Regd No.:-19BQ1A05M3

JAVA ASSIGNMENT-1 (SET-5)

Section:-C.S.E.-D

① List and explain Java buzzwords. Which factors are making Java famous language.

A) Java Buzzwords :-

1) Simple :- Easy to code, write, read, modularity. No pointers

2) Object Oriented :- Class, objects, Encapsulation, inheritance, polymorphism.

3) Portable :- WORA (Write once run anywhere)

4) Platform independent :- Jave programs are executed anywhere when we have code and its platform independent means it can be executed in any Operating System. It contains intermediate code which genesates Byte code (JVM). Java Virtual Machine (or) JRE (Java Runtime Environment).

5) Robust :- Strong (with out errors), contains exception holding, it gives strength to your code, contains garbage collection.

6) Security :- It provides security for data and members using Abstraction and Encapsulation.

7) Multi-Threaded :- Concurrent execution, Threads.

8) High-Performance :- It is faster than C, C++. Java contains both compiler and interpretor.

9) Distributed :- TCP, P, RMI (Remote Method Innotation). Using internet we can execute the code present in other systems using ours systems.

10) Interpreted :- In Java the code is interpreted faster.

11) Dynamic :- We can execute programs dynamically. Java is capable of linking in new classes, libraries, methods and objects. It can also link native methods.

Factors making Java as famous language :-

1) Java is easy to learn.

2) Java is an object oriented programming language.

3) Java has Rich API

4) Java has powerful development tools like Eclipse and Netbeans.

5) It has great collection of Open Source Libraries.

6) Java has wonderful community support.

7) Java is free. Since java is free from the start i.e. you do not need to pay anything to create java application. This free thing also helped Java to become popular among large organizations.

8) Java has excellent documentation support - Javadocs.

9) Java is platform independent. The idea of platform independance is great and Java's tagline write once run anywhere was enticing enough to attract lots of new development in java.

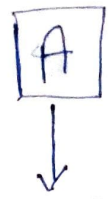
10) Java is Everywhere. It is on the desktop. It's on mobile, it's on the card and almost every where and so is Java programmers. This vast ability of Java programmers is another reason why organizations prefer to choose Java for their new development projects.

(2) What are the benefits of inheritance? Explain various forms of inheritance with suitable code segments.

A) Various forms of Inheritance :- Below are the various forms of Inheritance in Java

1) Single Inheritance :-

When a class extends another one class by only then we call it a single inheritance. The below flow diagram shows that class B extends only one class which is A. Here A is a parent class of B. B is child class of A.



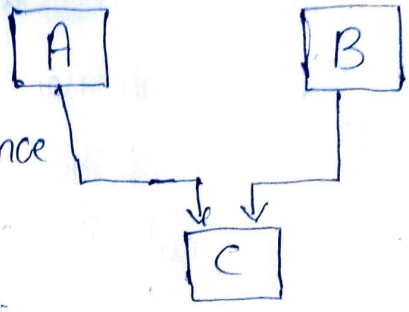
Single Inheritance

Example :-

```
class A
{ public void methodA()
  { System.out.println("Base Class method");
  }
}
class B extends A
{ public void methodB()
  { System.out.println("Child Class method");
  }
  public static void main(String args[])
  { B obj = new B();
    obj.methodA();
    obj.methodB();
  }
}
```

2) Multiple Inheritance :- Multiple Inheritance refers to the concept of one class extending (or) inheriting more than one true class. The problem with Multiple Inheritance is that the derived class will have to manage the dependancy on two base class.

* Multiple inheritance is very rarely used in Software projects. Using multiple inheritance often leads to the problems in hierarchy.
* Java does not allow Multiple Inheritance.



3) Multilevel Inheritance :-

Multilevel Inheritance refers to a mechanism in OO technology where one can inherit from a derived class, there by making this derived class the base class for the new class

Example :-


```
class X
{ public void method X()
  { System.out.println("Class X method");
  }
}
class Y extends X
{ public void method Y()
  { System.out.println("Class Y method");
  }
}
```

```
class C extends A
{ public void method()
  { System.out.println("Method of class C");
  }
}
class D extends A
{ public void methodD()
  { System.out.println("Method of class D");
  }
}
class JavaExample
{ public static void main(String args[])
  {
    B obj1 = new B();
    C obj2 = new C();
    D obj3 = new D();
    Obj1.methodA();
    Obj2.methodA();
    Obj3.methodA();
  }
}
```

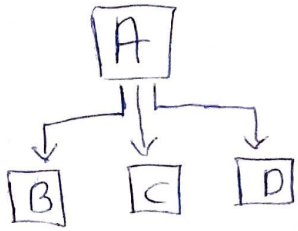
Output :-

```
method of class A
method of class A
method of class A
```

```
class Z extends Y
{ public void method Z()
  { System.out.println(" Class Z method ");
  }
  public static void main(String args[])
  {
     Z obj = new Z();
     obj.method X();
     obj.method Y();
     obj.method Z();
  }
}
```

4) Hierarchial Inheritance :-

In such kind of inheritance one class is inhereted by main sub classes. In below example class B, C and D inherits the same class A. A is parent class of B, C and D.



Example:-

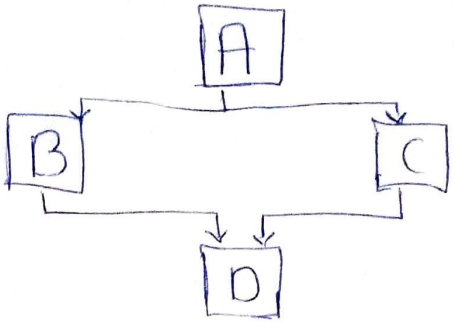
```
class A
{ public void method A()
  { System.out.println(" Method of class A");
  }
}
class B extends A
{ public void method B()
  { System.out.println(" Method of class B");
  }
}
```

5) Hybrid Inheritance :- In simple terms you can say that Hybrid Inheritance is a combination of single and multiple inheritance (or) more than one type of inheritance.

Example :-

```
class C
{
   public void disp()
   {
      System.out.println("C");
   }
}
class A extends C
{
   public void disp()
   {
      System.out.println("A");
   }
}
class B extends C
{
   public void disp()
   {
      System.out.println("B");
   }
}
class D extends A
{
   public void disp()
   {
      System.out.println("D");
   }
   public static void main(String args[]){
         D obj = new D();
         obj.disp();
   }
}
```

Output :- D



In the example program Class A and Class B extends Class C → Hierarchial inheritance

Class D extends class A → Single Inheritance

③ Define a class named movieMagic with the following discription:

Instance variables / data members:

int year - to store the year of release of a movie

String title - to store the title of the movie

float rating - to store the popularity rating of the movie

(minimum rating = 0.0 and maximum rating = 5.0)

Member methods:

(i) movieMagic() Default constructor to initialize numeric data members to 0 and string data member to " ".

(ii) void accept() To store year, title and rating inputs

(iii) void display() To display the title of a movie and a message based on the rating as per the table below:-

Rating	Message to be displayed
0.0 to 2.0	Flop
2.1 to 3.4	Semi-hit
3.5 to 4.5	Hit
4.6 to 5.0	Super Hit

Write a main method to create an object of the class and call the above members methods.

:A.) Program :-

```
import java.util.*;
class movieMagic {
    int year;
    float rating;
    String title;
    movieMagic() {
        year = 0;
        rating = 0.0f;
        title = "";
    }
    void accept() {
        Scanner sc = new Scanner(System.in);
        year = sc.nextInt();
        title = sc.next();
        rating = sc.nextFloat();
    }
    void display() {
        System.out.println("Title: " + title);
        if(rating > 0 && rating <= 2)
            System.out.println("Flop");
        else if(rating > 2 && rating <= 3.4)
            System.out.println("Semi-hit");
        else if(rating > 3.4 && rating <= 4.5)
            System.out.println("Hit");
        else if(rating > 4.5 && rating <= 5)
            System.out.println("Super-hit");
```

```
        else
            System.out.println("In-valid Rating!!");
    }
    public static void main(String args[])
    {
        movieMagic obj = new movieMagic();
        obj.accept;
        obj.display;
    }
}
```

Output :-

Output - 1 :-

2016
Brahmotsavam
5.0
Title : Brahmotsavam
Super-hit

Output - 2 :-

2020
XYZ
2.5
Title : XYZ
Semi-hit

(4) Write a class to overload a function num_calc() as follows:-
i) void num_calc(int num, char ch) with one integer argument and one character argument, computes the square of integer argument if choice ch is 's' otherwise its cube
ii) void num_calc(int a, int b, char ch) with two integer arguments and one character argument. It computes the product of integer arguments if ch is 'p' else add the integers.
iii) void num_calc(String s1, String s2) with two string arguments, which prints whether the strings are equal or not.

A) Program :-

```
import java.util.scanner;
public class Overloading {
    void num_calc(int num, char ch) {
        int op = 0;
        if(ch == 's')
            op = num * num;
        else
            op = num * num * num;
        System.out.println(op);
    }
    void num_calc(int a, int b, char ch) {
        int op;
        if(ch == 'p')
            op = a * b;
        else
            op = a + b;
        System.out.println(op);
    }
```

```
Void num_calc (String s1, String s2) {
        if(s1.equals(s2))
            System.out.println("Both Strings are same");
        else
            System.out.println("Both Strings are not same");
}
public static void main (String args[]) {
        Overloading obj = new Overloading();
        Scanner sc = new Scanner (System.in);
        int num = sc.nextInt();
        char ch1 = sc.next().charAt(0);
        int a = sc.nextInt();
        int b = sc.nextInt();
        char ch2 = sc.next().charAt(0);
        String s1 = sc.next();
        String s2 = sc.next();
        obj.num_calc(num, ch1);
        obj.num_calc(a, b, ch2);
        obj.num_calc(s1,s2);
    }
}
```

Output :-

Output - 1 :- (Input)

```
4
S
2
3
P
```

Hello

String

Output :-

16

6

Both Strings are not same.

Output - 2 :-

Input :-

5

a

5

4

b

Java

Java

Output :-

125

9

Both strings are same.