# Lesson 1: Getting started

This first lesson will make sure that both you and your computer are prepared correctly to handle all the programming tasks you'll be completing as you progress through this course!

Let's get started. You'll be needing a text editor. If you're looking for something with a nice and beautiful theme, we recommend [Sublime Text 3](https://www.sublimetext.com/3), if you want something a little more plain you can simply go for Notepad++(https://notepad-plus-plus.org/download/v6.8.8.html).

The next step is to install a C \*\*compiler\*\* onto your computer. "What is a compiler?", you ask... Well, as mentioned earlier, C is a low-level programming language meaning that the commands in C are closely related to the instructions that your \*\*computer processing unit\*\* would run. \*\*HOWEVER\*\*, this still means that the language is undirectly unreadable by the CPU, as such, you need to first compile your code such that it is translated into a form that your CPU can run.

Download the compiler [here](http://www.mingw.org/), you can see the button for "Download Installer" at the top right corner of the page.

Open your note editor and create a new file, save it with the C file extension, for example: "lesson1.c". We've decided to adhere to traditions so your first program will be a "Hello, world!" program, first this first lesson, just type out everything we tell you to, as the ultimate aim of this section is to teach you how to compile and run your code.

```c

#include <stdio.h>

int main(void)

{

printf("Hello, world");

}

```

That's it! A fully working program, but how do we run it? We can't simply just double-click on the file and hope it would do something useful... In other worlds, we must first compile it.

1. First, open your command-line or terminal. To do this on Windows, press Windows + R, then type cmd into the box and hit 'OK'. Meet the command-line interface, this scary black window, you'll become good friends with it on your programming journeys.
   1. Navigate to the directory on your computer in which you saved the file.
   2. For example, if I am currently in "C:/users/Documents" and I want to go into "C:/users/Documents/code", I would type "cd code" to move into the "code" folder.
   3. If you wish to move up a folder, on Windows you can type "cd..", on an Mac or Linux type "cd .." (make sure to include that space).
2. Next, we shall compile the code.
   1. Type "gcc" followed by the name of your file, then followed by the name you wish the compiled code to be.
   2. For example "gcc lesson1.c output".
   3. The compile file is now called output, and to run it you can type "output" and see the results of your program.

If you found this all simple enough, the real programming begins next lesson, as well as an introduction to what everything means, that program we just wrote together will all make sense very soon!

### Extra task

Now that you know how to compile your code, spend a few minutes exploring your computer through the command-line, move around! Pick a distant folder and try to move into it from where you currently are! Try opening other files through the command-line… next time you want to listen to a music file, navigate there by command-line!