How to Install GCC Compiler on Ubuntu 22.04

DC Team

Is anyone interested in Linux programming? software development should start by installing the GCC (GNU Compiler Collection) on Ubuntu. You may build and execute programs written in C, and C++. Also, other programming languages use the robust compiler suite GCC. This guide will show you how to install GCC on a Ubuntu. A step-by-step so that you have all the tools you need to start writing and creating software immediately.

A free software compiler system called GCC (GNU Compiler Collection). That can compile some computer languages including C, C++, Objective-C, and Fortran. A trustworthy compiler like GCC is necessary for programmers. Who wants to create C or C++ applications on a Linux-based platform?

Watch Video At: https://youtu.be/wUt1Il8ctLQ

Precondition

- Installed is Ubuntu 20.04 or 22.04.
- Access to a command line or terminal window.
- An individual with root or sudo rights.

How Do I Install GCC on Ubuntu?

On Ubuntu, installing the GCC Compiler is a simple procedure. Begin by opening the Terminal on your Ubuntu machine. Next, run a command to make sure your package lists are current. Next, run a short program to install GCC. Verify the GCC version to make sure the installation was successful. Now that you’ve finished, your Ubuntu system has the GCC compiler installed. You’re prepared to create, compile, and run C and C++ code for your programming projects. The best way to install GCC on Ubuntu depends on your customization requirements. Else whether a project needs a particular GCC version.

Medium 1: Installing GCC Compiler from Ubuntu Repositories in this Method.

The apt package manager is the quickest way to install GCC on Ubuntu. Nevertheless. There are drawbacks to installing GCC from the Ubuntu repositories, including a lack of customization options and potential dependency conflicts.

GCC compiler installation through Ubuntu repository:

Step 1: Update Package Repository

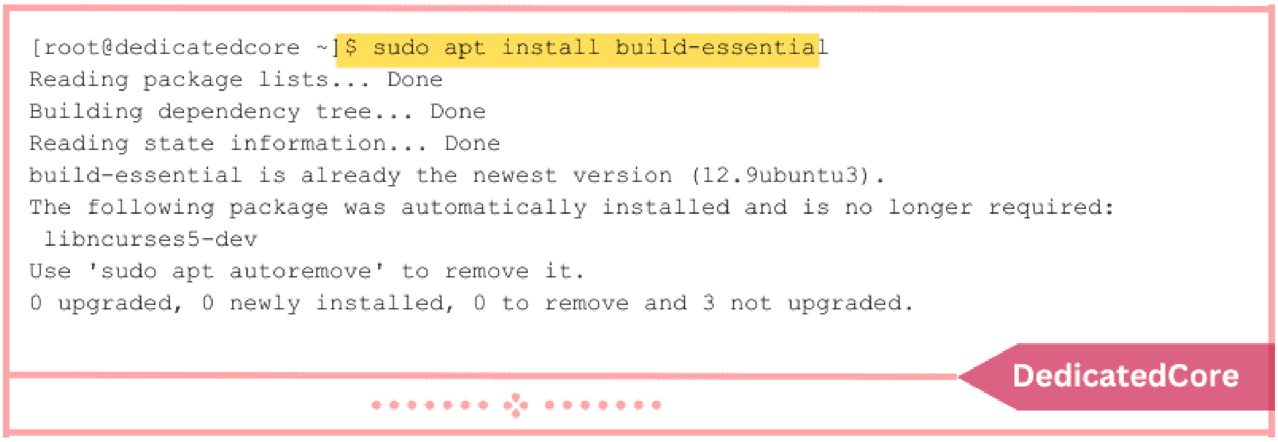
Use the command below to update the Ubuntu package repository:

sudo apt update

Step 2. Install build-essential Package

Install the package known as build-essential:

sudo apt install build-essential



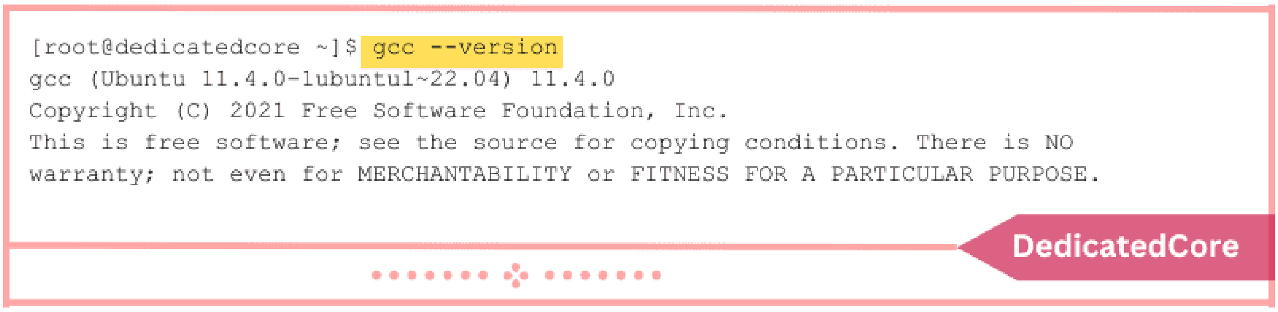
```
[root@dedicatedcore ~]$ sudo apt install build-essential
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
build-essential is already the newest version (12.9ubuntu3).
The following package was automatically installed and is no longer required:
 libncurses5-dev
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
```

The GCC compiler, along with other tools needed for software development, is included in the build-essential package.

Step 3: Determine GCC Version

To determine the GCC version, use the command:

gcc --version



```
[root@dedicatedcore ~]$ gcc --version
gcc (Ubuntu 11.4.0-1ubuntu1~22.04) 11.4.0
Copyright (C) 2021 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.
```

In this case, the GCC version is 11.4.0.

Medium 2: Installing several GCC Versions on Ubuntu using this Method

Users can install a certain GCC version (or multiple GCC versions) quickly and easily by using GCC PPAs (Personal Package Archives).

To do this:

Step 1: Update Repository First

Update the repository for Ubuntu packages:

sudo apt update

Step 2: Install Software-Properties Common Package

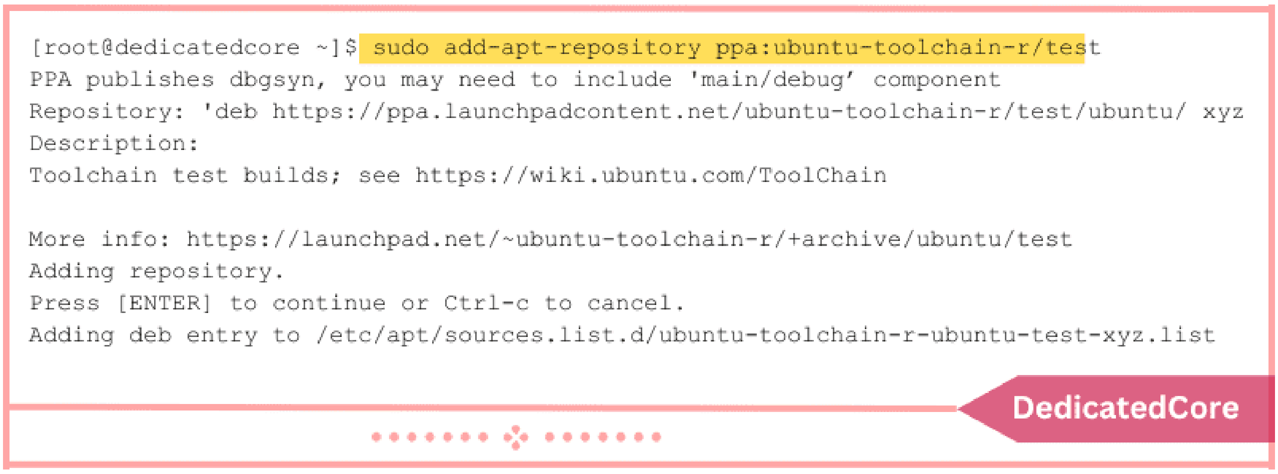
Execute the following command to install the software-properties-common package:

sudo apt install software-properties-common

Step 3: In the Below Step Include the GCC PPA

Include the GCC PPA, which contains every GCC compiler version:

sudo add-apt-repository ppa:ubuntu-toolchain-r/test



```
[root@dedicatedcore ~]$ sudo add-apt-repository ppa:ubuntu-toolchain-r/test
PPA publishes dbgsyn, you may need to include 'main/debug' component
Repository: 'deb https://ppa.launchpadcontent.net/ubuntu-toolchain-r/test/ubuntu/ xyz
Description:
Toolchain test builds; see https://wiki.ubuntu.com/ToolChain

More info: https://launchpad.net/~ubuntu-toolchain-r/+archive/ubuntu/test
Adding repository.
Press [ENTER] to continue or Ctrl-c to cancel.
Adding deb entry to /etc/apt/sources.list.d/ubuntu-toolchain-r-ubuntu-test-xyz.list
```

Step 4: Now, Add the PPA Packages

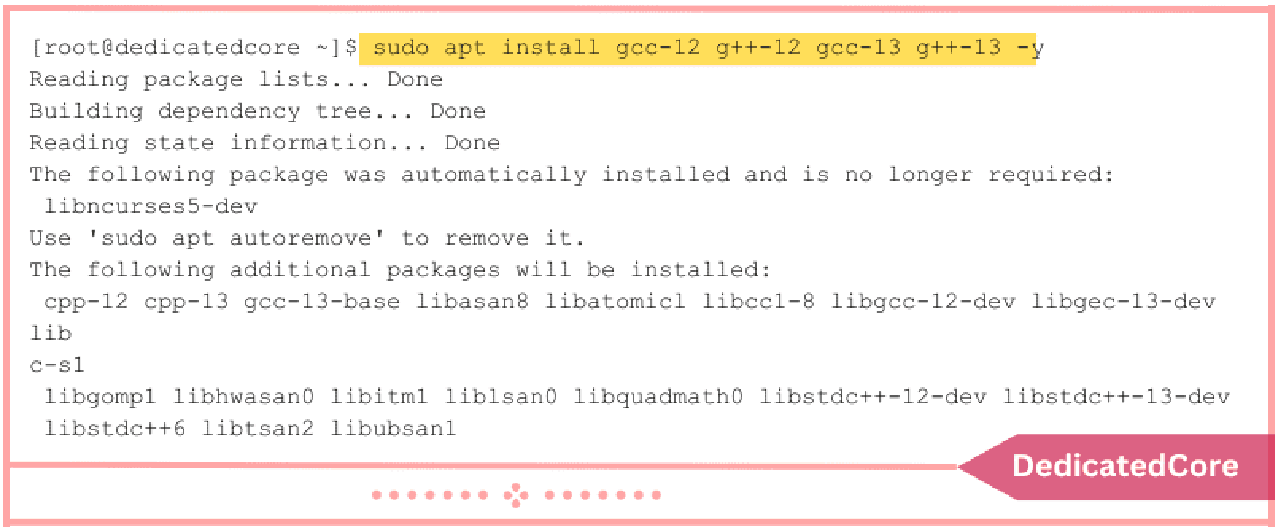
Add the PPA packages to the updated package list:

sudo apt update

Step 5: The next thing is to Install One or More GCC Versions

Use the command listed below to install one or more GCC versions. For instance, type: to install GCC 12 and GCC 13.

sudo apt install gcc-12 g++-12 gcc-13 g++-13 -y



```
[root@dedicatedcore ~]$ sudo apt install gcc-12 g++-12 gcc-13 g++-13 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
 libncurses5-dev
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
 cpp-12 cpp-13 gcc-13-base libasan8 libatomicl libccl-8 libgcc-12-dev libgec-13-dev
lib
c-sl
 libgompl libhwasan0 libitml liblsan0 libquadmath0 libstdc++-12-dev libstdc++-13-dev
 libstdc++6 libtsan2 libubsanl
```

Step 6: Solution to Handle Several GCC Versions

Users can handle several GCC versions with the update-alternatives tool: The update-alternatives filesystem should now include the GCC 12 alternative.

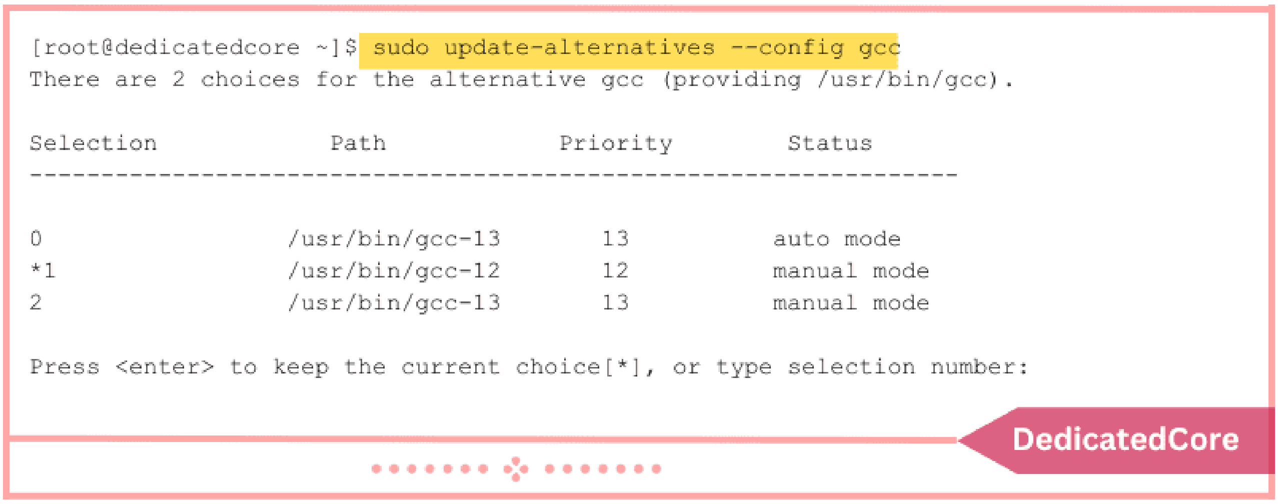
sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-12 12 --slave /usr/bin/g++ g++ /usr/bin/g++-12
The update-alternatives filesystem should now include the GCC 13 alternative.

sudo update-alternatives --install /usr/bin/gcc gcc /usr/bin/gcc-13 13 --slave /usr/bin/g++ g++ /usr/bin/g++-13

Step 7: Switch Between Installed GCC Versions

To switch between the installed GCC versions, use the update-alternatives tool:

sudo update-alternatives --config gcc



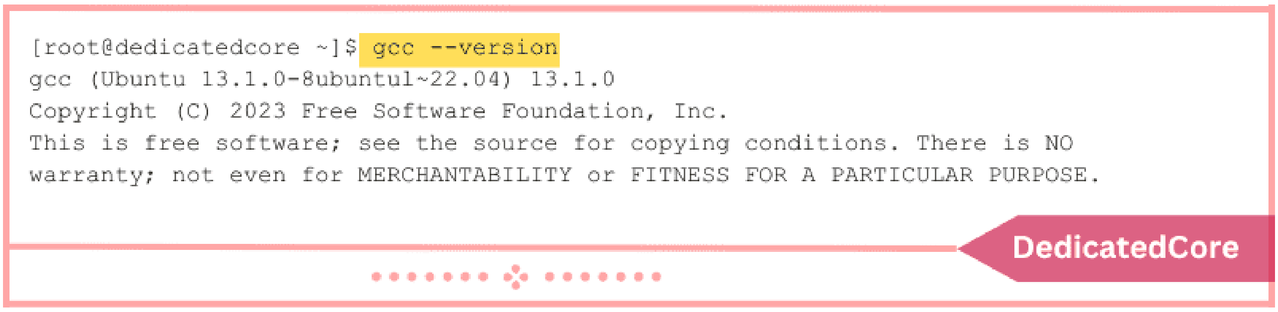
```
[root@dedicatedcore ~]$ sudo update-alternatives --config gcc
There are 2 choices for the alternative gcc (providing /usr/bin/gcc).

Selection            Path             Priority        Status
-----------------------------------------------------------------

0                  /usr/bin/gcc-13      13          auto mode
*1                 /usr/bin/gcc-12      12          manual mode
2                  /usr/bin/gcc-13      13          manual mode

Press <enter> to keep the current choice[*], or type selection number:
```

The system prompts the user to choose the default GCC version from a list of the installed GCC versions. For the version you wish to use, enter the selection number.

Step 8: Determine the Most Recent GCC Version

Use the following command to determine the most recent GCC version:

gcc --version



```
[root@dedicatedcore ~]$ gcc --version
gcc (Ubuntu 13.1.0-8ubuntu1~22.04) 13.1.0
Copyright (C) 2023 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.
```

In this example, the default GCC version is 13.1.0.

Medium 3: Install GCC Compiler from Source on Ubuntu

A sophisticated installation technique that enables users to alter the GCC configuration is building GCC from the source. Additionally, you can modify GCC's install sites or tailor it to a particular piece of hardware.

Installing the GCC compiler and obtaining the GCC source code from authorized repositories:

Step 1: Install Required Dependencies

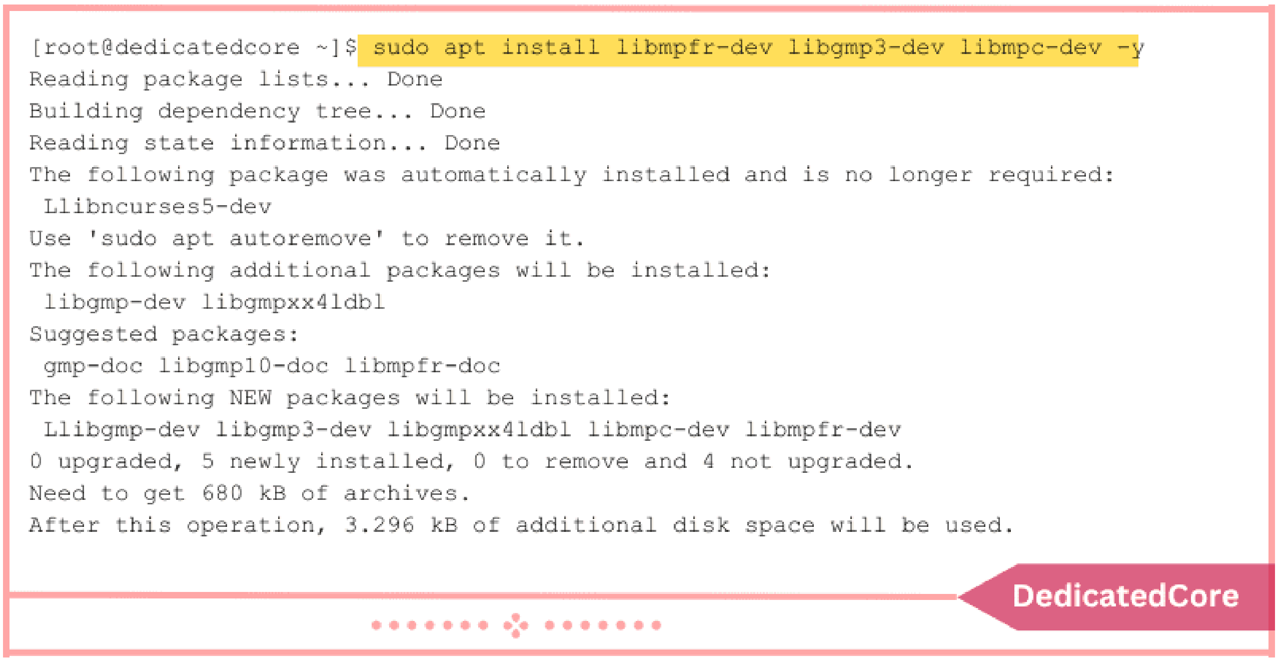
Run the Command below to Install the required Dependencies:

sudo apt install build-essential

Step 2: Install Packages Below

To help with GCC source compilation, install the libgmp3-dev, libmpfr-dev, and libmpc-dev packages:

sudo apt install libmpfr-dev libgmp3-dev libmpc-dev -y

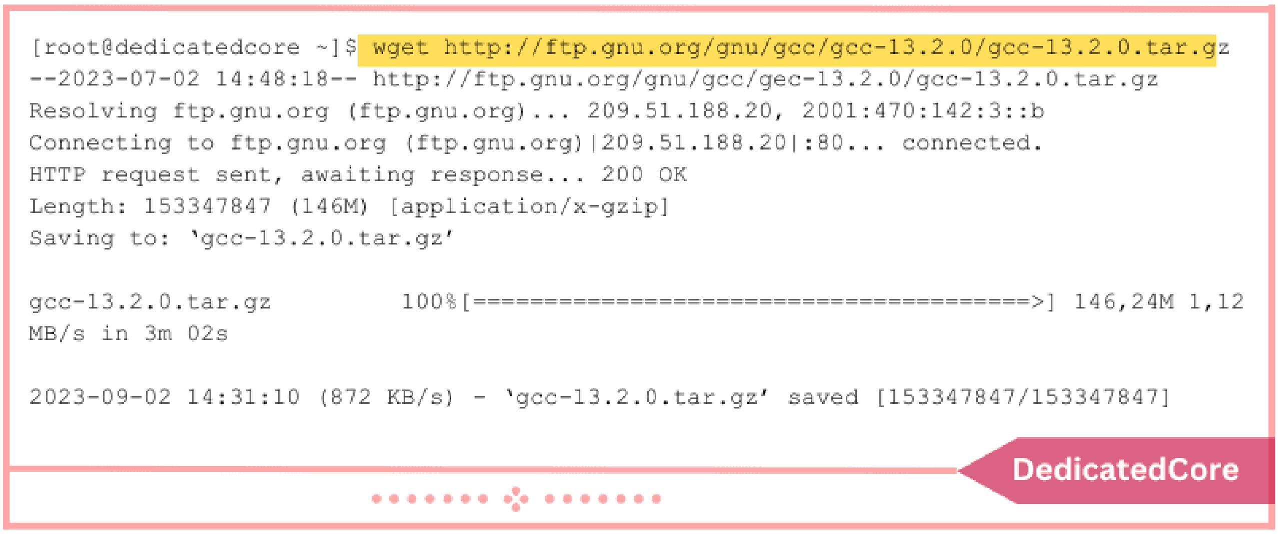


```
[root@dedicatedcore ~]$ sudo apt install libmpfr-dev libgmp3-dev libmpc-dev -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
 Llibncurses5-dev
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
 libgmp-dev libgmpxx4ldbl
Suggested packages:
 gmp-doc libgmp10-doc libmpfr-doc
The following NEW packages will be installed:
 Llibgmp-dev libgmp3-dev libgmpxx4ldbl libmpc-dev libmpfr-dev
0 upgraded, 5 newly installed, 0 to remove and 4 not upgraded.
Need to get 680 kB of archives.
After this operation, 3.296 kB of additional disk space will be used.
```

Step 3: Using Wget Command Download the GCC Source Code

Download the GCC source code from the GCC website using the wget command. The source code for GCC version 13.2.0 is present in the file gcc-13.2.0.tar.gz in the following example:

wget http://ftp.gnu.org/gnu/gcc/gcc-13.2.0/gcc-13.2.0.tar.gz



```
[root@dedicatedcore ~]$ wget http://ftp.gnu.org/gnu/gcc/gcc-13.2.0/gcc-13.2.0.tar.gz
--2023-07-02 14:48:18-- http://ftp.gnu.org/gnu/gcc/gec-13.2.0/gcc-13.2.0.tar.gz
Resolving ftp.gnu.org (ftp.gnu.org)... 209.51.188.20, 2001:470:142:3::b
Connecting to ftp.gnu.org (ftp.gnu.org)|209.51.188.20|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 153347847 (146M) [application/x-gzip]
Saving to: 'gcc-13.2.0.tar.gz'

gcc-13.2.0.tar.gz         100%[======================================>] 146,24M 1,12
MB/s in 3m 02s

2023-09-02 14:31:10 (872 KB/s) - 'gcc-13.2.0.tar.gz' saved [153347847/153347847]
```

To reflect the GCC version you want to install, change the URL in the command.

Step 4: Extract the GCC Build Files

Utilize the following command to extract the GCC build files:

tar -xf gcc-13.2.0.tar.gz

Step 5: Opening Directory of GCC

Open the directory of the GCC with the following command:

cd gcc-13.2.0

Step 6: Setting Up the GCC Build's Options

Set the GCC build's options:

./configure -v --build=x86_64-linux-gnu --host=x86_64-linux-gnu --target=x86_64-linux-gnu --prefix=/usr/local/gcc-13.2.0 --enable-checking=release --enable-languages=c,c++ --disable-multilib --program-suffix=-13.2.0
To explore the options available and suggested configuration methods, go to the official GNU GCC configuration page.

Step 7: Use the "make" Command to Launch the GCC Build Process

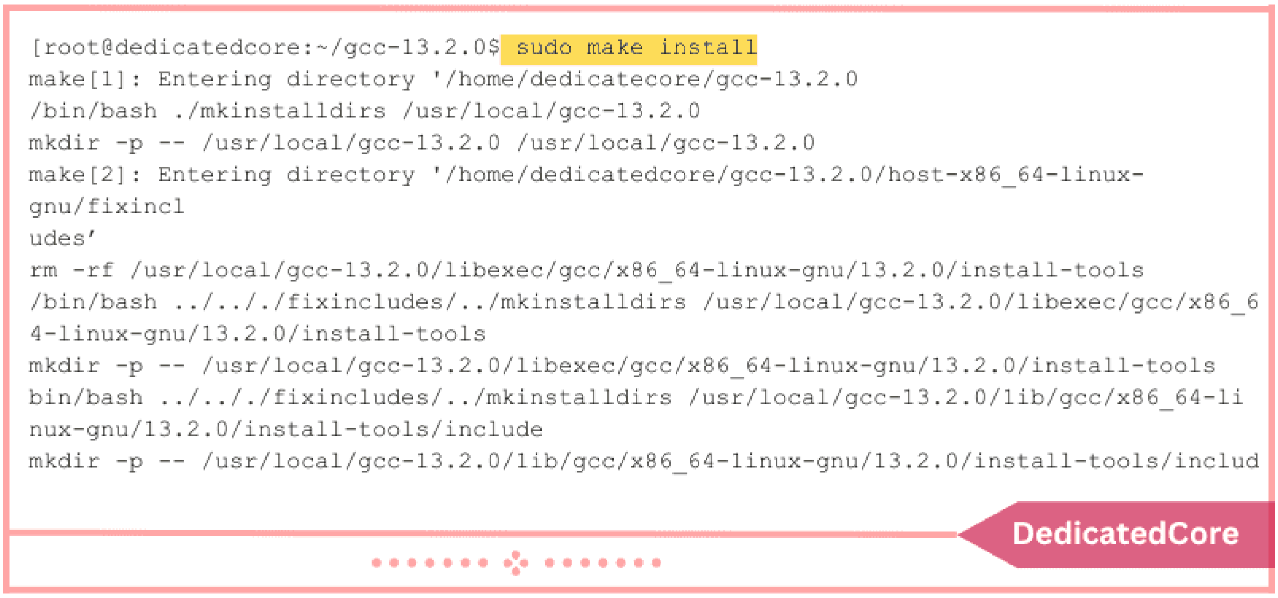
Launch the GCC build process using the make command:

make -j3
It can take a long time and a lot of resources to complete the build process. The j3 instruction instructs the machine to perform the task using three cores. Adapt the number of cores to the setup and capabilities of your system.

Step 8: Install GCC Once the Previous Procedure is Done

Once the building procedure is finished, run the following command to install GCC:

sudo make install

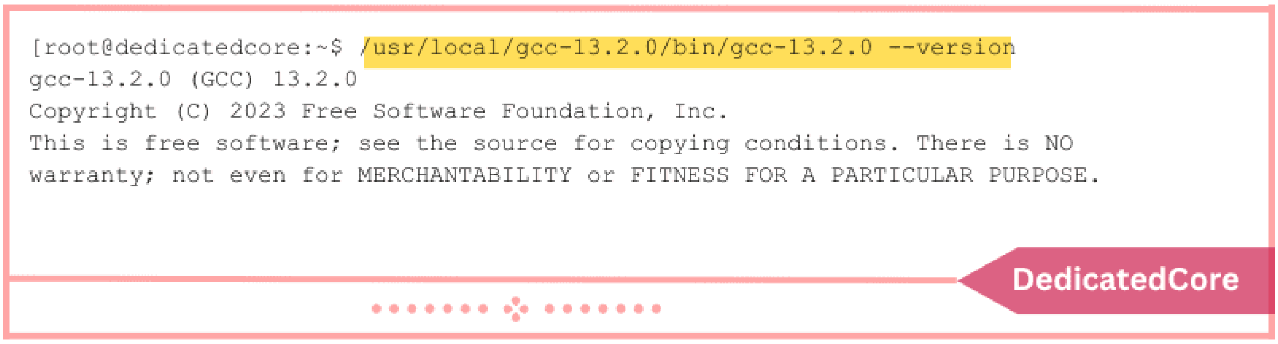


```
[root@dedicatedcore:~/gcc-13.2.0$ sudo make install
make[1]: Entering directory '/home/dedicatecore/gcc-13.2.0
/bin/bash ./mkinstalldirs /usr/local/gcc-13.2.0
mkdir -p -- /usr/local/gcc-13.2.0 /usr/local/gcc-13.2.0
make[2]: Entering directory '/home/dedicatedcore/gcc-13.2.0/host-x86_64-linux-
gnu/fixincl
udes'
rm -rf /usr/local/gcc-13.2.0/libexec/gcc/x86_64-linux-gnu/13.2.0/install-tools
/bin/bash ../.././fixincludes/../mkinstalldirs /usr/local/gcc-13.2.0/libexec/gcc/x86_6
4-linux-gnu/13.2.0/install-tools
mkdir -p -- /usr/local/gcc-13.2.0/libexec/gcc/x86_64-linux-gnu/13.2.0/install-tools
bin/bash ../.././fixincludes/../mkinstalldirs /usr/local/gcc-13.2.0/lib/gcc/x86_64-li
nux-gnu/13.2.0/install-tools/include
mkdir -p -- /usr/local/gcc-13.2.0/lib/gcc/x86_64-linux-gnu/13.2.0/install-tools/includ
```

Step 9: Finally, Check GCC Version

Type the command below to check the GCC version:

/usr/local/gcc-13.2.0/bin/gcc-13.2.0 --version



```
[root@dedicatedcore:~$ /usr/local/gcc-13.2.0/bin/gcc-13.2.0 --version
gcc-13.2.0 (GCC) 13.2.0
Copyright (C) 2023 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.
```

The installation of GCC 13.2.0 has been verified by the system.

Conclusion

As a programmer or developer, install the GCC compiler on your Ubuntu system. It is an essential first step. You may compile and execute code written in a variety of programming languages using GCC's extensive toolkit. You've given yourself the necessary tools to delve into the realm of Ubuntu. Software creation by following the simple steps suggested in this guide.

Your creativity and problem-solving abilities may now be fully unleashed. Hope you understand install gcc for Ubuntu here in detail. As you develop, assemble, and run your programs with assurance. You now understand how to use three distinct techniques to install the GCC compiler on Ubuntu. Using GCC on an Ubuntu machine enables you to do a wide range of programming. Also debugging, and system administration operations, as well as compile and run C and C++ code.