

# Command Line

As previously mentioned [here](#), you will be writing your program in C++ using **gedit**, a simple text editor for Linux. If you have not already opened gedit, do so now. Then for each program you write for the tutorial, follow these instructions:

- Copy the input provided, if any, and paste it into a new file in the editor.
- Save your file.
  - Go to File > Save...
  - If this is your first tutorial save, create a new folder somewhere within your home directory structure. Click the Create Folder button, and name it something exciting and descriptive, like Tutorial. If you choose, you may further organize your individual tutorial programs into their own directories—we leave it up to you.
  - Give your file a descriptive name with a file type of .cpp. For example, for your first program, you might name the file HelloWorld.cpp. Among other things, the .cpp will tell gedit that the content of this file is C++ code, and it will give you useful editing features accordingly.
- Add your code.
- Compile your code.
  - *Compiling* is converting it from the (basically) English version you typed into the language understood by the machine (which is all 0s and 1s).
  - The output of a successful compilation will be an *executable* file. This is what you will use to run, or *execute*, your program.
  - To compile, do the following:
    - Make sure your files are saved!
    - Bring up your terminal window.
    - Navigate to the directory where you saved your file using the commands we talked about [here](#) under **Linux Commands**.
    - Once you're in the same directory as your file, begin typing in the following:
    - `g++ <filename> -o <executableName>`
    - Replace <filename> with the name of the file you have been editing, and replace <executableName> with something similarly descriptive. For example, for your HelloWorld program, you might compile with the following:
    - `g++ HelloWorld.cpp -o HelloWorld`
    - *Something to note: editing a file in gedit creates a backup save file, which will be visible when you list the contents of a directory. It is named exactly the same as the edited file, except it will have a tilde (~) appended to it. You can safely ignore such files.*
  - Check the output in the terminal to see if the build was a success. If it was not, one or more errors will be printed to the terminal. See if you can understand and then fix the given errors. If you can't, please ask us! Compiler errors can be cryptic.
- Run your program.
  - Here's where the executable comes in! Still in your terminal, in the same directory as your program files, type in the following:
  - `./executableName`

- Replace executableName with the name you gave your executable when you compiled. Using the above example, to run your HelloWorld program you would enter ./HelloWorld at the command line.
- Hit Enter to run the command.
- Output from the execution will be printed to the terminal.
- Many things can go wrong when you are programming, so please ask for help *before* you get frustrated!

## Source