2017 Fall : COMP-SCI 5590 - Special Topics

Python/Deep Learning Programming

Lesson 1

**Course Plan**

The first tutorial will focus on installation and making one familiar with python programming concepts. This course is designed keeping in mind for students with little programming background. The course is going to cover all major concepts of Python Programming keeping in mind data scientists.

1. **Installation**

1.1Installing python

First we need to install Python on machine. For different Operating systems we need to install python in different way.

Windows:

<https://www.python.org/downloads/>

Download the required version. Then follow the instructions and click continue. Python will be installed.

UBUNTU Link:

1. First check if you have python version 3 or not. Open command terminal, type

python3

Do you see the acknowledgment that you do? done.

1. If you don't; install using following command line.

sudo apt-get install python3

We can install any version of Python 2.7 or python 3.6 as both are equally popular in outside world. We will still prefer Python 3.6 as it is the latest one and if we require modules which only support 2.7 version, we will switch. Through IDE it is very easy to do it.

1.2 Installing Pycharm Community Edition

As part of this we will install the IDE ((Integrated development Environment) for better programming experience.

Windows link:

<https://www.jetbrains.com/pycharm/download/#section=windows>

UNIX Link:

<https://www.jetbrains.com/pycharm/download/#section=linux>

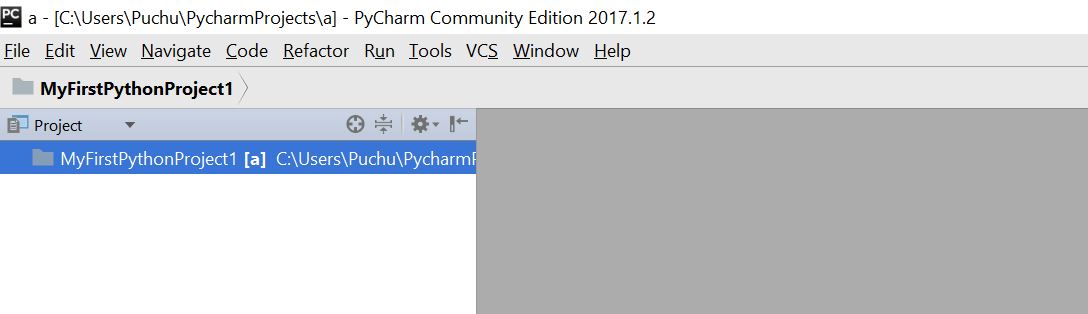
Find more on tutorials on Pycharm at

<https://confluence.jetbrains.com/display/PYH/PyCharm+Tutorials>

1. **Simple hello Program**

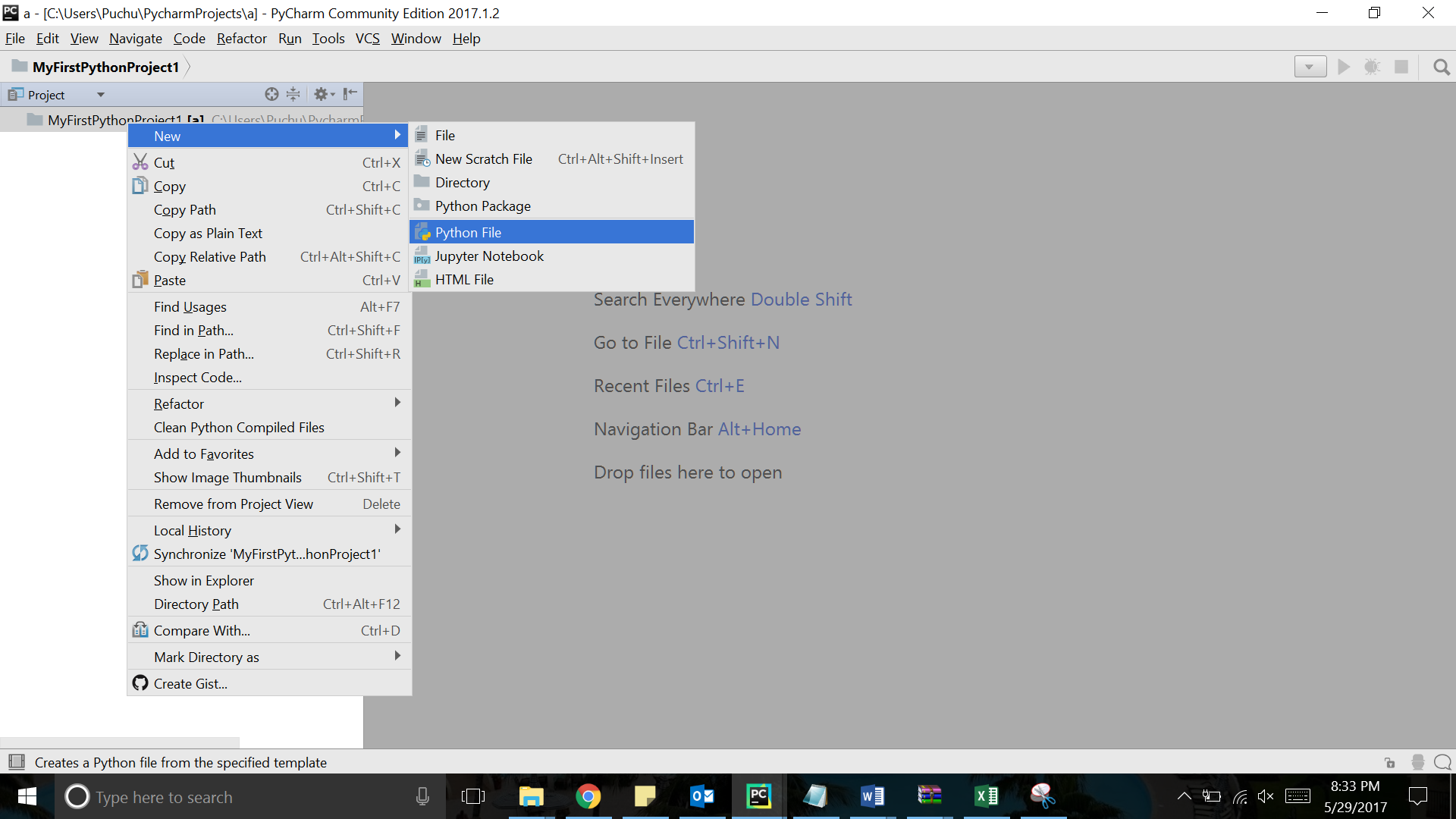
For this we need to first in PyCharm

File - > New Project -> Give your Project name as per you



Right click on MyFirstPythonProject1 in Project explorer.

Select New -> Python File -> give your file name as hello.py

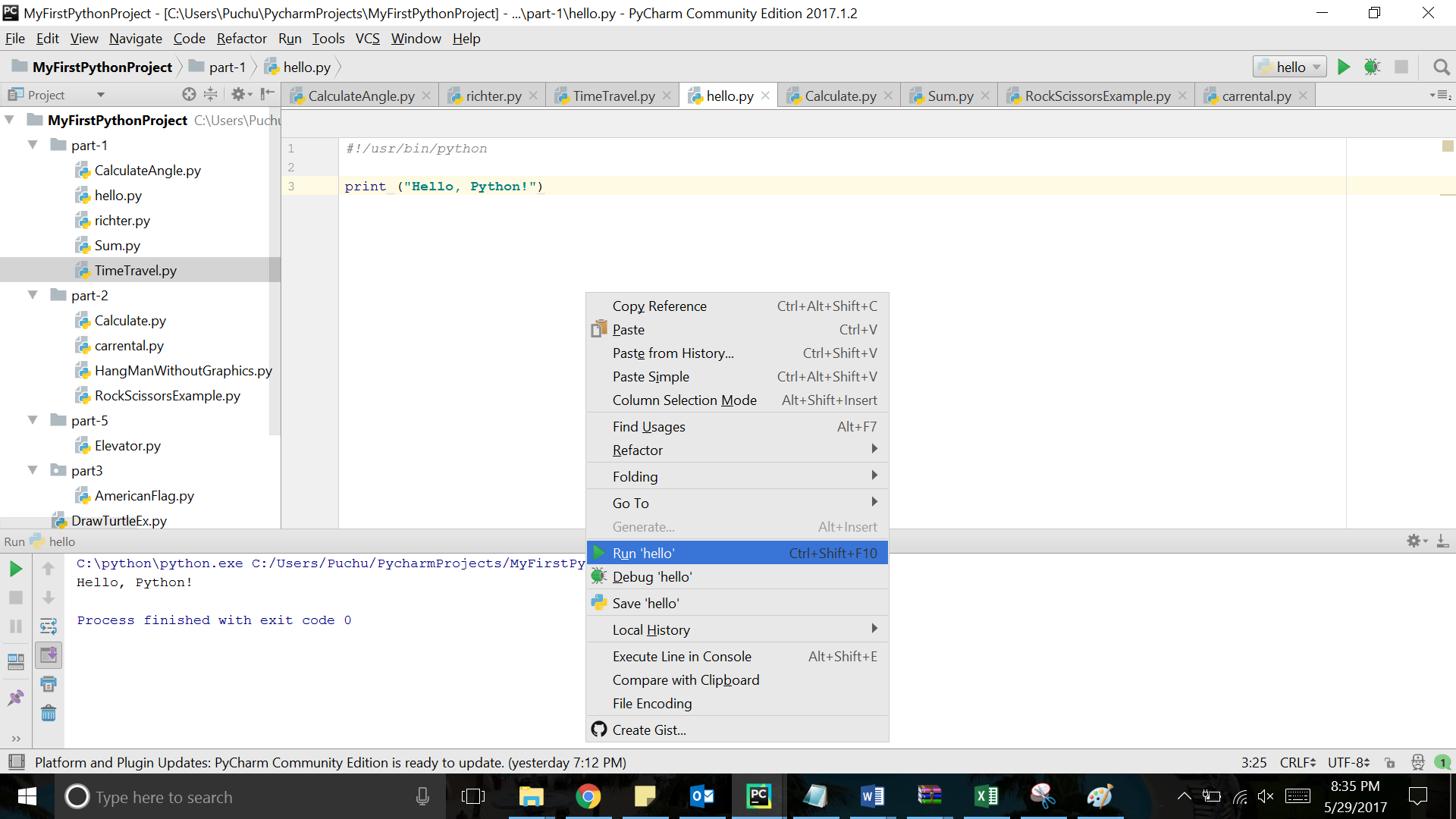


Rewrite the following code in that. It will simply print hello Python on Console Screen of Pycharm IDE.

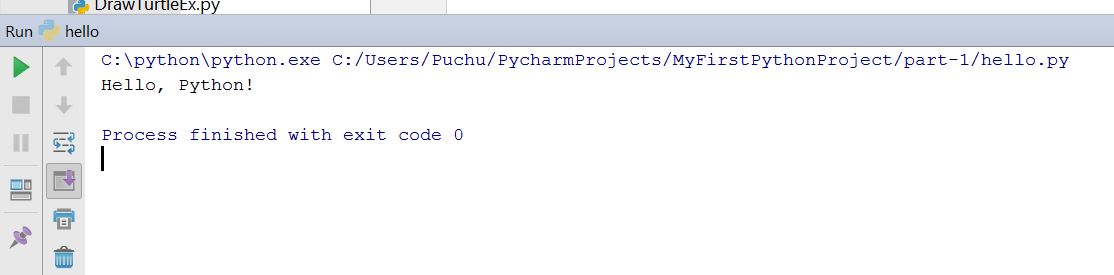
Sample Program:



In order to execute the above program click on Run on right click on the needed python file. Please remember that all python modules name end with .py extension.



Output:



Upon this step you should be able to create any python file and should know the execution of python. Python is interpreted language and hence there is no byte code generation unlike other languages like Java, C etc.

1. **Declaring Variables**

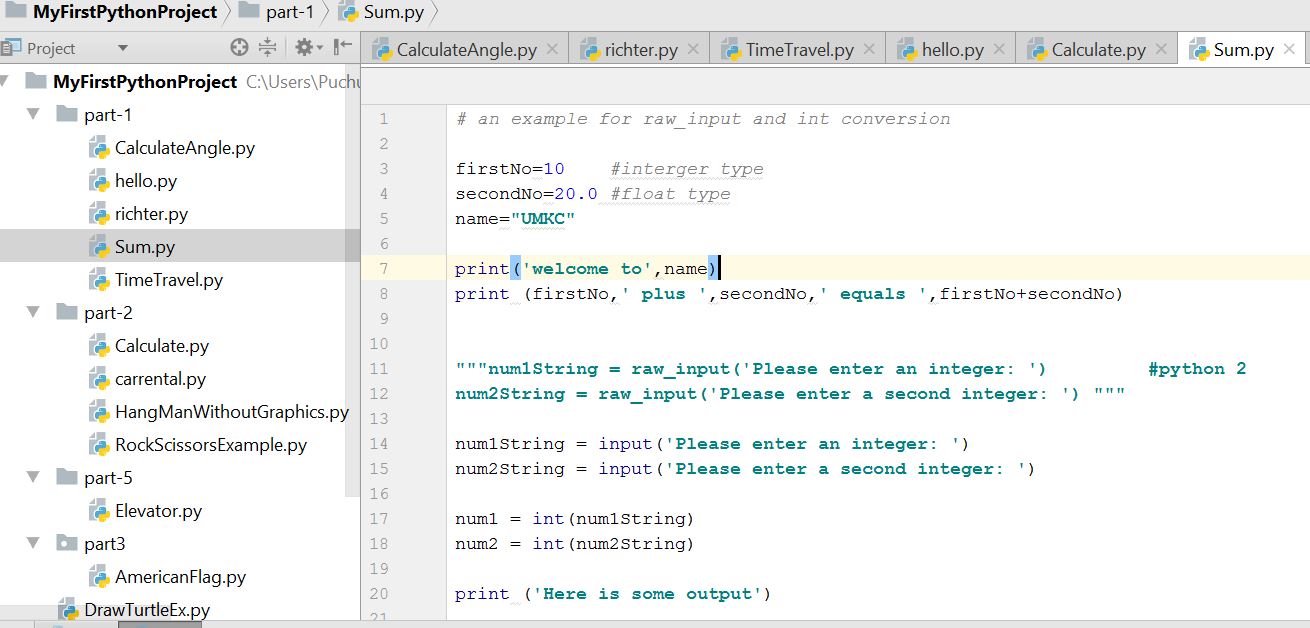
Python supports various types of variables. Following are the main categories:

* Numbers : int, long, float, complex
* String : str
* List : collection of elements with varied size and items .Given in square brackets
* Tuple : read only list i.e. collection of elements which cannot be updated
* Dictionary :key value pairs

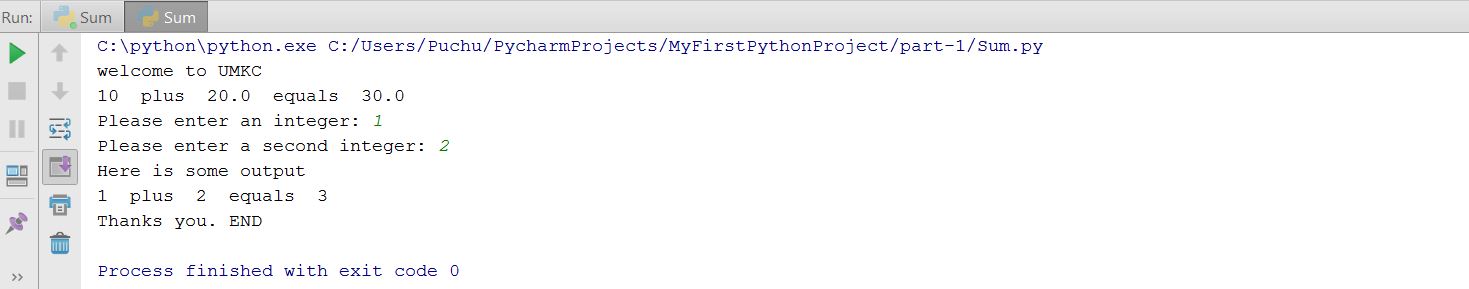
We can convert the basic primitive types to another one using various in built functions. Many times when we take input from user, It is considered to be String type and we need to convert it to another type.

Sample Program:

Take two integer input from the user. Convert it to integer and then sum it. Print the sum



Output:



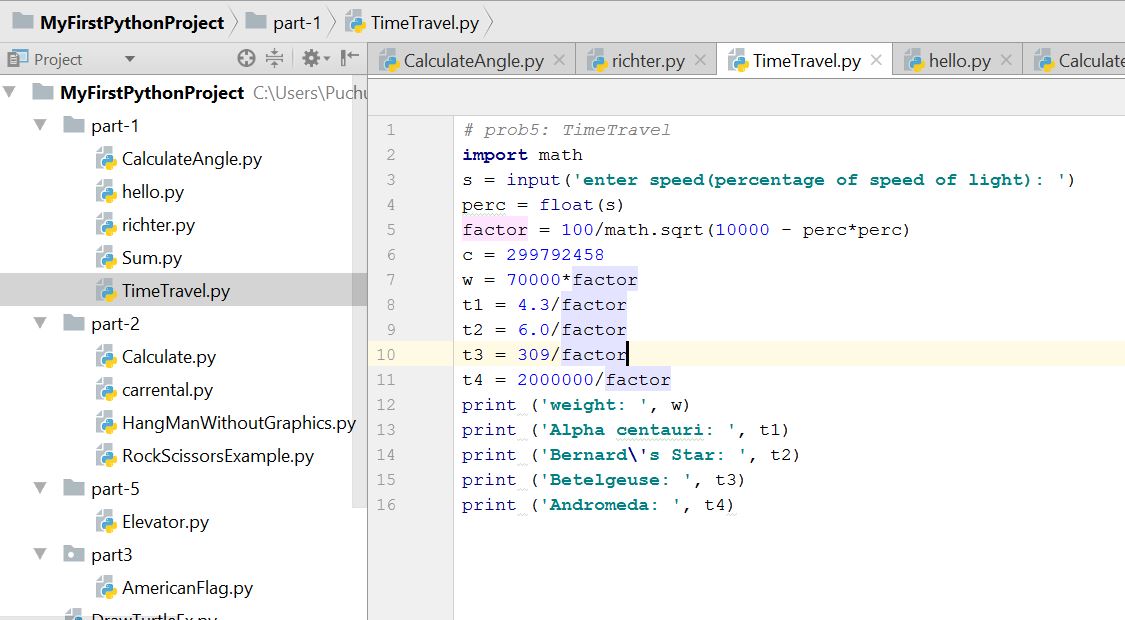
1. **Operations**

We will learn the various operations we can perform on variables. These are the broad categories:

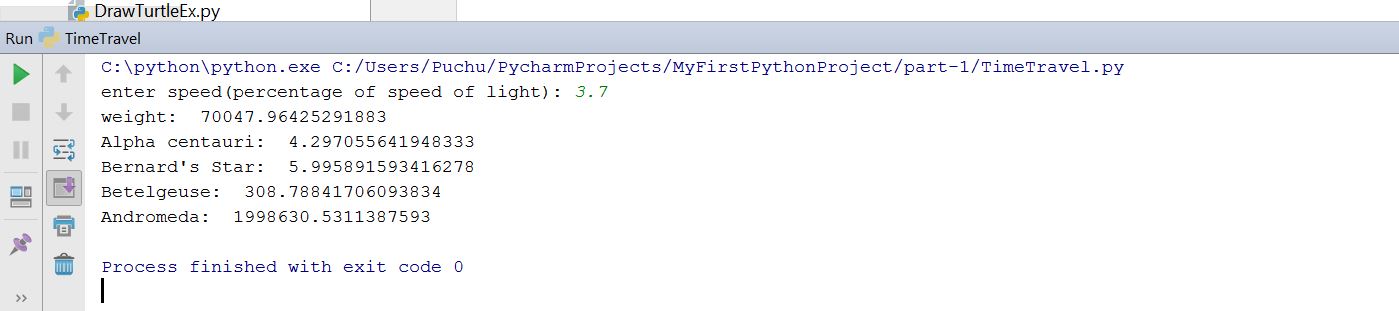
* Arithmetic Operators : +,-
* Comparison (Relational) Operators
* Assignment Operators
* Logical Operators
* Bitwise Operators
* Membership Operators
* Identity Operators

Sample Program:

Input some numbers and do some simple arithmetic operations based upon formulae to calculate distance between stars



Output:



1. **Import**

Use classes & functions defined in another file to get additional functionality

Three formats of the command:

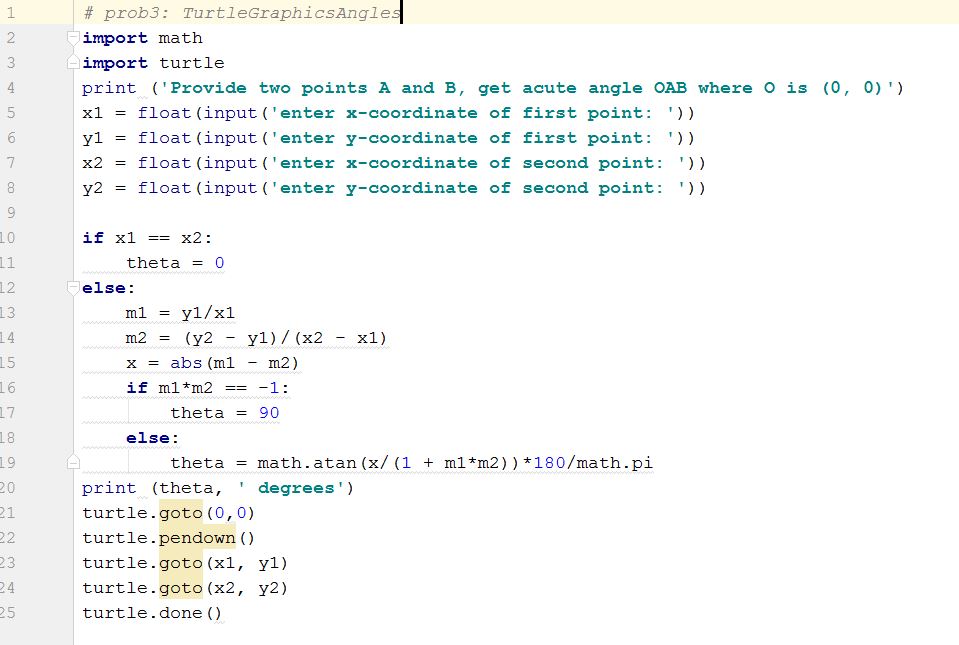
*import somefile*

*from somefile import \**

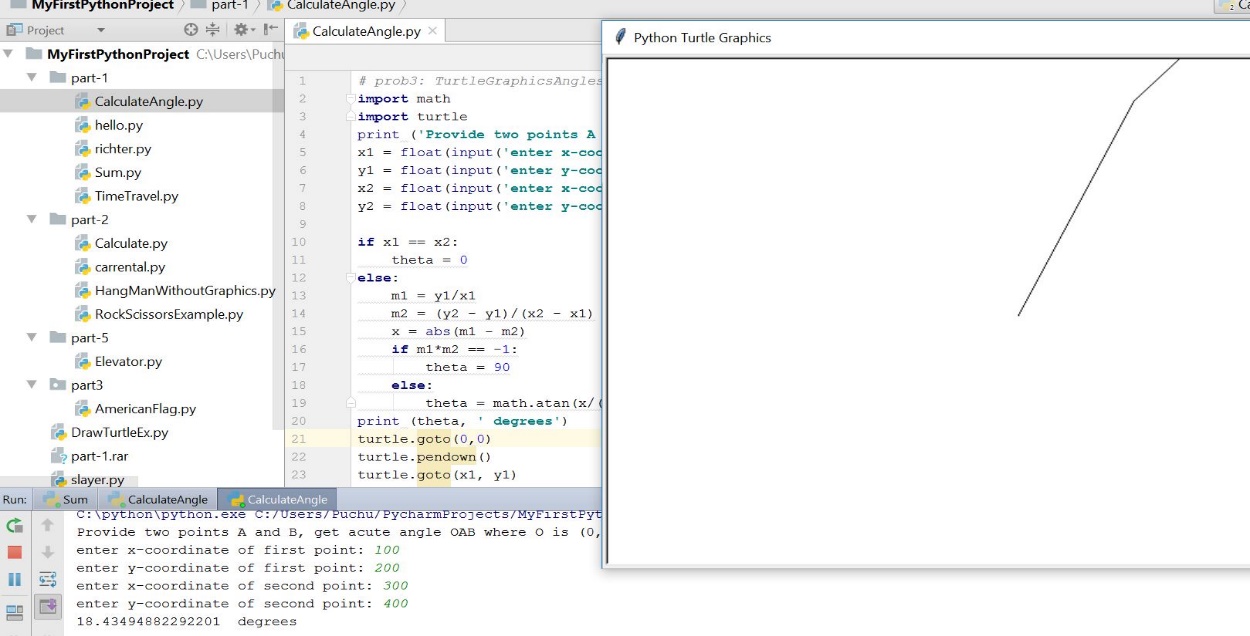
*from somefile import className*

**Sample Program:**

Import Turtle graphics library which is used for basic canvas drawing and then take coordinates from user as (x1,y1) and (x2,y2). Calculate angle between them and display them on canvas.

****

**Output:**

****

1. **Control Statements (Basic Overview)**

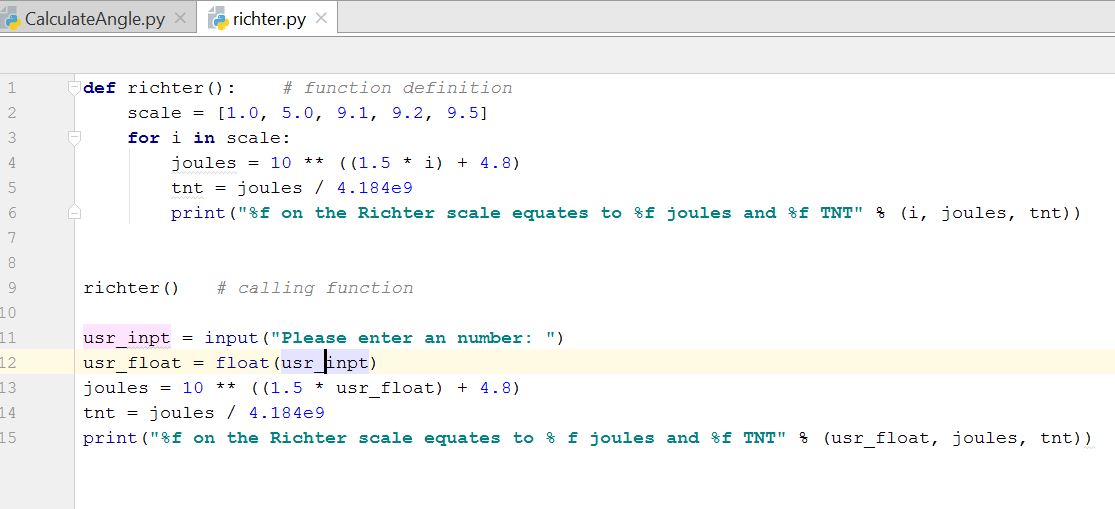
As part of this section we will cover the following conditional and loop statements:

* If statement
* [if...else statements](https://www.tutorialspoint.com/python/python_if_else.htm)
* While loop
* For loop

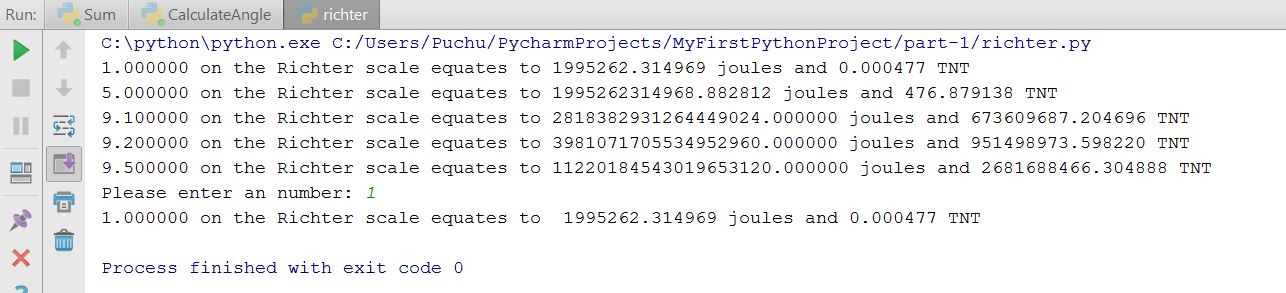
It will be covered in more detail in next session. We will learn the basic concepts here.

**Sample Program:**

For every single value on Richter scale calculate the energy in terms of joules and TNT required. We require for loop for this as the code will become more verbose.

****

**Output:**



**In-Class Exercise**

1. State differences between Python 2 and Python 3 version.
2. Write a python program to
3. Take radius of a circle as input from user and calculate area , circumference and print it.
4. Check whether a number input by the user is even or odd
5. Write a python program to

This is a number guess exercise. First pick a random digit via program i.e 0,1,2,3,4,5,6,7,8,9

Ask the user to guess the digit randomly picked by your program. Then print whether the number guess by the user is perfect or below the random number or above the random number. Also your program should explain the rules of this number guess game to the user.

\*\*\* *use random module. You need to use import statement here*.

Suppose the digit generated by program is 8

*Sample input:*

Guess the digit: 7

*Sample output:*

Your answer is low than required

*Sample input:*

Guess the digit: 8

*Sample output:*

Your answer is PERFECT!! Congratulations!!

*Sample input:*

Guess the digit: 9

*Sample output:*

Your answer is high than required

**Submission Guidelines:**

* Submit your code at Github and properly document it. Submit your screenshots as well.
* Properly document your code
* Use google form link submission to submit your in class assignment: