# Python Object

Python is an object oriented programming language. So its main focus is on objects unlike procedure oriented programming languages which mainly focuses on functions.

In object oriented programming language, object is simply a collection of data (variables) and methods (functions) that act on those data.

## Python Class

A class is a blueprint for the object. Let's understand it by an example:

Suppose a class is a prototype of a building. A building contains all the details about the floor, doors, windows, etc. we can make another buildings (as many as we want) based on these details. So building is a class and we can create many objects from a class.

An object is also called an instance of a class and the process of creating this object is known as instantiation.

Python classes contain all the standard features of Object Oriented Programming. A python class is a mixture of class mechanism of C++ and Modula-3.

## Define a class in Python

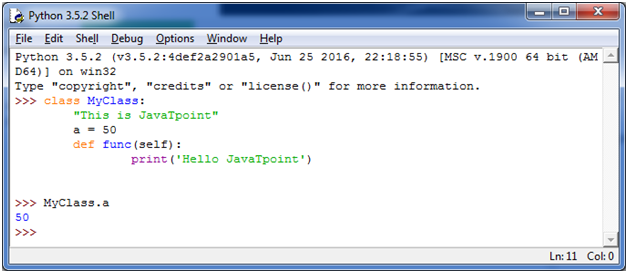
In Python, a class is defined by using a keyword class like a function definition begins with the keyword def.

**Syntax of a class definition:**

1. class ClassName:
2. <statement-1>
3. .
4. .
5. .
6. <statement-N>

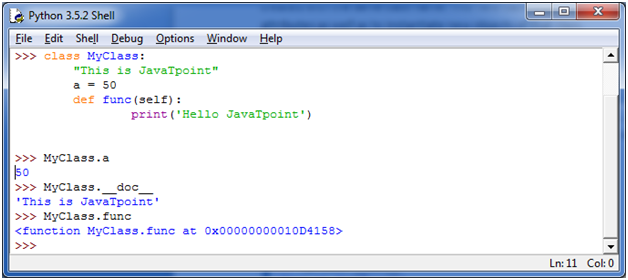
A class creates a new local namespace to define its all attribute. These attributes may be data or functions.

**See this example:**



There are also some special attributes that begins with double underscore (\_\_). For example: \_\_doc\_\_ attribute. It is used to fetch the docstring of that class. When we define a class, a new class object is created with the same class name. This new class object provides a facility to access the different attributes as well as to instantiate new objects of that class.

**See this example:**

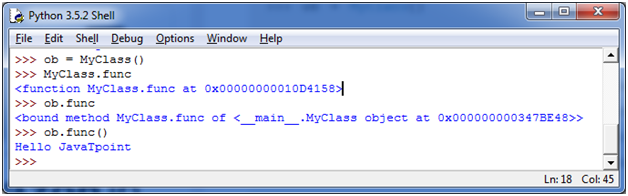


## Create an Object in Python

We can create new object instances of the classes. The procedure to create an object is similar to a function call.

Let's take an example to create a new instance object "ob". We can access attributes of objects by using the object name prefix.

**See this example:**



Here, attributes may be data or method. Method of an object is corresponding functions of that class. For example: MyClass.func is a function object and ob.func is a method object.

## Python Object Class Example

1. class Student:
2. def \_\_init\_\_(self, rollno, name):
3. self.rollno = rollno
4. self.name = name
5. def displayStudent(self):
6. print "rollno : ", self.rollno,  ", name: ", self.name
7. emp1 = Student(121, "Ajeet")
8. emp2 = Student(122, "Sonoo")
9. emp1.displayStudent()
10. emp2.displayStudent()

**Output:**

1. rollno :  121 , name:  Ajeet
2. rollno :  122 , name:  Sonoo