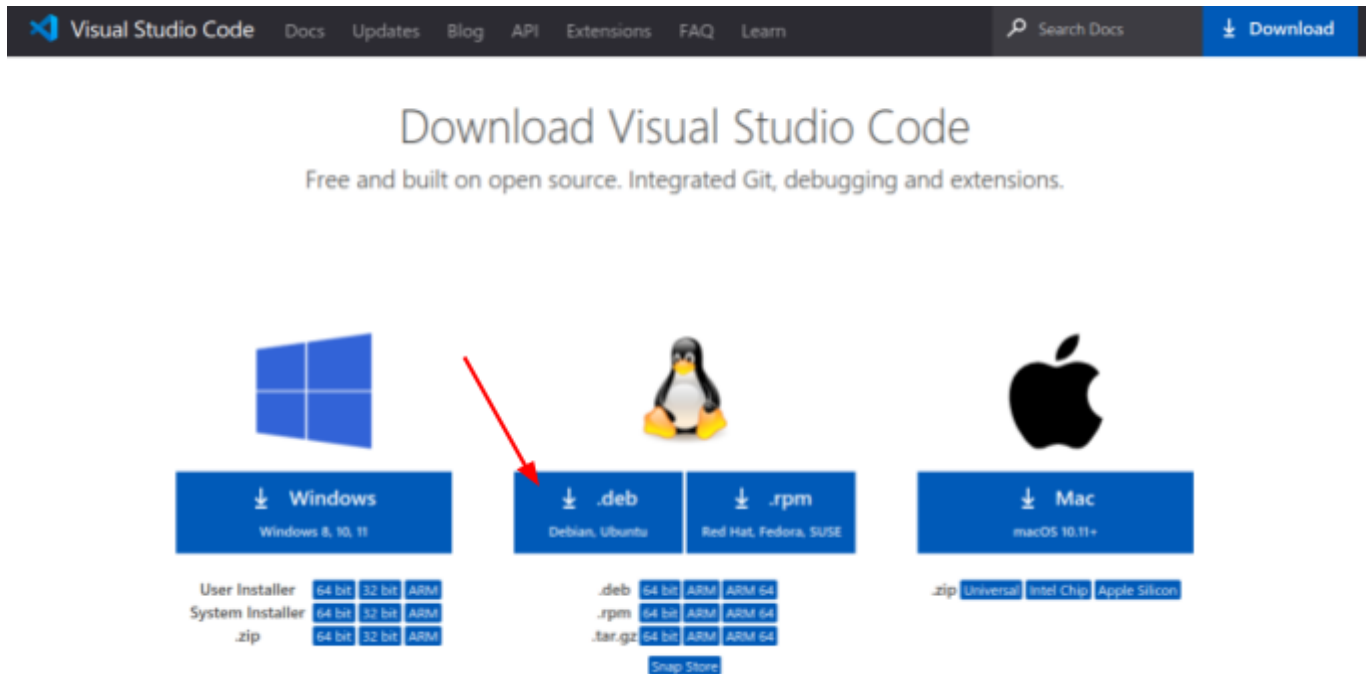


# Configuration of Visual Studio Code for C programming on Ubuntu.

## Step 1:

Download Visual Studio Code Editor from the following link.

<https://code.visualstudio.com/download#>



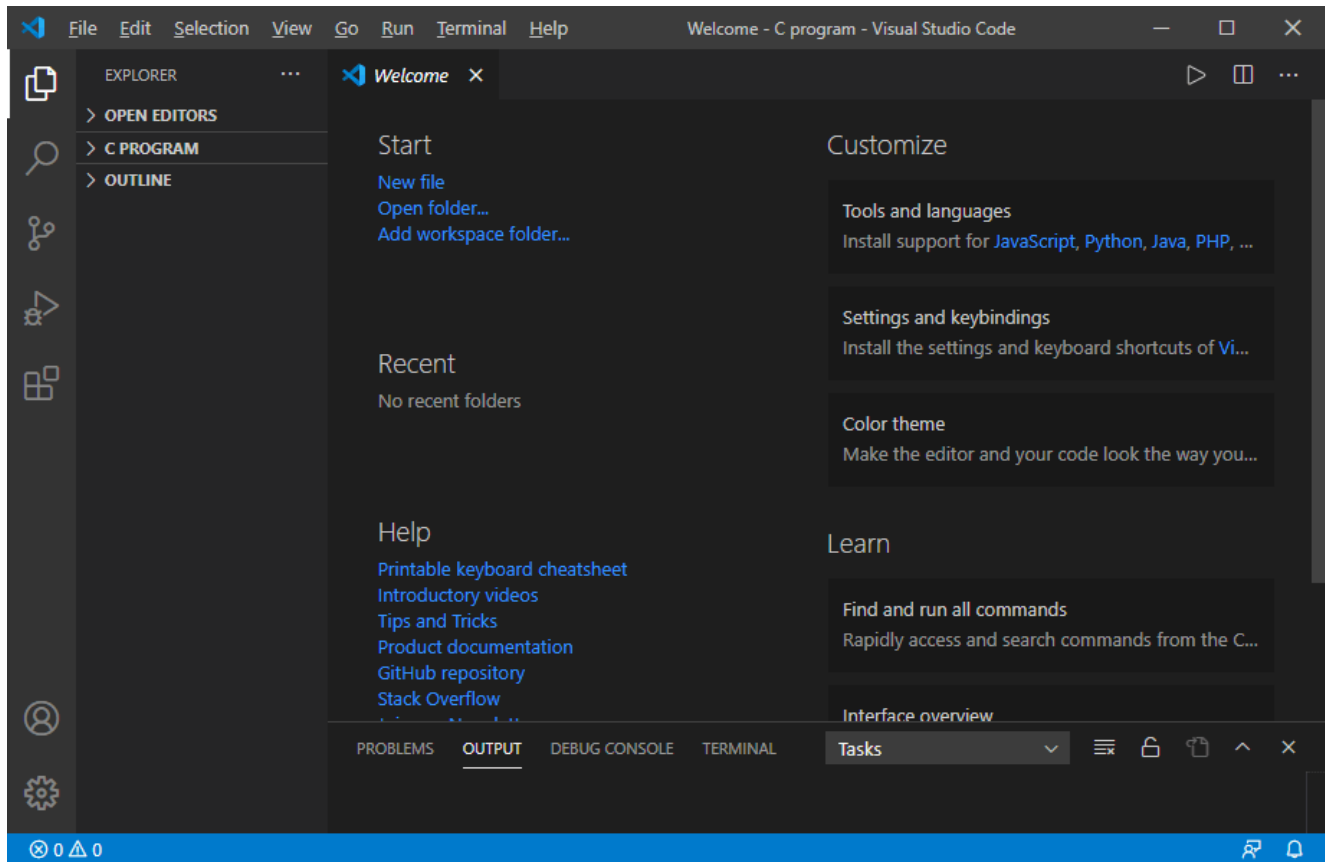
Then run the following command to install vs code.

```
sudo apt install /filepath/filename.deb
```

As shown below.

```
usman@usman-7G-Series: ~  
usman~$ sudo apt install /home/usman/Downloads/code_1.70.2-1660629410_amd64.deb  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
Note, selecting 'code' instead of '/home/usman/Downloads/code_1.70.2-1660629410_amd64.deb'  
code is already the newest version (1.70.2-1660629410).  
The following packages were automatically installed and are no longer required:  
  hplip-data libfprint-2-tod1 libfwupdplugin1 libimagequant0 libllvm10  
  python3-olefile python3-pil python3-renderpm python3-reportlab  
  python3-reportlab-accel shim  
Use 'sudo apt autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 20 not upgraded.  
usman~$
```

We have already installed the Visual Studio Code in our system. The user interface of VS code look like the following:

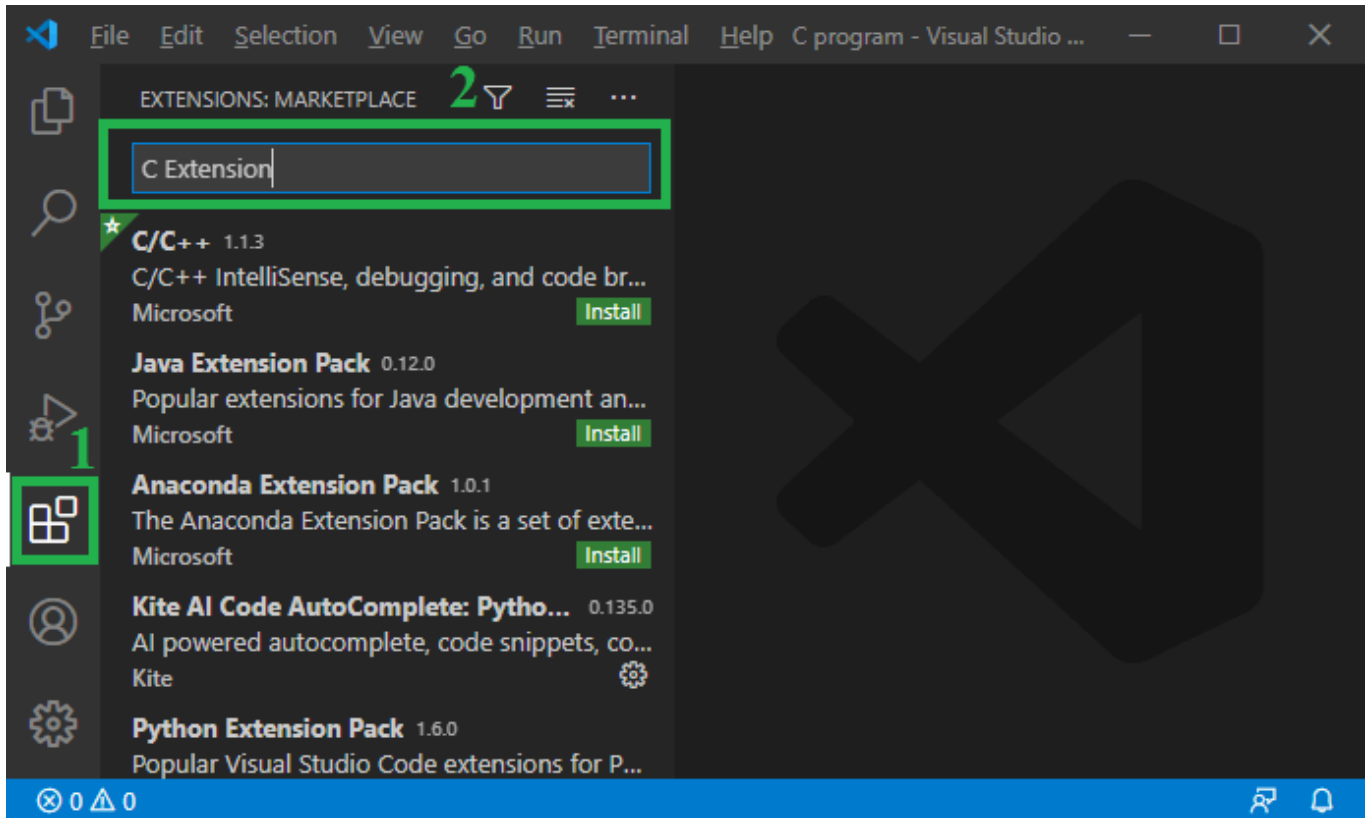


## Step 2:

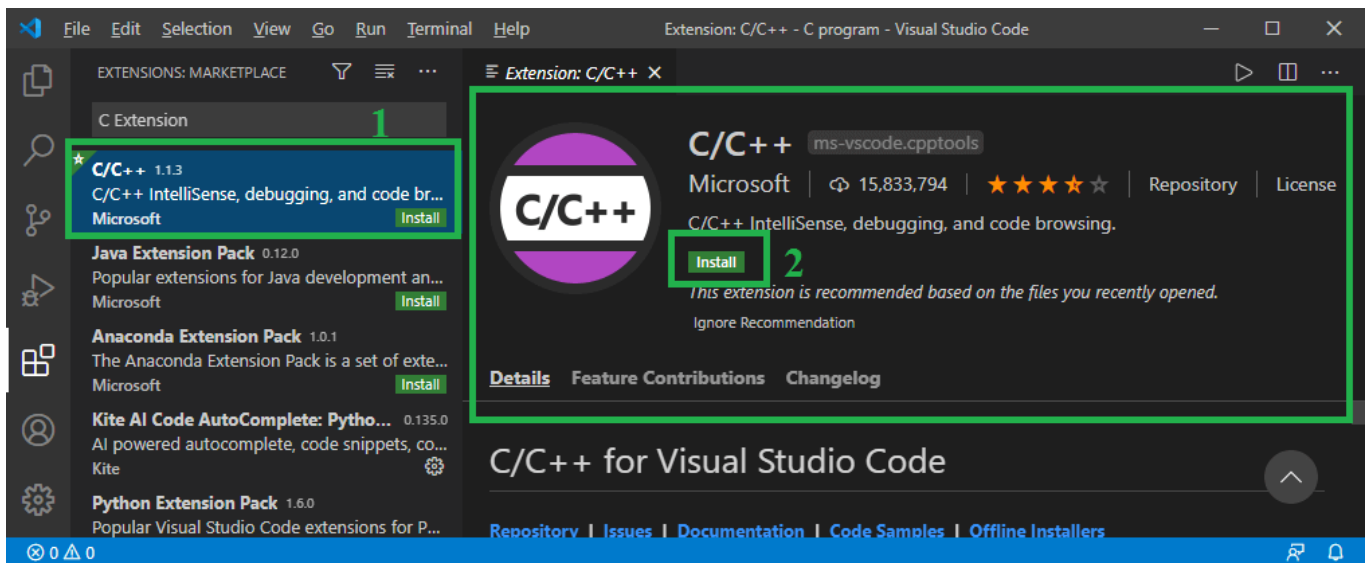
### Download & Install the C/C++ Extension

1. Download the C/C++ Extension. It is an extension provided by Microsoft that support visual studio code. It helps in IntelliSense, debugging and code browsing of the programming code in the visual studio.

We need to click on the extension button that displays a sidebar for downloading and installing the C/C++ extension in the visual studio code. In the sidebar, type C Extension.

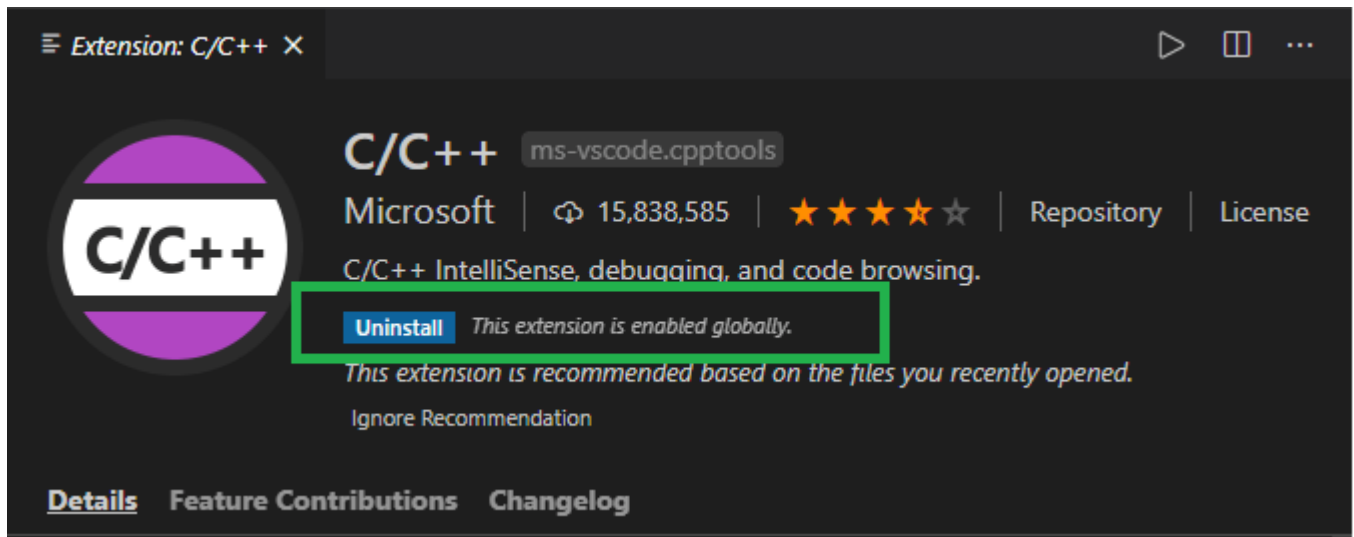


2. After that, click on the C/C++



In this image, click on the Install button to install the C/C++ extension.

3. After clicking the Install button, it shows the below image.



In this image, we can see it shows the Uninstall button that means the C/C++ extension has been successfully downloaded in the visual studio code.

In this image, we can see it shows the Uninstall button that means the C/C++ extension has been successfully downloaded in the visual studio code.

### Step 3:

#### Download and Install Compiler Extension

In order to run a C program in Linux, you need to have a C compiler present on your systems. The most popular compiler is gcc ([GNU Compiler Collection](#)).

You can install gcc using your distribution's package manager. In Debian and Ubuntu-based Linux distributions, use the apt command:

```
sudo apt update
```

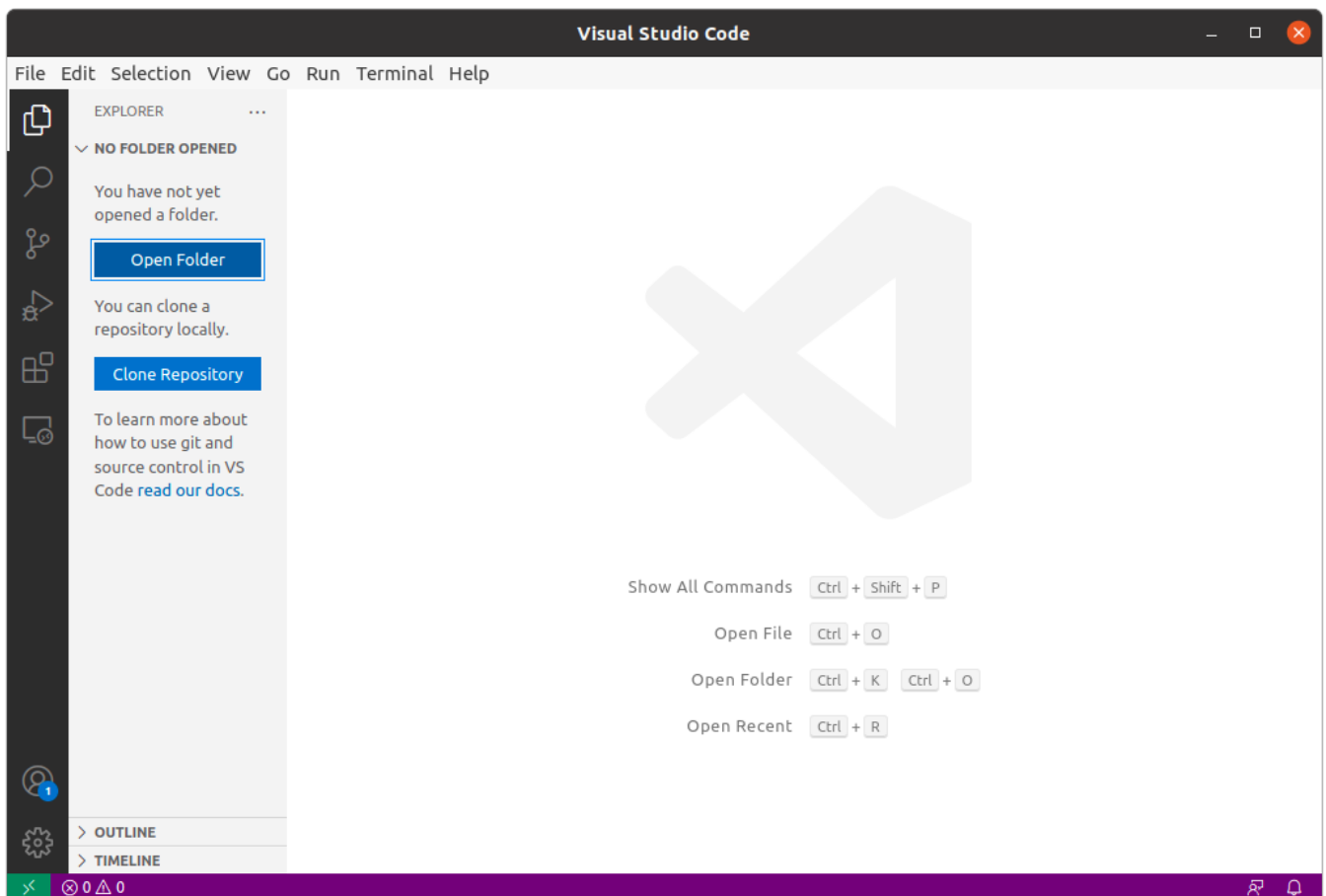
```
usman@usman-7G-Series: ~  
usman~$ sudo apt update  
Hit:1 http://deb.anydesk.com all InRelease  
Hit:2 http://pk.archive.ubuntu.com/ubuntu focal InRelease  
Hit:3 http://packages.microsoft.com/repos/code stable InRelease  
Hit:4 http://security.ubuntu.com/ubuntu focal-security InRelease  
Hit:5 https://linux.teamviewer.com/deb stable InRelease  
Hit:6 http://ppa.launchpad.net/obsproject/obs-studio/ubuntu focal InRelease  
Hit:7 http://pk.archive.ubuntu.com/ubuntu focal-updates InRelease  
Hit:8 https://dl.google.com/linux/chrome/deb stable InRelease  
Hit:9 http://ppa.launchpad.net/umang/indicator-stickynotes/ubuntu focal InRelease  
Ign:10 https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/5.0 InRelease  
Hit:11 https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/5.0 Release  
Ign:13 https://packages.microsoft.com/repos/ms-teams stable InRelease  
Hit:14 http://pk.archive.ubuntu.com/ubuntu focal-backports InRelease  
Get:15 https://packages.microsoft.com/repos/ms-teams stable Release [17.0 kB]  
Get:16 https://packages.microsoft.com/repos/ms-teams stable Release.gpg [492 B]
```

```
sudo apt install gcc
```

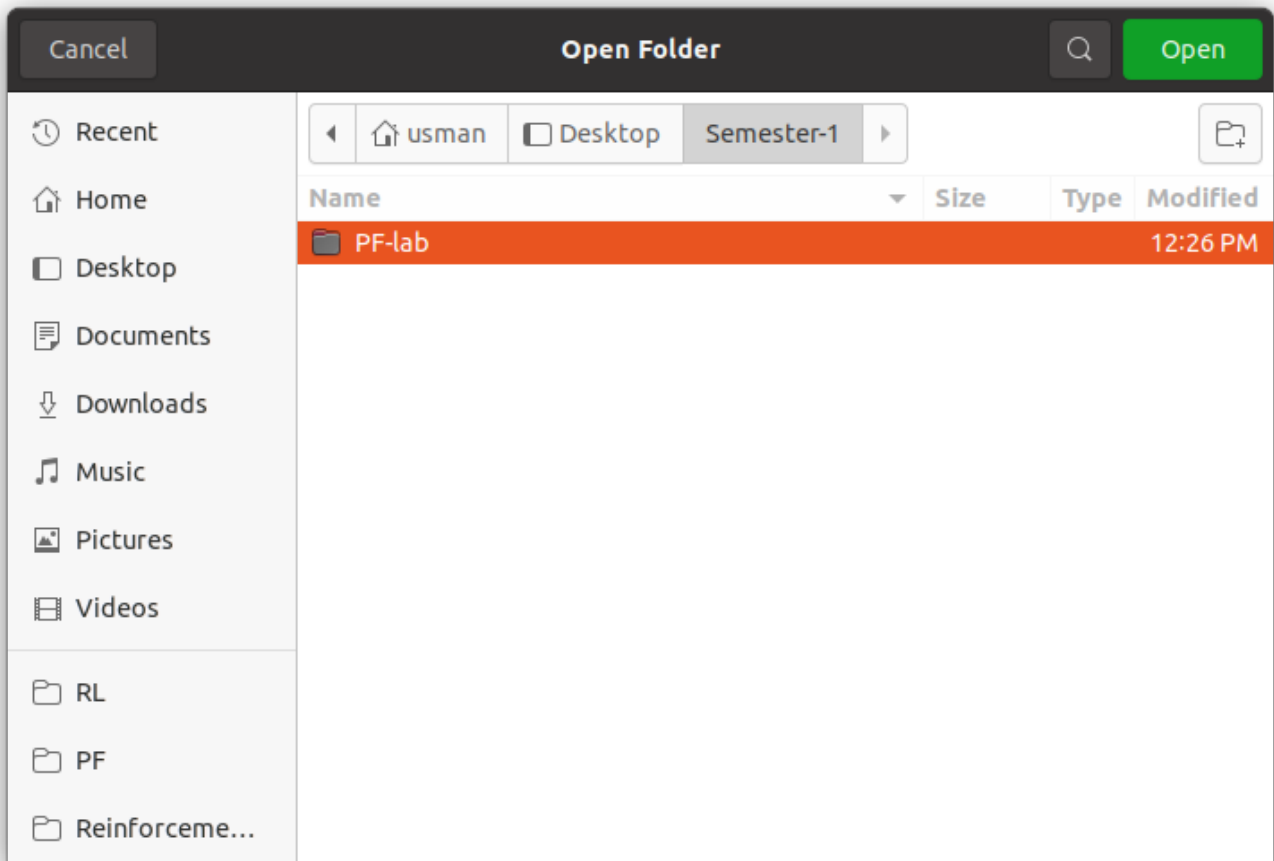
```
usman@usman-7G-Series: ~  
usman~$ sudo apt install gcc  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
gcc is already the newest version (4:9.3.0-1ubuntu2).  
The following packages were automatically installed and are no longer required:  
  hplip-data libfprint-2-tod1 libfwupdplugin1 libimagequant0 libllvm10  
  python3-olefile python3-pil python3-renderpm python3-reportlab  
  python3-reportlab-accel shim  
Use 'sudo apt autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 20 not upgraded.  
usman~$
```

## Start Coding in the Visual Studio Code Editor

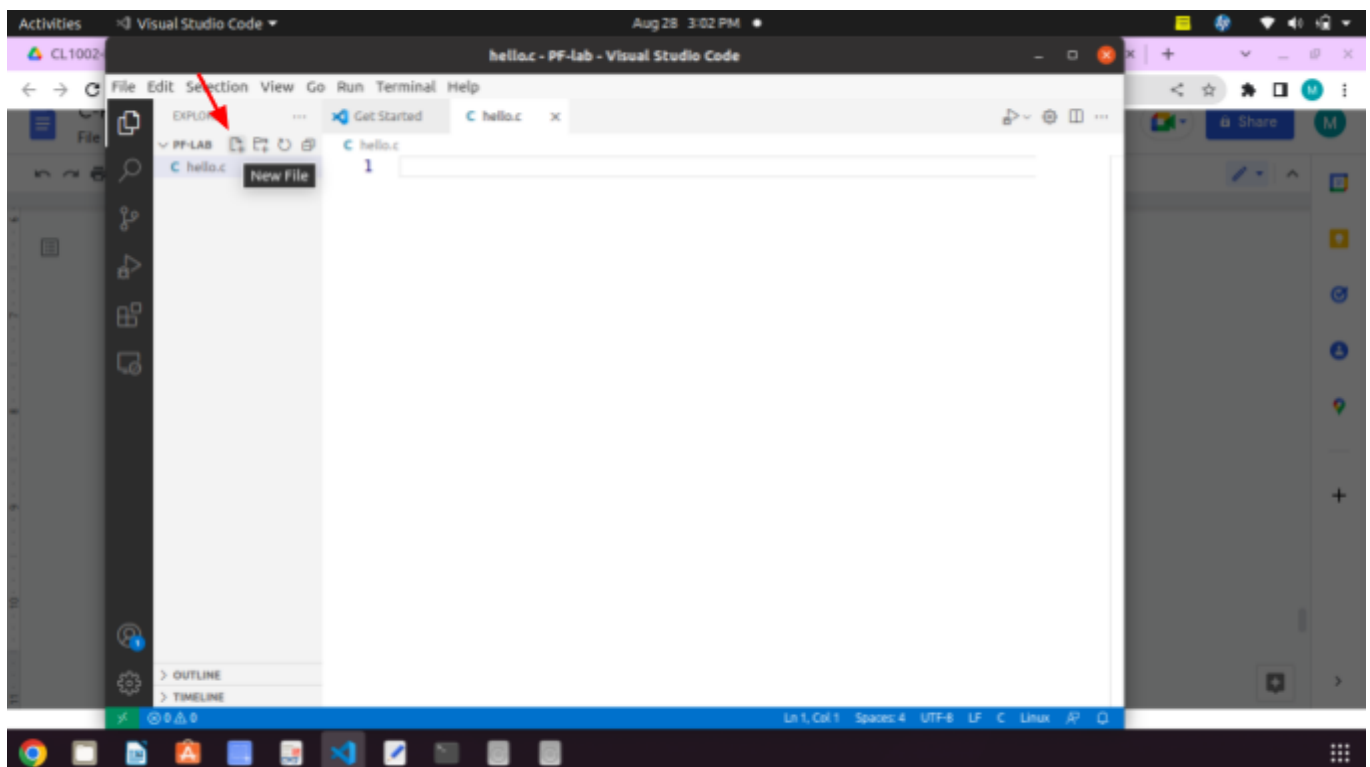
Open the VS Code and Click on Open Folder.



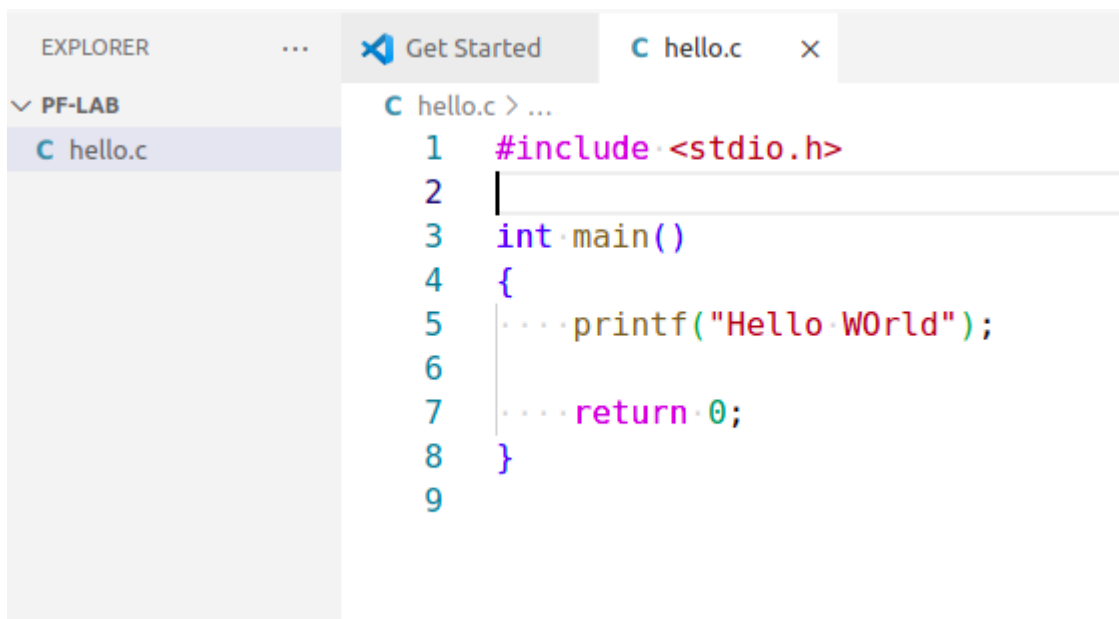
Click on Open.



Move the mouse over the PF-LAB folder; it shows a + Click on the button and write the file name as hello.c, as shown below.



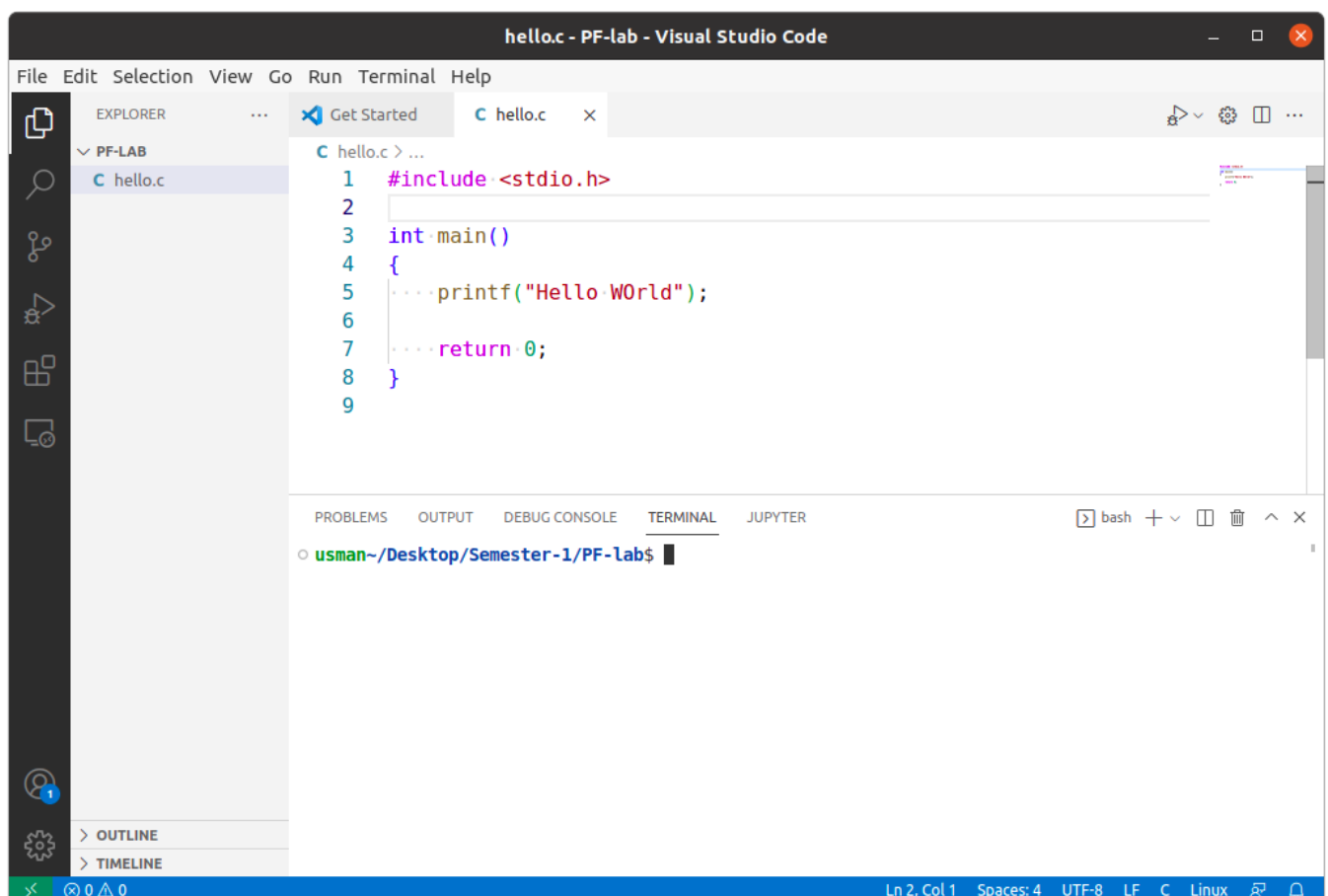
Now write and understand simple C programming in the VS Code editor.



The screenshot shows the Visual Studio Code editor interface. The Explorer sidebar on the left shows a folder named 'PF-LAB' containing a file 'hello.c'. The main editor area has a tab for 'hello.c' and displays the following C code:

```
C hello.c > ...
1  #include <stdio.h>
2  |
3  int main()
4  {
5      printf("Hello World");
6
7      return 0;
8  }
9
```

After Writing the Code Click on Terminal Tab to open the Terminal.



To Compile the Code type

```
gcc hello.c -o hello
```

As shown below.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
● usman~/Desktop/Semester-1/PF-lab$ gcc hello.c -o hello
○ usman~/Desktop/Semester-1/PF-lab$
```

To Execute the Code type

```
./hello
```

It shows the following output.

```
Hello World
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
● usman~/Desktop/Semester-1/PF-lab$ ./hello
Hello World
○ usman~/Desktop/Semester-1/PF-lab$
```

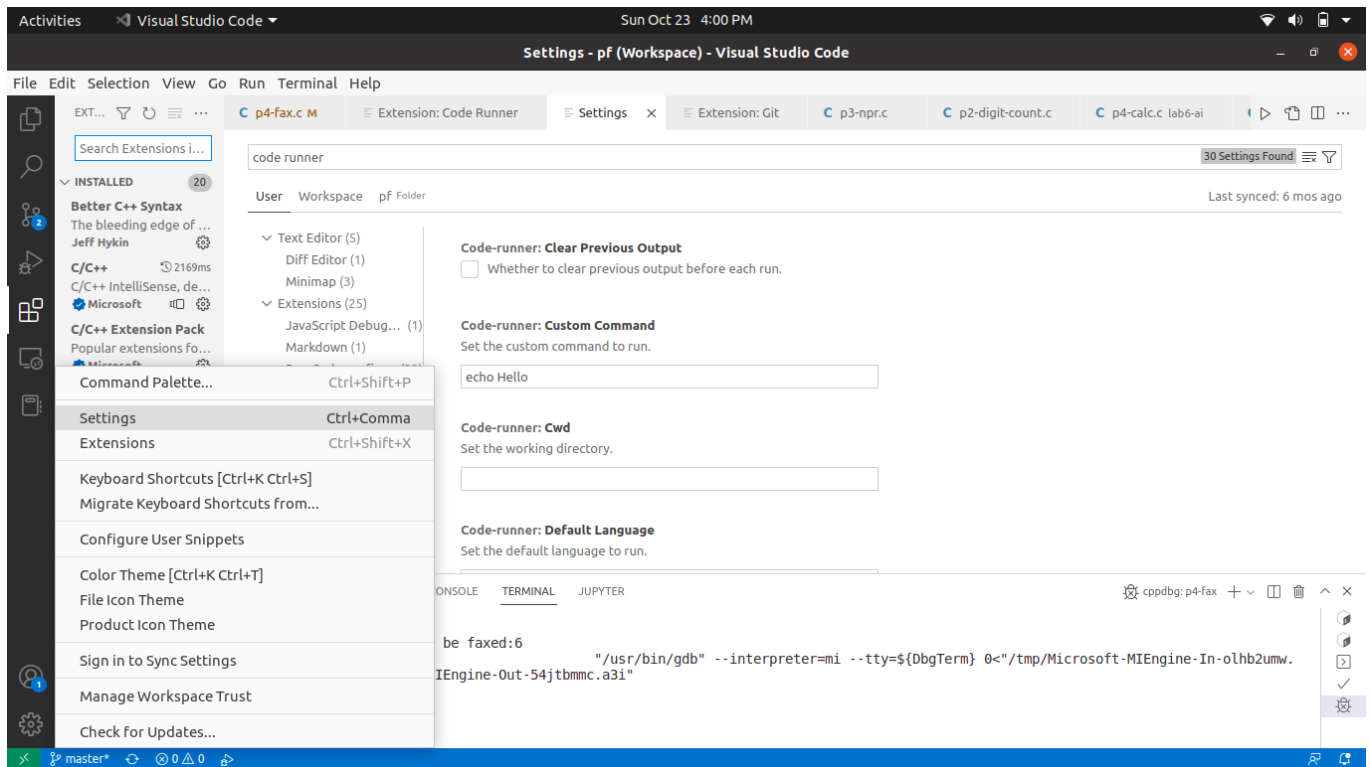
To directly compile and execute the c program:

Firstly install the Code runner Extension as shown below.

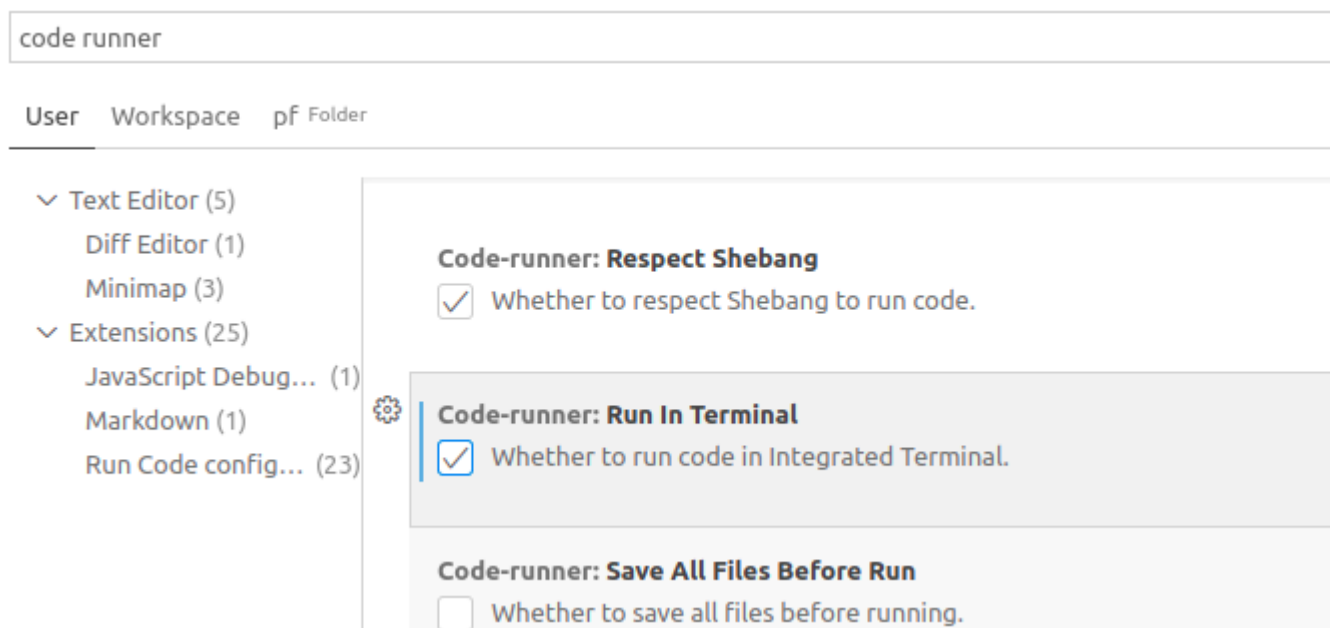


Secondly open the settings and search for code runner as shown below.






Make sure to tick the checkbox **Run in Terminal**



Now you can directly compile and execute the code. with the help of Run Code button as shown below.  
Or through the keyboard shortcut (**Ctrl+Alt+N**).

C escape.c x C p1-table.c C p2-table.c C p3-factors.c C p5-ncr.c p4-unit-v2.out C p4-unit-v2.c C p4-bill.c

```
pf > lab4 > C escape.c > main()
1  #include <stdio.h>
2  int main()
3  {
4      printf("Hello World\n");
5      . . .
6      return 0;
7  }
8
```



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

Code-lab4 + v

```
usman~/pf/lab4$ cd ~/home/usman/pf/lab4/* && gcc escape.c -o escape && ~/home/usman/pf/lab4/"escape
Hello World
usman~/pf/lab4$
```