

Practical-1

Aim: Write a program to convert rupees to dollar. 60 rupees=1 dollar.

```
class r2d {
    float rs,dollar;
    void convert(float rs){
        dollar = rs/60;
        System.out.println("Dollar is " +dollar);
    }
}

class d2r {
    float rs,dollar;
    void convert(float dollar){
        rs = dollar * 60;
        System.out.println("Rs is " +rs);
    }
}

class conversion {
    public static void main(String args[]){
        r2d r = new r2d();
        d2r d = new d2r();
        r.convert(120);
        d.convert(2);
    }
}
```

```
E:\160410116133\2150704-java-practicals-master\practical 1>javac conversion.java

E:\160410116133\2150704-java-practicals-master\practical 1>java conversion
Dollar is 2.0
Rs is 120.0
```

Aim: Write a program that calculates percentage marks of the student if marks of 6 subjects are given.

```
import java.util.*;
class inpt
{
    float s1[]= new float [6];
    float n,sum=0;
    void inp()
    {
        Scanner value= new Scanner(System.in);
        System.out.println("No of Subjects you want to enter marks for:");
        n=value.nextFloat();
        if(n!=6)
        {
            System.out.println("ERROR");
        }
        else
        {
            System.out.println("Enter Marks of Subjects:");
            for(int i=0;i<n;i++)
            {
                s1[i]=value.nextFloat();
            }
        }
    }

    void marks()
    {
        float sum=0,per;
        for(int i=0;i<6;i++)
        {
            sum=sum+s1[i];
        }
        per=sum/6;
        System.out.println("Percent=" +per);
    }
}
class abc
{
    public static void main(String args[])
    {
        float a;
        inpt ABC= new inpt();
        ABC.inp();
    }
}
```

```
        ABC.marks();  
    }  
}
```

```
E:\160410116133\2150704-java-practicals-master\practical 1>javac abc.java  
E:\160410116133\2150704-java-practicals-master\practical 1>java abc  
No of Subjects you want to enter marks for:  
6  
Enter Marks of Subjects:  
89  
78  
67  
90  
81  
60  
Percent=77.5
```

Practical-2

Aim: Write a program to find length of string and print second half of the string

```
import java.util.*;
class input{
    public static void main(String args[]){
        int i;
        String a;
        Scanner AB = new Scanner(System.in);
        a = AB.nextLine();
        int l = a.length();
        char def[] = new char[a.length()];
        System.out.println("Length is = " +l);
        System.out.println("Half of string is : " +a.substring(l/2));
        a.getChars(0, a.length(),def,0);
        for(i=a.length()/2;i<a.length();i++){
            System.out.print(" "+def[i]);
        }
        char x[] = a.toCharArray();
        for(i=x.length/2;i<x.length;i++){
            System.out.println(" "+x[i]);
        }
    }
}
```

```
E:\160410116133\2150704-java-practicals-master\practical 2>javac input.java
E:\160410116133\2150704-java-practicals-master\practical 2>java input
Hello World
Hello World
Hello World
Length is = 11
Half of string is : World
 W o r l d
W
o
r
l
d
```

Aim: Write a program to enter two numbers and perform mathematical operations on them.

```
import java.util.*;
public class Cla
{
    public static void main(String[] args)
    {
        int a,b;
        float res;
        Scanner scan=new Scanner(System.in);
        System.out.print("enter the value of a");
        a=scan.nextInt();
        System.out.println("enter the value of b");
        b=scan.nextInt();
        res=a+b;
        System.out.println("addition==" +res);
        res=a-b;
        System.out.println("subtraction==" +res);
        res=a*b;
        System.out.println("multiplication==" +res);
        res=(float)a/(float)b;
        System.out.println("division==" +res);
    }
}
```

```
E:\160410116133\2150704-java-practicals-master\practical 2>javac Cla.java
E:\160410116133\2150704-java-practicals-master\practical 2>java Cla
enter the value of a20
enter the value of b
10
addition==30.0
subtraction==10.0
multiplication==200.0
division==2.0
```

Practical-3

Aim: Write a program to accept a line and check how many consonants and vowels are there in line

```
import java.util.*;
class s1{
    public static void main(String args[]){
        int i;
        String a;
        Scanner AB = new Scanner(System.in);
        a = AB.nextLine();
        System.out.println(" " +a);
        for(i=0;i<a.length();i++){
            if(a.charAt(i)=='a' || a.charAt(i)=='e' || a.charAt(i)=='i' || a.charAt(i)=='o' ||
a.charAt(i)=='u' || a.charAt(i)=='A' || a.charAt(i)=='E' || a.charAt(i)=='I' || a.charAt(i)=='
'O' || a.charAt(i)=='U'){
                System.out.println("Vowels are " +a.charAt(i));
            }
            else{
                System.out.println("Consonants are " +a.charAt(i));
            }
        }
    }
}
```

```
D:\Sem 5\00PJ\pracs\practical 3>javac s1.java
D:\Sem 5\00PJ\pracs\practical 3>java s1
rahul soni
rahul soni
Consonants are r
Vowels are a
Consonants are h
Vowels are u
Consonants are l
Consonants are
Consonants are s
Vowels are o
Consonants are n
Vowels are i
```

Aim: Write a program to count the number of words that start with capital letters

```
import java.util.*;
class r1{
    public static void main(String args[]){
        int i,word=0,space=0;
        String a;
        Scanner AB = new Scanner(System.in);
        a = AB.nextLine();
        System.out.println(" " +a);
        for(i=0;i<a.length();i++){
            char ch = a.charAt(i);
            if(ch == 32){
                space++;
            }
            if (Character.isUpperCase(ch)
                && (i == 0 || Character.isWhitespace(a.charAt(i - 1)))) {
                word++;
            }
        }
        System.out.println("Total capital words" +word);
        System.out.println("Total spaces" +space);
    }
}
```

```
D:\Sem 5\00PJ\pracs\practical 3>javac r1.java
D:\Sem 5\00PJ\pracs\practical 3>java r1
hEllo This iS a New WOrld
hEllo This iS a New WOrld
Total capital words3
Total spaces5
```

Practical-4

Aim: Write a program to find that given number or string is palindrome or not.

```
import java.util.*;
class input{
    String name = "";
    void set(String a){
        name = a;
    }
    void get(){
        System.out.println(" " +name);
    }
}
class palindrome {
    public static void main(String args[]){
        String s1,s2;
        int i;
        input a1 = new input();
        Scanner AB = new Scanner(System.in);
        s1 = AB.nextLine();
        a1.set(s1);
        a1.get();
        s2 = "";
        for(i=s1.length()-1;i>=0;i--){
            s2 = s2+s1.charAt(i);
        }
        if(s2.equalsIgnoreCase(s1)){
            System.out.println("Palindrome");
        }
        else{
            System.out.println("Not Palindrome");
        }
    }
}
```

```
E:\160410116133\2150704-java-practicals-master\practical 4>javac palindrome.java
E:\160410116133\2150704-java-practicals-master\practical 4>java palindrome
hannah
hannah
Palindrome
E:\160410116133\2150704-java-practicals-master\practical 4>java palindrome
rahul
rahul
Not Palindrome
```


Aim: Create a class which asks the user to enter a sentence, and it should display count of each vowel type in the sentence. The program should continue till user enters a word “quit”. Display the total count of each vowel for all sentences.

```
import java.util.*;
class quit{
    public static void main(String args[]){
        String s1,s2;
        int i,count=0;
        Scanner AB = new Scanner(System.in);
        s1 = AB.nextLine();
        while(AB.hasNext()){
            s2 = AB.nextLine();
            if(s2.equalsIgnoreCase("quit"))
                break;
            s1 = s1+s2;
        }

        System.out.println("Ans: " +s1);
        for(i=0;i<s1.length();i++){
            if(s1.charAt(i)=='a' || s1.charAt(i)=='e' || s1.charAt(i)=='i' || s1.charAt(i)=='o' || s1.charAt(i)=='u' || s1.charAt(i)=='A' || s1.charAt(i)=='E' || s1.charAt(i)=='I' || s1.charAt(i)=='O' || s1.charAt(i)=='U'){
                count++;
            }
        }

        System.out.println("Number of vowels " +count);
    }
}
```

```
E:\160410116133\2150704-java-practicals-master\practical 4>javac quit.java
E:\160410116133\2150704-java-practicals-master\practical 4>java quit
hello this a new world
quit
Ans: hello this a new world
Number of vowels 6

E:\160410116133\2150704-java-practicals-master\practical 4>java quit
hello this is a new world
quit
Ans: hello this is a new world
Number of vowels 7
```

Practical-5

Aim: Write an interactive program to print a string entered in a pyramid form. For instance, the string “stream” has to be displayed as follows:

S
S t
S t r
S t r e
S t r e a
S t r e a m

```
import java.util.*;
class pattern{
    public static void main(String args[]){
        String s1,s2="";
        int i,j,k=50;
        Scanner AB = new Scanner(System.in);
        s1 = AB.nextLine();
        for(i=0;i<s1.length();i++){
            for(j=0;j<k;j++){
                System.out.print(" ");
            }
            k=k-1;
            for(j=0;j!=i+1;j++){
                System.out.print(" "+s1.charAt(j));
            }
            System.out.println();
        }
    }
}
```

```
E:\160410116133\2150704-java-practicals-master\practical 5>javac pattern.java
```

```
E:\160410116133\2150704-java-practicals-master\practical 5>java pattern
```

```
stream
```

```

      S
    S t
  S t r
S t r e
S t r e a
S t r e a m
```

Aim: Write an interactive program to print a diamond shape. For example, if user enters the number 3, the diamond will be as follows:

```

      *
     **
    ***
     **
      *
  
```

```

import java.util.*;
class pattern2{
    public static void main(String args[]){
        String s1,s2="";
        int i,j,k=50;
        Scanner AB = new Scanner(System.in);
        s1 = AB.nextLine();
        for(i=0;i<s1.length();i++){
            for(j=0;j<k;j++){
                System.out.print(" ");
            }

            k=k-1;
            for(j=0;j!=i+1;j++){
                System.out.print(" "+s1.charAt(j));
            }

            System.out.println();
        }
        k=52-s1.length();
        for(i=s1.length();i!=0;i--){
            for(j=0;j!=k;j++){
                System.out.print(" ");
            }

            k=k+1;
            for(j=i-1;j!=0;j--){
                System.out.print(" "+s1.charAt(j));
            }

            System.out.println();
        }
    }
}
  
```

```

E:\160410116133\2150704-java-practicals-master\practical 5>javac pattern2.java
E:\160410116133\2150704-java-practicals-master\practical 5>java pattern2
***
  
```

```

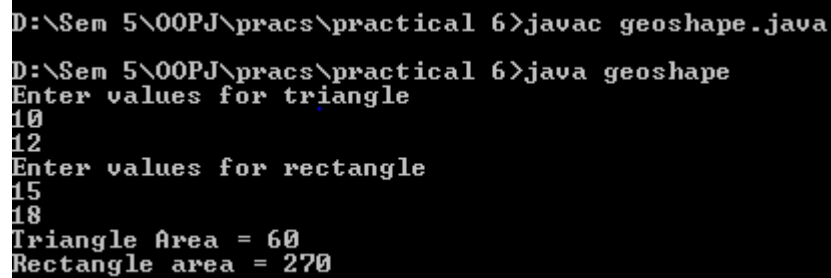
      *
     **
    ***
     **
      *
  
```

Practical-6

Aim: Create one Abstract Class Shape that has two variables for dimensions and one Abstract method called area (). Create two subclass Rectangle and Triangle of Shape class and find the area of Rectangle and Triangle. Create appropriate Constructors for data.

```
import java.util.*;
abstract class shape {
    int d1,d2;
    abstract void area(int d1 , int d2);
}
class triangle extends shape{
    int base,height,area;
    triangle(){
        base = 0;
        height = 0;
        area = 0;
    }
    void set(int b, int h){
        base = b;
        height = h;
    }
    void area(int base,int height){
        area = (base*height)/2;
        System.out.println("Triangle Area = " +area);
    }
}
class rectangle extends shape{
    int length,breadth,area;
    rectangle(){
        length = 0;
        breadth = 0;
        area = 0;
    }
    void set(int l, int b){
        length = l;
        breadth = b;
    }
    void area(int length, int breadth){
        area = (length*breadth);
        System.out.println("Rectangle area = " +area);
    }
}
```

```
public class geoshape{
    public static void main(String args[]){
        int a,b,c,d;
        Scanner AB = new Scanner(System.in);
        System.out.println("Enter values for triangle");
        a = AB.nextInt();
        b = AB.nextInt();
        System.out.println("Enter values for rectangle");
        c = AB.nextInt();
        d = AB.nextInt();
        triangle t1 = new triangle();
        rectangle r1 = new rectangle();
        t1.set(a,b);
        t1.area(a,b);
        r1.set(c,d);
        r1.area(c,d);
    }
}
```



```
D:\Sem 5\00PJ\pracs\practical 6>javac geoshape.java
D:\Sem 5\00PJ\pracs\practical 6>java geoshape
Enter values for triangle
10
12
Enter values for rectangle
15
18
Triangle Area = 60
Rectangle area = 270
```

Practical-7

Aim: WAP to demonstrate concept of different types of access modifiers using package.

```
//demo
package pk1;
import java.util.*;
public class Demo {
    public static void main(String args[])
    {
        Protection ob1 = new Protection();
        Derived ob2 = new Derived();
        SamePackage ob3 = new SamePackage();
    }
}

//demo2
package pk2;
public class Demo2
{
    public static void main(String args[])
    {
        Protection2 ob1 = new Protection2();
        OtherPackage ob2 = new OtherPackage();
    }
}

//Derived
package pk1;
class Derived extends Protection
{
    Derived()
    {
        System.out.println("derived constructor");
        System.out.println("n = " + n);
        //System.out.println("n_pri = " + n_pri);
        System.out.println("n_pro = " + n_pro);
        System.out.println("n_pub = " + n_pub);
    }
}

//Other package
package pk2;
class OtherPackage
{
    OtherPackage(){
        pk1.Protection p = new pk1.Protection();
    }
}
```

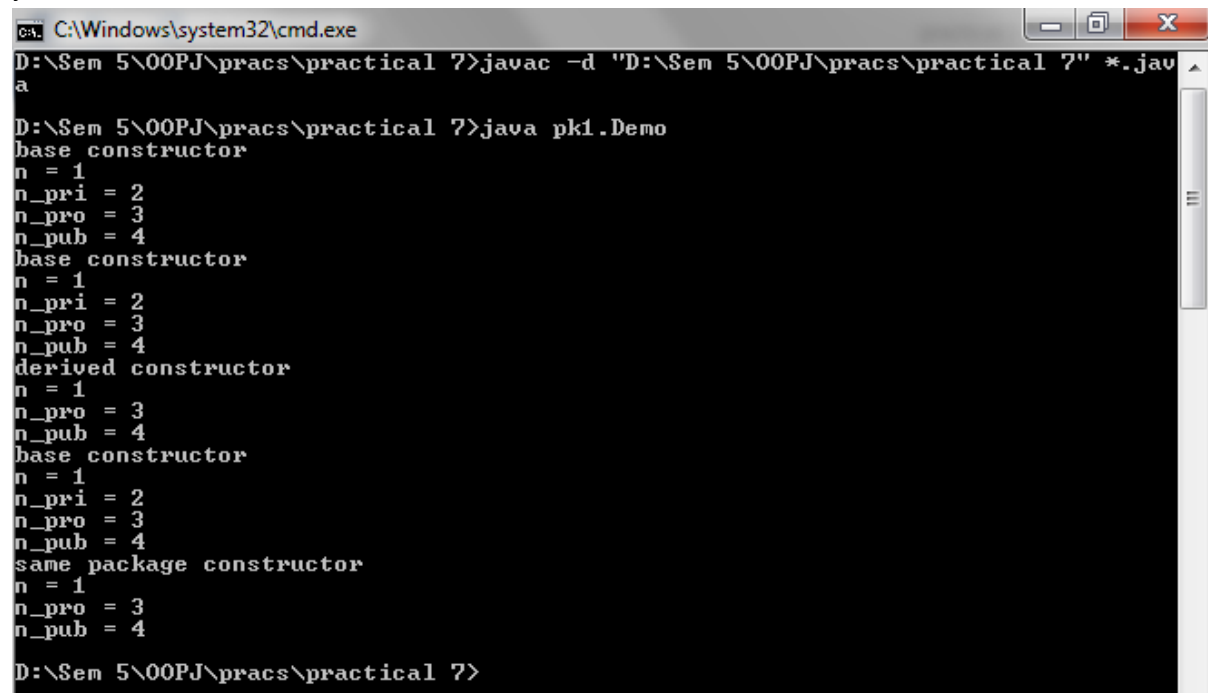
```
        System.out.println("other package constructor");
        // System.out.println("n = " + p.n);
        // System.out.println("n_pri = " + p.n_pri);
        // System.out.println("n_pro = " + p.n_pro);
        System.out.println("n_pub = " + p.n_pub);
    }
}

//Protection
package pk1;
public class Protection
{
    int n = 1;
    private int n_pri = 2;
    protected int n_pro = 3;
    public int n_pub = 4;
    public Protection()
    {
        System.out.println("base constructor");
        System.out.println("n = " + n);
        System.out.println("n_pri = " + n_pri);
        System.out.println("n_pro = " + n_pro);
        System.out.println("n_pub = " + n_pub);
    }
}

//Protection2
package pk2;
class Protection2 extends pk1.Protection
{
    Protection2()
    {
        System.out.println("derived other package constructor");
        // System.out.println("n = " + n);
        // System.out.println("n_pri = " + n_pri);
        System.out.println("n_pro = " + n_pro);
        System.out.println("n_pub = " + n_pub);
    }
}

//samepackage
package pk1;
class SamePackage {
    SamePackage() {
        Protection p = new Protection();
    }
}
```

```
        System.out.println("same package constructor");
        System.out.println("n = " + p.n);
        //System.out.println("n_pri = " + p.n_pri);
        System.out.println("n_pro = " + p.n_pro);
        System.out.println("n_pub = " + p.n_pub);
    }
}
```



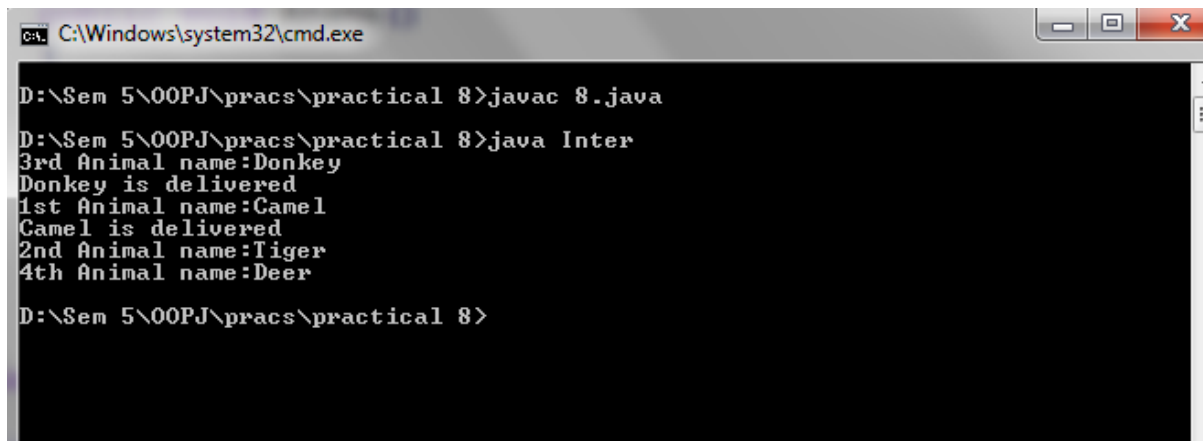
```
C:\Windows\system32\cmd.exe
D:\Sem 5\00PJ\pracs\practical 7>javac -d "D:\Sem 5\00PJ\pracs\practical 7" *.java
D:\Sem 5\00PJ\pracs\practical 7>java pk1.Demo
base constructor
n = 1
n_pro = 3
n_pub = 4
base constructor
n = 1
n_pro = 3
n_pub = 4
derived constructor
n = 1
n_pro = 3
n_pub = 4
base constructor
n = 1
n_pro = 3
n_pub = 4
same package constructor
n = 1
n_pro = 3
n_pub = 4
D:\Sem 5\00PJ\pracs\practical 7>
```


Practical 8

AIM: The Transport interface declares a deliver () method. The abstract class Animal is the super class of the Tiger, Camel, Deer and Donkey classes. The Transport interface is implemented by the Camel and Donkey classes. Write a test program that initialize an array of four Animal objects. If the object implements the Transport interface, the deliver () method is invoked.

```
interface Transport
{
    void Deliever();
}
abstract class Animal
{
    abstract void Aname();
}
class Tiger extends Animal
{
    void Aname()
    {
        System.out.println("2nd Animal name:Tiger");
    }
}
class Camel extends Animal implements Transport
{
    public void Aname()
    {
        System.out.println("1st Animal name:Camel");
    }
    public void Deliever()
    {
        System.out.println("Camel is delivered");
    }
}
class Donkey extends Animal implements Transport
{
    public void Aname()
    {
        System.out.println("3rd Animal name:Donkey");
    }
    public void Deliever()
    {
        System.out.println("Donkey is delivered");
    }
}
class Deer extends Animal
{
    void Aname()
    {
        System.out.println("4th Animal name:Deer");
    }
}
class Inter
```

```
{
    public static void main(String args[])
    {
        Transport Trans;
        Tiger T=new Tiger();
        Camel C=new Camel();
        Donkey D=new Donkey();
        Deer D1=new Deer();
        Trans=D;
        D.Aname();
        Trans.Deliever();
        Trans=C;
        C.Aname();
        Trans.Deliever();
        T.Aname();
        D1.Aname();
    }
}
```



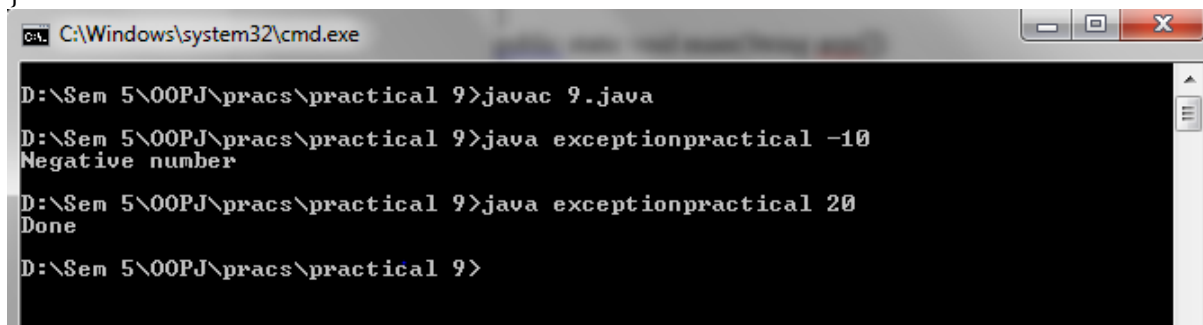
The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe". The user has navigated to the directory "D:\Sem 5\00PJ\pracs\practical 8" and executed the following commands and their outputs:

```
D:\Sem 5\00PJ\pracs\practical 8>javac 8.java
D:\Sem 5\00PJ\pracs\practical 8>java Inter
3rd Animal name:Donkey
Donkey is delivered
1st Animal name:Camel
Camel is delivered
2nd Animal name:Tiger
4th Animal name:Deer
D:\Sem 5\00PJ\pracs\practical 8>
```

Practical 9

AIM: Write a program that generates custom exception if any integer value given from its command line arguments is negative.

```
class CustomException extends Exception
{
    String name;
    CustomException(String S)
    {
        name=S;
    }
    public String toString()
    {
        return "Negative number";
    }
}
class exceptionpractical
{
    static void check(int n) throws CustomException
    {
        if(n<0)
            throw new CustomException("Exception");
        else
            System.out.println("Done");
    }
    public static void main(String args[])
    {
        int i=Integer.parseInt(args[0]);
        try
        {
            check(i);
        }
        catch(CustomException e)
        {
            System.out.println(e);
        }
    }
}
```



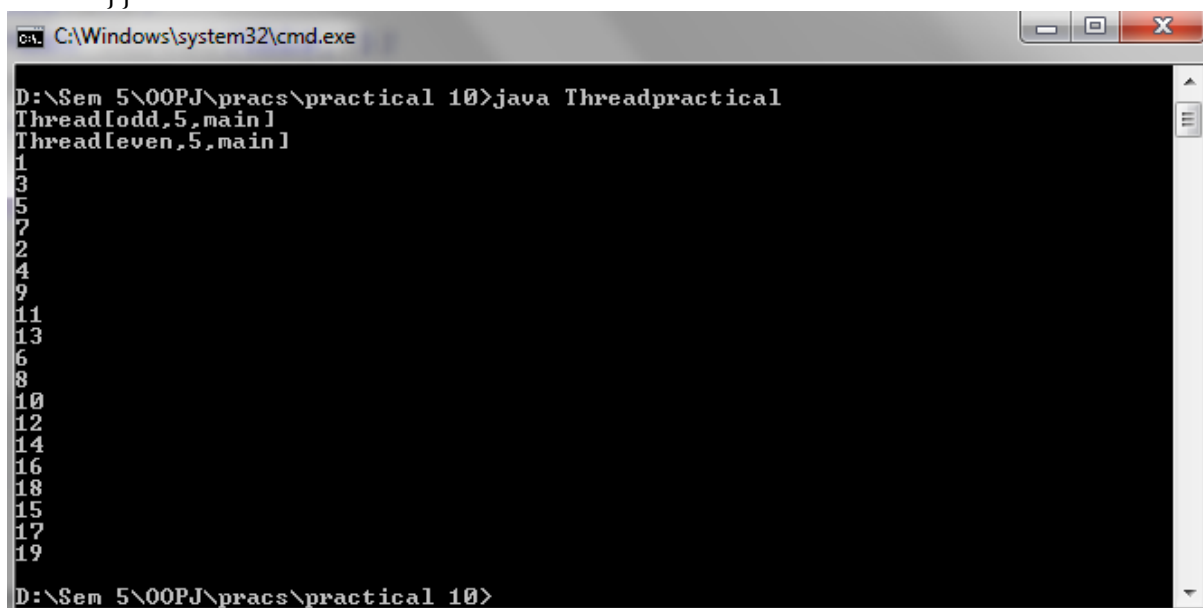
```
C:\Windows\system32\cmd.exe
D:\Sem 5\00PJ\pracs\practical 9>javac 9.java
D:\Sem 5\00PJ\pracs\practical 9>java exceptionpractical -10
Negative number
D:\Sem 5\00PJ\pracs\practical 9>java exceptionpractical 20
Done
D:\Sem 5\00PJ\pracs\practical 9>
```

Practical 10

Aim: Write a program to create two threads, one thread will print odd numbers and second thread will print even numbers between 1 to 20 numbers.

```
class MultiThread implements Runnable
{
    Thread t;
    String s;
    MultiThread(String s){
        this.s=s;
        t=new Thread(this,s);
        System.out.println(t);
        t.start();
    }
    public void run(){
        for(int i=1;i<20;i++){
            if(s.equalsIgnoreCase("odd") && i%2!=0){
                System.out.println(i);
            }
            else if(s.equalsIgnoreCase("even") && i%2==0){
                System.out.println(i);
            }
        }
    }
}

class Threadpractical
{
    public static void main(String args[]){
        new MultiThread("odd");
        new MultiThread("even");
    }
}
```



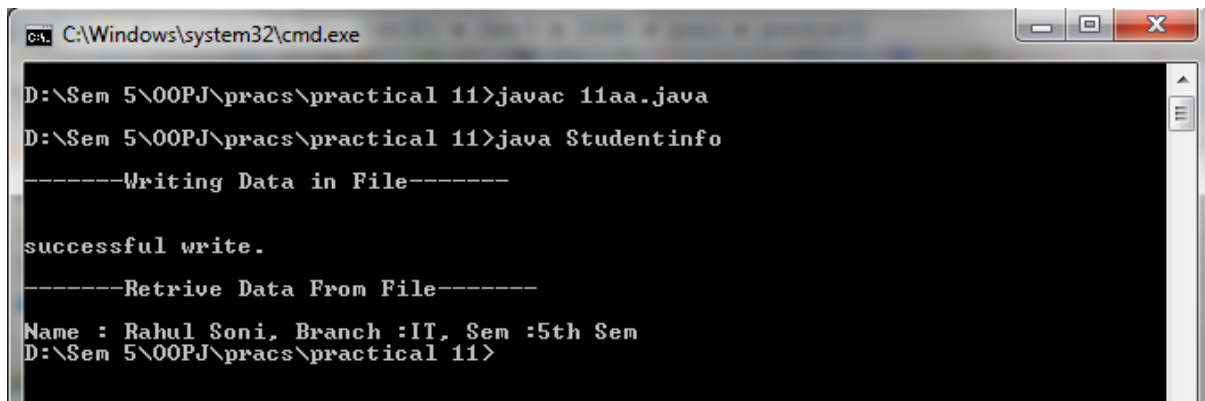
```
C:\Windows\system32\cmd.exe
D:\Sem 5\00PJ\pracs\practical 10>java Threadpractical
Thread[odd,5,main]
Thread[even,5,main]
1
3
5
7
9
11
13
15
17
19
2
4
6
8
10
12
14
16
18
D:\Sem 5\00PJ\pracs\practical 10>
```

PRACTICAL – 11

AIM: Create a class called Student. Write a student manager program to manipulate the student information from files by using FileInputStream and FileOutputStream.

```
import java.io.FileInputStream;
import java.io.FileOutputStream;
class Studentinfo {
public static void main(String[] args) {
System.out.println(" ");
System.out.println("-----Writing Data in File-----");
System.out.println(" ");
try {
FileOutputStream fout = new FileOutputStream("studentinfo.txt");
String str = "Name : Rahul Soni, Branch :IT, Sem :5th Sem";
byte b[] = str.getBytes();
fout.write(b);
fout.close();
System.out.println(" ");
System.out.println("successful write.");
}
catch (Exception e)
{
System.out.println();
}
System.out.println(" ");
System.out.println("-----Retrive Data From File-----");
try {
FileInputStream fin = new FileInputStream("studentinfo.txt"); int i = 0;
System.out.println(" ");
while ((i = fin.read()) != -1)
{
```

```
System.out.print((char) i);  
}  
fin.close();  
}  
catch (Exception e)  
{  
    System.out.println();  
}  
}  
}
```



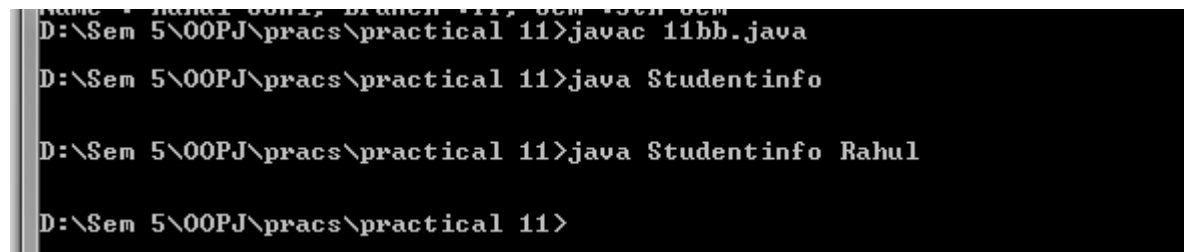
The screenshot shows a Windows command prompt window titled "C:\Windows\system32\cmd.exe". The user has navigated to the directory "D:\Sem 5\00PJ\pracs\practical 11". The commands and their outputs are as follows:

```
D:\Sem 5\00PJ\pracs\practical 11>javac 11aa.java  
D:\Sem 5\00PJ\pracs\practical 11>java Studentinfo  
-----Writing Data in File-----  
successful write.  
-----Retrive Data From File-----  
Name : Rahul Soni, Branch :IT, Sem :5th Sem  
D:\Sem 5\00PJ\pracs\practical 11>
```

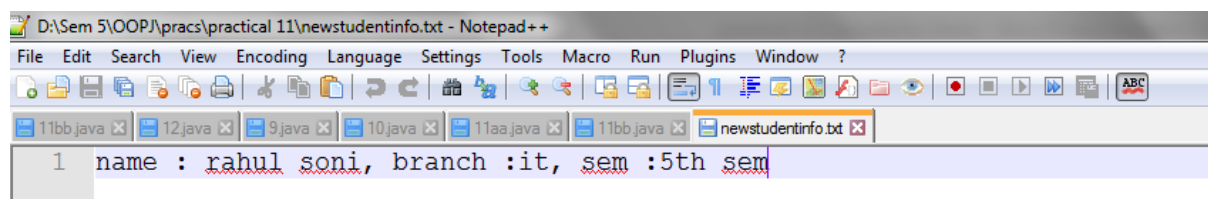
<11.2>

Aim: Refine the student manager program to manipulate the student information from files by using the BufferedReader and BufferedWriter.

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
class Studentinfo {
public static void main(String[] argv) throws Exception {
System.out.println(" ");
BufferedReader bufRead = new BufferedReader(new FileReader("studentinfo.txt"));
BufferedWriter bufWrite = new BufferedWriter(new FileWriter("newstudentinfo.txt"));
int i;
do {
i = bufRead.read();
if (i != -1) {
if (Character.isUpperCase((char) i)) { bufWrite.write(Character.toLowerCase((char) i));
} else { bufWrite.write((char) i);
}
}
} while (i != -1); bufRead.close(); bufWrite.close();
}
}
```

Output:

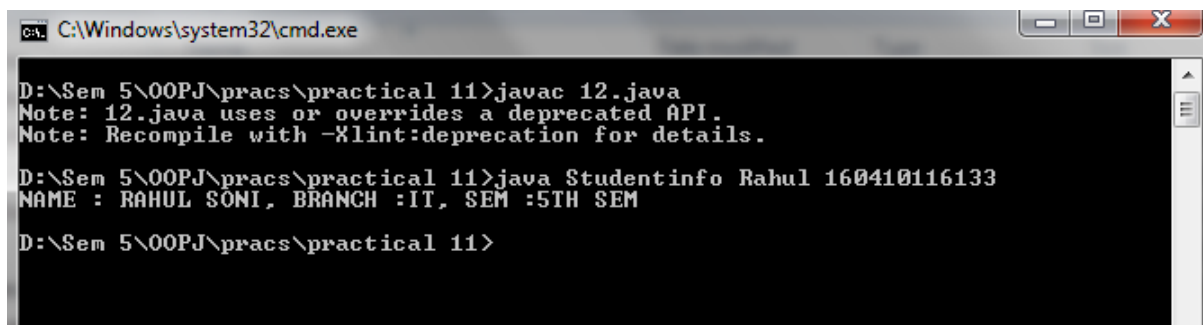
```
D:\Sem 5\OOPJ\pracs\practical 11>javac 11bb.java
D:\Sem 5\OOPJ\pracs\practical 11>java Studentinfo
D:\Sem 5\OOPJ\pracs\practical 11>java Studentinfo Rahul
D:\Sem 5\OOPJ\pracs\practical 11>
```



PRACTICAL -12

AIM: Refine the student manager program to manipulate the student information from files by using the DataInputStream and DataOutputStream. Assume suitable data.

```
import java.io.*;
class Studentinfo
{
    public static void main(String args[]) throws IOException
    { DataInputStream dataIS = new DataInputStream(new FileInputStream("studentinfo.txt"));
      DataOutputStream dataOS = new DataOutputStream(new FileOutputStream("newstudentinfo.txt"));
      String str;
      while ((str = dataIS.readLine()) != null)
      {
          String upper = str.toUpperCase();
          System.out.println(upper);
          dataOS.writeBytes(upper + " ");
      }
      dataIS.close();
      dataOS.close();
    }
}
```



```
C:\Windows\system32\cmd.exe

D:\Sem 5\OOPJ\pracs\practical 11>javac 12.java
Note: 12.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

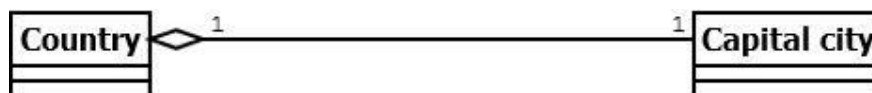
D:\Sem 5\OOPJ\pracs\practical 11>java Studentinfo Rahul 160410116133
NAME : RAHUL SONI, BRANCH :IT, SEM :5TH SEM

D:\Sem 5\OOPJ\pracs\practical 11>
```

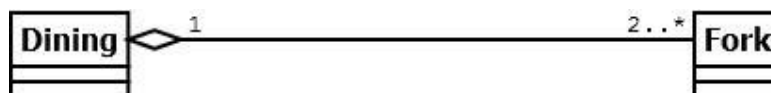

PRACTICAL - 13

AIM: Categorize the following relationships into generalization, aggregation or association.

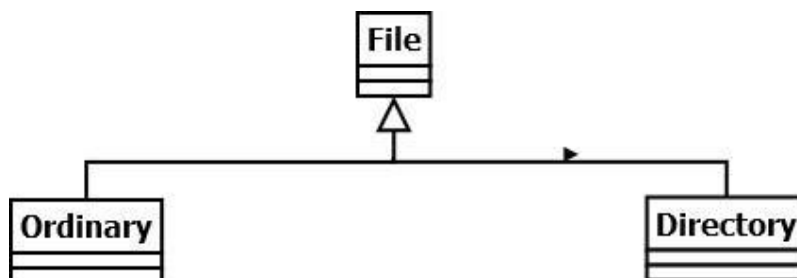
[A] A country has a capital city



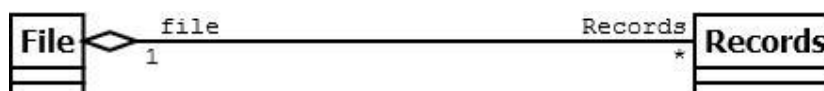
[B] A dining philosopher uses a fork



[C] A file is an ordinary file or a directory file



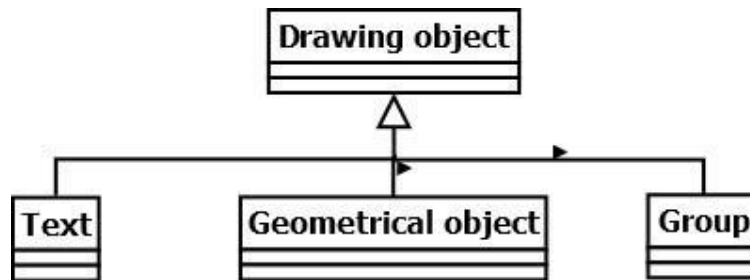
[D] Files contains records



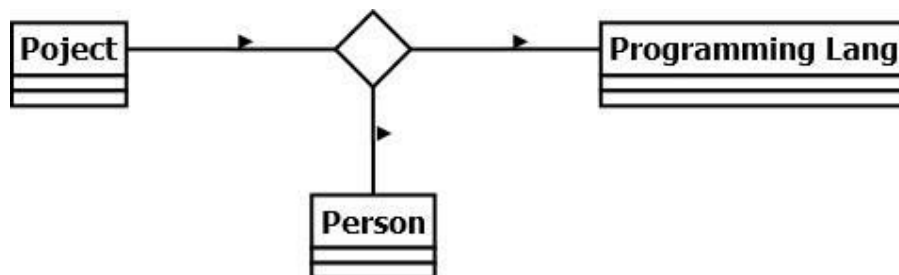
[E] A polygon is composed of an ordered set of points



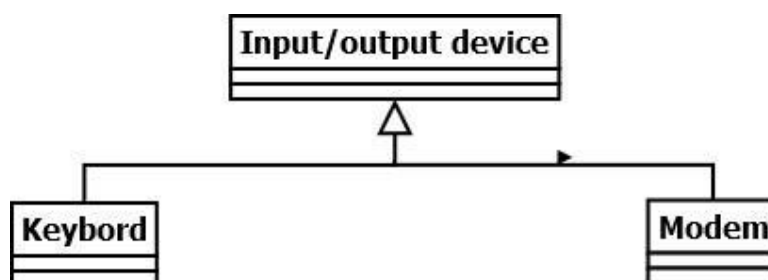
[F] A drawing object is text, a geometrical object, or a group



[G] A person uses a computer language on an object



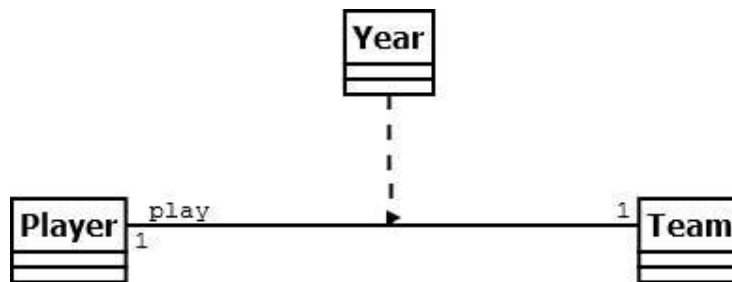
[H] Modems and keyboards are input/output devices



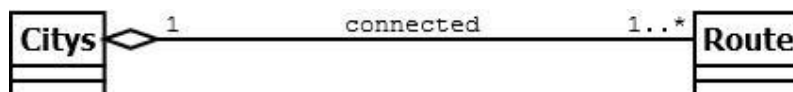
[I] Classes may have several attributes



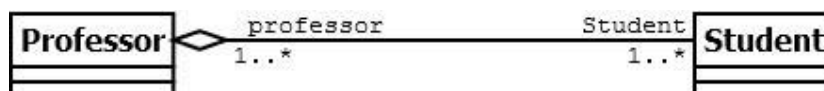
[J] A person plays for a team in a certain year



[K] A route connects two cities

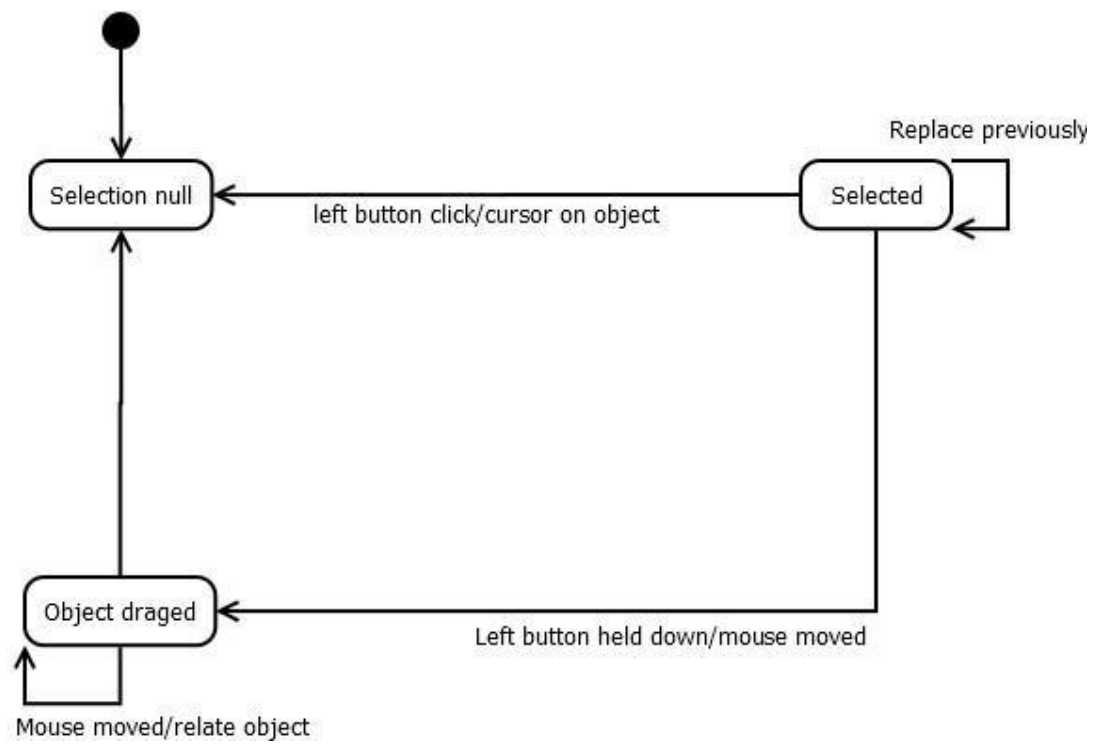


[L] A student takes a course from a professor.



PRACTICAL – 14

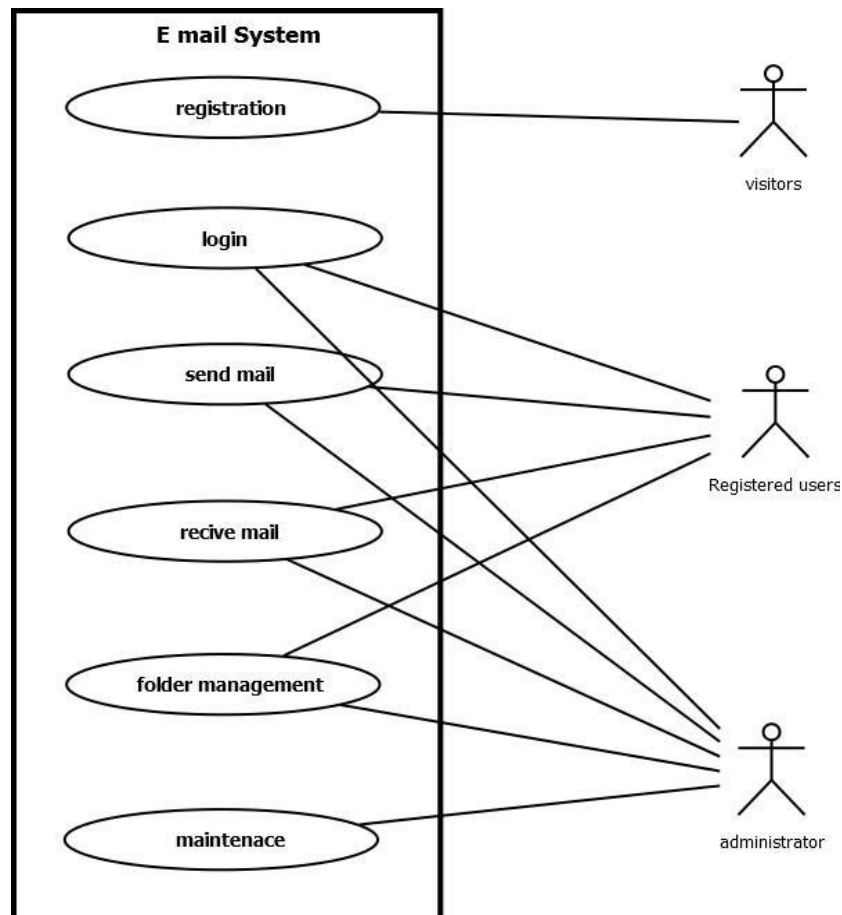
AIM: Prepare a state diagram for an interactive diagram editor for selecting and dragging objects.

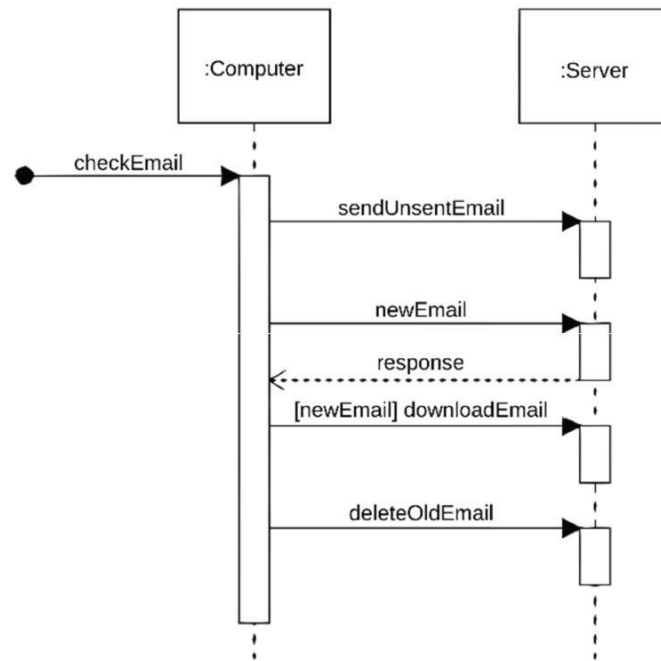


<14.2>

AIM: Prepare a use case diagram and sequence diagram for a computer email system.

OUTPUT

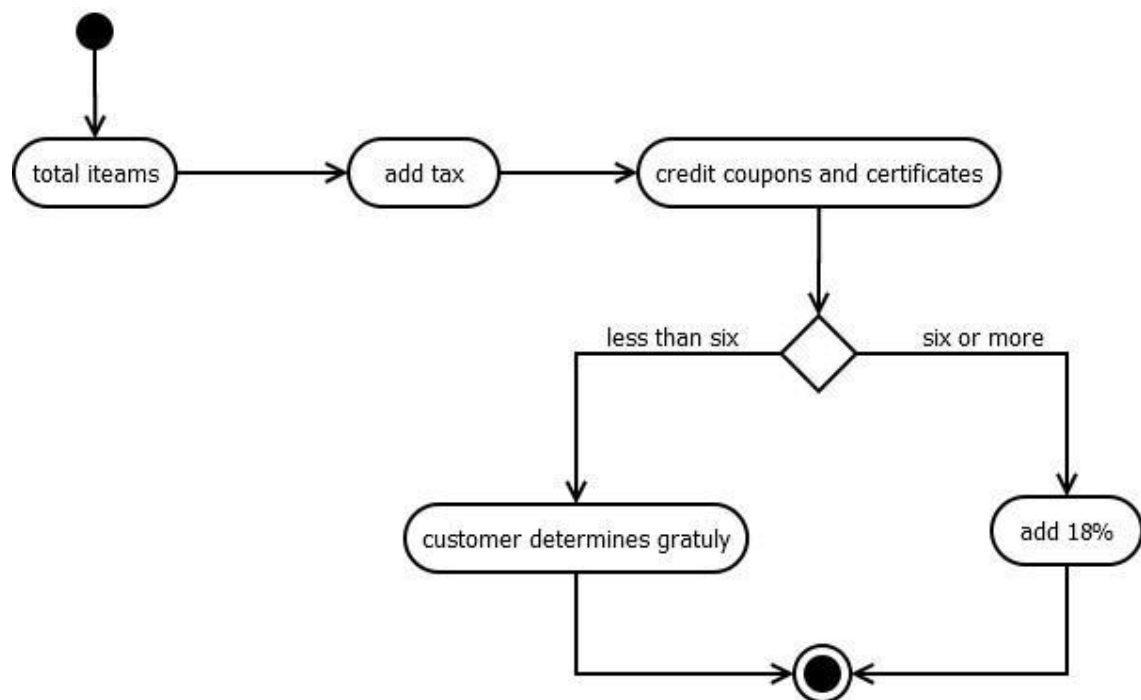




PRACTICAL - 15

AIM: Prepare an activity diagram for computing a restaurant bill, there should be charge for each delivered item. The total amount should be subject to tax and service charge of 18% for group of 6 and more. For smaller groups there should be a blank entry. Any coupons or gift certificates submitted by the customer should be subtracted.

OUTPUT



PRACTICAL -16

AIM: Prepare a sequence diagram for issuing a book in the library management system.

OUTPUT

