N. Nitishkumar
II CSE-C
19BQ1A05G2

SET-3

1) What is data abstraction? Differentiate data and procedural abstraction. Write inheritance hierarchy for the Superclass Quadrilateral, parallelogram, square & Rectangle. Calculate Area of Square, rect & parallelogram.

A) Data abstraction is the process of hiding certain details and sharing only essential information to the user.

Difference between data & procedoral obstraction:

Procedoral obstractions are normally characterized In a programming language as "function/subfunction" or "procedure". It is procedural abstraction is making use of methods that accepts formal parameters and hiding the implentation from the user. Methods are used to capture the procedoral patterns, abstracting over Behaviour.

Eg:- String str = "Hello world";
String str1 = str.substring(0,5);

Inheritance hierarchy:-

```
import java.util.Scanner;
abstract class Quadrilateral{
    public abstract double area(int l, int b);
}
class parallelogram extends Quadrilateral{
    Public double area(int l, int b){
    return "l*b";}
class Rectangle extends Quadrilateral{
    Public double are (int l, int b){
            return l*b";
}
class Square extends Quadrilateral{
    public double area(int l, int b){
            return l*b;}
    Public class Area{
    Public static void main(String[] args){
```

```
Square S = new Square();
System.out.println(s.area(s,s));
Parallelogram p = new parallelogram();
System.out.println(p.area(s,6));
 Rectangle r = new Rectangle();
System.out.println(r.area(10,s));
}
```

2) what is importance of Constructor?

A) Constructor:- Constructor is a method like a block of code which is called by Java Runtime during object. Creator using new() operator. Constructor are special in case that have same name as the class they are part of. They are also special in a sense that they are called by JVM automatically when you create a object.

Benefits:-

i) one reason is to initialize your object with default or initial state since default values are primitives.

ii) one more reason is create constructor to inform the world about dependencies a class needs to do it job

Ex:-

```
class Main{
  Private int x;
    // Constructor
  private Main(){
    System.out.println("Constructor called");
    x=5;
  }
  Public Static Void main(String[] args){
    Main obj = new Main();
    System.out.println("value of x = "+obj.x);
  }
}
```

Constructor Overloading:- It is a techinque of having more than one constructor with different parameter lists.

```
Public class OverloadingExample
{
  private int rollNum;
  OverloadingExample()
  { rollNum = 100;
  }
  OverloadingExample (int rnum)
  {
    this();
    rollNum = rollNum + rNum;
  }
  Public int getRollNum(){
        return rollNum;
  }
  Public void setRollNum(int rollNum){
        this.rollNum = rollNum;
  }
  Public static void main(String args[])
  {
     OverloadingExample obj = new OverloadingExample(10);
    System.out.println(obj.getRollNum());
  }
}
```

Static Methods:- Static methods are also called class methods. It is because a static method belongs to the class rather than object of a class.

Ex:-
```
Public class Myclass{
    Public static void sample(){
        System.out.println("Hello");
    }
    Public static void main(string args[]){
        Myclass.sample();
    }
}
```

Nesting Members:- A Nested class is a member and a such has the same access rights as any other member. As the use of Nested classes that enable you used in one place, increase the use of encapsulation, and create more readable and maintable code.

Ex:- class Outerclass {
    //...
    class Nestedclass {
        //...
    }
}

3A.
```
Public class BookFair{
    String Bname;
    double price;
    BookFair(String Bname, double price){
        this.Bname = Bname;
        this.price = price;
    }
    Public class BookFair{
        String Bname;
        double price;
    BookFair(){}
    Public void input(){
        Scanner input = new Scanner(System.in);
            Bname = input.next();
            price = input.double();
    }
    Public void calculate(){ double discound;
        if (price <= 1000)
            discount = (2* price)/100;
        else if (price > 1000 && price <= 3000)
            discount = (10* price)/100;
```

```
        else discount = (15*price)/100;
             price := price - discount;
    }
    public void display() {
    System.out.print(this.Banana+"of price:"+this.price);
    }
    Public static void main(String[] args){
        Book Fair B = new Book Fair();
            b.input();
            b.calculate();
            b.display();
    }
}
```

4) Write a program to accept word check. print whether the word is a palindrome or only special words.

```
import java.util.Scanner;
class test
{
public static void main()
{
    Scanner s = new Scanner(System.in);
    System.out.println("Enter a word");
     String w = s.next();
      int i = w.length();
     String w1 = " ";
//char ch1, ch2;
for (int k=0; k<i; k++)
{
    ch1 = w.charAt(k);
      w1 = ch1+w1;
} if (w.equals(w1) == true)
```

```
System.Out.println("It is palindrome word");
    else if (w.charAt(0) == w.charAt(i-1))
System.out.println("It is only special word");
else.
System.out.println("It is not a special word");
}
}
```