

# Exercise 0: Installing and writing the first program

By: Rahul Sharma

10 April, 2018

The first thing to do before jumping into the python programming is to get the computer ready for the programming environment. Well, this might sound unnecessary because we already have python installed on our systems but this is also as important as learning the programming itself. There are three things required by you to make your system ready to run the python programs.

1. The python compiler
2. A text editor
3. A terminal

The python compiler that we have been using comes as a part of the Anaconda 2 software package. We are working on Python version 2.7 here. Remember, the version of the language is very important. I'll skip the part of the installation of python as it is pretty straight forward ([www.python.org](http://www.python.org) or [www.anaconda.com](http://www.anaconda.com) for Anaconda version 2).

The text editor is the software that you will be using to write your python programs in and debug it. Some of the good examples of text editors are: Visual Studio Code (auto code completion) and Sublime Text 3. We will not be using IDE (Interactive Development Environments) like Visual Studio Community Edition or PyCharm for these classes. I want to keep it as simple as possible. You can download the editors and suit yourself with whichever you require. I will be using Sublime Text 3, but Visual Studio Code is the latest and better text editor.

The terminal (or command prompt in Windows) is where you will be calling the python compiler to compile and run your code. Which means, the terminal is responsible to show the outputs or carry out the operations that you have written in your python code using the text editor. The typical syntax for running a python code is as follows:

```
$ python filename.py
```

Don't worry about the '\$' sign for now. Python files end with a '.py' extension. Instead of running the code from file you can write and run the code in the command line itself. Writing the 'python' command in the command prompt will initialise the python 'interpreter' where you can write and run the code 'line by line' since python is an interpreted language and not a procedural one like we have been familiar with the C programming where we cannot run the C code line by line as the entry point is always the `main()` function from where it follows a procedure to carry out different operations.

One of the main advantages of interpreted languages like Python is that here the things get done really quickly because there is less effort in terms of time and lines of code needed to carry out the same functionality when using a procedural programming instead. However, this has some disadvantages too: the biggest of which is that, python is relatively slower than the procedural languages.

With this, we will move on to the first example and exercise:

1. Create a folder and name it PythonProgramming on your system, preferably NOT in C: drive
2. Create a file `ex1.py` and open it with your favourite text editor.
3. Type in the following code and run the file in the command prompt:

```
print "Hello World!"
print "Hello Again"
print "I like typing this."
```

```
print "This is fun."  
print 'Yay! Printing.'  
print "I'd much rather you 'not'."  
print 'I "said" do not touch this.'
```

4. Try to put comments in the code wherever possible. A '#' (octothrope, hash, pound, mesh) character in front of any statement in Python makes the statement as a comment which does not get printed in the final run of the program.

*Next lecture: Numbers and Math*