Data Flair

How to Install C – Learn to Install GCC Compiler for Ubuntu

How to Install C Programming Language?

Still stuck with the same problem of how to install C? You cannot install a programming language. Since C is a programming language, you directly start writing programs in it.

Then what do you install? The answer is- you need to install a Compiler, i.e. GCC Compiler, to compile and run your programs using it. This compiler recognizes the language by its syntax.

Steps to Install C Language

In this installation tutorial, we will help you to write and compile your own programs in the C programming language using terminal for Linux Operating System.

1. Environment for C programming

The terminal command prompt is pre-installed in your LINUX operating system. Our task now is to install the GCC compiler using terminal to execute the C programs.

2. How to install GCC compiler using terminal in Linux?

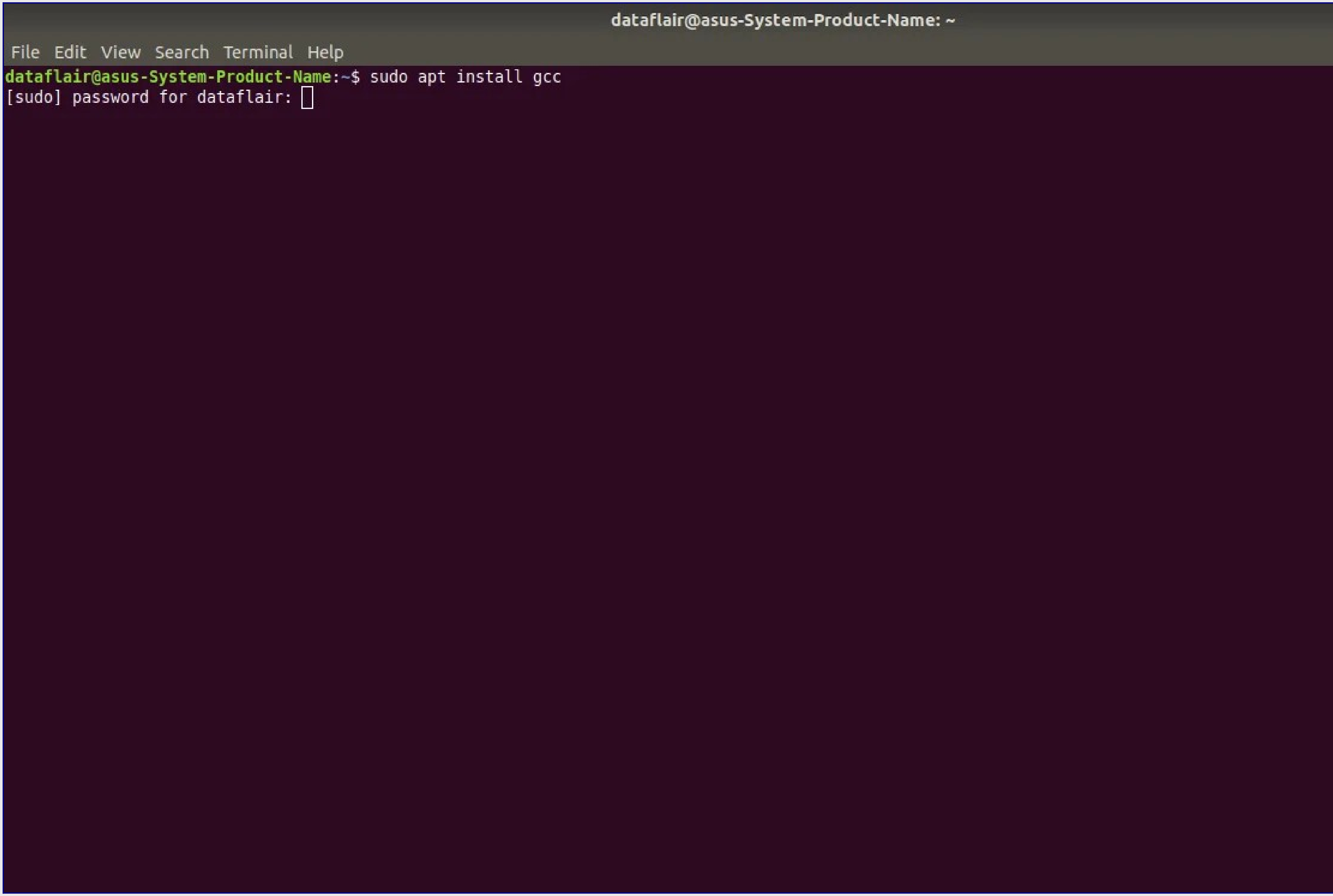
You can find the terminal icon on your Desktop screen. If not, you can find it through the search menu. A shortcut to open terminal is: Press Ctrl + Alt + T.

Open terminal so that we can proceed with the further steps to install the GCC Compiler.

The main command for installing the GCC compiler using terminal on Ubuntu is:

```
sudo apt install GCC
```

Here, GCC is the C compiler.

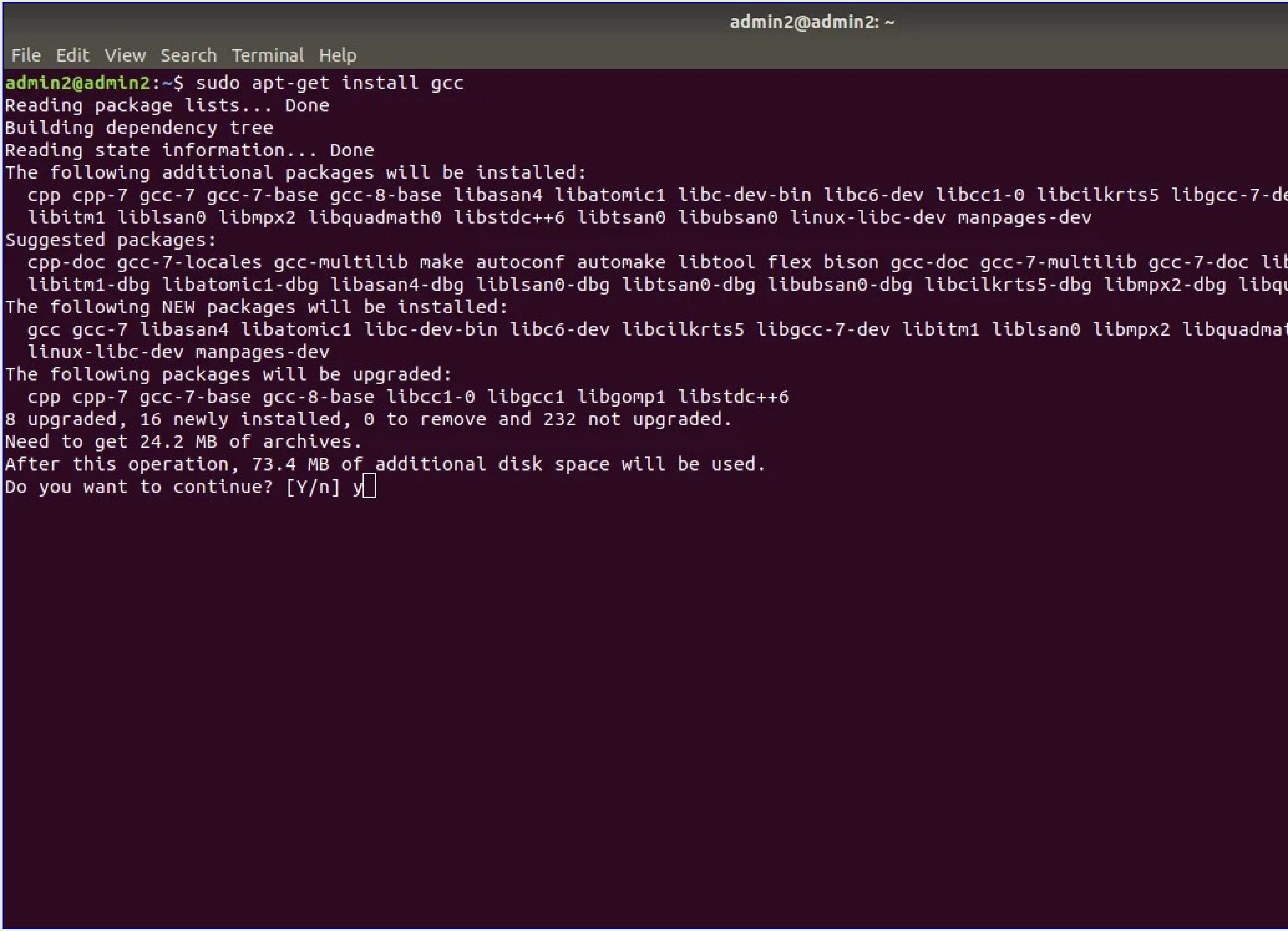


```
dataflair@asus-System-Product-Name:~$ sudo apt install gcc
[sudo] password for dataflair: 
```

GCC is the C compiler

If you do not have permission to install the GCC compiler, this message would be displayed where you are required to enter the admin password to get the permission granted:

If you have permission, then the installation process will proceed as follows:

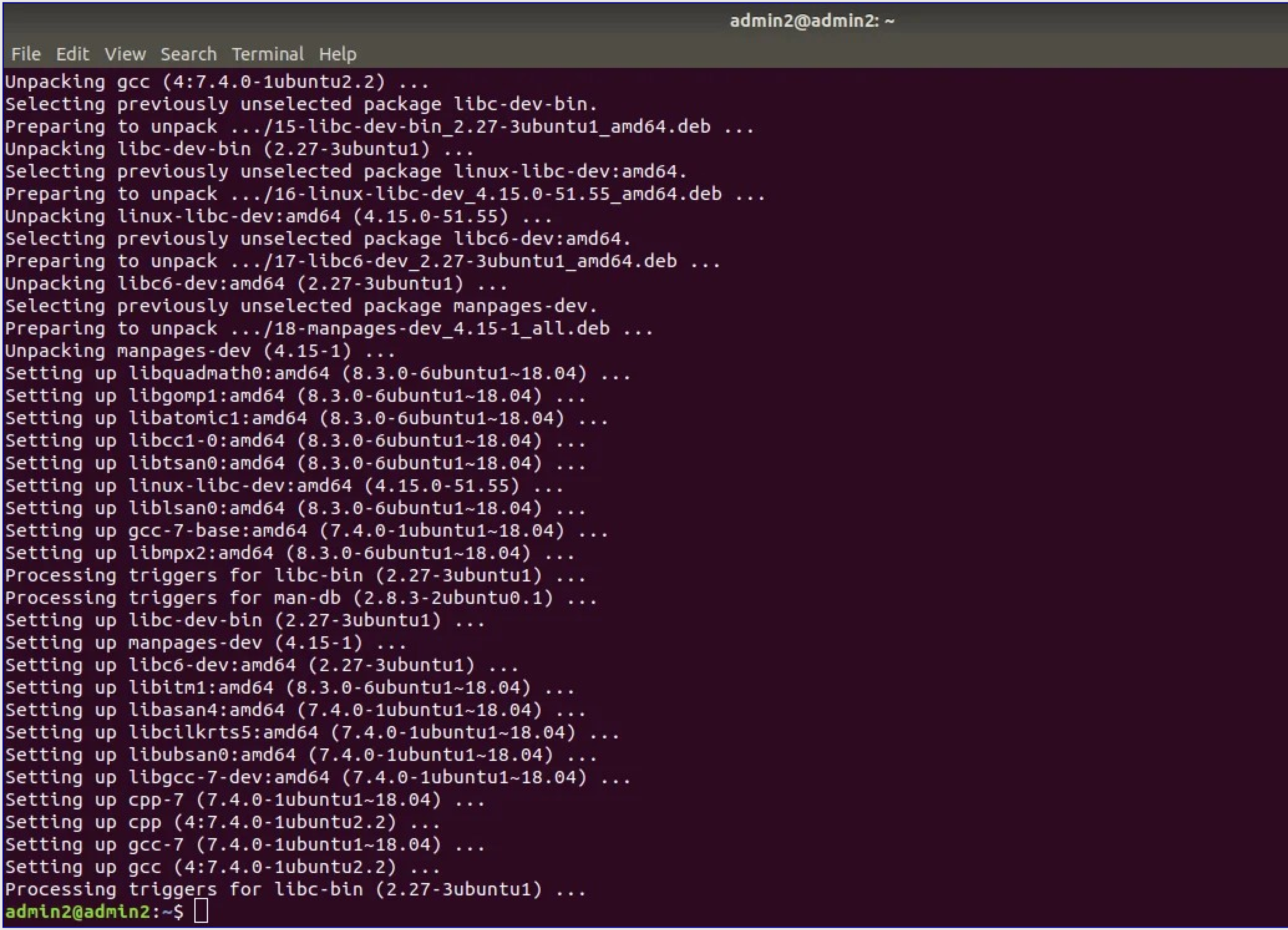


```
admin2@admin2:~$ sudo apt-get install gcc
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  cpp cpp-7 gcc-7 gcc-7-base gcc-8-base libasan4 libatomic1 libc-dev-bin libc6-dev libcc1-0 libcilkrts5 libgcc-7-de
  libitm1 liblsan0 libmpx2 libquadmath0 libstdc++6 libtsan0 libubsan0 linux-libc-dev manpages-dev
Suggested packages:
  cpp-doc gcc-7-locales gcc-multilib make autoconf automake libtool flex bison gcc-doc gcc-7-multilib gcc-7-doc lib
  libitm1-dbg libatomic1-dbg libasan4-dbg liblsan0-dbg libtsan0-dbg libubsan0-dbg libcilkrts5-dbg libmpx2-dbg libqu
The following NEW packages will be installed:
  gcc gcc-7 libasan4 libatomic1 libc-dev-bin libc6-dev libcilkrts5 libgcc-7-dev libitm1 liblsan0 libmpx2 libquadmat
  linux-libc-dev manpages-dev
The following packages will be upgraded:
  cpp cpp-7 gcc-7-base gcc-8-base libcc1-0 libgcc1 libgomp1 libstdc++6
8 upgraded, 16 newly installed, 0 to remove and 232 not upgraded.
Need to get 24.2 MB of archives.
After this operation, 73.4 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

GCC Compiler Installation Process

Type ‘y’ when the command prompt asks “Do you want to continue?” and then press Enter.

Thereafter, your installation will begin



```
Unpacking gcc (4:7.4.0-1ubuntu2.2) ...
Selecting previously unselected package libc-dev-bin.
Preparing to unpack .../15-libc-dev-bin_2.27-3ubuntu1_amd64.deb ...
Unpacking libc-dev-bin (2.27-3ubuntu1) ...
Selecting previously unselected package linux-libc-dev:amd64.
Preparing to unpack .../16-linux-libc-dev_4.15.0-51.55_amd64.deb ...
Unpacking linux-libc-dev:amd64 (4.15.0-51.55) ...
Selecting previously unselected package libc6-dev:amd64.
Preparing to unpack .../17-libc6-dev_2.27-3ubuntu1_amd64.deb ...
Unpacking libc6-dev:amd64 (2.27-3ubuntu1) ...
Selecting previously unselected package manpages-dev.
Preparing to unpack .../18-manpages-dev_4.15-1_all.deb ...
Unpacking manpages-dev (4.15-1) ...
Setting up libquadmath0:amd64 (8.3.0-6ubuntu1~18.04) ...
Setting up libgomp1:amd64 (8.3.0-6ubuntu1~18.04) ...
Setting up libatomic1:amd64 (8.3.0-6ubuntu1~18.04) ...
Setting up libcc1-0:amd64 (8.3.0-6ubuntu1~18.04) ...
Setting up libtsan0:amd64 (8.3.0-6ubuntu1~18.04) ...
Setting up linux-libc-dev:amd64 (4.15.0-51.55) ...
Setting up liblsan0:amd64 (8.3.0-6ubuntu1~18.04) ...
Setting up gcc-7-base:amd64 (7.4.0-1ubuntu1~18.04) ...
Setting up libmpx2:amd64 (8.3.0-6ubuntu1~18.04) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Setting up libc-dev-bin (2.27-3ubuntu1) ...
Setting up manpages-dev (4.15-1) ...
Setting up libc6-dev:amd64 (2.27-3ubuntu1) ...
Setting up libitm1:amd64 (8.3.0-6ubuntu1~18.04) ...
Setting up libasan4:amd64 (7.4.0-1ubuntu1~18.04) ...
Setting up libcilkrts5:amd64 (7.4.0-1ubuntu1~18.04) ...
Setting up libubsan0:amd64 (7.4.0-1ubuntu1~18.04) ...
Setting up libgcc-7-dev:amd64 (7.4.0-1ubuntu1~18.04) ...
Setting up cpp-7 (7.4.0-1ubuntu1~18.04) ...
Setting up cpp (4:7.4.0-1ubuntu2.2) ...
Setting up gcc-7 (7.4.0-1ubuntu1~18.04) ...
Setting up gcc (4:7.4.0-1ubuntu2.2) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
admin2@admin2:~$ 
```

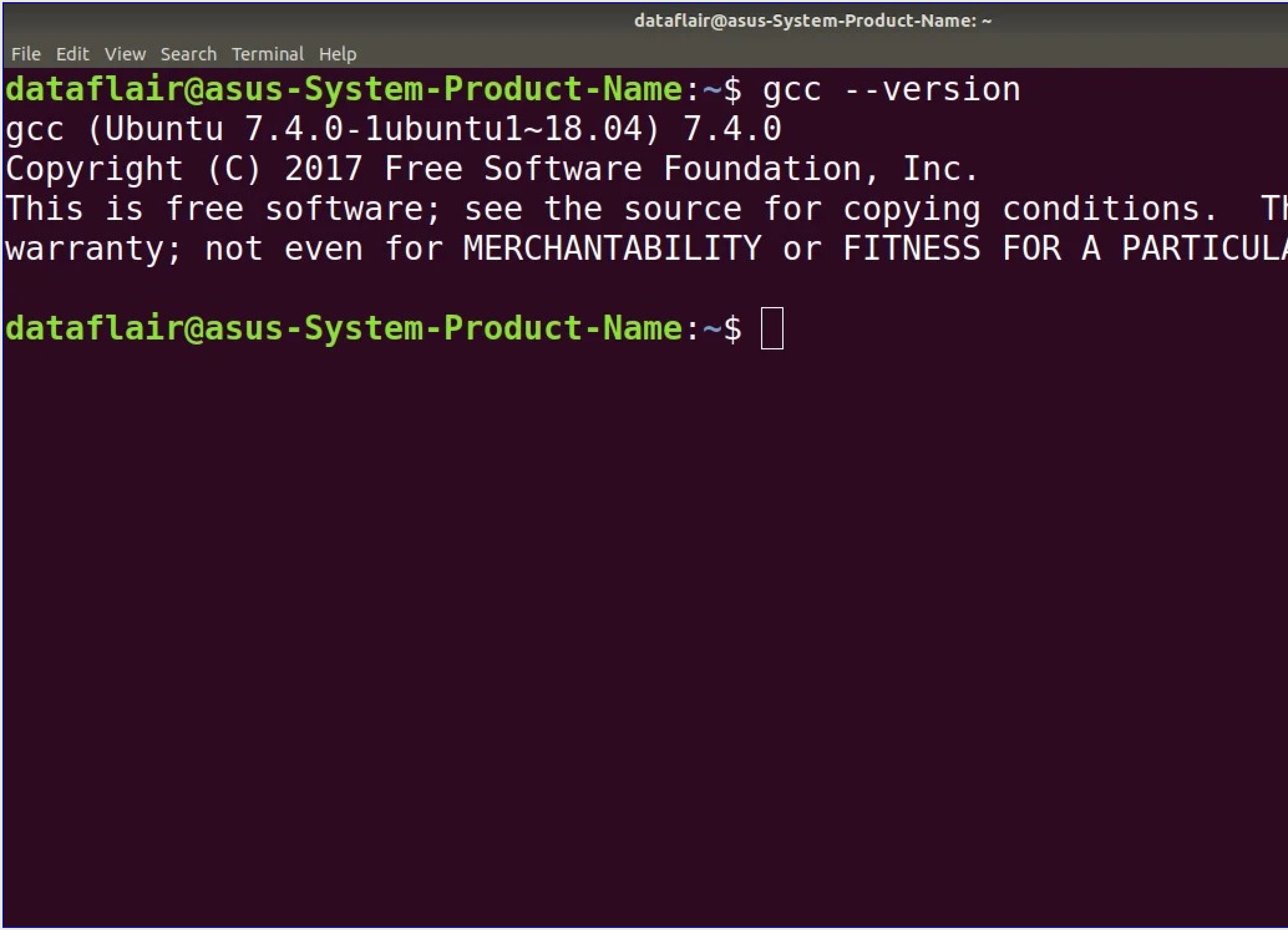
gcc installation begins

3. How to check the installed compiler version?

- You can find the terminal icon on your Desktop screen or on the search menu. A shortcut to open terminal is: Press Ctrl + Alt + T simultaneously.
- Now the terminal will open.
- The ‘gcc’ command will help you throughout your coding journey. It is a very important command that comes pre-installed on your latest Ubuntu version.
- Write the following command to know the version of the GCC compiler that you installed

GCC — version

After pressing enter, the version of the terminal software would be displayed.



dataflair@asus-System-Product-Name: ~
File Edit View Search Terminal Help

```
dataflair@asus-System-Product-Name:~$ gcc --version
gcc (Ubuntu 7.4.0-1ubuntu1~18.04) 7.4.0
Copyright (C) 2017 Free Software Foundation, Inc.
This is free software; see the source for copying conditions.  Th
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULA

dataflair@asus-System-Product-Name:~$ 
```

Displaying the version of the terminal software

This version is 7.4.0

If you don’t see such a message on your display screen, then it probably means that you haven’t successfully installed the GCC compiler.

4. How to create a C program?

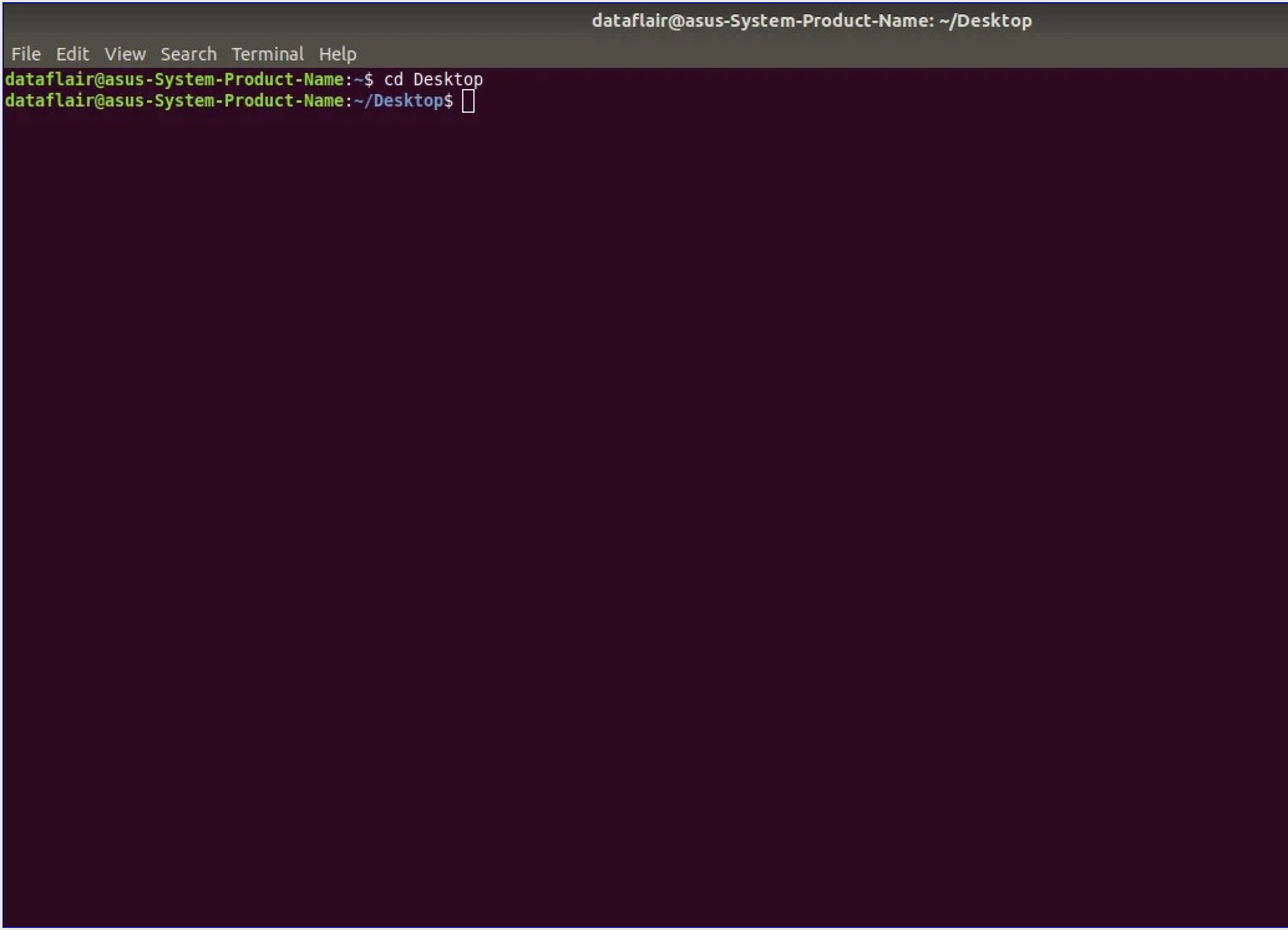
In order to create a C program, use the ‘touch’ command and give the name of the file with .c extension

But before that, you need to choose the directory you are working on.

Most of the programmers generally prefer to work on the Desktop folder as it proves to be quite convenient to write, modify and run your programs instantly.

In order to locate your files on the Desktop folder and to change the reference of the current directory to Desktop, use this command:

```
cd Desktop
```



dataflair@asus-System-Product-Name: ~/Desktop
File Edit View Search Terminal Help

```
dataflair@asus-System-Product-Name:~$ cd Desktop
dataflair@asus-System-Product-Name:~/Desktop$ 
```

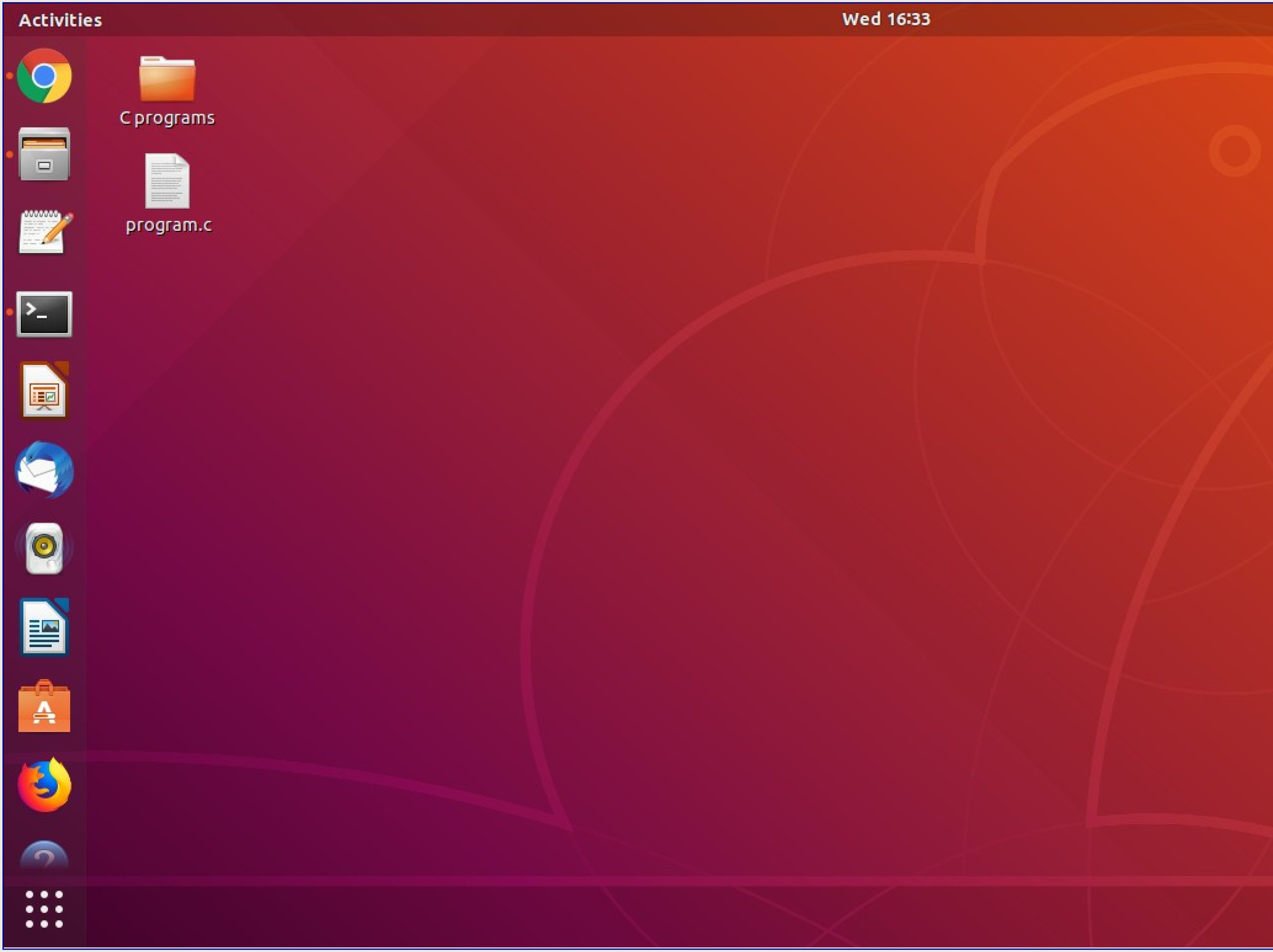
Inserting cd Desktop command

Key takeaway: Commands are case sensitive

The command for creating a program in C is:

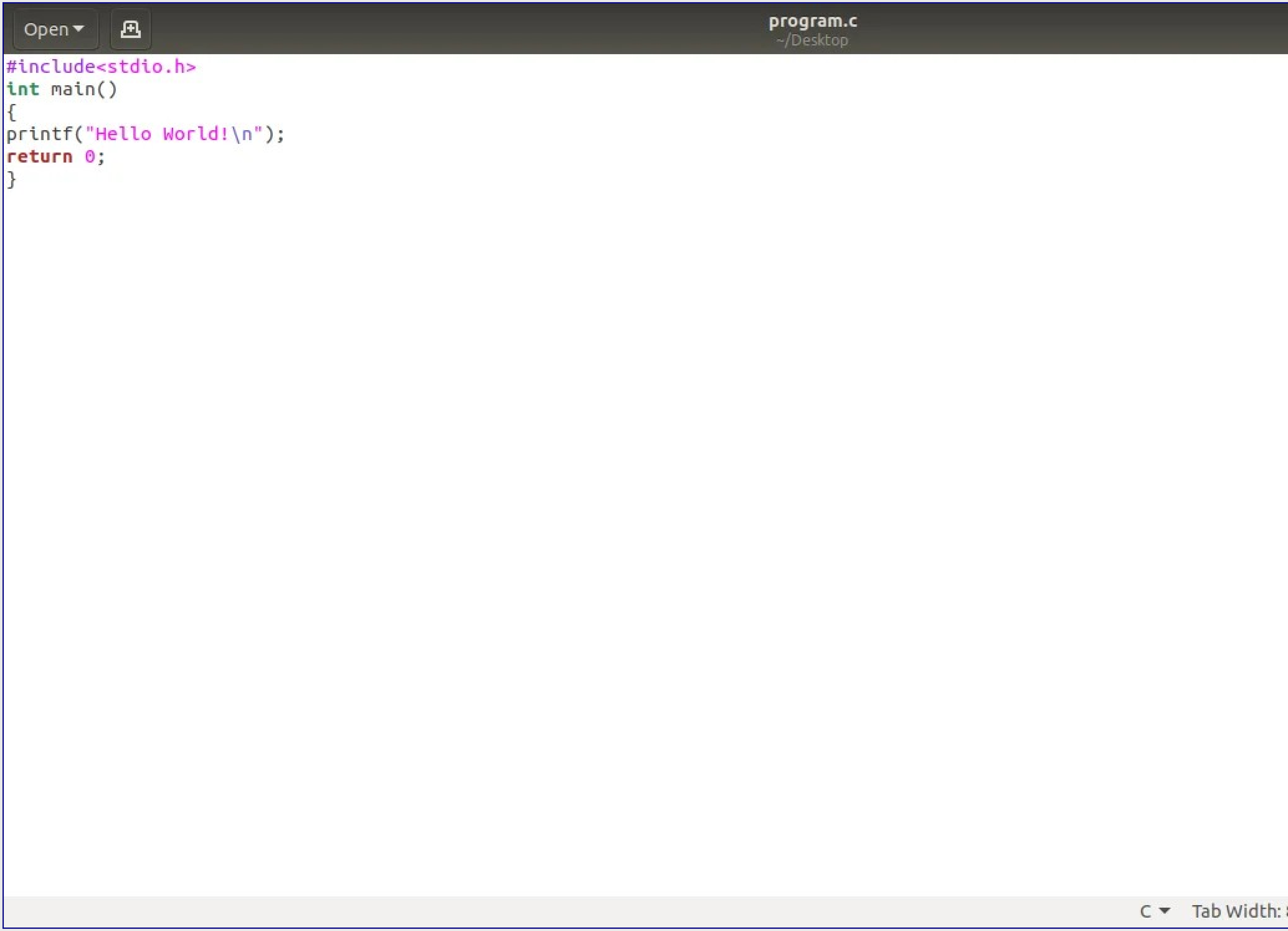
```
touch program.c
```

Now, a file has been created in our Desktop folder called program.c


Activities
Wed 16:33
C programs
program.c

The file is created on the desktop

Open this file and write a basic code – **“Hello World!”**



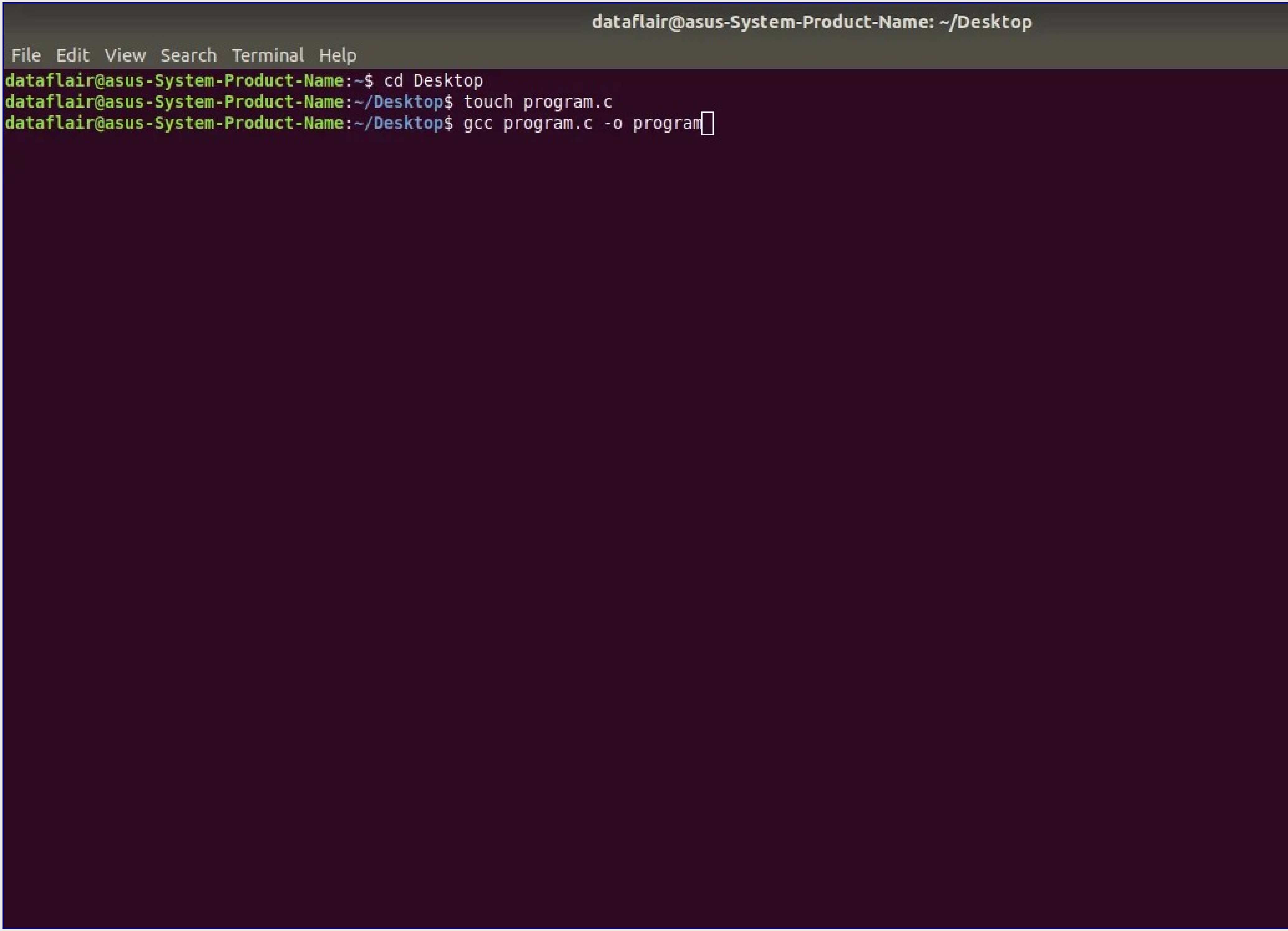
```
#include<stdio.h>
int main()
{
printf("Hello World!\n");
return 0;
}
```

5. How to compile and run a C program?

Now that we have created a C program called program.c, it is now time to compile it and run it in terminal.

To compile the code, we use the GCC command:

```
GCC program.c -o program
```


dataflair@asus-System-Product-Name: ~/Desktop
File Edit View Search Terminal Help

```
dataflair@asus-System-Product-Name:~$ cd Desktop
dataflair@asus-System-Product-Name:~/Desktop$ touch program.c
dataflair@asus-System-Product-Name:~/Desktop$ gcc program.c -o program
```

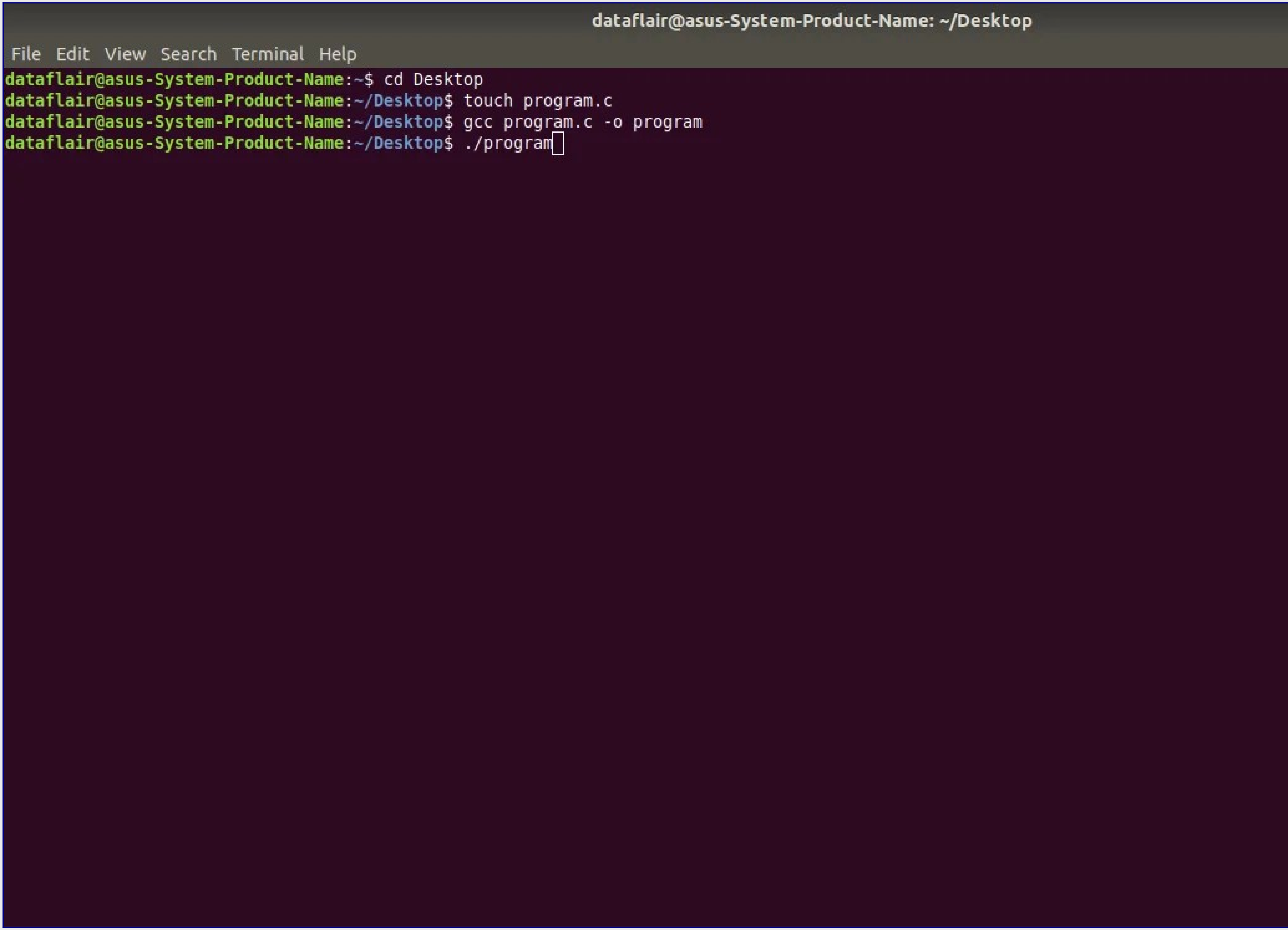
Compile code using GCC command

Here, the ‘gcc’ command is followed by the file name with .c extension and ‘-o’ and the name of the executable file, which is, ‘program’ in this case.

Key takeaway: The executable file name can be different from the source file name

In order to run the program, use the command

```
./program
```

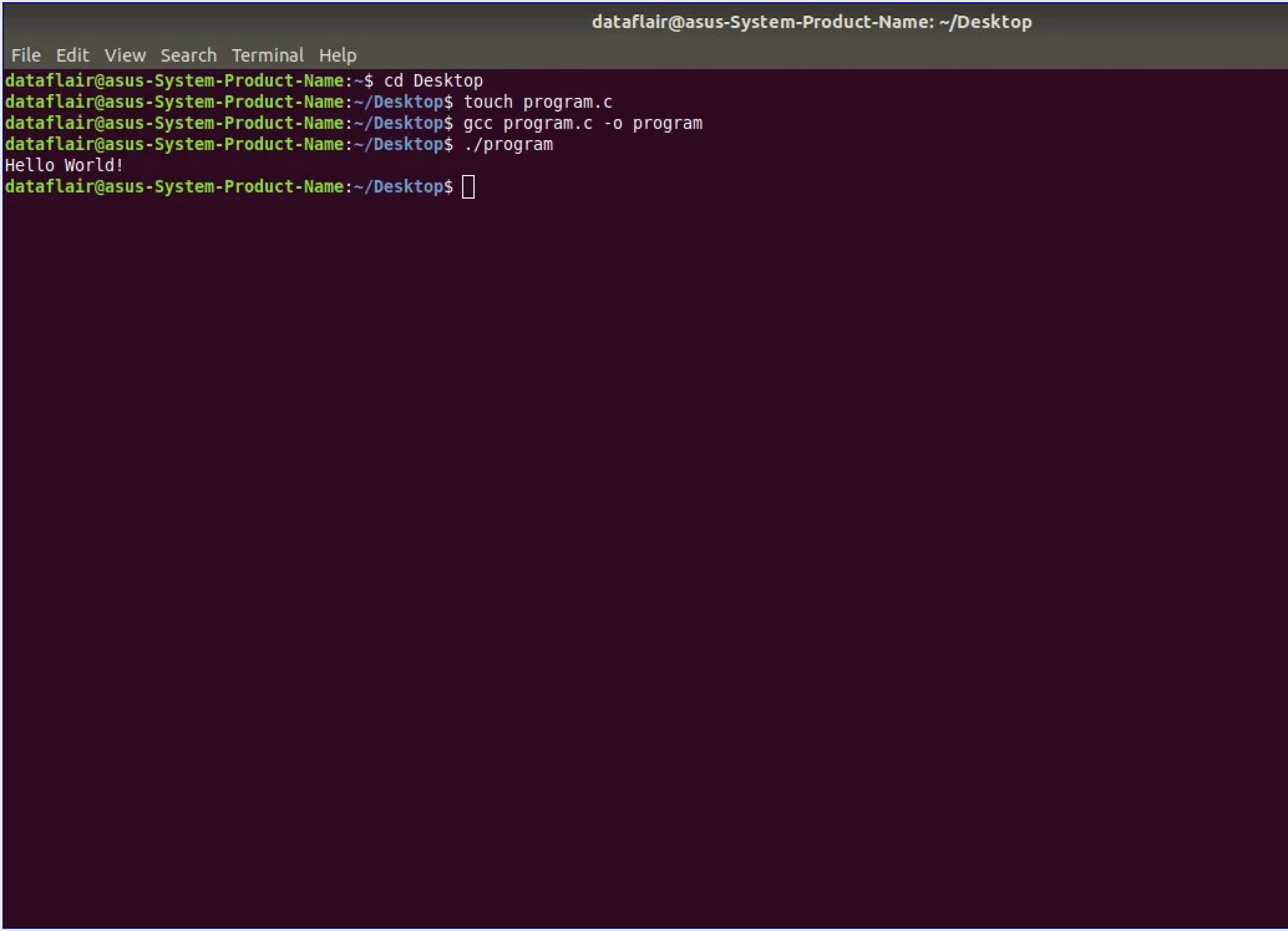

dataflair@asus-System-Product-Name: ~/Desktop
File Edit View Search Terminal Help

```
dataflair@asus-System-Product-Name:~$ cd Desktop
dataflair@asus-System-Product-Name:~/Desktop$ touch program.c
dataflair@asus-System-Product-Name:~/Desktop$ gcc program.c -o program
dataflair@asus-System-Product-Name:~/Desktop$ ./program
```

Run the code using ./program command

Here, the period symbol ‘.’ followed by a backslash is followed by the name of the executable file, that is, ‘program’ in this case

The program output would be:


dataflair@asus-System-Product-Name: ~/Desktop
File Edit View Search Terminal Help

```
dataflair@asus-System-Product-Name:~$ cd Desktop
dataflair@asus-System-Product-Name:~/Desktop$ touch program.c
dataflair@asus-System-Product-Name:~/Desktop$ gcc program.c -o program
dataflair@asus-System-Product-Name:~/Desktop$ ./program
Hello World!
dataflair@asus-System-Product-Name:~/Desktop$ 
```

Output of Program

Summary

In this way, you can install the GCC compiler using terminal when working on the LINUX OS. We inferred that it is pretty easy to install the GCC compiler in LINUX as it requires only one command. Thereafter, we saw how to check the version of the installed compiler. Then, we saw how to create a program in C with the help of certain commands. Finally, we learned how to compile and run C programs using terminal.

We hope you found this tutorial helpful for installing the GCC compiler using terminal for LINUX!

If you have any queries regarding the installation process or if you’re finding any difficulties to install C in your system feel free to leave a comment below.

Don’t forget to check Applications of C Programming That Will Make You Fall In Love With C