Java assignment-1

19BQ1A05N7
CSE-D

Set-3

1) What is data abstraction? Differentiate data and procedural abstraction. Write inheritance hierarchy for the super class Quadrilateral parallelogram, square and rectangle. Calculate area of square, rectangle and parallelogram.

A) Abstraction means displaying only essential information and hiding the irrelevant information. Data abstraction refers to providing only essential information about the data, hiding the background details. For example man driving a car only knows accelerator will increase the speed of car and applying brakes will decrease the speed of car, steering of car helps to rotates/turns the car to left or right. But he doesn't known actually how its functioning or mechanism of accelerator, steering, engine or internal parts of the car.

Procedural Abstraction:

Procedural abstraction provides mechanisms for abstracting well defined procedures or operations as entities. The implementation of the procedure requires a number of steps to be performed. A simple example is a debit card operation which performs various steps to debit certain amount from the bank account. Hence at the banking level, credit and debit become well defined procedural abstractions. These are extensively used by requirements analysts, as well as designers and programmers.

Procedural abstractions are normally characterized in a programming language as function/subfunction or procedure abstraction.

Data Abstraction:

This principle is at the case of object orientation. In this form of abstraction, instead of just focusing on operations, we focus on data first and then the operations that manipulate the data. A simple example is queue data and the associated operations add() and delete(). Both add() and delete() operations manipulates queue data. In a simple procedural abstraction there would be only add and delete operations separately but their association with the queue data will not be captured. The advantage of data abstraction over procedural abstraction is that the data and the association operations get specified together and hence it is easy to modify the code when data changes.

Inheritance hierarchy for the super class Quadrilateral, Parallelogram, Square, Rectangle:

```
import java.util.Scanner;
class QuadrilateralTest {
    double a;
    double b;
    Scanner sc = new Scanner(System.in);
}
class Parallelogram extends QuadrilateralTest {
    public void input() {
        System.out.println("Enter First side of Parallelogram");
        a = sc.nextDouble();
        System.out.println("Enter Second side of Parallelogram");
        b = sc.nextDouble();
    }
    public double area() {
        return a*b;
    }
}
```

```
class Rectangle extends QuadrilateralTest{
    Public void input(){
        System.out.println("Enter the sides of Rectangle");
        a= sc.nextDouble();
        System.out.println("Enter the sides of Rectangle");
        b= sc.nextDouble();
    }
    Public double area(){
        return a*b;
    }
}
class Square extends QuadrilateralTest{
    Public void input(){
        System.out.println("Enter side value of square");
        a= sc.nextDouble();
    }
    Public double area(){
        return a*a;
    }
}
Public class Quadrilateral{
    Public static void main(String.args[]){
        Parallelogram obj1 = new Parallelogram();
        Rectangle obj2 = new Rectangle();
        Square obj3= new square();
        obj1.input();
        System.out.println("\n Area of Parallelogram"+obj1.area()+"\n");
        obj2.input();
        System.out.println("\n Area of Rectangle"+obj2.area()+"\n");
        obj3.input();
        System.out.println("\n Area of square"+obj3.area()+"\n");
    }
}
```

output:

Enter the first side of parallelogram
2
Enter the second side of parallelogram
3
Area of the Parallelogram 6.0
Enter the sides of Rectangle
9
Enter the sides of Rectangle
5
Area of the Rectangle 45.0
Enter side value of square
7
Area of the square 49.0

2) What is the importance of Constructor? Write a java program to perform constructor overloading. Describe the usage of static members and nesting members with suitable example programs in java.

A) In simple word, constructor is a method like a block of code which is called by Java runtime during object creation using new() operator. Constructor are special in the sense that they have the 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 an object.

Importance of Constructor:

One reason is to initialize your object with default an initial state since default values for primitives may not be what we are looking for. One more reason we create constructor is to inform the world about dependencies, a class needs to do its job. Anyone by looking at our constructors should be able to figure out, what he needs in order to use this class.

Constructor can be overloaded:

This means we can have more than one constructor in our class (all with the same name) until they have different method signature, which comprises type of argument and order type of argument. Here is an example of constructor overloading. Here we have three constructors but all with a different set of parameters.

Example of Constructor overloading:

```
class student {
    int id;
    String name;
    int age;
    // creating two arg constructor
    Student(int i, String n){
    id = i;
    name = n;
    }
    // creating three arg constructor
     Student (int i, String n, int a){
     id = i;
     name = n;
     age = a;
    }
    void display(){
        System.out.println(id+" "+name+" "+age);}
    public static void main(String.arg s[]){
        student S1 = new student(95, "Kaethik");
        Student S2 = new student(20, "Ramu", 19);
        S1.display();
        S2.display();
}
}
```

Output:

95 Kaethik 0

20 Ramu 19

Static Members in Java:

In Java, static members are those which belongs to the class and we can access these members without instantiating the class. The static keyword can be used with methods, Fields, classes (inner/nested), blocks

Use of static members in Java, when a member is declared static, it can be accessed before any objects of its class are created, and without reference to any object.

Example of static method:

```
Class Test
{
   //static method
   Static void PS()
   {
      system.out.println("from PS");
   }
   public static void main(string[] args)
   {
      PS();
   }
}
```

Output:

From PS

Nesting members in Java:

They enable you to logically group classes that are only used in one place, thus this increases the use of encapsulation and creates more readable and maintainable code. As a member of its enclosing class, a nested class can be declared private, public, Protected or package.

Example of static nested class

```
Class outerclass
{
    static int outer_P=10;
    int outer_Q = 20;
    private static int outer_private = 30;
    static class staticNestedClass
    {
        void display()
        {
            System.out.println ("outer_P = "+outer_P);
            system.out.println ("outer_private = " + outer_private);
        }
    }
}
Public class staticNestedclassDemo
{
    public static void main (String[] args)
    {
        Outerclass.staticNestedClass . nestedobject = new outerclass.
                                                    staticNestedClass();
        nestedobject.display();
    }
}
```

output:

```
outer_P = 10
outer_private = 30
```

3) Define a Class named Bookfair with following description

Instance Variable/Data members:

String Bname - Stores the name of the book.
double price - stores the price of the book.

Member Methods:

(i) Bookfair() - Default constructor to initialize data members
(ii) void Input() - To input and store the name and price of the book
(iii) void calculate() - To calculate the price after discount. Discount is calculated based on the following criteria.

Price	Discount
less than or equal to Rs1000	2% of price
More than Rs1000 and less than Or equal to Rs 3000	10% of price
More than Rs 3000	15% of price

(iv) void display() - To display the name and price of the book after discount.

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

A) Program

```
import java.util.Scanner;
class Bookfair{
        String Bname;
        double price;
        Scanner SC = new Scanner (System.in);
        public Bookfair(){}
        public Bookfair( string Bname, double price){
            this.Bname = Bname;
            this.price = price;
        }
        public void calculate(){
            if
```

```
    Public void Input(){
        System.out.println("Enter Book Name:");
        Bname = sc.nextLine();
        System.out.println("Enter Book price:");
        Price = sc.nextDouble();
    }
    public void calculation(){
        if (price <= 1000.00){
            price = price - (price * 0.02);
        }
        else if (price > 1000.00 && price <= 3000.00){
            price = price - (price * 0.1);
        }
        else {
            price = price - (Price * 0.15);
        }
    }
    public void display(){
        System.out.println("Name of the book = " + Bname);
        System.out.println("Price of the book after discount
                                = " + price);
    }
}
public class Bfair{
    Public static void main (String.args[]){
        Bookfair book1;
        book1 = new Bookfair();
        book1.Input();
        book1.calculate();
        book1.display();
    }
}
```

Output 1:

Enter Book Name:
NDA
Enter Book Price:
570
Name of the book = NDA
Price of the book after discount = 558.60

Output 2:

Enter Book Name:
Untold story
Enter Book price: 1200
Name of the book = Untold story
Price of the book after discount = 1080.00

Output 3:

Enter Book Name:
Rachna Bisht Rawat
Enter Book price: 3420
Name of the book = Rachna Bisht Rawat
Price of the book after discount = 2907.00

4) Special words are those words whic starts and ends with the same letter.

Examples:
EXISTENCE
COMIC
WINDOW

Palindrome words are those words which read the same from left to right and vice-versa.

Examples:
MALAYALAM
MADAM
LEVEL
ROTATOR
CIVIC
All palindromes are special words, but all special words are not palindromes.
Write a program to accept a word check and print whether the word is a palindrome or only special word.

A) Program

```
import java.util.*;
Public class check
{
  public static void main (String[] args) {
      String original, reverse = "";
      Scanner sc = new Scanner (System.in);
      System.out.println(" Enter a string to be reversed :");
      original = sc.nextLine();
      int length = original.length();
      For (int i = length-1; i>=0; i--)
         reverse = reverse + original.charAt(i);
      System.out.println(" Reverse of the string :" + reverse);
      if(original.equals(reverse) && (original.substring(0,1).
                                      equals(original.substring(length-1))))
         System.out.println ("palindrome");
      else if (original.substring(0,1).equals(original,substring(length-1)))
          system.out.println ("special word");
      else
          system.out.println ("None");
  }
```

Output 1:

Enter a string tourreversed:
CIVIC
Reverse of a string: CIVIC
Palindrome

Output 2:

Enter a string to be reversed:
WINDOW
Reverse of the string: WODNIW
Special word

Output 3:

Enter a string to be reversed:
MADAM
Reverse of a string: MADAM
Palindrome.