# **Assignment # 06:**

Question 1:

**OBJECT ORIENTED PROGRAMMING:** Object-oriented programming (OOP) refers to a type of computer programming (software design) in which programmers define the data type of a data structure, and also the types of operations (functions) that can be applied to the data structure.

Question 2:

**List down the Benefits of OOP?**

1. Modularity for easier troubleshooting
2. Reuse of code through inheritance
3. Flexibility through polymorphism
4. Effective problem solving

Question 3:

**Differentiate between function and method?**

**FUNCTION:** A function is a piece of code that is called by name. It can be passed data to operate on (i.e. the parameters) and can optionally return data (the return value). All data that is passed to a function is explicitly passed.

**METHOD:** A method is a piece of 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:

1. A method is implicitly passed the object on which it was called.
2. 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).

Question 4:

**CLASS:** In object-oriented programming, a class is a blueprint for creating objects (a particular data structure), providing initial values for state (member variables or attributes), and implementations of behavior (member functions or methods).

**OBJECT:** In object-oriented programming (OOP), objects are the things you think about first in designing a program and they are also the units of code that are eventually derived from the process. In between, each object is made into a generic class of object and even more generic classes are defined so that objects can share models and reuse the class definitions in their code. Each object is an instance of a particular class or subclass with the class's own methods or procedures and data variables. An object is what actually runs in the computer.

**ATTRIBUTE:** In object-oriented programming (OOP), objects are the things you think about first in designing a program and they are also the units of code that are eventually derived from the process. ... Each object is an instance of a particular class or subclass with the class's own methods or procedures and data variables.

**BEHAVIOR:** A class's behavior determines how an instance of that class operates; for example, how it will "react" if asked to do something by another class or object or if its internal state changes. Behavior is the only way objects can do anything to themselves or have anything done to them.

Question 5:

Write a code in python in which create a class named it Car which have 5 attributes such like (model, color and name etc.) and 3 methods. And create 5 object instance from that class.

**class** **Car**():

**def** \_\_init\_\_(self,model,color,name,warranty,price):

self.model = model

self.color = color

self.name = name

self.warranty = warranty

self.price = price

car1 = Car(2016,"Grey","Acura RDX","2 year",9126000)

car2 = Car(2019,"Black","Audi A7","3 year",34750000)

car3 = Car(2018,"Black","Audi A6","3 year",24750000)

car4 = Car(2015,"Red","Audi A3","2 year",6000000)

car5 = Car(2017,"Light Grey","Audi A5","2.5 year",10000000)

print("Model : ",car1.model)

print("Color : ",car1.color)

print("Name : ",car1.name)

print("Warranty : ",car1.warranty)

print("Price : ",car1.price,"PKR")

print("**\n**")

print("Model : ",car2.model)

print("Color : ",car2.color)

print("Name : ",car2.name)

print("Warranty : ",car2.warranty)

print("Price : ",car2.price,"PKR")

print("**\n**")

print("Model : ",car3.model)

print("Color : ",car3.color)

print("Name : ",car3.name)

print("Warranty : ",car3.warranty)

print("Price : ",car3.price,"PKR")

print("**\n**")

print("Model : ",car4.model)

print("Color : ",car4.color)

print("Name : ",car4.name)

print("Warranty : ",car4.warranty)

print("Price : ",car4.price,"PKR")

print("**\n**")

print("Model : ",car5.model)

print("Color : ",car5.color)

print("Name : ",car5.name)

print("Warranty : ",car5.warranty)

print("Price : ",car5.price,"PKR")

Model : 2016

Color : Grey

Name : Acura RDX

Warranty : 2 year

Price : 9126000 PKR

Model : 2019

Color : Black

Name : Audi A7

Warranty : 3 year

Price : 34750000 PKR

Model : 2018

Color : Black

Name : Audi A6

Warranty : 3 year

Price : 24750000 PKR

Model : 2015

Color : Red

Name : Audi A3

Warranty : 2 year

Price : 6000000 PKR

Model : 2017

Color : Light Grey

Name : Audi A5

Warranty : 2.5 year

Price : 10000000 PKR