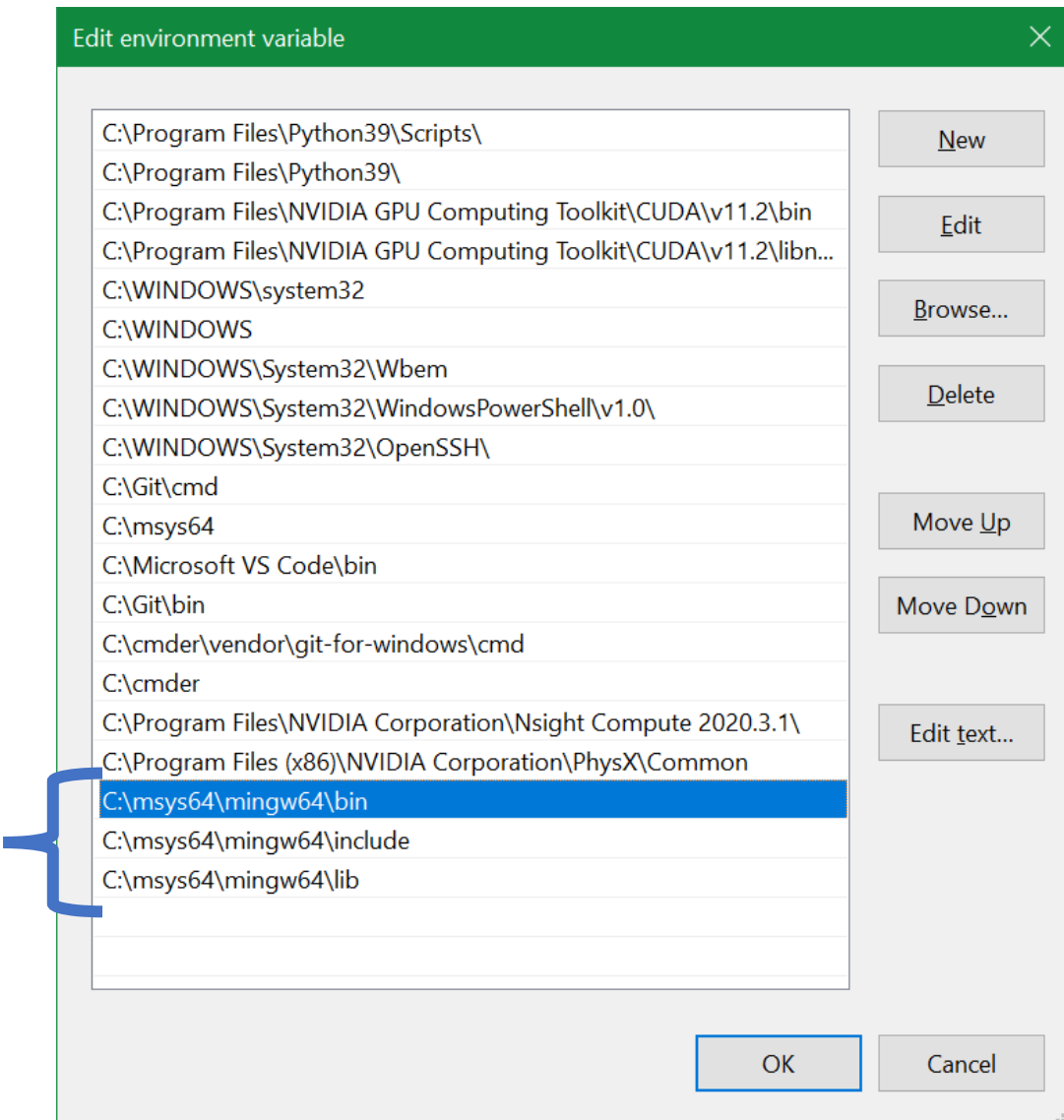
5. In the “System variables” section in the low part of the window, search for the line that has the word “Path”, then select it by the mouse, then press the “EDIT” button.



6. The following window will be displayed

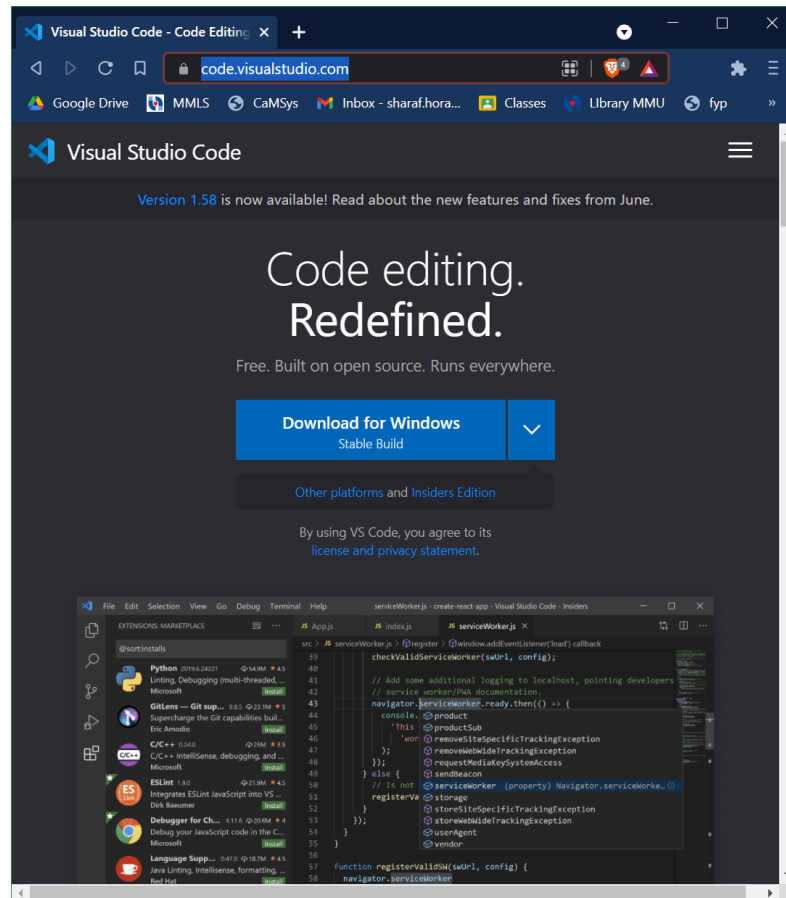


7. Click the “New” button, when a new line is displayed, press on the “Browse” button and navigate to the following directory ( C:\msys64\mingw64\bin).
8. Do the same thing for the following two directories (C:\msys64\mingw64\include) and (C:\msys64\mingw64\lib)
9. When done, press the “OK” key for all the three windows opened so far.

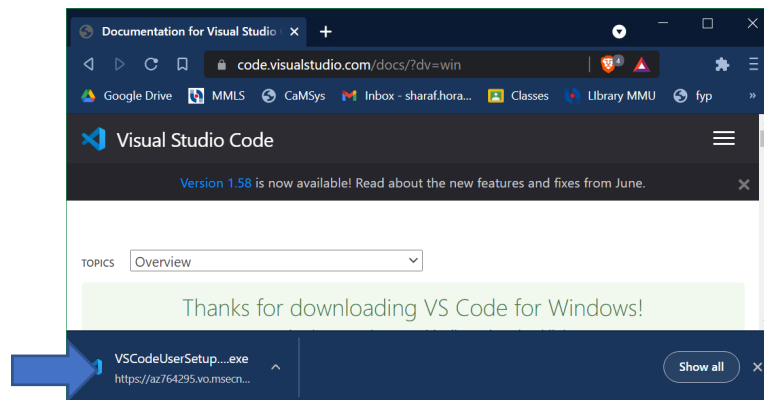


# Installing Visual Studio Code

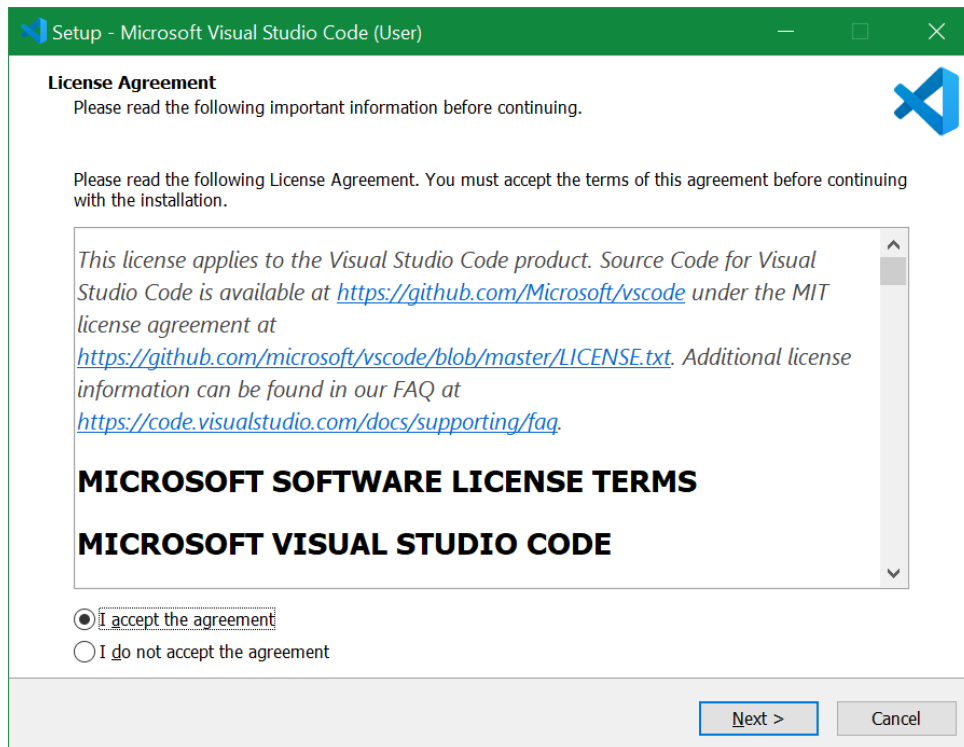
1. Visit the web page <https://code.visualstudio.com/>
2. Download the latest stable version for Windows into your download folder



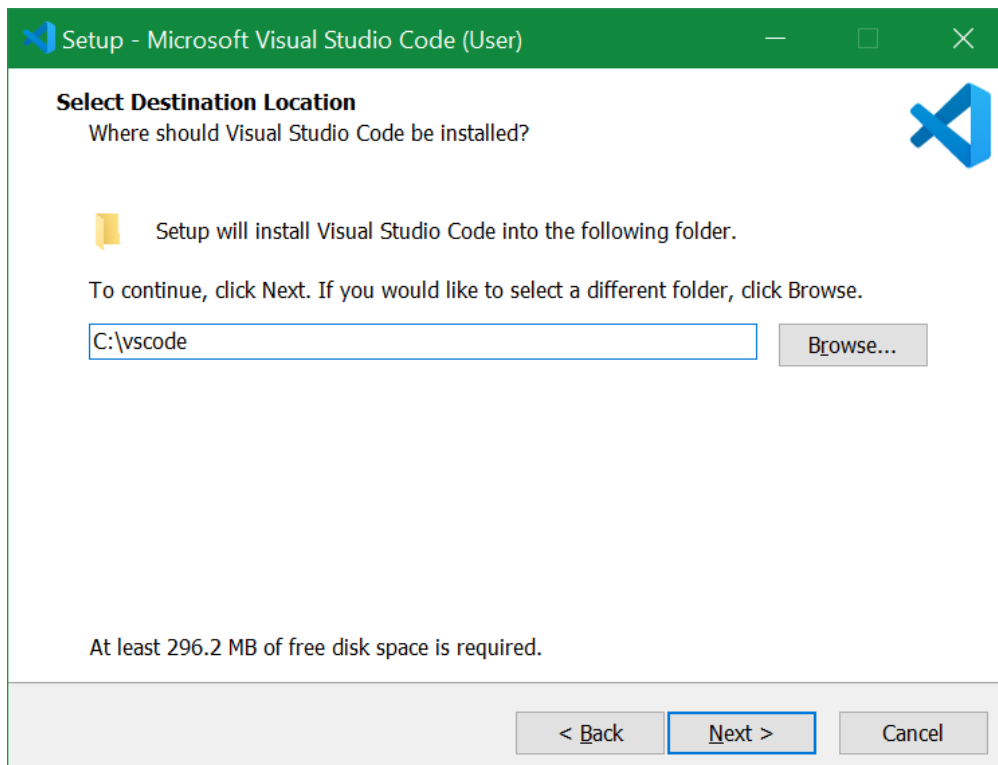
3. When done click the mouse on the name of the file downloaded



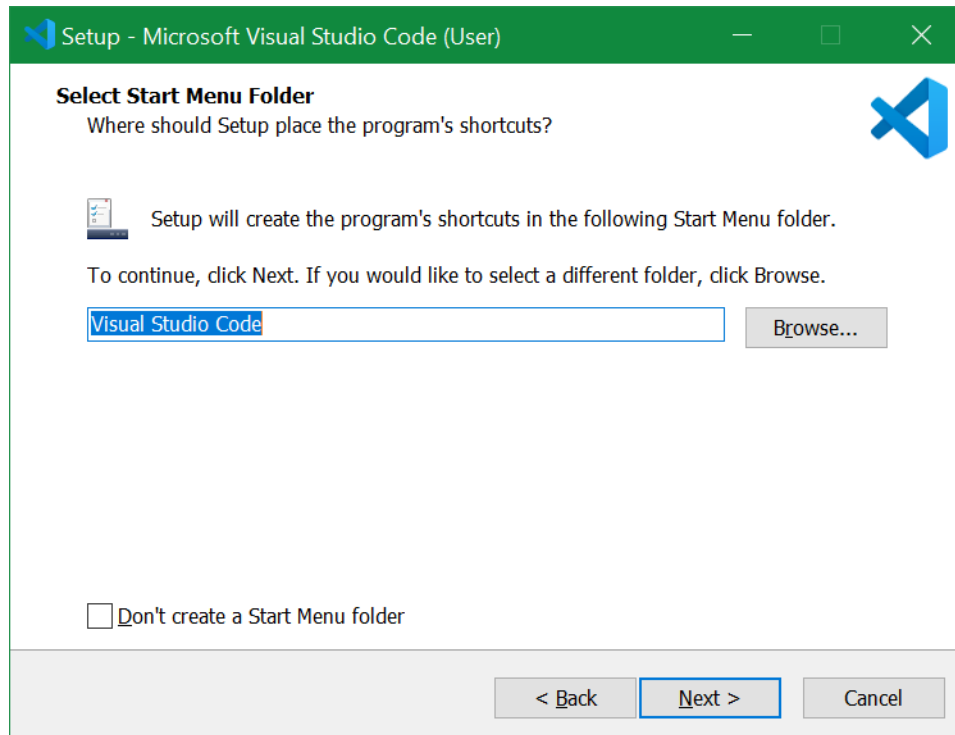
4. Accept the agreement and press “Next”



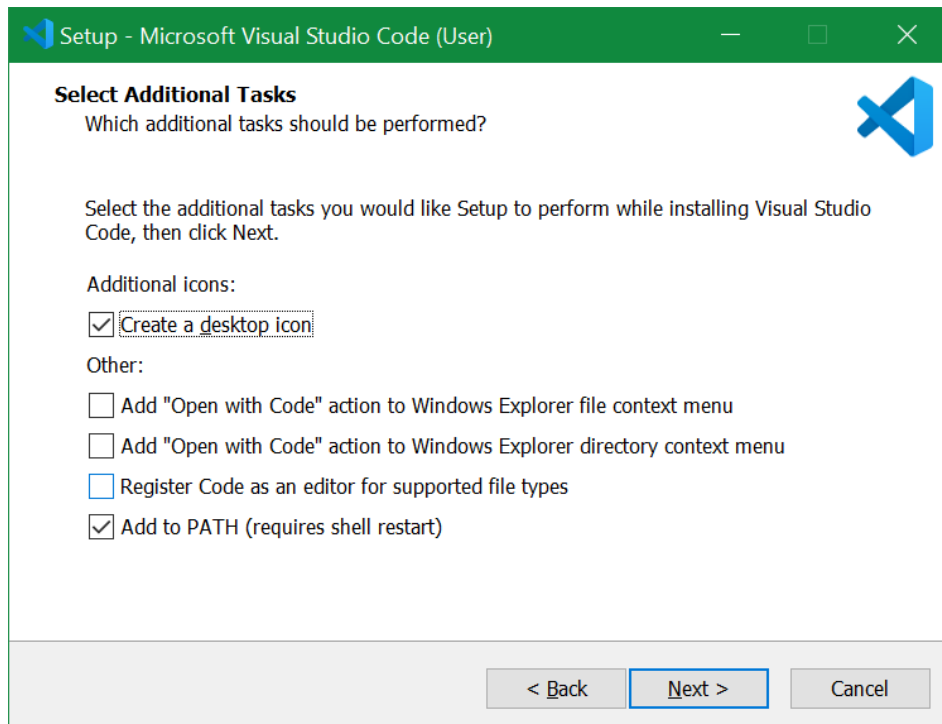
5. Change the name of the file and path to (c:\vscode) then press “Next”



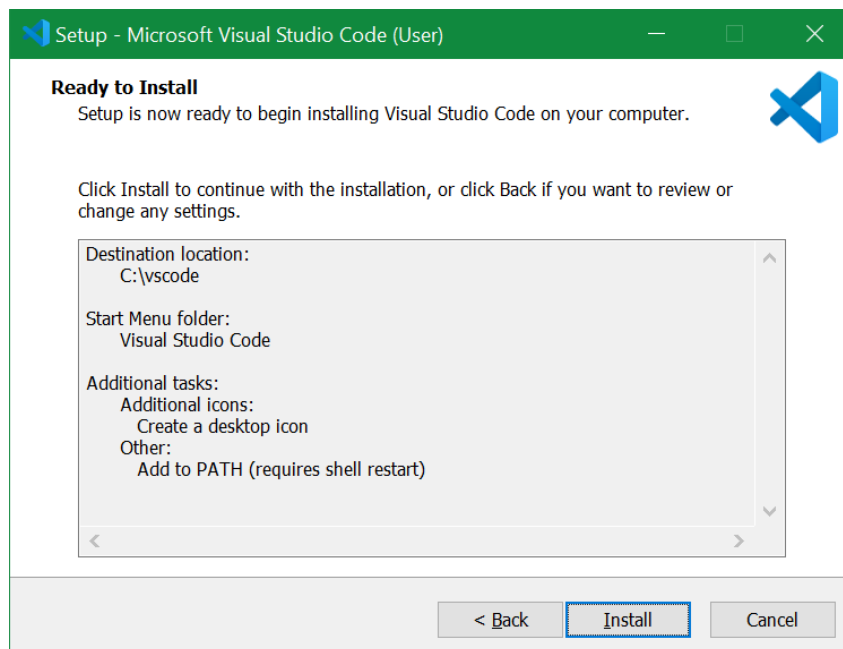
6. Press the “Next” button



7. Check the “Create a desktop icon” check box and press the “Next” button.



8. Press the Install button



9. Wait until the installation is complete

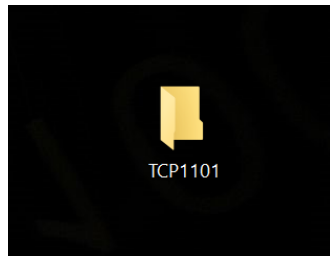
10. The vscode is installed and ready to be used.



11. To test the program, double click the Visual Studio Code button on the Desktop of your PC.

# Running your first C++ program

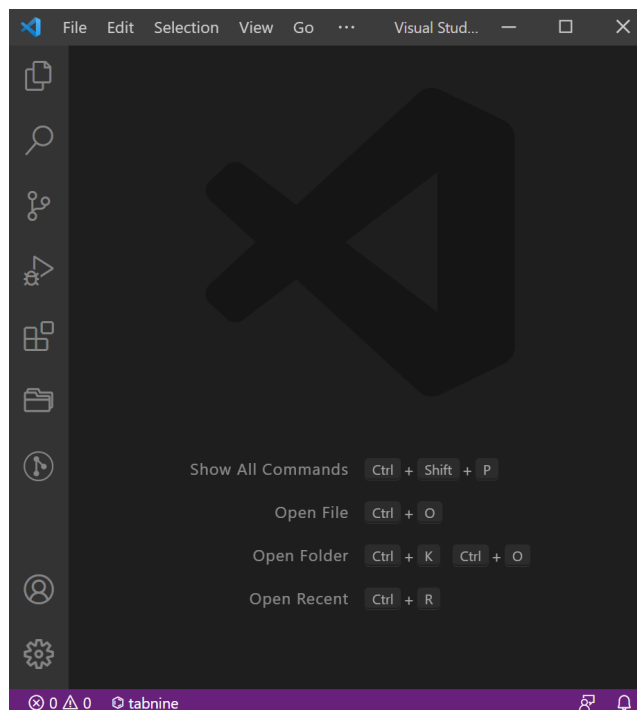
1. Create a new folder on your Desktop and name it TCP1101



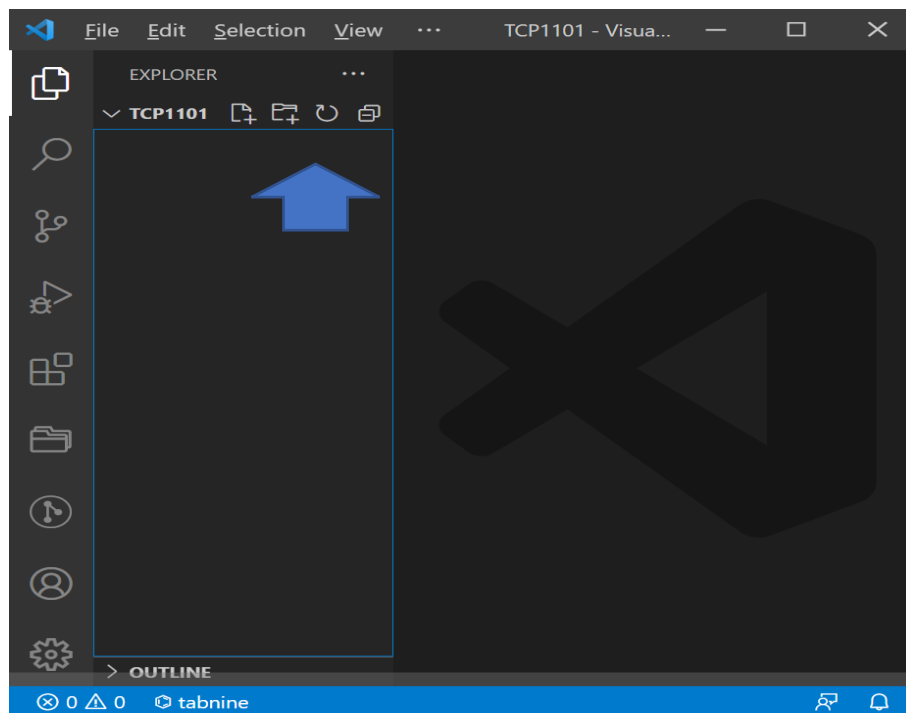
2. Run the Visual Studio Code program by clicking the icon on the Desktop



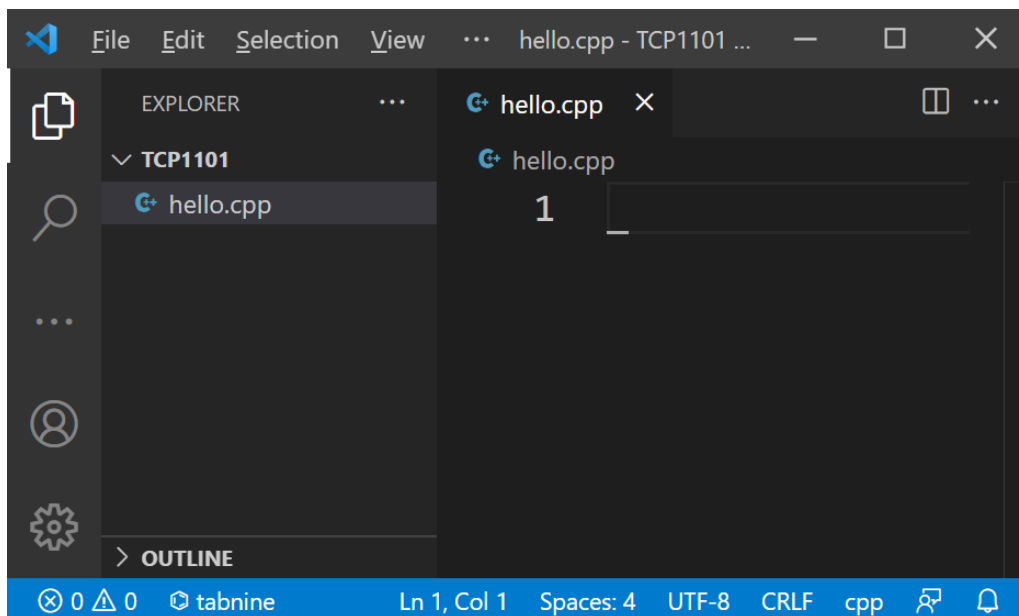
3. When the program opens, select from the menu the item "File" and choose "Open Folder".



4. Select the folder "TCP1101" form the Desktop then press the button "Select Folder"



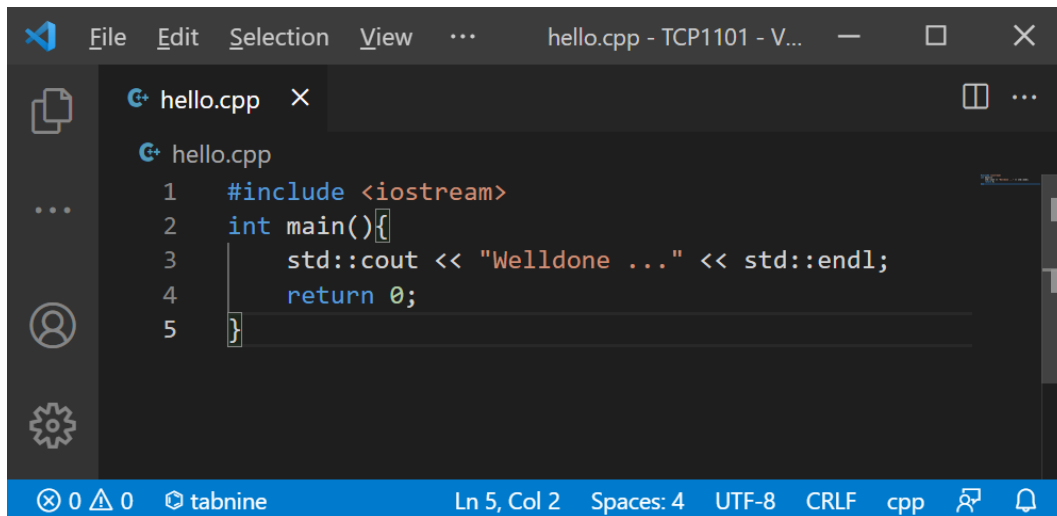
5. Press on the icon pointed at by the blue arrow in the figure above
6. Name the file as "hello.cpp" then press "Enter" key



7. Type the following C++ program

```
#include <iostream>
int main(){
    std::cout << "Welldone ..." << std::endl;
    return 0;
}
```

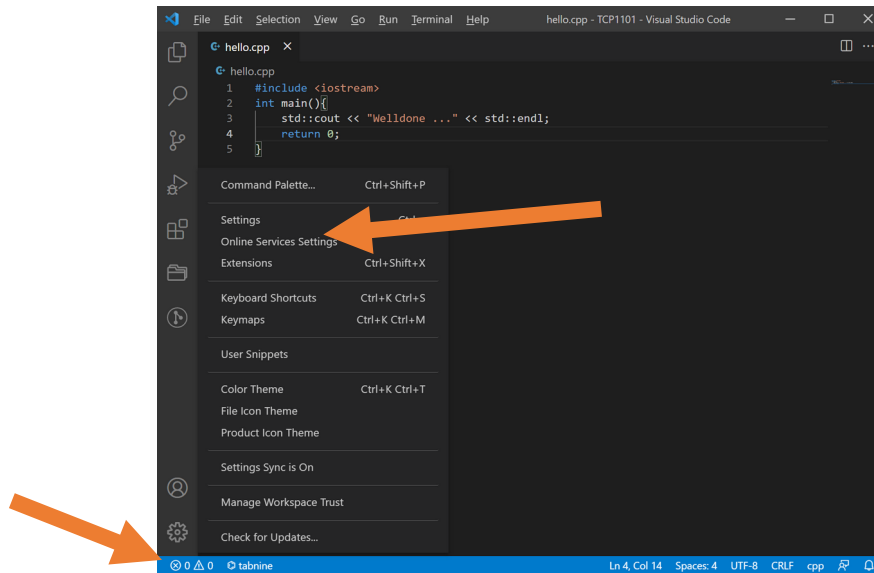
8. Then press CTRL+S to save the file



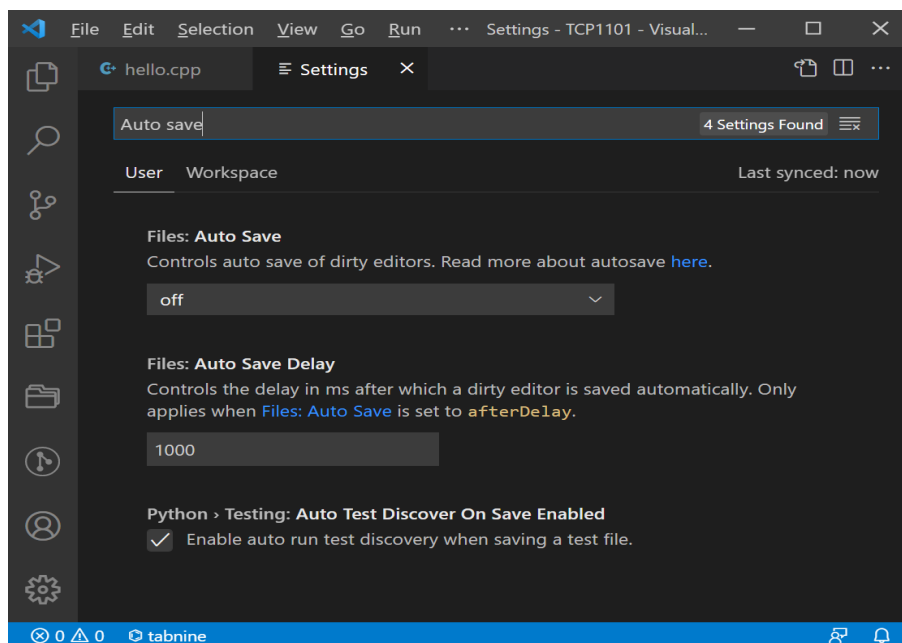
9. To compile the program:
- Select from the menu the option Terminal → New Terminal
  - Type in the terminal the following command → `g++ hello.cpp`
  - To run the program, type the following command → `a`  
Then press Enter.

# Configure the VSCode

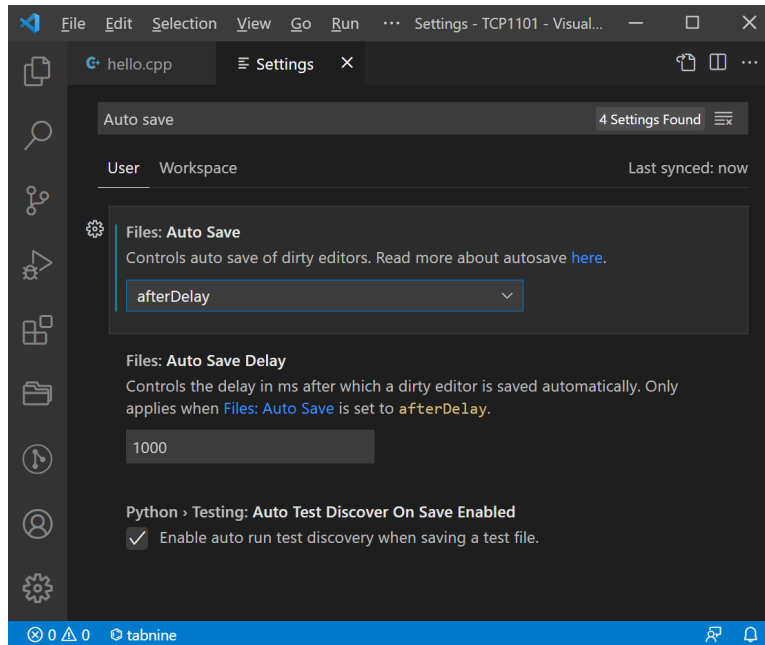
1. Select the flower-shaped icon at the bottom of the left side of the VSCode then choose "settings"



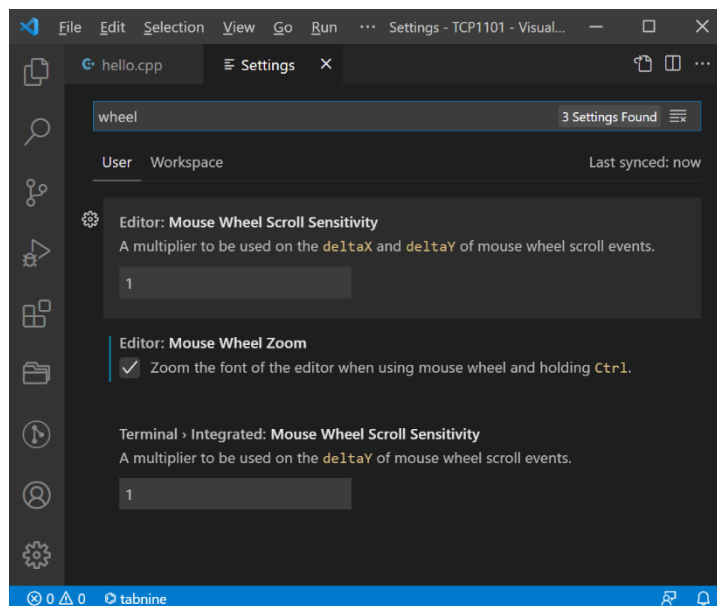
2. In The search settings field type the word "autosave"



3. Change the "Auto save" from "Off" into "AfterDelay"



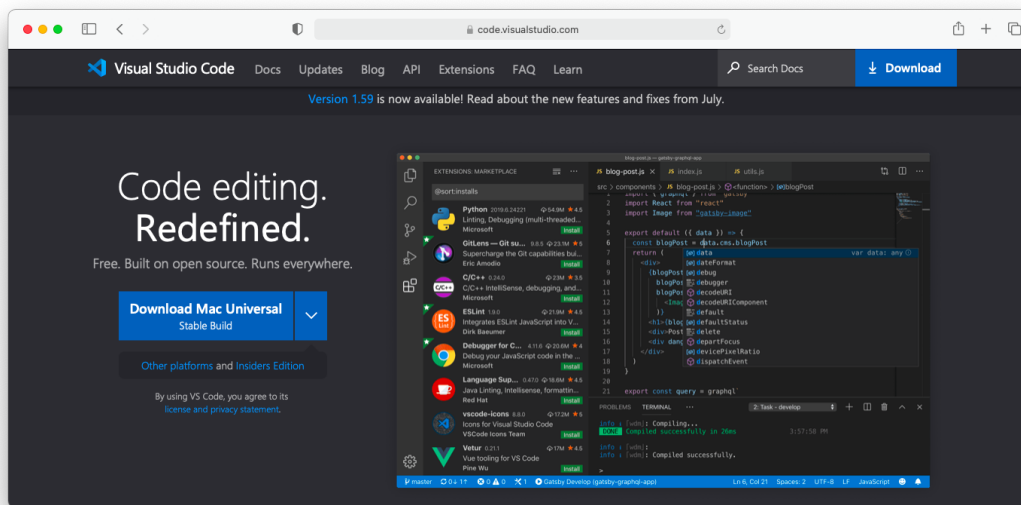
4. Then type the word “Wheel”, and check the box “Mouse Wheel Zoom”



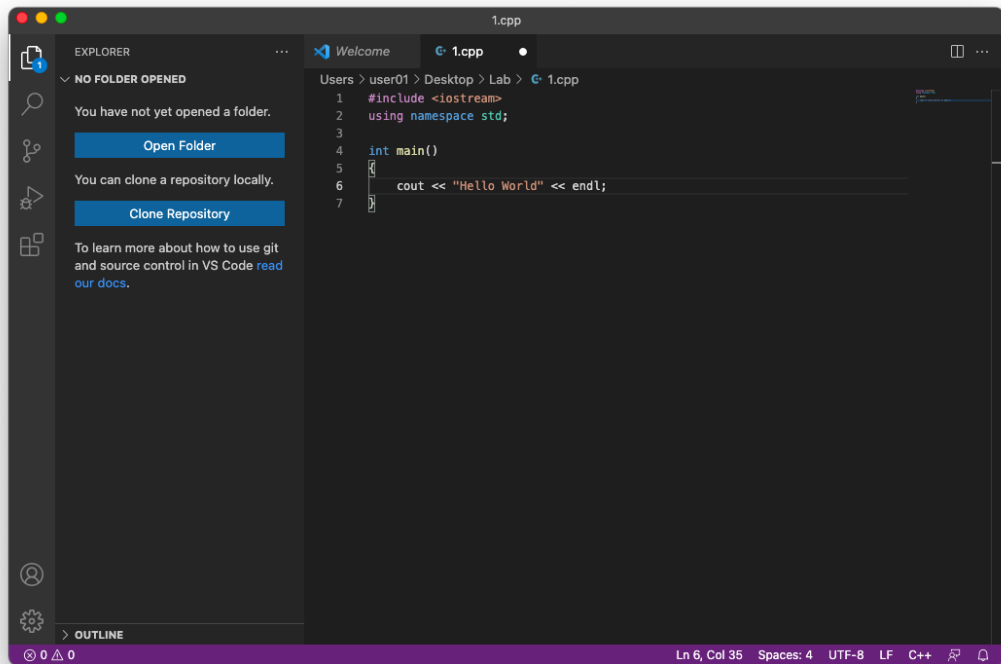
5. Then type the word Terminal and check the box “Debug> Terminal: Clear Before Reusing”

## B. MacOS

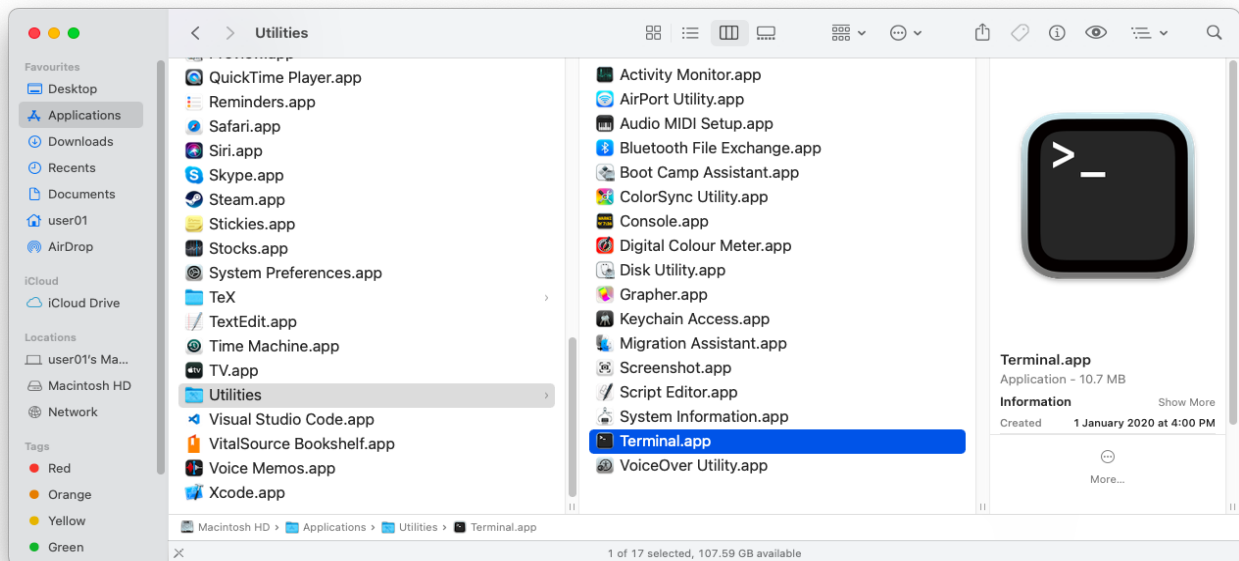
1. Install XCode from the Mac App Store
2. Download Visual Studio Code from <https://code.visualstudio.com>



3. Drag Visual Studio Code.app from the Download folder to the Applications folder.
4. Double click to launch Visual Studio Code.app
5. Opening a new file and saving your C++ source code with .cpp is the same as the instructions for Microsoft Windows



6. To compile and run the codes, double click to launch Terminal.app





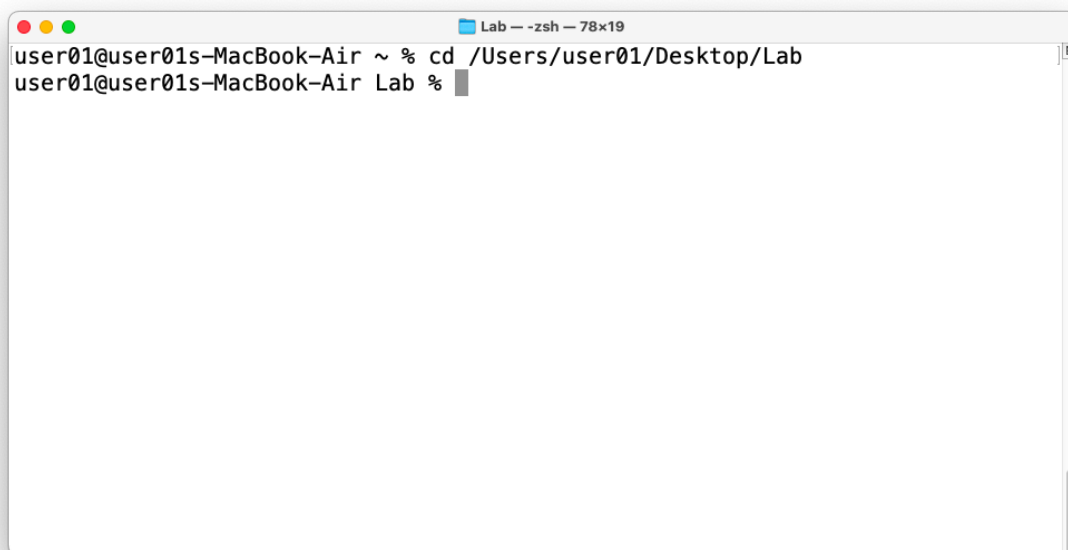
7. Type `cd` and drag the folder where your `.cpp` code is stored to the terminal.



```
user01@user01s-MacBook-Air ~ % cd
```

A terminal window titled "user01 - zsh - 83x16" on a MacBook-Air. The prompt is "user01@user01s-MacBook-Air ~ %". The command "cd" has been typed, and the cursor is at the end of the line.

For example, if `1.cpp` is stored in a folder names `Lab` on the Desktop, the terminal becomes as shown below after dragging to folder to the terminal and typing the enter key.

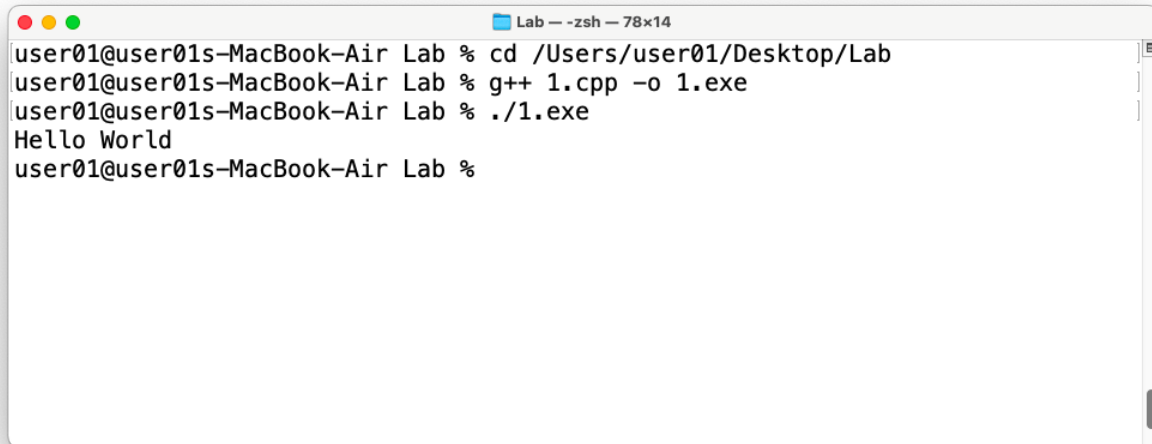


```
user01@user01s-MacBook-Air ~ % cd /Users/user01/Desktop/Lab
user01@user01s-MacBook-Air Lab %
```

A terminal window titled "Lab - zsh - 78x19" on a MacBook-Air. The prompt is "user01@user01s-MacBook-Air ~ %". The command "cd /Users/user01/Desktop/Lab" has been typed. The prompt has changed to "user01@user01s-MacBook-Air Lab %", indicating the directory change was successful.

8. Type `g++ 1.cpp -o 1.exe` to compile 1.cpp

9. Type `./1.exe` to run 1.exe



```
Lab — -zsh — 78x14
user01@user01s-MacBook-Air Lab % cd /Users/user01/Desktop/Lab
user01@user01s-MacBook-Air Lab % g++ 1.cpp -o 1.exe
user01@user01s-MacBook-Air Lab % ./1.exe
Hello World
user01@user01s-MacBook-Air Lab %
```

A terminal window titled "Lab — -zsh — 78x14" showing a series of commands and their outputs. The user navigates to the directory /Users/user01/Desktop/Lab, compiles the file 1.cpp into 1.exe using g++, and then runs 1.exe, which outputs "Hello World".