15-09-2020

S. SAI SRI VINAY REDDY
19BQ1A05J7
2nd CSE-D

JAVA PROGRAMMING
ASSIGNMENT - 1

1. Write about the role of JVM and Java API in developing the Platform independent java program with suitable example?

Ans:- The meaning of Platform-independent is that the java Compiled code(byte code). can run on all operating systems. while the role of JVM in independent platform is that it acts as a virtual processor, which processes the Byte code to machine code to instructions for various platforms

i.e. Programs written in Java are compiled into the Java Byte code, which is then interpreted by a special Java Interpreter for specific platform

Here, Java is platform-independent, but JVM is the Platform dependent.

For example, If we are running Mac OS X, we will have a different JVM than if we are running windows or Some other operatingsystem This can be verified while downloading the JDK which gives a list of OS targeted files. Hence, we conclude that the programing language, we write in any JDK is same, while the JDK file we use is platform dependent. There fore, JVM is platform dependent & Java is platform independent.

resources!-

https://www.geeksforgeeks.org
https://www.Brainly.in

Java API (Application Programing Interface) is a list of all classes that are the part of Java development kit (JDK) It includes all Java packages, classes and interfaces along with their methods, fields and Constructors. These pre-written classes provide a tremendous amount of functionality to a programer.

for example processing's reference is an API. It's the classes and functions we used to write Processing code, Similarly, the Java API is the list of classes and functions we use to write Java code The point is that an API is a collection of things we can do when writing code

references:

https //happycoding.io

https://resources saylor.org

2. Explain the concept of classes and Nested classes in Java with an example?

Ans: class - A class is a user defined blueprint or prototype from which objects are created. It represents the set of properties or methods that are common to all objects of one type. The components of a class are

→ Modifiers: A class can be public or has default access.

→ classname: The name should begin with a initial letter (Capitalized by convention)

-> Also a class can contain subclass, superclass or an

* Interface also

General structures of a class:

```
Public class classname{
        instance/class variable declaration;
        Default Constructor (optional);
        parameterised Constructor (if any;)
        Methods;
        and any other components;
        }
```

```
public class StaticNestedClassDemo{
    Public static void main(string a[]){
        outerclass.StaticNestedclass nestedObject;
        nestedObject = new OuterClass StaticNestedclass();
        nestedObject.display();
    }
}
```

Output:

```
    Outer_x = 10;
    Outer_Private = 30
```

```
// program for Innerclass
// we can access non static members of Outer class also
class Outerclass{
    static int outer_X = 10;
    int outer_y = 20;
    Private int outer_private = 30;
    class InnerClass{
        void display(){
            System.out.println("outer_x = "+ outer_x);
            System.out.println("outer_y = " + Outer_y);
            System.out.println("outer_private = "+ outer_private
        }
    }
};
Public class InnerClassDemo{
    Public static void main(string a[]){
        outerClass outerObject = new Outerclass();
        Outerclass.Innerclass innerObject =
                outerObject.newInnerClass();
```

```
public class StaticNestedClassDemo{
    Public static void main(string a[]){
        outerclass.StaticNestedclass nestedObject;
        nestedObject = new OuterClass StaticNestedclass();
        nestedObject display();
    }
}
```

Output:

```
Outer_x = 10;
Outer_private = 30
```

```
// program for Innerclass
// we can access non static members of outer class also.
class Outerclass{
    static int outer_x = 10;
    int outer_y = 20;
    Private int outer_private = 30;
    class InnerClass{
        void display(){
            System.out.println("outer_x = "+ outer_x);
            System.out.println("outer_y = " + outer_y);
            System.out.println("outer_private ="+ outer_private)
        }
    }
}
Public class InnerClassDemo{
    Public static void main(string a[]){
        OuterClass outerObject = new Outerclass();
        Outerclass.Innerclass innerObject =
                outerObject.newInnerClass();
    innerObject.display();
    }
}
```

Output:

```
    outer_x: 10
    outer_y: 20
    outer_private: 30
```

3 Design a class RailwayTicket with the following description

Intance Variables/data members :

String name: to store name of customer

String coach: to store type of coach.

long mobno: to store customers mobile number

int amt: to store basic amount of ticket

int totalamt: to store the amount to be paid after uploading the original amount.

Methods:

void accept()

void update()

void display()

Type of coaches	Amount.
First_AC	700
Second_AC	500
Third_AC	250
sleeper	None.

write the main() method to create an object of class and call the above methods.

```
class  RailwayTicket {
   Private String name;
   Private String coach.
   Private long mobno;
   Private int amt;
   Private int totamt;
   Public void accept (String name, String coach, long mobno, int amt)
   {
      this.name = name;
      this.coach = coach;
      this.mobno = mobno;
      this.amt = amt;
   }
   Public void update(){   → Ignore Case
      if (coach.compareTo("first_ac")==0)
      this.totalamt = amt + 700;      → Ignore Case
      else if (coach.compareTo("second_ac")==0)
      this.totalamt = amt + 500;  → Ignore Case
      else if (coach.compareTo("third-ac")==0)
      this.total amt = amt + 250;
                               → Ignore Case
      else if (coach.compareTo("sleeper")==0)
      this.totalamt = amt + 0;
      else
      this.totalamt = 0;
   }
  public void display(){
      if (totalamt ==0)
      System.out.println("Invalid Coach type ... Try again!!");
      else {
      System.out.println("Name : "+ name + "\n Coach type :" +
                           Coach+ "total amount : Rs." + totalamt
                           + "/-\n" + "Mobile : +91" + mobno +
      }                    "\n THANKYOU .... SAFE JOURNEY");
}
```

```
Public class Assignment{
    Public static void main (String args[]) {
    System.out.println("\t\t\t\t INDIANRAILWAYS \n \t\t\t ");
    System.out.println("\t\t\t SOUTH CENTRAL RAILWAYS");
    System.out.println("\t\t\t GUNTUR RAILWAY STATION");

    Scanner sc= new Scanner (System.in);
    System.out.println ("Enter passenger Name :");
    String name = sc.nextLine();
    System.out.println("\n 1. First_AC\n 2. Second_AC \n
                         3. Third_AC\n 4. Sleeper\n Enter coach");

    System.out.println ("numeric characters not allowed");
    String coach= sc.nextLine();
    System.out.print ("enter mobile number: +91");
    long mobno = sc.nextLong();
    System.out.println(" Base amount : 150/-");
    final int amt = 150;

    RailwayTicket ticket= new RailwayTicket();
    ticket.accept(name, coach, mobno, amt);
    ticket.update();
    ticket.display();
    System.out.println(" #STAYHOME... STAYSAFE# ");
  }
}
```

4. Design a class to overload a function volume() as follows.

i) double volume(double r) - with 'r' radius as an argument, return the volume of sphere using the formula.

$V = \frac{4}{3} \times \frac{22}{7} \times r^3$

ii) double volume (double h, double r) - with height 'h' and radius 'r' as the arguments returns the volume of cylinder using the formula:

$V = \frac{22}{7} \times r^2 \times h.$

iii) double volume (double l, double b, double h) - with length l breadth b, height h as the arguments, returns the volume of a cuboid using the formula

v= l×b×h.

```
class Volume{
    public double volume (double r){
        double V=(4/3)*(22/7)*(r*r*r);
        return v;
    }
    public double volume (double h, double r){
        double V=(22/7)*(r*r)*h;
        return v;
    }
    public double volume (double l, double b; double h) {
        double V= l* b* h;
        return v;
    }
}
public class Assignment {
    public static void main (String args[]){
        volume vol = new Volume();
        Scanner sc = new Scanner (System.in);
        System.out.println("1.volume of sphere\n 2.volume of
                cylinder\n 3.volume of cuboid\n Enter choice:");
        int ch= sc.nextInt();
        switch(ch){
            case 1:
                System.out.println ("vol.of sphere\n radius :");
                double r = sc.nextDouble();
                double v1 = vol.volume(r);
```

```
Public class Assignment {
    Public static void main (String args[]) {
    System.out.println("\t\t\t\t INDIANRAILWAYS \n \t\t\t ");
    System.out.println("\t\t\t SOUTH CENTRAL RAILWAYS");
    System.out.println("\t\t\t GUNTUR RAILWAY STATION");

    Scanner sc= new Scanner (System.in);
    System.out.println ("Enter passenger Name :");
    String name = sc.nextLine();
    System.out.println("\n 1. First_AC \n 2. Second_AC \n
                          3. Third_AC \n 4. Sleeper \n Enter coach");

    System.out.println ("numeric characters not allowed");
    String coach= sc.nextLine();
    System.out.print ("enter mobile number: +91");
    long mobno = sc.nextLong();
    System.out.println(" Base amount : 150/-");
    final int amt = 150;

    RailwayTicket ticket = new RailwayTicket();
    ticket.accept(name, coach, mobno, amt);
    ticket.update();
    ticket.display();
    System.out.println(" #STAYHOME... STAYSAFE# ");
  }
}
```

4. Design a class to overload a function volume() as follows.

i) double volume(double r) - with 'r' radius as an argument, return the volume of sphere using the formula.

$$V = \frac{4}{3} \times \frac{22}{7} \times r^3$$

ii) double volume (double h, double r) - with height 'h' and radius 'r' as the arguments returns the volume of cylinder using the formula: $V = \frac{22}{7} \times r^2 \times h.$

iii) double volume (double l, double b, double h) - with length l
breadth b, height h as the arguments, returns the volume
of a cuboid using the formula
v- lxbxh.

```
class Volume{
    Public double volume (double r){
        double v=(4/3)*(22/7)*(r*r*r);
        return v;
    }
    Public double volume (double h, double r){
        double v=(22/7)*(r*r)*h;
        return v;
    }
    public double volume (double l, double b; double h) {
        double v= l* b * h;
        return v;
    }
}
Public class Assignment {
    Public static void main (String args[]){
        volume vol = new Volume();
        Scanner sc = new Scanner (System.in);
        System.out.println("1.volume of sphere\n 2.volume of
                cylinder\n 3.volume of cuboid\n Enter choice:");
        int ch= sc.nextInt();
        switch(ch){
            case 1:
                System.out.println("vol.of sphere\n radius :");
                double r = sc.nextDouble();
                double v1 = vol.volume(r);
    System.out.println("result :"+V1);
case 2:
      System.out.println("volume of cylinder");
      System.out.println(" Enter height & radius");
      double h = sc.nextDouble();
      double r1 = sc.nextDouble();
      double V2 = vol.Volume(h,r1);
      ~~break~~;
      System.out.println("result :" + V2);
      break;
case 3:
      System.out.println("Volume of cuboid:");
      System.out.println("Enter l,b,h");
      double l = sc.nextDouble();
      double b = sc.nextDouble();
      double h1 = sc.nextDouble();
      double V3 = Vol.volume(l,b,h1);
      System.out.println("result :" + V3);
      break;
default:
      System.out.println("choice out of Range");
    }
  }
}
```