2100090109.

Assignment : 2

1) Difference between Array and ArrayList

Array	ArrayList
An array is a dynamically created objects it serves as a Container that holds the Constant number of values of the same type,	The ArrayList is a class of java collections framework. It contains classes
Array is static in size	ArrayList is dynamic in size
An array is a fixed length data structure	ArrayList is a Variable length datastructure it can be resized itself when needed
The Array can store both objects and primitive type	we cannot store primitive type in ArrayList
Array can be multi-dimensional	ArrayList is always single-dimensional

2) Explain Dynamic Method Dispatch with Example.

Dynamic method dispatch:- is the mechanism in which a call to an overridden method is resolved at run time instead of Compile-time. This is an important concept because of how java implements run-time Polymorphism

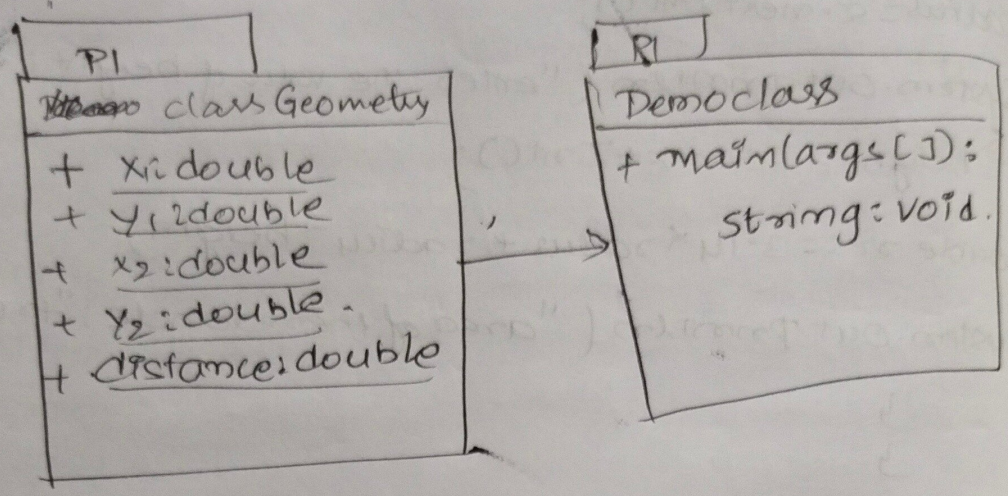
Java uses the principle of a superclass reference variable can refer to a subclass object' to resolve calls to overridden methods at runtime

Example:-

```
Package pack1;
import java.util.scanner;
Public class Shape
  {
  Public static int radius;
  Public void area()
  {
System.Out.println("enter the value of radius");
  Scanner sc= new Scanner(system.in);
   radius=sc.nextInt();
   int res = radius*radius;
System.Out.println("The area of the rectangle
                    is :"+res));
        }
        }
Class rectangle extends shape
   {
   int height;
    Public void area()
    {
  Scanner sc= new scanner(system.in);
System.Out.println("enter the radius");
```

```
radius = sc.nextInt();
System.out.println("enter the value of height:");
    height = sc.nextInt();
double res = 3.14 * radius * radius * height;
System.out.println("area of the rectangle:"+res);
    }
    }
class Overridden
{
public static void main(String[] args)
{
    shape q;
    q = new shape();
    q.area();
    q = new rectangle()
    q.area();
    }
    }
```

3> Develop a static method distance() in class geometry which accepts x_1 y_1 x_2 y_2 representing the coordinates of two points of type double and then returns the distance between them and call the method from main() in Demo class. Assume & inputs command line. Draw the class diagram.



~~Class~~.

```
Package P1;
import java.util.scanner;
public class Democlass
{
public static void main (string[] args)
{
sc.x1 = Double.parseDouble(args[0]);
sc.y1 = Double.parseDouble(args[1]);
sc.x2 = Double.parseDouble(args[2]);
sc.y2 = Double.parseDouble(args[3]);
Class Geometry sc=new ClassGeometry();
System.out.println(" The distance of 4 points
is: "+sc.distance());
}
}
```

```
package P1;
public class classGeometry
{
public static double x1;
public static double y1;
public static double x2;
public static double y2;

public static double distance()
{
double dis = Math.sqrt( y2-y1)*(y2-y1)+(x2-x1)*(x2-x1);

    return dis;

}
}
```

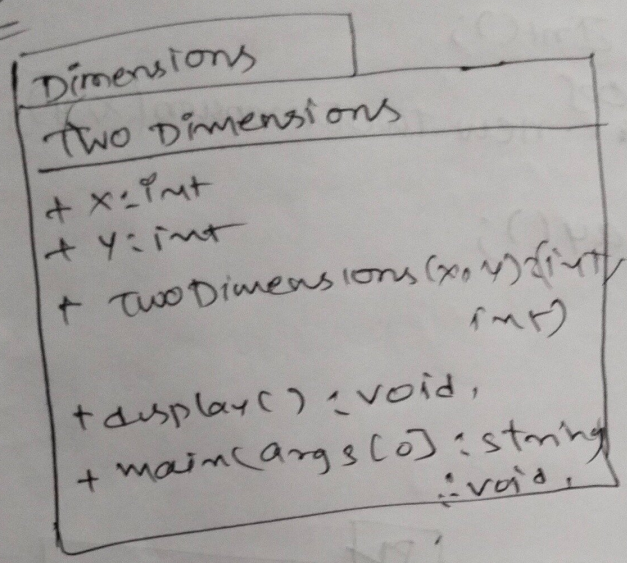
(4) • Constructors :

Constructors
+ loan : int + years : int + rate : int + display() : void + main(args[]) : String:void + Constructors() : + Constructors (int, int, int) : (int, int, int) + display() : void

```
Package constructors;
import java.util.scanner;
Public class Constructors
{
Public static int loan;
Public static int years;
Public static int rate;
  Public Constructors()
  {
  loan=0;
  years=0;
  rate=0;
  }
  Public Constructors (int i, int j, int k)
  {
  loan = i;
  years=j;
  rate =k;
  }
Public void display()
{
System.Out.println("loan:"+loan+"years:"+
        years+"rate:"+rate);
}
public static void main(string[] args)
{
scanner sc=new Scanner(system.in);
constructors obj = new Constructors();
obj.display();
```

```
Constructors obj1=new Constructors(100000,2,30);
    obj1.display();
}
}
```

5)

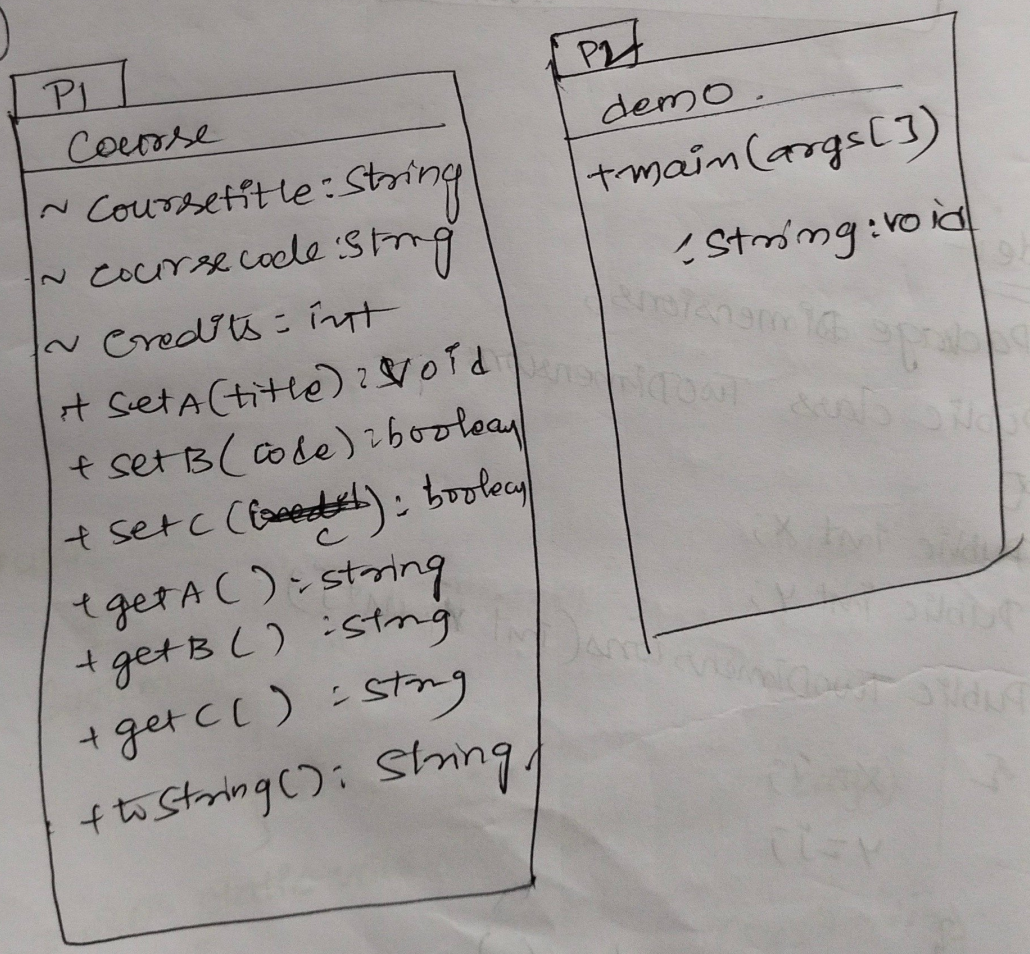


code:-

```
Package Dimensions;
Public class TwoDimensions
{
Public int x;
Public int y;
Public TwoDimensions(int i, int j)
{   x=i;
    y=j;
}
Public void display()
{
System.out.println("value of x"+x+"value of y"+y)
}
```

```
public static void main (string[] args)
{
    Scanner sc= new Scanner(system.in)

        X=sc.nextInt();
        Y=sc.nextInt();
TwoDimension obj = new TwoDimension(x, y);

        obj.Display();

        }
    }
```

(6)



```
Package ~~Course~~ P;
Public class course
{
Private String courseitle;
Private String courseode;
Private int Credits;
Public void setA( string title)
{
Courseitle= title;
}
Public boolean setB(String code)
{
if (code.matches("[A-Z.A-Z]+"))
{
courseode=code;
return true;
}
return false;
}
Public boolean setC( int c)
{
if (c>0&&c<6)
{
credit =c;
return true;
}
return false;
}
```

```
Public string getA()
  {
  return Course title;
  }
Public String getB()
  {
  return coursecode;
  }
Public String getC()
  {
  return Credits+points;
  }
Public string toString()
{
String str = String.format("CourseTitle:
%s%n coursecode %s%n course credit %s %n",
getA(), getB(), getC());
return str;
}
}

Package P2;
import java.util.scanner;
import P1.course;
Public class demo{
Public static void main(String[] args)
  {
  Course sc=new Course();
```

```
Scanner s1= new Scanner(System.in);
    sc.setA(s1.nextlong());
    sc.setB(s1.next());
    sc.setC(s1.nextInt());
    System.out.println(sc);
        }
    }
```

numbers

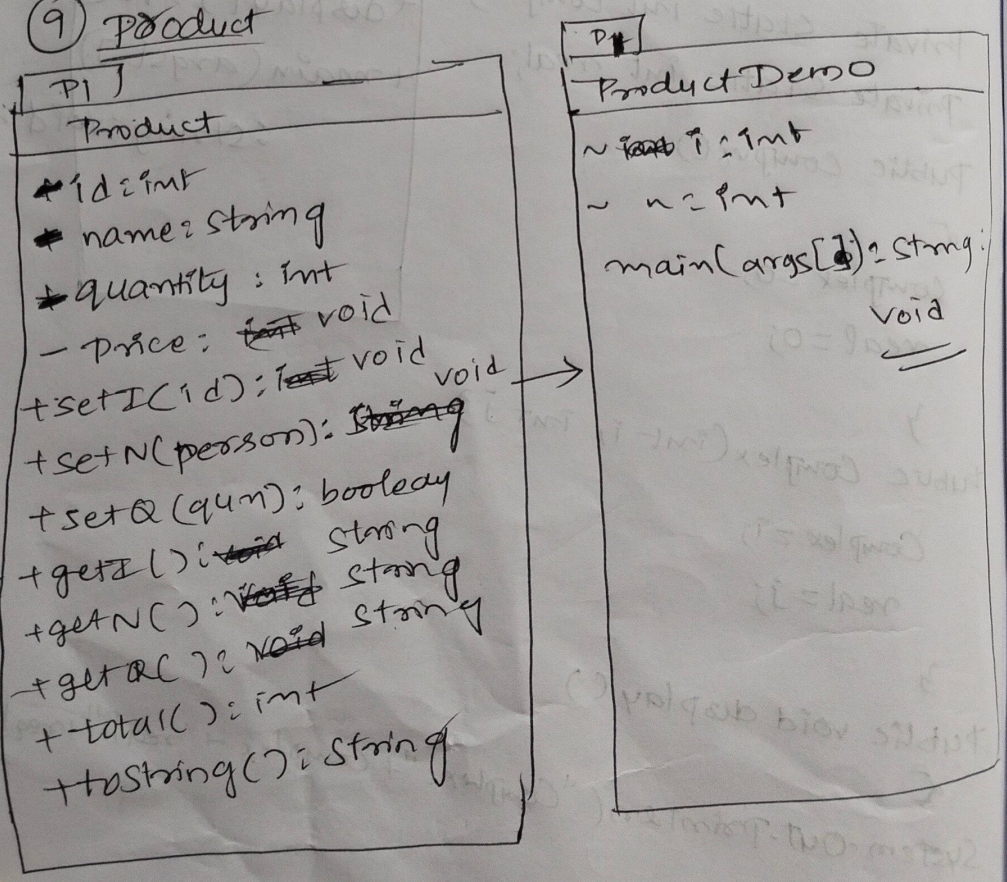
- complex : int
- Real : int
+ Complex() : void
+ Complex(i,j) : int,int
+ display() : void
+ main(args[])
: String : void

(7)

```
package numbers;
import java.util.scanner;
public class Complex
    {
    Private static int Complex;
    Private static int real;
    public complex()
    {
     complex=0;
     real=0;
    }
public Complex(int i, int j)
     Complex=i;
      real=j;
    }
public void display()
    {
System.Out.Println("Complex:"+ Complex +"real"+real);
    }
```

```
Public static void main(string[] args)
  {
   Complex sc = new Complex();
     sc.display();
   Scanner cs = new Scanner (system.in);
   Complex=cs.nextInt();
   real=cs.nextInt();
   Complex obj = new Complex(Complex, real)
     obj.display();
        }
      }
```

(9) Product



```
Package P1;
Public class product.
{
Private int ID;
Private String name;
Private int quantity
Private int price;
Public int setI(int id)
{ ID=id;
~~set~~ }
Public int setN(String person)
{ name=person;
}.
Public boolean setQ(int qun)
{ if (qun>0)
{ quantity=qun;
return true;
}
else
{ quantity =0;
return false;
}
}
```

```
Public boolean setP(int pri)
{
 if (pri>0)
  {
  price=pri;
   return true;
      }
  else
  {
   Price=0;
   return false;
      }
    }
Public String getI()
  {
  return id;
      }
Public String getN()
   {
   return name;
      }
Public int getQ()
      {
      return quantity;
      }
Public int getP()
      {
      return price;
      }
```

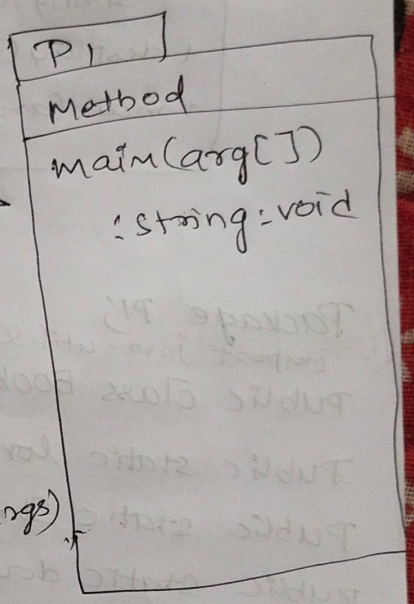
```
public int total()
{
  return quantity * price;
  }
public String toString()
  {
String str = String.format("%s%n %s%n %d %n %d %n
  %d", getI(), getN(), getQ(), getP(), total());

    return str;
    }
  }

package pack;
import java.util.Scanner;
public class ProductDemo
  {
public static void main(String args[])
  {
  Product z = new Product();
  int i = integer.parseInt(args[0]);
  int n = integer.parseInt(args[1]);
    Scanner sc = new Scanner(system.in);
      z.setI(sc.next());
      z.setN(sc.next());
      z.setQ(sc.nextInt());
      z.setP(sc.nextInt());
    System.out.println(z);
    }
  }
```

(1) Package P1;

```
Public class Student
{
Private int id;
Private string name;
Private double mobile-

  public void setA (int idn)
  {
    id = idn;
  }
Public void setB (string names)
{ name = names;
  }
Public void setC (double mbl)
  {
    mobile = mbl;
  }
  Public int getA()
  {
    return id;
  }
  Public String getB()
  {
    return name;
  }
```

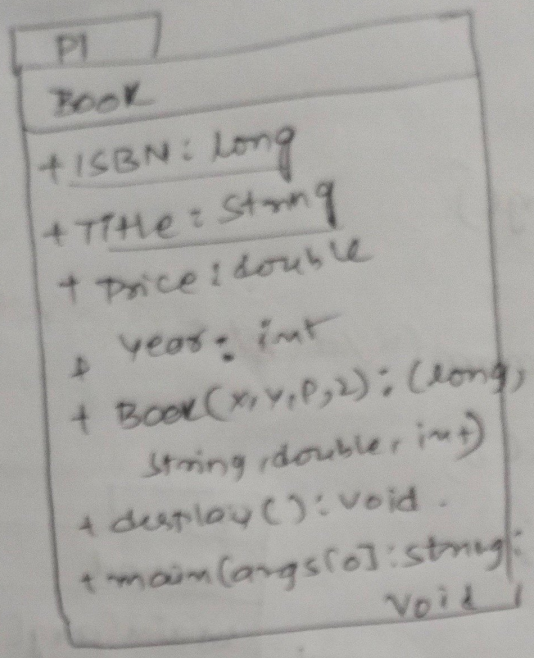
P1
student
+id: int +name: String +mobile: double +setA (idn): int: vo +setB (names): String: vo +setC (mbl): double: vo +getA( ): int +getB( ): string +getC( ): dousle +toString( ): string

```
Public double getC()
{
return mobile;
}
Public String toString()
{
String str= String.format("%d %s %n %f",
getA(), getB(), getC());
return str;
}
}
```




```
Package pack1;
import java.util.scanner;
Public class Method {
Public static void main(String[] args)
{
Student s1= new Student();
Scanner sc= new Scanner(System.in)
s1.setA(sc.nextInt());
s1.setB(sc.next());
s1.setC(sc.nextDouble());
System.out.println(s1);
}
}
```

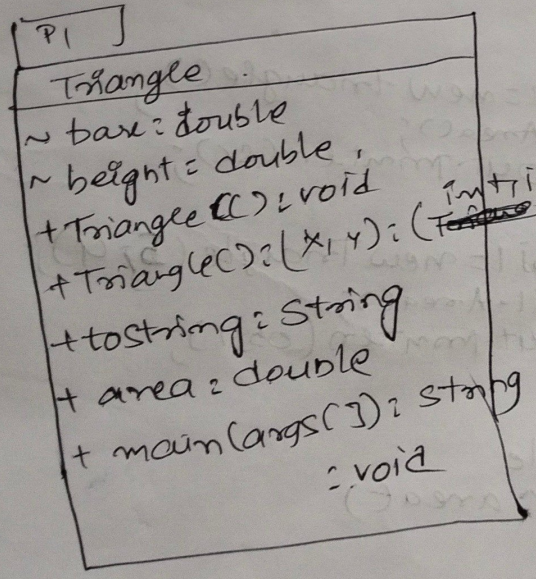
(12)




```
Package P1;
import java.util.Scanner;
Public class Book;
Public static long ISBN;
Public static String Title;
Public static double Price;
Public static Integer Year;
Public Book(long x, String y, double p,
                                int z)
{

ISBN = x;
Title = y;
Price = p;
Year = z;

}
```

```
Public void display()
{
System.out.Println("ISBN"+ISBN+"title"+
title+"Price"+Price+"year"+year);
}
Public static void main (string[] args)
{
scanner sc= new scanner (system.in);
ISBN =sc.nextlong();
title = sc.next();
Price = sc.nextDouble();
year = sc.nextInt();
Book Obj = new Book (ISBN,title,price,year);
Obj.Display();
}
```

(13)

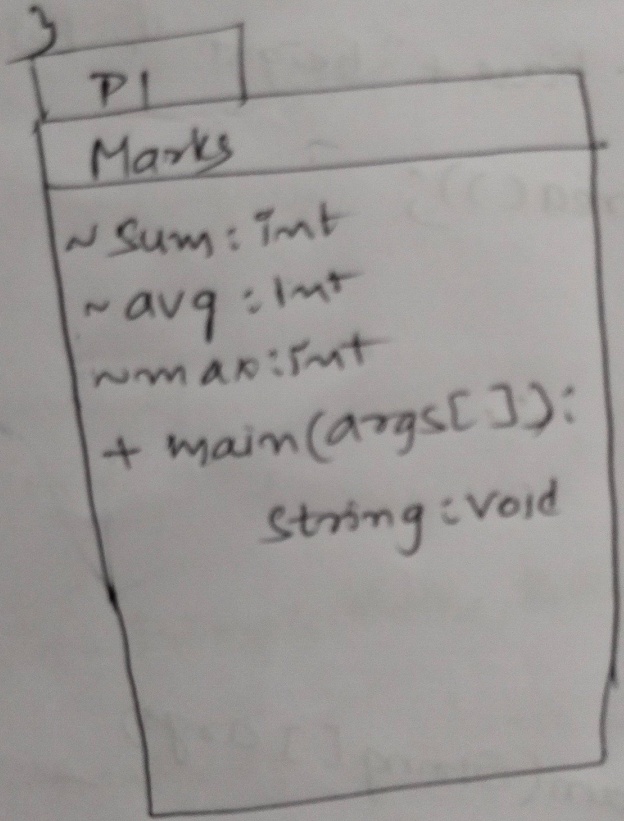


```
Package P1;

Public class Triangle
{
    Private double base;
    Private double height;

Public triangle()
{
    base=0;
    height =0;
}
Public Triangle(double x, double y)
{
    base=x;
    height=y;
}
Public static void main (String[] args)
{
    triangle sc=new triangle();
        sc.Area();
    system.out.println(sc);

    Triangle obj1=new Triangle(5,4);
        obj1.Area();
    System.out.println(obj1);

}
Public void double area()
{
```

```
double res = 0.5*base*height;
        return res;
    }
Public String toString()
{
String str ("base value" + base + " height " + height +
        "area" area());
    }
}
```

---

(14) Package P1;

```
Public class Marks
{
Public static void main(String [] args)
{
int sum=0;
int max = args[0];
for (int i =0; i<args.length; i++)
    int x = Integer.parseInteger[i];
        if (args[i] > max)
        {
            max = args[i];
        }
        sum = sum + x;
}
```

```
int avg=sum/args*length;

System.out.println("sum="+sum);
System.out.println("avg="+avg);
System.out.println("max="+max);
System.out.println
}
}
```