



Name: Muhammad Ammar

Std.Id: 16602

BSSE- 3rd Semester

Department of Computer Science
Semester Assignment Spring 2020
Object Oriented Programming

Q#1: What is Class and role of object in a Class, explain in detail with the help of a suitable program?

Ans:

Class

- It is the building block of object oriented programming.
- It is a user defined prototype.
- It is a blue print for creating different objects.
- It Contains variables and methods.
- It represents the set of properties or methods that are common to all objects of one type.

In general, class declarations can include these components, in order:

1. **Modifiers:** A class can be public or has default access.
2. **Class name:** The name should begin with an initial letter (capitalized by convention).
3. **Superclass (if any):** The name of the class's parent (superclass), if any, preceded by the keyword extends. A class can only extend (subclass) one parent.
4. **Interfaces (if any):** A comma-separated list of interfaces implemented by the class, if any, preceded by the keyword implements. A class can implement more than one interface.
5. **Body:** The class body surrounded by braces, { }

Syntax:

```
Class <class_name>{  
    field;  
    method; }  
}
```

Object

- An object is an instance of a class.
- A product devised from the blueprint.
- All of the objects created from a class shares the attributes and behaviors of that class.
- A single class can have multiple objects.

An object consists of:

- **State:** It is represented by attributes of an object. It also reflects the properties of an object.

- **Behavior:** It is represented by methods of an object. It also reflects the response of an object with other objects.
- **Identity:** It gives a unique name to an object and enables one object to interact with other objects.

Real Life Examples:

<i>Class Name</i>	<i>Object</i>
Airplanes	F16
Snakes	Cobra
Cats	Lion
Cars	Honda

Simple Program to understand Class & Object

```
class Student { //Defining a Student class.
int id=16602; //field or data member or instance variable
String name="Ammar";
public static void main (String args[]) { //creating main method inside the Student class
Student s1=new Student(); //creating an object of Student
System.out.println(s1.id); //accessing member through reference variable
System.out.println(s1.name);
}
}
```

Output

16602
Ammar

Q#2: Write a program about table printing which takes input from the user on the basis of OOP and explain in detail.

Ans:

Program

```
1. import java.util.Scanner;
2. public class Multiplication_Table
3. {
4.     public static void main(String[] args)
5.     {
6.         Scanner s = new Scanner(System.in);
7.         System.out.print("Enter a number for you want a table:");
8.         int x=s.nextInt();
9.         for(int i=1; i <= 10; i++)
10.        {
11.            System.out.println(x+" * "+i+" = "+x*i);
12.        }
13.    }}
```

Output of Program

Enter a number for you want a table:7

7 * 1 = 7

7 * 2 = 14

7 * 3 = 21

7 * 4 = 28

7 * 5 = 35

7 * 6 = 42

7 * 7 = 49

7 * 8 = 56

7 * 9 = 63

7 * 10 = 70

Q#3: Write a program about any 2 cars which can calculate the performance of both of them and explain in detail.

Ans:

Program

```
public abstract class Car
{
public abstract void start();
    public void stop(){
System.out.println("Stopping Car in abstract class");  }}
class FourWheeler extends Car{  @Override
public void start()
{      System.out.println("Starting Four Wheeler");          }      }
class SixWheeler extends Car{  @Override  public void start()
{      System.out.println("Starting Six Wheeler");  }}
```

Main // main function

```
package oopsconcept; public class VehicleTesting
{  public static void main(String[] args)
{      Car myFourWheeler = new FourWheeler();
        Car mySixWheeler = new mySixWheeler();
        myFourWheeler.start();
myFourWheeler.stop();
mySixWheeler.start();
        mySixWheeler.stop();  }}
```

Output of Program

Stopping Car in abstract class
Starting Four Wheeler
Starting Six Wheeler