***Assignment # 6***

***Q 1: Define Object Oriented Programming Language?***

OOP is a programing language which uses “Class” and “Objects” to make a program. Class define the attributes and function of objects and class is implemented by the use of Object.

Objects contains data, in the form of fields (attributes or properties) and code, in the form of procedures (methods).

Class is a template of object. An object may have multiple attribute and functions so by making its Class we can make it reusable. It may use multiple times when it needs. We just need to instantiate it and pass the Class its attributes.

Object is instantiate of class. We introduce the Class to the program by object and collect the data the data stored in it.

***Q 2: List down the Benefits of OOP?***

* OOP provides a clear modular structure for programs.
* Implementation details are hidden from other modules.
* It is easy to maintain and modify existing code.
* New objects can be created with small differences to existing ones.
* objects, methods, instance, message passing, inheritance are some important properties provided by these particular languages
* It implements real life scenario.
* In OOP, programmer not only defines data types but also deals with operations applied for data structures.

***Q 3: Differentiate between function and method?***

A function is a code that is called by name. It can be passed data to operate on and can optionally return.

A method is a code that is called by a name that is associated with an object. In most respects it is identical to a function except for two key differences:

* A method is implicitly passed the object on which it was called.
* A method is able to operate on data that is contained within the class (remembering that an object is an instance of a class - the class is the definition, the object is an instance of that data).

***Q 4: Define the following terms:***

1. **Class:**

A class is an entity that determines how an object will behave and what the object will contain.

In other words:

it is a blueprint that defines the nature of a future object for creating new objects, providing initial values(attributes), and implementations of behavior of object.

1. **Object:**

An object is a self-contained component which consists of methods to make a particular type of data useful. Object determines the behavior of the class. When you send a message to an object, you are asking the object to invoke or execute one of its methods.

1. **Attribute** A class attribute is a Python variable that belongs to a class rather than a particular object. It is shared between all the objects of this class and it is defined outside the constructor function, \_\_init\_\_(self,...), of the class
2. **BEHAVIOR :** Objects in Python are generally classified according to their behaviors and the features that they implement. For example, all of the sequence types such as strings, lists, and tuples are grouped together merely because they all happen to support a common set of sequence operations such as *s*[*n*], len(*s*), etc. All basic interpreter operations are implemented through special object methods. The names of special methods are always preceded and followed by double underscores (\_\_). These methods are automatically triggered by the interpreter as a program executes.