ASSIGNMENT - I

M. Geethika
CSE - C

Set - 3

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

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

Abstraction is one of the major pillars.

Data Abstraction :-

It means while designing the classes itself, you need to identify only those attributes of class which are relevant to that domain. For example :- if person is an entity, it can have many attributes such as first name, surname, salary, spouse, age, height, weight, Blood group. But if you are developing an application for healthcare domain, then you want to choose only those attributes which are related to healthcare (eg. height, blood group) and ignore the rest. on the other hand, if you are developing an application for social survey, those attributes are not required. In this way, identifying the required attributes and ignoring the rest is the data abstraction.

Procedual Abstraction :-

The necessary part is "what the procedure does and ignoring how it does it". most of the languages by default support it. In Java, one class can call methods of other class without knowing its implementation details.

Inheritance hrerarchy program:

```
public class Quadrilateral
{
 protected int x1,x2,x3,x4,y1,y2,y3,y4;
 protected void setcoordinate(int a, int b, int c, int d, int e, int f, int g, int h)
{
 x1=a;
 y1=b;
 x2=c;
 y2=d;
 x3=e;
 y3=f;
x4=g;
y4=h;
}
}
public class square extends Quadrilateral{
square(int a, int b, int c, int d, int e, int f, int g, int h)
{
set coordinate(a,b,c,d,e,f,g,h);
}
int area()
{
int d=(int)Math.sqrt[(x1-x2)*(x1-x2)+(y1-y2)*(y1-y2)];
return d;
}
}
public class Rectangle extends Quadrilateral{
Rectangle(int a, int b, int c, int d, int e, int f, int g, int h){
Set coordinate(a,b,c,d,e,f,g,h);
}
int area()
{
int d1=(int)Math.sqrt((x1-x2)*(x1-x2)+(y1-y2)*(y1-y2));
```

```
int d2 = (int)Math.sqrt((x1-x4)*(x1-x4)+(y1-y4)*(y1-y4));
return d1*d2;
}
}

public class parallelogram extends Quadrilateral {
private int height;
Parallelogram(int a, int b, int c, int d, int e, int f, int g, int h, int height)
{
Set coordinate (a,b,c,d,e,f,g,h);
this.height = height;
}
int area()
{
int d1 = (int)Math.sqrt((x1-x2)*(x1-x2)+(y1-y2)*(y1-y2));
return d1*height;
}
}

public class testQuadrilateral{
public static void main (String[] args){
Square                    sq=new
Square(10,10,20,10,20,10,20);
System.out.println("Area of the Square is "+sq.area());
Rectangle                 rec=new
Rectangle(10,10,30,10,30,20,10,20);
System.out.println("Area of the Rectangle is "+rec.area());
Parallelogram             p=new
Parallelogram(10,10,30,10,20,0,20,8);
System.out.println("Area of the parallelogram is "+p.area());
}
}
```

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.

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. one reason is to initialize your object with default or initial state since default values for primitives may not be. one more reason you create constructor is to inform the world about dependencies, a class needs to do its job.

Program to perform constructor overloading :-

Suppose we have a 'student' class and while making its object, we want to pass a name of it and if nothing is passed. then name should be "unknown". And yes! we can do this by having two constructors.

```
class Student {
    private string name;
    public Student( string n){
        name = n;
    }
    public student(){
        name = "unknown";
    }
```

```
    public void printName(){
        System.out.println(name);
    }
}
class Out{
    public static void main(String[] args){
        Student a = new Student("XYZ");
        Student b = new Student();
        a.printName();
        b.printName();
    }
}
```

output:-

XYZ
Unknown.

Static nested class:-

A static class - that is created inside a class is called static nested class in Java. It cannot access non-static data members and methods. It can be accessed by outer class name.

It can access static data members of outer class including private

Static nested class cannot access non-static (instance) data member.

example:-

```
class Testouter1{
static intdata =30;
static class Inner{
void msg(){
    system.out.println("data is"+data);
}
}
public static void main (string args[]){
Testouter1.Inner obj = new Testouter1.Inner();
obj.msg();
}
}
```

Output:-

data is 30.

3) Define a class named Bookfair with the following description

Instance variables / Data members:

string ename - stores the names of the book.

double price - stores the price of the book.

Member methods:

(i) Book fair() - Default constructor to initialize data members

(ii) void input() - To input and store the name and the 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 RS 1000	2% of price
more than RS 1000 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 the class and call the above member methods.

```
import java.io.*;
class Bookfair
{
string Bname; double price;
Book fair()
{
Bname = " ";
price = 0.00;
}
void Input() throws IO Exception
{
BufferedReader br = new
BufferedReader (new Inpot StreamReader (system.in));
system.out.println("Enter the name of the book ");
Bname = br.readline();
System.out.println("Enter the price of the book");
Price = Double.parseDouble( br.readline());
}
```

```
void caculate()
{
if(price <= 1000)
price = price -(price* 0.02);
else if( price > 1000 && price <= 3000 )
price = price(price* 0.1);
else
price = price -(price* 0.05);
}
void Display()
{
System.out.println ("Book Name \t"+ "Total price");
System.out.println (Bname +"\t "+ price );
}
public static void
main (String []args)throws IO exception
{
Book fair obj= new Book fair();
obj. Input();
obj. caculate();
obj. Display();
}
}.
```

4) Special words are those words which 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 weather the word is a palindrome or any special word.

```
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<l; k++)
```

```
{
    ch1 = w.charAt(k);
    w1 = ch1 + w1;
}
    if (w1.equals(w) == true)
    System.out.println("It is palindrome word");
    else if( w.charAt(0) == w.charAt(i-1))
    System.out.println("It is only a Special word");
else
    System.out.println("It is not a Special word");
    }
}
```

Resources:-

-> https://www.bschools.com/java/java_abstract

-> https://www.javapoint.com/java_constructor

-> https://javalearners1.blogspot.com/2016/10/book-fair-program.html .