Slip 1

Q1) Write a Program to print all Prime numbers in an array of ‘n’ elements. (use command line arguments)

```
import java.util.Scanner;
public class PrimeNumbers
{
public static void main (String[] args){
    int[] array = new int [20];
   int n=Integer.parseInt(args[0]);
    Scanner in = new Scanner (System.in);
    System.out.println("Enter the elements of the array: ");
    for(int i=0; i<n; i++)
    {
        array[i] = in.nextInt();
    }
    for(int i=0; i<n; i++){
        boolean isPrime = true;
        for (int j=2; j<i; j++){
            if(i%j==0){
                isPrime = false;
                break;
            }
        }
        if(isPrime)
            System.out.println(i + " are the prime numbers in the array ");
    }
}
}
```

Q2) Define an abstract class Staff with protected members id and name. Define a parameterized constructor. Define one subclass OfficeStaff with member department. Create n objects of OfficeStaff and display all detail

```
import java.io.*;

abstract class Staff
{
String name,id;
}
```

```
class OfficeStaff extends Staff
{
String department;
double salary;
public void accept() throws IOException
{
System.out.println("Enter the name, id, department and salary: ");
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
name=br.readLine();
id=br.readLine();
department=br.readLine();
salary=Double.parseDouble(br.readLine());
}
public void display(){

System.out.println("Name: "+name);
System.out.println("Address: "+id);
System.out.println("Department: "+department);
System.out.println("Salary: "+salary);
System.out.println("----------------------");
}
}
public class StaffInfo
 {
public static void main(String [] args) throws IOException
{
int i;
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
System.out.println("Office Staff Information");
System.out.println("Enter the number of Staff: ");
int n=Integer.parseInt(br.readLine());
OfficeStaff [] l=new OfficeStaff[n];
for(i=0;i<n;i++)
{
l[i]=new OfficeStaff();
l[i].accept();
}
for(i=0;i<n;i++)
{
l[i].display();
}
```

```
}
}
```

Slip 2

Q.1]Write a program to read the First Name and Last Name of a person, his weight and height using command line arguments. Calculate the BMI Index which is defined as the individual's body mass divided by the square of their height. (Hint : BMI = Wts. In kgs / (ht)2)

```
public class BMI
 {
   public static void main(String[] args) throws Exception
 {
     float weight = Float.parseFloat(args[2]); //Weight
     float height= Float.parseFloat(args[3]); //Height
      System.out.println("First Name"+"  "+args[0]);
      System.out.println("Last Name"+"  "+args[1]);
      System.out.println("Weight"+"  "+args[2]);
      System.out.println("Height"+"  "+args[3]);

       float bmi = weight/((height*height)/10);
       System.out.println("Your BMI is: "+bmi);

       if(bmi < 18.5) {
          System.out.println("You are underweight");
       }else if (bmi < 25) {
          System.out.println("You are normal");
       }else if (bmi < 30) {
          System.out.println("You are overweight");
       }
      else {
          System.out.println("You are obese");
       }
   }
}
```

Q2) Define a class CricketPlayer (name,no_of_innings,no_of_times_notout, totatruns, bat_avg). Create an array of n player objects .Calculate the batting average for each player using static
method avg(). Define a static sort method which sorts the array on the basis of average. Display

the player details in sorted order

```
import java.io.*;
class Cricket {
String name;
int inning, tofnotout, totalruns;
float batavg;
public Cricket(){
name=null;
inning=0;
tofnotout=0;
totalruns=0;
batavg=0;
}
public void get() throws IOException{
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
System.out.println("Enter the name, no of innings, no of times not out,
total runs: ");
name=br.readLine();
inning=Integer.parseInt(br.readLine());
tofnotout=Integer.parseInt(br.readLine());
totalruns=Integer.parseInt(br.readLine());
}
public void put(){
System.out.println("Name="+name);
System.out.println("no of innings="+inning);
System.out.println("no times notout="+tofnotout);
System.out.println("total runs="+totalruns);
System.out.println("bat avg="+batavg);
}
static void avg(int n, Cricket c[]){
try{
for(int i=0;i<n;i++){
c[i].batavg=c[i].totalruns/c[i].inning;
}
}catch(ArithmeticException e){
System.out.println("Invalid arg");

}
}
static void sort(int n, Cricket c[]){
String temp1;
int temp2,temp3,temp4;
```

```
float temp5;
for(int i=0;i<n;i++){
for(int j=i+1;j<n;j++){
if(c[i].batavg<c[j].batavg){
temp1=c[i].name;
c[i].name=c[j].name;
c[j].name=temp1;
temp2=c[i].inning;
c[i].inning=c[j].inning;
c[j].inning=temp2;
temp3=c[i].tofnotout;
c[i].tofnotout=c[j].tofnotout;
c[j].tofnotout=temp3;
temp4=c[i].totalruns;
c[i].totalruns=c[j].totalruns;
c[j].totalruns=temp4;
temp5=c[i].batavg;
c[i].batavg=c[j].batavg;
c[j].batavg=temp5;
}
}
}
}
}
public class a4sa1 {
public static void main(String args[])throws IOException{
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
System.out.println("Enter the limit:");
int n=Integer.parseInt(br.readLine());
Cricket c[]=new Cricket[n];
for(int i=0;i<n;i++){
c[i]=new Cricket();
c[i].get();
}
Cricket.avg(n,c);
Cricket.sort(n, c);
for(int i=0;i<n;i++){
c[i].put();
}
}

}
```

Output:

```
Enter the limit:
2
Enter the name, no of innings, no of times not out, total runs:
Virat
3
1
100
Enter the name, no of innings, no of times not out, total runs:
Rohit
2
1
120
Name=Rohit
no of innings=2
no times notout=1
total runs=120
bat avg=60.0
Name=Virat
no of innings=3
no times notout=1
total runs=100
bat avg=33.0
```

Slip 3

Q.1] Write a program to accept 'n' name of cities from the user and sort them in ascending order.

```
import java.util.Scanner;
public class CitySort
{
   public static void main(String[] args)
   {
      int n;
      String temp;
      Scanner s = new Scanner(System.in);
      System.out.print("Enter number of names you want to enter:");
      n = s.nextInt();
      String names[] = new String[n];
      Scanner s1 = new Scanner(System.in);
      System.out.println("Enter all the names:");
      for(int i = 0; i < n; i++)
      {
         names[i] = s1.nextLine();
      }
      for (int i = 0; i < n; i++)
      {
         for (int j = i + 1; j < n; j++)
         {
            if (names[i].compareTo(names[j])>0)
            {
               temp = names[i];
               names[i] = names[j];
               names[j] = temp;
            }
         }
      }
      System.out.print("Names in Sorted Order:");
      for (int i = 0; i < n - 1; i++)
      {
         System.out.print(names[i] + ",");
      }
      System.out.print(names[n - 1]);
```

```
    }
}
```
Output:

```
$ javac CitySort.java
$ java CitySort

Enter number of names you want to enter:5
Enter all the names:
bryan
adam
rock
chris
scott
Names in Sorted Order:adam,bryan,chris,rock,scott
```

Q2) Define a class patient (patient_name, patient_age, patient_oxy_level,patient_HRCT_report). Create an object of patient. Handle appropriate exception while patient oxygen level less than 95% and HRCT scan report greater than 10, then throw user defined Exception “Patient is Covid Positive(+) and Need to Hospitalized” otherwise display its information.

```
import java.util.*;
class Patient
{
String name;
int age;
int oxylevel;
int HRCTreport;
patient(String name, int age, int oxylevel, int HRCTreport)
{
this.name = name;
this.age = age;
this.oxylevel = oxylevel;
this.HRCTreport = HRCTreport;
}
}
public class Main extends Exception
{
public static void main(String[] args)
{
Scanner sc = new Scanner(System.in);
System.out.println("How many patient you want insert:");
int number = sc.nextInt();
```

```
patient[] ob = new patient[number];
for(int j=0; j<number; j++)
{
System.out.println("Enter Name ");
String name = sc.next();
System.out.println("Enter Age ");
int age = sc.nextInt();
System.out.println("Enter oxygen level");
int oxylevel = sc.nextInt();
System.out.println("Enter HRCT report");
int HRCTreport = sc.nextInt();
ob[j] = new patient(name, age, oxylevel, HRCTreport);
}

for(int j=0; j<number; j++)
{
if(ob[j].oxylevel < 95 && ob[j].HRCTreport > 10)
try
{

throw new NullPointerException("\n");
}
catch(Exception e)
{
System.out.println("Patient is Covid Positive(+) and Need to
Hospitalized\

n");

}
else
{
System.out.println("name: "+ob[j].name);
System.out.println("age " + ob[j].age);
System.out.println("oxygen level " +ob[j].oxylevel);
System.out.println("HRCT report " + ob[j].HRCTreport);
System.out.println("\n");
}
}
}
}
```

Output:

How many patient you want insert:
2
Enter Name
abc
Enter Age
109
Enter oxygen level
70
Enter HRCT report
10
Enter Name
xyz
Enter Age
111
Enter oxygen level
60
Enter HRCT report
9
name: abc
age 109
oxygen level 70
HRCT report 10

name: xyz
age 111
oxygen level 60
HRCT report 9

Slip 4

Q1) Write a program to print an array after changing the rows and columns of a given
two-dimensional array.

```
import java.util.Scanner;
public class Transpose
{
   public static void main(String args[])
   {
        int i, j;
        System.out.println("Enter total rows and columns: ");
        Scanner s = new Scanner(System.in);
        int row = s.nextInt();
        int column = s.nextInt();
        int array[][] = new int[row][column];
        System.out.println("Enter matrix:");
        for(i = 0; i < row; i++)
        {
           for(j = 0; j < column; j++)
           {
        array[i][j] = s.nextInt();
        System.out.print(" ");
         }
        }
        System.out.println("The above matrix before Transpose is ");
        for(i = 0; i < row; i++)
        {
           for(j = 0; j < column; j++)
         {
        System.out.print(array[i][j]+" ");
         }
         System.out.println(" ");
      }
        System.out.println("The above matrix after Transpose is ");
        for(i = 0; i < column; i++)
        {
```

```
            for(j = 0; j < row; j++)
          {
             System.out.print(array[j][i]+" ");
          }
          System.out.println(" ");
       }
    }
}
```

Output:

```
$ javac Transpose.java
$ java Transpose

Enter total rows and columns:
3 3
Enter matrix:
1
2
3
4
5
6
7
8
9
The above matrix before Transpose is
1 2 3
4 5 6
7 8 9
The above matrix after Transpose is
1 4 7
2 5 8
3 6 9
```

Q2) Write a program to design a screen using Awt that will take a user name and password. If the user name and password are not same, raise an Exception with appropriate message. User can have 3 login chances only. Use clear button to clear the TextFields.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class InvalidPasswordException extends Exception
{
```

```
}
class Slip4 extends JFrame implements ActionListener
{
   JLabel name, pass;
   JTextField nameText;
   JPasswordField passText;
   JButton login, end;
   static int cnt=0;
      Slip4()
   {
       name = new JLabel("Name : ");
       pass = new JLabel("Password : ");
       nameText = new JTextField(20);
       passText = new JPasswordField(20);
       login = new JButton("Login");
       end = new JButton("End");
       login.addActionListener(this);
        end.addActionListener(this);
                        setLayout(new GridLayout(3,2));
        add(name);
        add(nameText);
        add(pass);
        add(passText);
        add(login);
        add(end);
        setTitle("Login Check");
        setSize(300,300);
                             setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }
    public void actionPerformed(ActionEvent e)
    {
            if(e.getSource()==end)
        {
                    System.exit(0);

        }
        if(e.getSource()==login)
        {
          try
          {
             String user = nameText.getText();
             String pass = new String(passText.getPassword());
                          if(user.compareTo(pass)==0)
```

```
            {
JOptionPane.showMessageDialog(null,"Login
Successful","Login",JOptionPane.INFORMATION_MESSAGE);
                System.exit(0);
System.out.println("login successful");
             }
             else
             {
                throw new InvalidPasswordException();
             }
            }
            catch(Exception e1)
            {

        cnt++;
              JOptionPane.showMessageDialog(null,"Login
Failed","Login",JOptionPane.ERROR_MESSAGE);
                nameText.setText("");
                passText.setText("");
                nameText.requestFocus();
                if(cnt == 3)
               {
        JOptionPane.showMessageDialog(null,"3
AttemptsOver","Login",JOptionPane.ERROR_MESSAGE);
                  System.exit(0);

            }
       }
  }
}
 public static void main(String args[])
{
            new Slip4();
}
}
```

Slip 5

Q.1] Write a program for multilevel inheritance such that Country is inherited from Continent. State is inherited from Country. Display the place, State, Country and Continent.

```
import java.io.InputStreamReader;
import java.io.BufferedReader;
import java.io.IOException;
class Continent
{
String con;
InputStreamReader i = new InputStreamReader(System.in);
BufferedReader r = new BufferedReader(i);
void con_input() throws IOException
{
System.out.println("Enter Continent Name: ");
con = r.readLine();
}
}
class Country extends Continent
{
String cou ;
void cou_input() throws IOException
{
System.out.println("Enter Country Name: ");
cou = r.readLine();
}
}
class State extends Country
{
String sta;
void sta_input() throws IOException
{
System.out.println("Enter State Name: ");
sta = r.readLine();
}
}
```

```
class Main extends State
{
String pla;
void pla_input()throws IOException
{
System.out.println("Enter Place Name : ");
pla = r.readLine();

}

public static void main( String argsp[] )throws IOException
{
Main s = new Main();
s.con_input();
s.cou_input();
s.sta_input();
s.pla_input();
System.out.println("\n\nContinent: "+s.con);
System.out.println("Country: "+s.cou);
System.out.println("State: "+s.sta);
System.out.println("Place :" + s.pla);
}
}
```

Output:

```
Enter Continent Name:
Asia
Enter Country Name:
India
Enter State Name:
Maharashtra
Enter Place Name :
Pune

Continent: Asia
Country: India
State: Maharashtra
Place :Pune
```

Q.2] Write a menu driven program to perform the following operations on multidimensional array
ie matrices :
 Addition

 Multiplication
 Exit

```
import java.util.Scanner;
public class Matrix
{
   public static void main(String args[])
   {
      //Scanner class to take input
      Scanner scan = new Scanner(System.in);
      int row, col;
      int mat1[][] = new int[3][3];
      int mat2[][] = new int[3][3];
      int add[][] = new int[3][3];
      int prod[][] = new int[3][3];

      //Entering first matrix
      System.out.println("Enter the 3x3 matrix elements for 1st matrix : ");
      // Loop to take array elements as user input for first matrixn i.e
mat1
      for(row=0;row<3;row++)
         for(col=0;col<3;col++)
            mat1[row][col] = scan.nextInt();

      //print the elements of first matrix i.e mat1
      System.out.print("1st matrix : ");
      for(row=0;row<3;row++)
      {
         // Used for formatting
         System.out.print("\n");
         for(col=0;col<3;col++)
         {
            System.out.print(mat1[row][col]+" ");
         }
      }
      //Entering second matrix
      System.out.println("\nEnter the 3x3 matrix elements for 2nd matrix :
");
      // Loop to take array elements as user input for second matrix
      for(row=0;row<3;row++)
         for(col=0;col<3;col++)
            mat2[row][col] = scan.nextInt();

      //print the elements of second matrix i.e mat2
```

```
System.out.print("2nd matrix : ");
for(row=0;row<3;row++)
{
   // Used for formatting
   System.out.print("\n");
   for(col=0;col<3;col++)
   {
      System.out.print(mat2[row][col]+" ");
   }
}
int res[][] = new int[3][3], operationHolder = 0;
int choice ;
while(true)
{
   //Prints the menu to choose operation from
   System.out.println("\n\nBASIC MATRIX OPERATIONS");
   System.out.println("_____________________");
   System.out.println("1. Addition of two matrices");
   System.out.println("2. Multiplication of two matrices");
   System.out.println("3. Exit");
   System.out.println("_____________________");
   System.out.println("Enter your choice : ");
   choice = scan.nextInt();
   // Switch cases to run the menu
   switch(choice)
   {
      case 1:   System.out.print("Addition of matrix : ");
                for(row=0;row<3;row++)
                {
                  // Used for formatting
                 System.out.print("\n");
                for(col=0;col<3;col++)
               {
                 add[row][col]=mat1[row][col] + mat2[row][col];
                 System.out.print(add[row][col]+" ");
               }
              }
            break;
      case 2: System.out.print("Addition of matrix : ");
                for(row=0;row<3;row++)
                {
                  // Used for formatting
                 System.out.print("\n");
                for(col=0;col<3;col++)
```

```
                    {
                      prod[row][col] = 0;
                      for(int i = 0; i<3; i++)
                      prod[row][col]+=mat1[row][i]*mat2[i][col];
                      System.out.print(prod[row][col]+" ");
                    }
                   }
                  break;
            case 3: System.out.println("Exited from the program");
                  return;
            default: System.out.println("Wrong input, please try again!!");
          }
        }

    }

}
```

Slip 6

Q.1) Write a program to display the Employee(Empid, Empname, Empdesignation, Empsal)
information using toString().

```
class Employee
{
   int id,salary;
   String name;
   String desig;
   Employee(int id, String name, int salary ,String desig)
   {
       this.id=id;
       this.name=name;
       this.salary=salary;
       this.desig=desig;
   }
   public String toString() // overrides toString() method
   {
      return id+" "+name+" "+salary+" "+desig;
   }
   public static void main(String args[])
   {
      Employee E1=new Employee(111,"Rakesh",50000,"Manager");
      Employee E2=new Employee(112,"Suresh",25000,"Engineer");
      // both will print Employee.toString()
      System.out.println("Employee details: "+E1);
      System.out.println("Employee details: "+E2);
    }
}
```

Q2) Create an abstract class “order” having members id, description. Create two subclasses
“PurchaseOrder” and “Sales Order” having members customer name and Vendor name

respectively. Define methods accept and display in all cases. Create 3 objects each of Purchase
Order and Sales Order and accept and display details

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
abstract class Order{
String id,description;
}
class PurchaseOrder extends Order{
String Customername,Vendorname;
public void accept() throws IOException{
System.out.println("Enter the id,description,names of customers and
vendors: ");
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
id=br.readLine();
description=br.readLine();
Customername=br.readLine();
Vendorname=br.readLine();
}
public void display(){
System.out.println("id: "+id);
System.out.println("Description: "+description);
System.out.println("Customername: "+Customername);
System.out.println("Vendorname: "+Vendorname);
System.out.println("----------");
}
}
class SalesOrder extends Order{
String Customername,Vendorname;
public void accept() throws IOException{
System.out.println("Enter the id,description,names of customers and
vendors: ");
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
id=br.readLine();
description=br.readLine();
Customername=br.readLine();
Vendorname=br.readLine();
}
public void display(){
System.out.println("id: "+id);
System.out.println("Description: "+description);
```

```
System.out.println("Customername: "+Customername);
System.out.println("Vendorname: "+Vendorname);
System.out.println("----------");
}
}
public class Main {
public static void main(String [] args) throws IOException{

int i;
System.out.println("Select Any One: ");
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
System.out.println("1.Purchase Order");
System.out.println("2.Sales Order");
int ch=Integer.parseInt(br.readLine());
switch(ch){
case 1:
System.out.println("Enter the number of purchase Orders: ");
int n=Integer.parseInt(br.readLine());
PurchaseOrder [] l=new PurchaseOrder[n];
for(i=0;i<n;i++){
l[i]=new PurchaseOrder();
l[i].accept();
}
for(i=0;i<n;i++){
l[i].display();
System.out.println ("Object is created");
}
break;
case 2:
System.out.println("Enter the number of sales orders: ");
int m=Integer.parseInt(br.readLine());
SalesOrder [] h=new SalesOrder[m];
for(i=0;i<m;i++){
h[i]=new SalesOrder();
h[i].accept();
}
for(i=0;i<m;i++){
h[i].display();
System.out.println(" Object is created ");
}
break;

}
```

```
}
}
```

Output:

```
Select Any One:
1.Purchase Order
2.Sales Order
1
Enter the number of purchase Orders:
2
Enter the id,description,names of customers and vendors:
1
Soap
Rahul
Gajni
Enter the id,description,names of customers and vendors:
2

Handwash
prince
abhishek
id: 1
Description: Soap
Customername: Rahul
Vendorname: Gajni
----------
Object is created
id: 2
Description: Handwash
Customername: prince
Vendorname: abhishek
----------
Object is created
```

Slip 8

Q1) Create a class Sphere, to calculate the volume and surface area of sphere.
(Hint : Surface area=4*3.14(r*r), Volume=(4/3)3.14(r*r*r))

```
class surfaceareaandvolume {
    public static void main(String[] args)
    {
        double r = 5.0, surfacearea = 0.0, volume = 0.0;
        surfacearea = 4 * 3.14 * (r * r);
        volume = ((double)4 / 3) * 3.14 * (r * r * r);

        System.out.println("surfacearea of sphere ="
                           + surfacearea);

        System.out.println("volume of sphere =" + volume);
    }
}
```

Q2) Design a screen to handle the Mouse Events such as MOUSE_MOVED
and MOUSE_CLICKED and display the position of the Mouse_Click in a TextField.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class MyFrame extends JFrame
{
        JTextField t,t1;
        JLabel l,l1;
        int x,y;
```

```
	Panel p;
	MyFrame(String title)
	{
		super(title);
		setLayout(new FlowLayout());

		p=new Panel();
		p.setLayout(new GridLayout(2,2,5,5));
		t=new JTextField(20);
		l= new JLabel("Co-ordinates of clicking");
		l1= new JLabel("Co-ordinates of movement");
		t1=new JTextField(20);
		p.add(l);
		p.add(t);
		p.add(l1);
		p.add(t1);
		add(p);
		addMouseListener(new MyClick());
		addMouseMotionListener(new MyMove());
		setSize(500,500);
		setVisible(true);
	}
	class MyClick extends MouseAdapter
	{
		public void mouseClicked(MouseEvent me)
		{
			x=me.getX();
			y=me.getY();
			t.setText("X="+x+" Y="+y);
		}
	}
	class MyMove extends MouseMotionAdapter
	{
		public void mouseMoved(MouseEvent me)
		{
			x=me.getX();
			y=me.getY();
			t1.setText("X="+ x +" Y="+y);
		}
	}
}
class Slip4
{
	public static void main(String args[])
```

```
        {
                MyFrame f = new MyFrame("Slip Number 4");
        }
}
```

Slip 10

Q1) Write a program to find the cube of given number using functional interface.

```
import java.util.*;
import java.lang.*;
class CubeCal
{
   public static void main (String[] args)
   {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number: ");
        int num = sc.nextInt();

        //Calculating the cube of the number
        int cube = (int) Math.pow(num,3);

        System.out.println("Cube of the number "+num+" is "+cube+" .");

   }
}
```

Q2) Write a program to create a package name student. Define class StudentInfo with method to
display information about student such as rollno, class, and percentage. Create another class
StudentPer with method to find percentage of the student. Accept student details like
rollno, name, class and marks of 6 subject from user.

```
package student;
```

```
public class StudentInfo
{
    public int r_no;
    public String name;
    public String class;
    public int a,b,c,d,e,f;
    int total=0;
    public Student(int roll, String nm, int m1,int m2,int m3,int m4,int m5,int
m6)
    {
        r_no = roll;
        name = nm;
        a = m1;
        b = m2;
        c = m3;
        d=m4;
        e=m5;
        f=m6;
       total= a+b+c+d+e+f;
    }
    public void displayper()
    {
        System.out.println("Roll_no : "+r_no);
        System.out.println("Name    : "+name);
        System.out.println("-----MARKS-------");
        System.out.println("Sub 1     : "+a);
        System.out.println("Sub 2     : "+b);
        System.out.println("Sub 3     : "+c);
        System.out.println("Sub 4     : "+d);
        System.out.println("Sub 5     : "+e);
        System.out.println("Sub 6     : "+f);

        System.out.println("Total     : "+total);
        System.out.println("percentage: "+total/3);
        System.out.println("-----------------");
    }
}

import student.StudentInfo;
import java.util.*;
import java.lang.*;
import java.io.*;
class StudentPer
{
```

```
    public static void main(String[] args)
    {
        String nm;
        int roll;
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter Roll no:= ");
        roll = sc.nextInt();
        System.out.print("Enter Name:= ");
        nm = sc.next();
        int m1,m2,m3,m4,m5,m6;
        System.out.print("Enter 6 sub mark:= ");
        m1 = sc.nextInt();
        m2 = sc.nextInt();
        m3 = sc.nextInt();
        m4=sc.nextInt();
        m5=sc.nextInt();
        m6=sc.nextInt();
        Student s = new Student( roll,nm,m1,m2,m3,m4,m5,m6);
        s.displayper();
    }
}
```

Slip 11

Q.1]Define an interface “Operation” which has method volume( ).Define a constant PI having a value
3.142 Create a class cylinder which implements this interface (members – radius,height). Create
one object and calculate the volume.

```
import java.io.*;
interface Operation
{
  double PI=3.14;
  void volume();
}
```

```
class Cylinder implements Operation
{


        public void volume()
        {
Scanner s= new Scanner(System.in);
        System.out.println("Enter the radius:");
        double r=s.nextDouble();
        System.out.println("Enter the height:");
        double h=s.nextDouble();
          double  vol=((22*r*r*h)/7);
           System.out.println("volume of Cylinder is: " +vol);
         }
  }
class CylinderVol
{
  public static void main(String args[])
   {
      Cylinder obj=new Cylinder();
      obj.volume();
   }
}
```

Q2) Write a program to accept the username and password from user if username and password are
not same then raise "Invalid Password" with appropriate msg.

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
class InvalidPasswordException extends Exception
{
}
class Slip11 extends JFrame implements ActionListener
{
   JLabel name, pass;
   JTextField nameText;
   JPasswordField passText;
   JButton login, end;
   static int cnt=0;
      Slip11()
   {
```

```
        name = new JLabel("Name : ");
        pass = new JLabel("Password : ");
        nameText = new JTextField(20);
        passText = new JPasswordField(20);
        login = new JButton("Login");
        end = new JButton("End");
        login.addActionListener(this);
         end.addActionListener(this);
         setLayout(new GridLayout(3,2));
         add(name);
         add(nameText);
         add(pass);
         add(passText);
         add(login);
         add(end);
         setTitle("Login Check");
         setSize(300,300);
                              setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
         setVisible(true);
    }
    public void actionPerformed(ActionEvent e)
    {
            if(e.getSource()==end)
        {
                    System.exit(0);

        }
        if(e.getSource()==login)
        {
          try
          {
             String user = nameText.getText();
             String pass = new String(passText.getPassword());
                          if(user.compareTo(pass)==0)
             {
JOptionPane.showMessageDialog(null,"Login
Successful","Login",JOptionPane.INFORMATION_MESSAGE);
                System.exit(0);
System.out.println("login successful");
              }
              else
              {
                 throw new InvalidPasswordException();
              }
```

```
            }
            catch(Exception e1)
            {

          cnt++;
                JOptionPane.showMessageDialog(null,"Login
Failed","Login",JOptionPane.ERROR_MESSAGE);
                nameText.setText("");
                passText.setText("");
                nameText.requestFocus();

        }
   }
}
  public static void main(String args[])
 {
            new Slip11();
 }
}
```

Slip 14

Q.1]Write a program to accept a number from the user, if number is zero then throw user defined
exception “Number is 0” otherwise check whether no is prime or not (Use static keyword).

```
import java.io.*;
class NumberZeroException extends Exception
{
   public String toString()
   {
      return("Number is 0");
   }
}

class PrimeNumber
{
   static int no;
   BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
   PrimeNumber()
   {
      try
      {
         System.out.println("Enter any integer to check prime ");
         no=Integer.parseInt(br.readLine());
         if(no==0)
            throw new NumberZeroException();
      }
```

```
        catch(NumberZeroException ex)
        {
            System.out.println(ex);
        }
        catch(IOException ex1)
        {
            System.out.println("Enter proper number");
        }
    }

    public void prime()
    {
        int cnt=0;
        for(int i=2;i<=no/2;i++)
            if(no%i==0)
            {
                cnt++;
                break;
            }
        if(cnt==0)
            System.out.println(a+" Number is prime");
        else
            System.out.println(a+" Number is not prime");
    }
    public static void main(String args[])
    {
        PrimeNumber pn=new PrimeNumber();
        pn.prime();
    }
}
```

Q2) Write a Java program to create a Package “SY” which has a class SYMarks (members –
ComputerTotal, MathsTotal, and ElectronicsTotal). Create another package TY which has a
class TYMarks (members – Theory, Practicals). Create ‘n’ objects of Student class (having
rollNumber, name, SYMarks and TYMarks). Add the marks of SY and TY computer subjects
and calculate the Grade (‘A’ for >= 70, ‘B’ for >= 60 ‘C’ for >= 50, Pass Class for > =40
else‘FAIL’) and display the result of the student in proper format.

```
package SY;
```

```
public class symark
{
	int ct,mt,et;
	public symark()
	{
		ct=0;mt=0;et=0;
	}
	public symark(int c,int m,int e)
	{
		this.ct=c;
		this.mt=m;
		this.et=e;
	}

}
```

package ty

```
package TY;
public class tymark
{
	int tt,pt;
	public tymark()
	{
		tt=0;pt=0;
	}
	public tymark(int t,int p)
	{
		this.tt=t;
		this.pt=p;
	}
}
```

prog

```
import java.io.*;
import SY.symark;
import TY.tymark;

public class set2b2
{
	private int rno;
	private String name;
	private symark sy1;
```

```
	private tymark ty1;
	private String grade;

	public set2b2()
	{
		rno=0;
		name="";
		sy1=new symark();
		ty1=new tymark();
		grade="";
	}
	public set2b2(int r,String na,int ct1,int mt1,int et1,int t1,int p,String g)
	{
		rno=r;
		name=na;
		sy1=new symark(ct1,mt1,et1);
		ty1=new tymark(t1,p);
		grade=g;

	}
	public void display(int ct1,int mt1,int et1,int t1,int p,String g)
	{
		System.out.println("roll no\t"+rno);
		System.out.println("name\t"+name);
		System.out.println("total of computer for sy:  "+ct1);
		System.out.println("total of maths for sy:  "+mt1);
		System.out.println("total of electronic for sy:  "+et1);
		System.out.println("total of theory for ty:  "+t1);
		System.out.println("total of practicals for ty:  "+p);
		System.out.println("grade:\n"+g);
	}

public static void main(String args[]) throws IOException
	{
		String g;
		BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
		System.out.println("How many records to be created:");
		int n=Integer.parseInt(br.readLine());
		set2b2 s[]=new set2b2[n+1];
		for(int i=0;i<n;i++)
		{
			System.out.println("Enter details of record no.:"+(i+1));
			System.out.println("Roll Number:");
			int r=Integer.parseInt(br.readLine());
			System.out.println("Name:");
			String na=br.readLine();
```

```
                System.out.println("Computer Total of SY:");
                int ct1=Integer.parseInt(br.readLine());
                System.out.println("Maths Total of SY:");
                int mt1=Integer.parseInt(br.readLine());
                System.out.println("Electronics Total of SY:");
                int et1=Integer.parseInt(br.readLine());
                System.out.println("Theory Total of TY:");
                int t1=Integer.parseInt(br.readLine());
                System.out.println("Practical Total of TY");
                int p=Integer.parseInt(br.readLine());
                double avg=((ct1+mt1+et1+t1+p)/5);
                if (avg>=70.0)
                        g="A";
                else if(avg<70.0 && avg>=60.0)
                        g="B";
                else if(avg<60.0 && avg>=50.0)
                        g="C";
                else if(avg<50.0 && avg>=40.0)
                        g="Pass Class";
                else
                        g="Fail";
                       s[i]=new set2b2(r,na,ct1,mt1,et1,t1,p,g);
                s[i].display(ct1,mt1,et1,t1,p,g);
            }
        }
}
/*
OUTPUT =
How many records to be created:
2
Enter details of record no.:1
Roll Number:
509
Name:
NADAF
Computer Total of SY:
89
Maths Total of SY:
75
Electronics Total of SY:
74
Theory Total of TY:
222
Practical Total of TY
247
roll no         509
nameNADAF
```

total of computer for sy: 89
total of maths for sy: 75
total of electronic for sy: 74
total of theory for ty: 222
total of practicals for ty: 247
grade:
A
Enter details of record no.:2
Roll Number:
506
Name:
MALI
Computer Total of SY:
90
Maths Total of SY:
58
Electronics Total of SY:
76
Theory Total of TY:
216
Practical Total of TY
220
roll no 506
nameMALI
total of computer for sy: 90
total of maths for sy: 58
total of electronic for sy: 76
total of theory for ty: 216
total of practicals for ty: 220
grade:
A
*/

Slip 16

Write a program to find the Square of given number using function interface.

```
import java.util.*;
import java.lang.*;
class SquareCal
{
   public static void main (String[] args)
   {
         Scanner sc = new Scanner(System.in);
         System.out.println("Enter the number: ");
         int num = sc.nextInt();

         //Calculating the cube of the number
         int sqr = (int) Math.pow(num,2);

         System.out.println("Cube of the number "+num+" is "+sqr+" .");

   }
}
```

Q2) Write a program to design a screen using Awt that,

```
import java.awt.*;
class MenuExample
{
    MenuExample(){
       Frame f= new Frame("Menu and MenuItem Example");
       MenuBar mb=new MenuBar();
```

```
        Menu menu=new Menu("File");
        Menu menu1=new Menu("Edit");
        Menu menu2=new Menu("About");
        MenuItem i1=new MenuItem("New");
        MenuItem i2=new MenuItem("Open");
        MenuItem i3=new MenuItem("Save");
        MenuItem i4=new MenuItem("Show About");
        MenuItem i5=new MenuItem("Exit");
        menu.add(i1);
        menu.add(i2);
        menu.add(i3);
        menu.add(i4);
        menu.add(i5);
        mb.add(menu);
        mb.add(menu1);
        mb.add(menu2);

        f.setMenuBar(mb);
        f.setSize(400,400);
        f.setLayout(null);
        f.setVisible(true);
}
public static void main(String args[])
{
new MenuExample();
}
}
```

Slip 17

Design a Super class Customer (name, phone-number). Derive a class Depositor(accno , balance) from Customer. Again, derive a class Borrower (loan-no, loan-amt) from Depositor. Write necessary member functions to read and display the details of ‘n’customers.

```
import java.io.*;

class cust
{
 String name;
 long pno;
 double bal;
 double l_bal;
 void get()
 {
 BufferedReader bc=new BufferedReader(new
InputStreamReader(System.in));
   try
   {
  System.out.print("\nEnter Customer Name: ");
  name=bc.readLine();
  System.out.print("\nEnter Customer Phone No.: ");
  pno=Long.parseLong(bc.readLine());
   }
    catch(Exception e)  {  }
 }
 void put()
 {
 System.out.println("\tCustomer Details: ");
 System.out.println("\tName\tPhone No.");
```

```
System.out.println("\t=======================================");
 System.out.println("\t"+name+"\t"+pno);
 }
}
   class deposit extends cust
   {
     int acno;

       void get_d()
       {
       BufferedReader bd=new BufferedReader(new
InputStreamReader(System.in));
         try
         {
         System.out.print("\nEnter Account No.: ");
         acno=Integer.parseInt(bd.readLine());
         System.out.print("\nEnter Account Balance.: ");
         bal=Double.parseDouble(bd.readLine());
         }
          catch(Exception e)  {  }
       }

       void put_d()
       {
        bal=bal-l_bal;
       System.out.println("\n\tDepositer Details: ");
        System.out.println("\tA/C No.\tBalance");

System.out.println("\t=======================================");
       System.out.println("\t"+acno+"\t"+bal);
        }
   }

   class borrower extends deposit
   {
     int l_no;

       void get_b()
       {
       get();
       get_d();

       BufferedReader b=new BufferedReader(new
InputStreamReader(System.in));
         try
         {
```

```
        System.out.print("\nEnter Loan No.: ");
        l_no=Integer.parseInt(b.readLine());
        System.out.print("\nEnter Loan Balance.: ");
        l_bal=Double.parseDouble(b.readLine());

         }
          catch(Exception e)  {  }
       }

       void put_b()
       {
       put();
       put_d();
       System.out.println("\n\tBorrower Details: ");
       System.out.println("\tLoan No.\tLoanBalance");

System.out.println("\t=======================================");
       System.out.println("\t"+l_no+"\t"+l_bal);
       }
   }
  class Bank
  {
    public static void main(String args[])
    {
     int n,i;
     BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
         try
         {
        System.out.print("\nHow many customer information enterd: ");
        n=Integer.parseInt(br.readLine());
        System.out.print("\nEnter "+n+" Customer Information: ");
        borrower obj[]=new borrower[n];
      for(i=0;i<n;i++)
      {
      obj[i]=new borrower();
      obj[i].get_b();
      }
      for(i=0;i<n;i++)
      {
       obj[i].put_b();
      }
        }
        catch(Exception e)   {    }
   }
 }
```

```
/*
How many customer information enterd: 2

Enter 2 Customer Information:
Enter Customer Name: Abc

Enter Customer Phone No.: 9076543566

Enter Account No.: 123

Enter Account Balance.: 20000

Enter Loan No.: 12

Enter Loan Balance.: 10000

Enter Customer Name: Pqr

Enter Customer Phone No.: 9876456733

Enter Account No.: 111

Enter Account Balance.: 50000

Enter Loan No.: 11

Enter Loan Balance.: 25000
        Customer Details:
        Name    Phone No.
        ========================================
        Abc     9076543566

        Depositer Details:
        A/C No. Balance
        ========================================
        123     10000.0

        Borrower Details:
        Loan No.        LoanBalance
        ========================================
        12      10000.0
        Customer Details:
        Name    Phone No.
        ========================================
        Pqr     9876456733
```

```
        Depositer Details:
        A/C No. Balance
        ========================================
        111     25000.0

        Borrower Details:
        Loan No.        LoanBalance
        ========================================
        11      25000.0

*/
```

Slip 18

Q.1] Write a program to implement Border Layout Manager.

```
import java.awt.*;
import javax.swing.*;

public class Border
{
JFrame f;
Border()
{
   f = new JFrame();

    // creating buttons
   JButton b1 = new JButton("NORTH");; // the button will be labeled as
NORTH
   JButton b2 = new JButton("SOUTH");; // the button will be labeled as
SOUTH
   JButton b3 = new JButton("EAST");; // the button will be labeled as
EAST
   JButton b4 = new JButton("WEST");; // the button will be labeled as
WEST
   JButton b5 = new JButton("CENTER");; // the button will be labeled as
CENTER

   f.add(b1, BorderLayout.NORTH); // b1 will be placed in the North
Direction
   f.add(b2, BorderLayout.SOUTH);  // b2 will be placed in the South
Direction
   f.add(b3, BorderLayout.EAST);  // b2 will be placed in the East
Direction
   f.add(b4, BorderLayout.WEST);  // b2 will be placed in the West
```

```
Direction
   f.add(b5, BorderLayout.CENTER);  // b2 will be placed in the Center

   f.setSize(300, 300);
   f.setVisible(true);
}
public static void main(String[] args) {
   new Border();
}
}
```

Q.2] Define a class CricketPlayer (name,no_of_innings,no_of_times_notout, totatruns, bat_avg). Create an array of n player objects. Calculate the batting average for each player using static method avg(). Define a static sort method which sorts the array on the basis of average. Display the player details in sorted order.

```
import java.io.*;
class Cricket {
String name;
int inning, tofnotout, totalruns;
float batavg;
public Cricket(){
name=null;
inning=0;
tofnotout=0;
totalruns=0;
batavg=0;
}
public void get() throws IOException{
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
System.out.println("Enter the name, no of innings, no of times not out,
total runs: ");
name=br.readLine();
inning=Integer.parseInt(br.readLine());
tofnotout=Integer.parseInt(br.readLine());
totalruns=Integer.parseInt(br.readLine());
}
public void put(){
System.out.println("Name="+name);
System.out.println("no of innings="+inning);
System.out.println("no times notout="+tofnotout);
System.out.println("total runs="+totalruns);
System.out.println("bat avg="+batavg);
```

```
}
static void avg(int n, Cricket c[]){
try{
for(int i=0;i<n;i++){
c[i].batavg=c[i].totalruns/c[i].inning;
}
}catch(ArithmeticException e){
System.out.println("Invalid arg");

}
}
static void sort(int n, Cricket c[]){
String temp1;
int temp2,temp3,temp4;
float temp5;
for(int i=0;i<n;i++){
for(int j=i+1;j<n;j++){
if(c[i].batavg<c[j].batavg){
temp1=c[i].name;
c[i].name=c[j].name;
c[j].name=temp1;
temp2=c[i].inning;
c[i].inning=c[j].inning;
c[j].inning=temp2;
temp3=c[i].tofnotout;
c[i].tofnotout=c[j].tofnotout;
c[j].tofnotout=temp3;
temp4=c[i].totalruns;
c[i].totalruns=c[j].totalruns;
c[j].totalruns=temp4;
temp5=c[i].batavg;
c[i].batavg=c[j].batavg;
c[j].batavg=temp5;
}
}
}
}
}
public class a4sa1 {
public static void main(String args[])throws IOException{
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
System.out.println("Enter the limit:");
int n=Integer.parseInt(br.readLine());
```

```
Cricket c[]=new Cricket[n];
for(int i=0;i<n;i++){
c[i]=new Cricket();
c[i].get();
}
Cricket.avg(n,c);
Cricket.sort(n, c);
for(int i=0;i<n;i++){
c[i].put();
}
}

}
```

Output:

```
Enter the limit:
2
Enter the name, no of innings, no of times not out, total runs:
Virat
3
1
100
Enter the name, no of innings, no of times not out, total runs:
Rohit
2
1
120
Name=Rohit
no of innings=2
no times notout=1
total runs=120
bat avg=60.0
Name=Virat
no of innings=3
no times notout=1
total runs=100
bat avg=33.0
```

Slip 19

Q.1]Write a program to accept the two dimensional array from user and display sum of its diagonal elements.

```
public class Main {

  public static void main(String[] args) {
    int[][] dataset = {{1, 5, 8},
                       {4, 3, 1},
                       {6, 5, 2}};

    System.out.println("Diagonal sum is " + DiagonalSum(dataset));
  }

  /**
  * Calculate the sum of diagonal elements.
  * @param a : 2-D array.
  * @return sum of diagonal elements.
  */
  private static int DiagonalSum(int[][] a) {
    int sum = 0;
    for (int i = 0; i < a.length; i++)
      for (int j = 0; j < a[0].length; j++) {
        // Check for main diagonal element.
```

```
        if (i == j) {
          sum += a[i][j];
        }

        // Check for secondary diagonal element.
        if (i + j == a.length - 1) {
          sum += a[i][j];
        }
      }
    return sum;
   }

}
```

Q2) Write a program which shows the combo box which includes list of T.Y.B.Sc.(Comp. Sci) subjects. Display the selected subject in a text field.

```
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import java.io.*;

class ComboBox_Demo extends JFrame
{
    JFrame frame = new JFrame(" ");
    JComboBox cb1, cb2;
    JTextField txtsub;
    Container cc;
    JLabel lblsub, sub;

    public ComboBox_Demo()
    {
        super(" COMBO LIST");
        txtsub = new JTextField(20);
        cb1 = new JComboBox();
        lblsub = new JLabel(" SELECT SUBJECT:");
        sub= new JLabel(" SUBJECT NAME:");
        cc = getContentPane();
        cc.setLayout(null);
        cc.add(lblsub);
        cc.add(cb1);
        cc.add(txtsub);
        cc.add(sub);

        setSize(500,300);
        setLocation(0,0);
```

```
        setResizable(false);
        setLayout(null);
        cc.setBackground(Color.pink);
        setVisible(true);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        lblsub.setBounds(50,50,250,40);
        cb1.setBounds(170,50,250,35);
        sub.setBounds(50,150,250,40);
        txtsub.setBounds(170,150,250,35);

        cb1.addItem("OPERATING SYSTEM");
        cb1.addItem("THEORATICAL COMPUTER SCIENCE");
        cb1.addItem("NETWORKING");
        cb1.addItem("INTERNET PROGRAMMING");
        cb1.addItem("PROGRAMMING IN JAVA I");
        cb1.addItem("BLOCKCHAIN");
        cb1.addItem("PYTHON");
        cb1.addItemListener(new ItemListener()
        {
            public void itemStateChanged(ItemEvent event)
            {
                if (event.getStateChange() == ItemEvent.SELECTED)
                {
                    try
                    {
                        String no=cb1.getSelectedItem().toString();
                        txtsub.setText(no);
                    }
                    catch (Exception ex)
                    {
                        ex.printStackTrace();
                    }
                }
            }
        });
    }
    public static void main(String[] args)
    {
        new ComboBox_Demo();
    }
}
```

Slip 20

Q1) Write a Program to illustrate multilevel Inheritance such that country is inherited from continent. State is inherited from country. Display the place, state, country and continent.

```
import java.io.InputStreamReader;
import java.io.BufferedReader;
import java.io.IOException;
class Continent
{
String con;
InputStreamReader i = new InputStreamReader(System.in);
BufferedReader r = new BufferedReader(i);
void con_input() throws IOException
{
System.out.println("Enter Continent Name: ");
con = r.readLine();
}
}
class Country extends Continent
{
String cou ;
void cou_input() throws IOException
{
```

```
System.out.println("Enter Country Name: ");
cou = r.readLine();
}
}
class State extends Country
{
String sta;
void sta_input() throws IOException
{
System.out.println("Enter State Name: ");
sta = r.readLine();
}
}
class Main extends State
{
String pla;
void pla_input()throws IOException
{
System.out.println("Enter Place Name : ");
pla = r.readLine();

}

public static void main( String argsp[] )throws IOException
{
Main s = new Main();
s.con_input();
s.cou_input();
s.sta_input();
s.pla_input();
System.out.println("\n\nContinent: "+s.con);
System.out.println("Country: "+s.cou);
System.out.println("State: "+s.sta);
System.out.println("Place :" + s.pla);
}
}
```

Output:

```
Enter Continent Name:
Asia
Enter Country Name:
India
Enter State Name:
```

Maharashtra
Enter Place Name :
Pune

Continent: Asia
Country: India
State: Maharashtra
Place :Pune

Q2) Write a package for Operation, which has two classes, Addition and Maximum. Addition has two methods add () and subtract (), which are used to add two integers and subtract two, float values respectively. Maximum has a method max () to display the maximum of two integers

Addition.java

```
package tybca;
public class Addition{
int a,b,add;
double x,y,z,sum;
public Addition(int a,int b)
{
add=a+b;
System.out.println("addition of 2 integer is :->"+add);
}
public Addition(double x, double y)
{

sub=x-y;
System.out.println("Subtraction is :->"+sub);
}
protected void finalize()
{
a=b=0;
x=y=z=0;
}
}
```

Maximum.java

```
package pack2;
public class Maximum
{
int a,b,c;
public Maximum(int a,int b,int c)
{
if((a>b) &&(a>c))
System.out.println(a+" is Maximum ");
else
```

```
{
if(b>c)
System.out.println(b+” is Maximum “);
else
System.out.println(c+” is Maximum “);
}
}
}
PackDemo.java
import tybca.*;
class PackDemo
{
public static void main(String args[])
{
Addition obj=new Maths(2,6);
Addition obj1=new Maths(1.0,2.0,3.0);
Maximum m =new Maximum(8,9,5);
}
}
```

Slip 21

Q1) Define a class MyDate(Day, Month, year) with methods to accept and display a MyDateobject.

Accept date as dd,mm,yyyy. Throw user defined exception "InvalidDateException" if the date is invalid.

```
import java.io.*;

class InvalidDayException extends Exception
{
        InvalidDayException()
        {
                System.out.println("You Entered Invalid Day ........");
        }
}
class InvalidMonthException extends Exception
{
        InvalidMonthException()
        {
                System.out.println("You Entered Invalid Month ........");
        }
}
 class MyClass
{
```

```
	int day,mon,yr;
	MyClass()
	{
		day=1;
		mon=1;
		yr=1991;
	}
	MyClass(int a, int b, int c)
	{
		day=a;
		mon=b;
		yr=c;
		System.out.println("You Entered Valid date...");
		System.out.println(day+"-"+mon+"-"+yr);
	}
}

class set4b1
{
	public static void main(String args[]) throws IOException
	{
		//BufferedReader bf= new BufferedReader(new
InputStreamReader(System.in));
		try
		{

		    int c = Integer.parseInt(args[0]);
		  int b = Integer.parseInt(args[1]);
		    int a = Integer.parseInt(args[2]);

		    boolean leap=(c%400==0) || (c%4==0) && (c%100!=0);

		    if(b<13&&b>0)
		    {
		    }
		    else
			throw new InvalidMonthException();

		    if(b==1||b==3||b==5||b==7||b==8||b==10||b==12)
		    {
			if(a<32 &&a>0)
			{
			}
			else
				throw new InvalidDayException();
		    }
```

```
                else if(b==4||b==6||b==9||b==11)
                {
                        if(a<31&&a>0)
                        {
                                System.out.println("Accepted....");
                        }
                        else
                                throw new InvalidDayException();
                }
                else
                {
                        if(leap && a>29)
                        {
                                System.out.println("Accepted.....");
                        }
                        else
                                throw new InvalidDayException();
                }

                        MyClass m= new MyClass(a,b,c);
                }
                catch(InvalidMonthException m)
                {
                }
                catch(InvalidDayException d)
                {
                }
        }
}
/*----------------------------------------
output =
[user@localhost ~]$ javac set4b1.java
[user@localhost ~]$ java set4b1 1991 6 23
Accepted....
You Entered Valid date...
23-6-1991
[user@localhost ~]$ java set4b1 1991 23 6
You Entered Invalid Month ........
[user@localhost ~]$ java set4b1 1991 6 31
You Entered Invalid Day ........
employee [user@localhost ~]$
*/
```

Q.2] Create an employee class(id,name,deptname,salary). Define a default and parameterized constructor. Use ‘this’ keyword to initialize instance variables. Keep a count of objects created. Create objects using parameterized constructor and display the object count after each

object is created. (Use static member and method). Also display the contents of each object.

```
import java.util.Scanner;

public class Employee {

int id;

String name;

String deptname;

float salary;

static int numberofobjects=0;

Employee(){

id=0;

name="";

deptname="";

salary=0;

}

Employee(int id,String name,String deptname,float salary ){

this.id=id;

this.name=name;

this.deptname=deptname;

this.salary=salary;

numberofobjects++;

}

public void display(){

System.out.println("Employee Id :"+id);

System.out.println("Employee name: "+name);

System.out.println("Employee Department: "+deptname);
```

```
System.out.println("Employee Salary :"+salary);
}
public static void main(String[] args){
int n=0;
Scanner sc=new Scanner(System.in);
System.out.print("How many employees you want to enter :");
n=sc.nextInt();
Employee[] ob=new Employee[n];
for(int i=0;i<n;i++){
sc= new Scanner(System.in);
System.out.println("Enter Id of employee "+(i+1)+" :");
int id=sc.nextInt();
System.out.println("Enter Name of employee "+(i+1)+" :");
sc.nextLine();
String name= sc.nextLine();
System.out.println("Enter dept name of employee "+(i+1)+" :");
String deptname=sc.nextLine();
System.out.println("Enter salary of employee "+(i+1)+" :");
float salary = sc.nextFloat();
ob[i]=new Employee(id,name,deptname,salary);
System.out.println("\nNumber of Objects : "+numberofobjects);

}
for(int i=0;i<n;i++)
{
ob[i].display();
```

```
}
}
}
```

Output:

```
How many employees you want to enter :2
Enter Id of employee 1 :
1
Enter Name of employee 1 :
rohit
Enter dept name of employee 1 :
abc
Enter salary of employee 1 :
1000000
Number of Objects : 1
Enter Id of employee 2 :
2
Enter Name of employee 2 :
bairwa
Enter dept name of employee 2 :
90000000
Enter salary of employee 2 :
78888888
Number of Objects : 2
```

Employee Id :1

Employee name: rohit

Employee Department: abc

Employee Salary :1000000.0

Employee Id :2

Employee name: bairwa

Employee Department: 90000000
Employee Salary :7.8888888E7

Slip 22

Q1) Write a program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea() that prints the area of the given shape. (use method overriding).

```
import java.util.*;
abstract class shape
{
int x,y;
abstract void area(double x,double y);
}
class Rectangle extends shape
{
void area(double x,double y)
{
System.out.println("Area of rectangle is :"+(x*y));
}
```

```
}
class Circle extends shape
{
void area(double x,double y)
{
System.out.println("Area of circle is :"+(3.14*x*x));
}
}
class Triangle extends shape
{
void area(double x,double y)
{
System.out.println("Area of triangle is :"+(0.5*x*y));
}
}
public class AbstactDDemo
{
public static void main(String[] args)
{
Rectangle r=new Rectangle();
r.area(2,5);
Circle c=new Circle();
c.area(5,5);
Triangle t=new Triangle();
t.area(2,5);
}
}
```

Output:

```
Area of rectangle is :10.0
Area of circle is :78.5
Area of triangle is :5.0
```

Q2) Write a program that handles all mouse events and shows the event name at the center of the Window, red in color when a mouse event is fired. (Use adapter classes).

```
import javax.swing.*;

import java.awt.*;

import javax.swing.event.*;

import java.awt.event.*;

class MouseEventPerformer extends JFrame implements
```

```
MouseListener
{
   JLabel l1;
   public MouseEventPerformer()
   {
      setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      setSize(300,300);
      setLayout(new FlowLayout(FlowLayout.CENTER));
      l1 = new JLabel();
      Font f = new Font("Verdana", Font.BOLD, 20);
      l1.setFont(f);
      l1.setForeground(Color.BLUE);
      add(l1);
      addMouseListener(this);
      setVisible(true);
   }
   public void mouseExited(MouseEvent m)
   {
      l1.setText("Mouse Exited");
   }
   public void mouseEntered(MouseEvent m)
   {
      l1.setText("Mouse Entered");
   }
   public void mouseReleased(MouseEvent m)
   {
```

```
        l1.setText("Mouse Released");
    }
    public void mousePressed(MouseEvent m)
    {
        l1.setText("Mouse Pressed");
    }
    public void mouseClicked(MouseEvent m)
    {
        l1.setText("Mouse Clicked");
    }
    public static void main(String[] args) {
        MouseEventPerformer mep = new MouseEventPerformer();
    }
}
```

Slip 23

Q1) Define a class MyNumber having one private int data member. Write a default constructor to initialize it to 0 and another constructor to initialize it to a value (Use this). Write methods isNegative, isPositive, isZero, isOdd, isEven. Create an object in main.Use command line arguments to pass a value to the Object.

```
class MyNumber
{       int n;
        MyNumber()
        {
                n=0;
        }
        MyNumber(int n)
        {
                this.n=n;
        }
        void isNegative()
        {
                if(n<0)
                        System.out.println(n+"is negative");
        }
```

```
        void isPositive()
    {
        if(n>0)
            System.out.println(n+"is positive");
    }
      void isZero()
      {
            if(n==0)
                System.out.println(n+"is zero");
      }
      void isOdd()
      {
            if(n%2!=0)
                System.out.println(n+"is odd");
      }
      void isEven()
      {
            if(n%2==0)
                System.out.println(n+"is even");
      }
      public static void main(String args[])
      {
            int n=Integer.parseInt(args[0]);
            MyNumber m=new MyNumber();
          m.isNegative();
          m.isPositive();
          m.isZero();
          m.isOdd();
          m.isEven();

            MyNumber m1=new MyNumber(10);
            m1.isNegative();
            m1.isZero();
            m1.isEven();
            m1.isOdd();
      }
}
```

Q.2] Write a simple currency converter, as shown in the figure. User can enter the amount of "Singapore Dollars", "US Dollars", or "Euros", in floating-point number. The converted values shall be displayed to 2 decimal places. Assume that 1 USD = 1.41 SGD, 1 USD = 0.92 Euro, 1 SGD = 0.65 Euro.

Slip 24

Q1) Create an abstract class 'Bank' with an abstract method 'getBalance'. Rs.100, Rs.150 and Rs.200 are deposited in banks A, B and C respectively. 'BankA', 'BankB' and 'BankC' are subclasses of class 'Bank', each having a method named 'getBalance'. Call this method by creating an object of each of the three classes.

```
abstract class Bank{
  public abstract void getBalance();
}
class BankA extends Bank{
  public void getBalance(){ System.out.println("Deposited Balance is = $100"); }
}
//Written and logic by Avtar Singh (Singha2k2)
class BankB extends Bank{
  public void getBalance(){ System.out.println("Deposited Balance is =
```

```
$150"); }
}
class BankC extends Bank{
   public void getBalance(){ System.out.println("Deposited Balance is =
$200"); }
}
class calculateBalance{
public static void main(String[] args){
  BankA Balance1 = new BankA();
  Balance1.getBalance();
  BankB Balance2 = new BankB();
  Balance2.getBalance();
  BankC Balance3 = new BankC();
  Balance3.getBalance();
  }
}
```

Q2) Program that displays three concentric circles where ever the user clicks the mouse on a frame.
The program must exit when user clicks ‘X’ on the frame

```
#include<stdio.h>
#include<graphics.h>
#include<conio.h>


int main(){
  int gd = DETECT,gm;
  int x ,y;
  initgraph(&gd, &gm, "C:\\TC\\BGI");
  /* Initialize center of circle with center of screen */
  x = getmaxx()/2;
  y = getmaxy()/2;
```

```
    outtextxy(240, 50, "Concentric Circles");
   /* Draw circles on screen */
   setcolor(RED);
   circle(x, y, 30);
   setcolor(GREEN);
   circle(x, y, 50);
   setcolor(YELLOW);
   circle(x, y, 70);
   setcolor(BLUE);
   circle(x, y, 90);
    getch();
   closegraph();
   return 0;
}
```

Slip 25

Q1) Create a class Student(rollno, name ,class, per), to read student information from the console and display them (Using BufferedReader class)

```
import java.io.* ;

class Except

{

public static void main(String args[])throws Exception

{

InputStreamReader r=new InputStreamReader(System.in);

BufferedReader br=new BufferedReader(r);

System.out.println(“Enter name:”);

String name = br.readLine();
```

```
System.out.println(“Enter roll no.:”);

String number=br.readLine();

System.out.println(“Enter marks:”);


String marks=br.readLine();

System.out.println(“name:”+name);

System.out.println(“Roll No.:”+number);

System.out.println(“Marks:”+marks);

}

}
```

Q2) Create the following GUI screen using appropriate layout manager. Accept the name, class hobbies from the user and display the selected options in a textbox.

```
import javax.swing.*;

import java.awt.*;

import java.awt.event.*;


class seta3 extends JFrame implements ActionListener

{

        private JLabel l1,l2,l3;

        private JButton b;

        private JRadioButton r1,r2,r3;

        private JCheckBox c1,c2,c3;

        private JTextField t1,t2;
```

```
private ButtonGroup b1;
private JPanel p1,p2;
private StringBuffer s1=new StringBuffer();

public seta3(String s)
{
        super(s);
        b1=new ButtonGroup();
        p1=new JPanel();
        p2=new JPanel();
        b=new JButton("Clear");
        b.addActionListener(this);

        r1=new JRadioButton("FY");
        r2=new JRadioButton("SY");
        r3=new JRadioButton("TY");

        b1.add(r1);
        b1.add(r2);
        b1.add(r3);
        r1.addActionListener(this);
        r2.addActionListener(this);
        r3.addActionListener(this);

        c1=new JCheckBox("Music");
        c2=new JCheckBox("Dance");
```

```
c3=new JCheckBox("Sports");

c1.addActionListener(this);
c2.addActionListener(this);
c3.addActionListener(this);

l1=new JLabel("Your Name");
l2=new JLabel("Your Class");
l3=new JLabel("Your Hobbies");
t1=new JTextField(20);
t2=new JTextField(30);

p1.setLayout(new GridLayout(5,2));
p1.add(l1);p1.add(t1);
p1.add(l2);p1.add(l3);
p1.add(r1);p1.add(c1);
p1.add(r2); p1.add(c2);
p1.add(r3);p1.add(c3);

p2.setLayout(new FlowLayout());
p2.add(b);
p2.add(t2);

setLayout(new BorderLayout());
add(p1,BorderLayout.NORTH);
add(p2,BorderLayout.EAST);
```

```
}
public void actionPerformed(ActionEvent e)
{
        if(e.getSource()==r1)
        {
                String s =t1.getText();
                s1.append("Name = ");
                s1.append(s);
                s1.append(" Class = FY");
        }
        else if(e.getSource()==r2)
        {
                String s =t1.getText();
                s1.append("Name = ");
                s1.append(s);
                s1.append(" Class = SY");
        }
        else if(e.getSource()==r3)
        {
                String s =t1.getText();
                s1.append("Name = ");
                s1.append(s);
                s1.append(" Class = TY");
        }
        else if(e.getSource()==c1)
        {
```

```
            s1.append(" Hobbies = Music");
        }
        else if(e.getSource()==c2)
        {
            s1.append(" Hobbies = Dance");
        }
        else if(e.getSource()==c3)
        {
            s1.append(" Hobbies = Sports");
        }
        t2.setText(new String(s1));
        //          t2.setText(s2);

        if(e.getSource()==b)
        {
            t2.setText(" ");
            t1.setText(" ");
        }

}
public static void main(String arg[])
{
        seta3 s=new seta3("Profile");
        s.setSize(400,200);
        s.setVisible(true);
        s.setLocation(400,400);
```

```
            s.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        }
}
```

Slip 27

Q1) Define an Employee class with suitable attributes having getSalary() method, which returns salary withdrawn by a particular employee. Write a class Manager which extends a class Employee, override the getSalary() method, which will return salary of manager by adding traveling allowance, house rent allowance etc.

```
class Employee
{
String name;
int sal;
Employee()
{
name=null;
sal=0;
}
```

```
Employee(String n,int s)
{
name=n;
sal=s;
}
int getSalary()
{
return sal;
}

}

class Manager extends Employee

{
int hra,ta;

Manager()
{
super();
hra=ta=0;
}
Manager(String n,int sal,int h,int t)
{super(n,sal);
hra=h; ta=t;
}
```

```
int getSalary()

{

return (super.getSalary()+hra+ta);

}

class EmpMan

{

public static void main(String args[])

{

Manager m1=new Manager(“Raj”,20000,200,800); //parameter
as(Name,salary,HRA,TA)

System.out.println(“Salary of Manager= “+m1.getSalary());

}

}
```

Q2) Write a program to accept a string as command line argument and check whether it is a file or directory. Also perform operations as follows: i)If it is a directory,delete all text files in that directory. Confirm delete operation from user before deleting text files. Also, display a count showing the number of files deleted, if any, from the directory. ii)If it is a file display various details of that file.

```
import java.io.*;



class set5a1

{

	public static void main(String args[]) throws IOException

	{

		BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));

		String dirname=args[0],ext;

		int ch,i,cnt=0;
```

```
File f1=new File(dirname);
ext="txt";
if(f1.isFile())
{
        System.out.println(f1+" is a File\n");
        System.out.println("Path      : "+f1.getPath());
        System.out.println("File Size : "+f1.length()+" bytes\n");
}

else if(f1.isDirectory())
{
        System.out.println(args[0]+" Is a Directory\n");
        System.out.println("Contents Of : "+dirname);
        String s[]=f1.list();

        for(i=0;i<s.length;i++)
        {
                File f=new File(dirname,s[i]);
                if(f.isFile())
                {
                        cnt++;
                        System.out.println(s[i]+" is a File\n");
                }
                else
                        System.out.println(s[i]+" is a Directory\n");
        }
```

```
                    System.out.println("Total Number Of Files :"+cnt);
        System.out.println("Do You Want To Delete Files With Extension 'txt' (1/0) : ?");
                    ch=Integer.parseInt(br.readLine());
                    if(ch==1)
                    {
                        for(i=0;i<s.length;i++)
                        {
                            File f=new File(dirname,s[i]);
                            if(f.isFile() && s[i].endsWith(ext))
                            {
                                System.out.println(s[i]+" -> deleted");
                                f.delete();
                            }
                        }
                    }
                }
        }
}
/*---------------------------------------------------------------------------
output =
[user@localhost ~]$ javac set5a1.java
[user@localhost ~]$ java set5a1 Diya
Diya Is a Directory
```

Contents Of : Diya

abc.txt is a File

def.txt is a File

soyal is a Directory

Total Number Of Files :2

Do You Want To Delete Files With Extension 'txt' (1/0) : ?

0

[user@localhost ~]$ java set5a1 Diya

Diya Is a Directory

Contents Of : Diya

abc.txt is a File

def.txt is a File

soyal is a Directory

Total Number Of Files :2

Do You Want To Delete Files With Extension 'txt' (1/0) : ?

1

abc.txt -> deleted

def.txt -> deleted

[user@localhost ~]$ java set5a1 Diya

Diya Is a Directory

Contents Of : Diya

soyal is a Directory

Total Number Of Files :0

Do You Want To Delete Files With Extension 'txt' (1/0) : ?

0

[user@localhost ~]$
*/

Slip 29

Q1) Write a program to create a class Customer(custno,custname,contactnumber,custaddr). Write a method to search the customer name with given contact number and display the details.

```
import java.io.*;

class Customer

{

public static void main(String arg[])

{

DataInputStream myinput=new DataInputStream(System.in);

String name;

int bill = 0,id = 0;

try

{

System.out.println("enter name of customer");

name = myinput.readLine();
```

```
System.out.println("enter bill");

bill = Integer.parseInt(myinput.readLine());

System.out.println("enter id");

id = Integer.parseInt(myinput.readLine());

System.out.println ("name of customer is"+name);

System.out.println ("bill of customer"+bill);

System.out.println ("id of customer"+id);

}

catch(Exception e)

{

System.out.println("wrong input error!!!");

}

}
}
```

Q.2]) Write a program to create a super class Vehicle having members Company and price. Derive two different classes LightMotorVehicle(mileage) and HeavyMotorVehicle (capacity_in_tons). Accept the information for "n" vehicles and display the information in appropriate form. While taking data, ask user about the type of vehicle first.

```
import java.util.*;


class Vehicle

{

        String company;

        double price;

        Scanner s=new Scanner(System.in);


        void accept()
```

```
	{
		System.out.println("Enter Company Name :: ");
		company=s.next();
		System.out.println("Enter Price :: ");
		price=s.nextDouble();
	}
	void display()
	{
		System.out.println("Company Name ::"+company);
		System.out.println("Price ::"+price);
	}

}

class LightMotorVehicle extends Vehicle
{
	double mileage;

	void accept()
	{
		super.accept();
		System.out.println("Enter Mileage :: ");
		mileage=s.nextDouble();
	}

	void display()
```

```
        {
                System.out.println("For Light Motor Vehicle :: ");
                super.display();
                System.out.println("Mileage ::"+mileage);
        }

}

class HeavyMotorVehicle extends Vehicle
{
        double capacity;

        void accept()
        {
                super.accept();
                System.out.println("Enter Capacity in tons :: ");
                capacity=s.nextDouble();
        }

        void display()
        {
                System.out.println("For Heavy Motor Vehicle :: ");
                super.display();
                System.out.println("Capacity in tons :: "+capacity);
        }
```

```
}

class vehicleDemo
{
	public static void main(String[] args)
	{
		Scanner s=new Scanner(System.in);
		System.out.println("Enter how many objects to be created ::
");
		int n=s.nextInt();

		LightMotorVehicle[] l=new LightMotorVehicle[n];
		HeavyMotorVehicle[] h=new HeavyMotorVehicle[n];

		int lcnt=0;
		int hcnt=0;

		for(int i=0;i<n;i++)
		{
			System.out.println("Enter \n 1.Light Motor Vehicle\n
2.Heavy Motor Vehicle");
			int ch=s.nextInt();

			if(ch==1)
			{
				l[lcnt]=new LightMotorVehicle();
```

```
                l[lcnt].accept();
                lcnt++;
            }
            else if(ch==2)
            {
                h[hcnt]=new HeavyMotorVehicle();
                h[hcnt].accept();
                hcnt++;
            }
        }

        for(int i=0;i<lcnt;i++)
        {
            l[i].display();

        }
        for(int i=0;i<hcnt;i++)
        {
            h[i].display();
        }
    }
}
```