

Name : Hamza Iqbal

ID : 14784

Program : BS (SE)

subject : Object Oriented
Programming

Section : B

Question No 1:

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

Answer:-

Before you create object in Java, you need to define class. A class is a blueprint for the object.

We can think of the class as a sketch (prototype) of a house.

It contains all the details about the floor, door, windows etc. Based on these description we build the house. House is an object.

Since many houses can be made from the same description we can create many object from a class.

For example :-

```
→ class   className {  
    // variables  
    // methods  
}
```

For example :-

```
class Lamp {  
    " instance variable private  
    Boolean ison;  
  
    " method public void  
    turnon () {  
        ison = true ;  
    }  
  
    " method  
    public void turn off () {  
        ison = false ;  
    }  
}
```

Here we have created class named Lamp.

The class has on variable named (ison) and the two method turnon () & turn off (). These variable and methods defined within class are called members of class.

In the example public & private know as the modifiers.

Object in class :-

An object are called an instance of a class.

For example, suppose [Animal] is a class then cat, Dog, Horse, and so on can be considered as object of [Animal] class.

```
className object = new classname();
```

Here we must using the constructor `className()` to create the object. Construction have the same name as the class and are similar to methods.

For example :-

// L1 object

```
Lamp L1 = new Lamp();
```

// L2 object

```
Lamp L2 = new Lamp();
```

We have created object named L1 and L2 using the constructor of Lamp class (Lamp()).

Object are used to access members of a class.

For example:

```
class Lamp
```

```
    void turn on () {  
        ison = true ;  
    }  
}
```

```
class classobject Example {  
    public static void main {  
        (string [] args) {  
            L1 turn on ();  
}
```

```
    }  
}
```

```
    void turn off () {
```

```
        " initialize variable with value.
```

```
        ison = false ;
```

```
        System.out.println ("Light on") + }
```

```
    }
```

```
class main {
```

```
    public static void main (string [] arg).
```

```
        " create object L1 & L2
```

```
        Lamp L1 = new Lamp ();
```

```
        Lamp L2 = new Lamp ();
```

```
        " call method turn on () & turn off ().
```

```
        L1 . turn on ();
```

```
        L2 . turn off ();
```

```
    }
```

```
}
```

Question No 2

Write a program about table printing
----- explain in detail.

Answer:-

```
import java.util.Scanner;
public class work {
    public static void main (String []arg) {
        Scanner in = new Scanner (System.in);
        int num1 = in.nextInt();
        for (int i = in.nextInt();
            System.out.println (num1 + "x" +
                (i+1) + " = " +
                (num1 * (i+1)));
        }
    }
}
```

Basics of OOP:-

Object-oriented programming has four basic concepts: Encapsulation, Abstraction, inheritance and polymorphism. Even if these concepts seem incredibly complex, understanding their general framework of how they work will help you understand the basics of computer systems.

1. Encapsulation
2. Abstraction
3. Inheritance
4. polymorphism

1. Encapsulation:-

The different object inside of one program will try to communicate with each other automatically. If a programmer want to stop object from interacting with each other, they need to be encapsulate in individual classes cannot change or input with the specific variable and function of an object.

Just like a pill "encapsulation" or contains the medication inside of its coating the principle of encapsulation work in a digital way to form a protective barrier around the information that separate it from the rest of the code. Programmers can replicate this object through out different part of the program or other programs

2. Abstraction:-

Abstraction is like an extension of encapsulation because it hides certain properties and methods from the outside code to make the interface of the object code simpler. Programmer use abstraction for several beneficial reasons - overall abstraction help isolate the impact of changes made to the code so that if something goes wrong, the change will only effect the variable shown and not the outside code.

3. Inheritance:-

Programmers can extend the functionality of the code existing classes or eliminate repetitive code. For instance, element of HTML code that include a text box, select field and and checkbox have contain properties and common with specific methods. Instead of ~~redefine~~ redefining the properties and method for each types of HTML element you can define them once in a generic

Object - Naming that object something line "HTML element" will cause other object to inherit its properties and methods so you can reduce unnecessary codes.

Polymorphism :-

The technique meaning "many form or shapes". allows programmers to render multiple HTML element depending on the types of object. This concept allows programmers to redefine the the way something work by changing how it is done or by changing how its the part in which it is done.

Terms of polymorphism are called overriding or overloading.

Question No 3

Answer: -

```
import java.util.Scanner;  
public class Exercise 12 {
```

```
    public static void main (String [] args) {  
        Scanner in = new Scanner (System.in);
```

~~Scanner~~

```
        System.out.print ("Input speed (KM/H) of  
                            first blue car:");
```

```
        int blueCar1 = in.nextInt();
```

```
        System.out.print ("Input speed (KM/H) of  
                            second Red car:");
```

```
        int RedCar2 = in.nextInt();
```

```
        System.out.println ("Performance of two  
                             car is: " + Car1 + Car2);
```

blue ← ↓ Red

```
    }  
}
```