

```
Terminal - mariana@mariana-VirtualBox: ~/embedded2023_2
File Edit View Terminal Tabs Help
mariana@mariana-VirtualBox:~$ pwd
/home/mariana
mariana@mariana-VirtualBox:~$ mkdir embedded2023_2
mariana@mariana-VirtualBox:~$ cd embedded2023_2
mariana@mariana-VirtualBox:~/embedded2023_2$ pwd
/home/mariana/embedded2023_2
mariana@mariana-VirtualBox:~/embedded2023_2$ ls -l
total 0
mariana@mariana-VirtualBox:~/embedded2023_2$ cat > hello.c
```

#### Basic commands in the Linux terminal

- pwd: print working directory.
- mkdir: make a new directory.
- cd: change directory.
- ls: lists all files and directories in current directory.

#### About the cat command

Reads data from the file and gives its content as output. It helps create, view, and concatenate files.

- To view a single file:  
cat file\_name
- To view multiple files:  
cat file\_name1 file\_name2
- Create a file and add content:  
cat > newfile\_name

#### Creation of a c program using the linux terminal



```
mariana@mariana-VirtualBox:~/embedded2023_2$ cat hello.c
#include <stdio.h>
#include <stdlib.h>

int main(){
    printf("Hello");
    return 0;
}
```

To run the program:

```
mariana@mariana-VirtualBox:~/embedded2023_2$ ./hello.c
bash: ./hello.c: Permission denied
```

Permission is denied because it only has reading permissions:

```
mariana@mariana-VirtualBox:~/embedded2023_2$ ls -l
total 8
-rwxrwxr-x 1 mariana mariana 85 ago  7 16:44 hello.c
-rw-rw-r-- 1 mariana mariana  7 ago  7 16:46 hello.txt
```

To habilitate compiling, we use the chmod +x

```
mariana@mariana-VirtualBox:~/embedded2023_2$ ./hello.c
bash: ./hello.c: Permission denied
mariana@mariana-VirtualBox:~/embedded2023_2$ chmod +x hello.c
```

About chmod

In Unix operating systems, the chmod command is used to change the access mode of a file. The name is an abbreviation of **change mode**, which states that every file and directory has a set of permissions that control the permissions like who can read, write, or execute the file. The permissions have three categories: read, write, and execute simultaneously represented by 'r', 'w' and 'x'.

chmod [options] [mode] [File\_name]

“chmod” in Linux options taken from geeksforgeeks website

Options	Description
`-R`	Apply the permission change recursively to all the files and directories within the specified directory.
`-v`	It will display a message for each file that is processed. while indicating the permission change that was made.
`-c`	It works same as `-v` but in this case it only displays messages for files whose permission is changed.
`-f`	It helps in avoiding display of error messages.
`-h`	Change the permissions of symbolic links instead of the files they point to.

## 1. Symbolic mode

Most common method used for specifying permissions. It is a combination of letters and operators to set or tell what to do with permissions.

- '+': Add permissions.
- '-': Remove permissions.
- '=': Set the permissions to the specified values.

The following letters can be used in symbolic mode:

Letters	Definition
'r'	Read permission
'w'	Write permission
'x'	Execute permission

The following letters for references are used:

Reference	Class
u	Owner
g	Group
o	Others
a	All (owner,groups,others)

Examples:

Read, write and execute permissions to the file owner: `chmod u+rw [file_name]`

Remove write permission for the group and others: `chmod go-w [file_name]`

Read and write for Owner, and Read-only for the group and other: `chmod u+rw,go+r [file_name]`

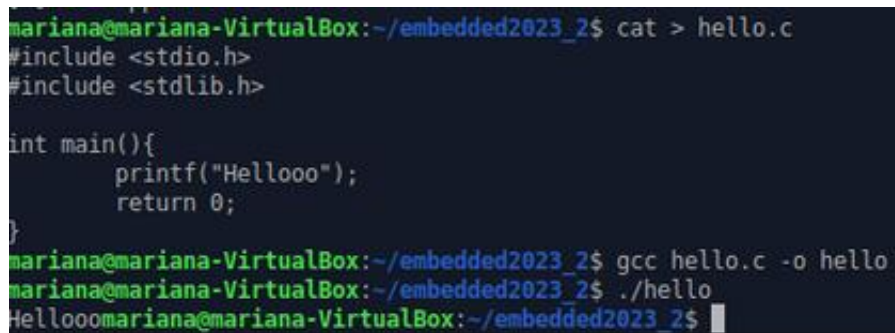
Viewing current permissions of a file or directory: `ls -l`

Now that we have access to run the file, we need to compile it first, this is done by:

`gcc file_name.c -o file_name`

Then, we run it:

`./file_name`



```
mariana@mariana-VirtualBox:~/embedded2023_2$ cat > hello.c
#include <stdio.h>
#include <stdlib.h>

int main(){
    printf("Hellooo");
    return 0;
}
mariana@mariana-VirtualBox:~/embedded2023_2$ gcc hello.c -o hello
mariana@mariana-VirtualBox:~/embedded2023_2$ ./hello
Hellooomariana@mariana-VirtualBox:~/embedded2023_2$
```