1.GuessMyNumber

```
import java.util.Random;
import java.util.Scanner;

public class GuessMyNumber {
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        Random rand = new Random();
        int compno = rand.nextInt(10);
        while(true)
        {
            int myguess = sc.nextInt();
            if(myguess==compno)
            {
                System.out.println("Your guess no:"+myguess+"is correct");
                break;
            }
            if(myguess<compno)
            {
                System.out.println("Your guess no:"+myguess+"is lower");

            }

            else
                System.out.println("Your guess no:"+myguess+"is higher");


        }
    }
}
```

2.Arrays are Symmetrical

```
import java.util.Scanner;

class Symmetrical
{
    int a[];
    int oddno;
    int evenno;
    public void read()
    {
        Scanner s= new Scanner(System.in);
        System.out.println("Enter the size of array");
        int size = s.nextInt();
        a = new int[size];
        System.out.println("Enter the array Elements");
        for(int i=0;i<size;i++)
        {
            a[i]=s.nextInt();
        }

    }
```

```
    public void compute() {
        for (int i = 0; i < a.length;i++) {
            if (a[i] % 2 == 0) {
                evenno++;
            } else {
                oddno++;

            }


        }
    }
        int getevenno()
        {
            return evenno;
        }
        int getoddno()
        {
            return oddno;
        }

}
public class CheckSymmetrical {
    public static void main(String[] args)
    {
        Symmetrical s1 = new Symmetrical();
        s1.read();
        s1.compute();
        Symmetrical s2 = new Symmetrical();
        s2.read();
        s2.compute();
        if(s1.getevenno()== s2.getevenno()&&s1.getoddno()== s1.getoddno())
        {
            System.out.println("Arrays are Symmetrical");
        }
        else {
            System.out.println("Arrays are not Symmetrical");
        }
    }
}


3.student
import java.util.Scanner;

class Student {
    private String USN;
    private String Name;
    private String Branch;
    private String Phone;

    public Student(int size) {
        USN = " ";
        Name = " ";
        Branch = " ";
        Phone = " ";
```

```
    }

    public Student(String pUSN, String Name, String Branch, String Phone) {
        USN = pUSN;
        this.Name = Name;
        this.Branch = Branch;
        this.Phone = Phone;
    }

    void display() {
        System.out.println("USN:" + USN);
        System.out.println("Name:" + Name);
        System.out.println("Branch:" + Branch);
        System.out.println("Phone:" + Phone);
    }
}
    public class StudentDemo{

        public static void main(String[] args)
        {
            Scanner sc=new Scanner(System.in);
            System.out.println("How many Students");
            int size= sc.nextInt();
            Student[] ob = new Student[size];
             sc=new Scanner(System.in);
            for(int i=0;i<ob.length;i++)
            {
                String USN,Name,Branch,Phone;
                System.out.println("Enter USN,Name,Branch,Phone");
                USN=sc.nextLine();
                Name=sc.nextLine();
                Branch=sc.nextLine();
                Phone=sc.nextLine();
                Student objs=new Student(USN,Name,Branch,Phone);
                ob[i]=objs;

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


4.Staff
import java.util.Scanner;

class staff
{
    String staffId;
    String Name;
    String Phone;
    double Salary;
    void read()
```

```
    {
        Scanner read =new Scanner(System.in);
        System.out.println("enter StaffId,Name,Phone,Salary");
        staffId=read.nextLine();
        Name= read.nextLine();
        Phone=read.nextLine();
        Salary=read.nextDouble();


    }
    void display()
    {
        System.out.println("StaffId:"+staffId);
        System.out.println("Name:"+Name);
        System.out.println("Phone:"+Phone);
        System.out.println("Salary"+Salary);

    }
}
class Teaching extends staff
{
    String domain;
    String publications;
    void read()
    {
        super.read();
        Scanner read= new Scanner(System.in);
        System.out.println("Enter Domain, Publications:");
        domain=read.nextLine();
        publications=read.nextLine();
    }

    @Override
    void display() {
        super.display();
        System.out.println("Domain:"+domain);
        System.out.println("Publications:"+publications);
    }
}
class Technical extends staff
{
    String skills;

    void read()
    {
        super.read();
        Scanner read= new Scanner(System.in);
        System.out.println("Enter skills:");
        skills=read.nextLine();

    }

    @Override
    void display() {
        super.display();
        System.out.println("Skills:"+skills);
```

```
    }
}

class Contract extends staff
{
    String period;

    void read()
    {
        super.read();
        Scanner read= new Scanner(System.in);
        System.out.println("Enter period:");
        period=read.nextLine();

    }

    @Override
    void display() {
        super.display();
        System.out.println("Period:"+period);

    }
}
public class StaffDemo {
    public static void main(String[] args)
    {
        Teaching T1= new Teaching();
        T1.read();
        T1.display();
        Technical tech=new Technical();
        tech.read();
        tech.display();
        Contract c1=new Contract();
        c1.read();
        c1.display();

    }
}
```

5.Converters

```
package converters;

import java.util.Scanner;

public class DistanceConvertor {
    public void converter()
    {
        System.out.println("Enter the distance in miles:");
        Scanner sc=new Scanner(System.in);
        int miles = sc.nextInt();
        double km=1.609*miles;
        System.out.println("The converted distance in kilometer:"+km);
        System.out.println("Enter the distance in km");
        Scanner read = new Scanner(System.in);
        int kilometer = read.nextInt();
```

```
        double Miles=kilometer/1.609;
        double meter=kilometer*1000;
        System.out.println("The Converted Distance in miles is:"+Miles+"and
the converted distance in meter is:"+meter);
        System.out.println("Enter the distance in meter");
        Scanner Read=new Scanner(System.in);
        int Meter=Read.nextInt();
        double Kilometer=Meter/1000;
        System.out.println("the Converted distance in Kilometer
is:"+Kilometer);

    }
}

package converters;

import java.util.Scanner;

public class TimeConvertor {
    public  void converter()
    {
        System.out.println("Enter the time in hours:");
        Scanner read= new Scanner(System.in);
        int hours =read.nextInt();
        double min=hours*60;
        double sec=hours*3600;
        System.out.println("converted time in min:"+min+"converted time in
sec:"+sec);
        System.out.println("Enter the time in min");
        Scanner sc= new Scanner(System.in);
        int minutes=sc.nextInt();
        double Sec=minutes*60;
        double Hours=min/60;
        System.out.println("converted time in sec:"+sec+"and in
hours:"+Hours);



    }
}
package Demo;
import converters.*;

public class Convertor {
    public static void main(String[] args)
    {
        DistanceConvertor dc1= new DistanceConvertor();
        dc1.converter();
        TimeConvertor tc1=new TimeConvertor();
        tc1.converter();
    }
}
```

6.Interface

import java.util.Scanner;

```
interface Resume
{
   void biodata();
}
class Teacher implements Resume
{
   String Perfo;
   String quali;
   String exp;
   String Achieve;
   void readdata()
   {
      Scanner s=new Scanner(System.in);
      System.out.println("enter the personal information:");
      Perfo=s.nextLine();
      System.out.println("Enter the quali:");
      quali=s.nextLine();
      System.out.println("Enter the exp:");
      exp=s.nextLine();
      System.out.println("enter the achieve:");
      Achieve=s.nextLine();
   }
   public void biodata()
   {
      System.out.println("Personal Information:"+Perfo);
      System.out.println("Qualification:"+quali);
      System.out.println("experience:"+exp);
      System.out.println("Achievements:"+Achieve);
   }
}
class Student implements Resume
{
   String perinfo;
   String result;
   String discip;
   void readata()
   {
      Scanner s=new Scanner(System.in);
      System.out.println("enter the personal information:");
      perinfo=s.nextLine();

      System.out.println("Enter the result:");
      result=s.nextLine();
      System.out.println("enter the discipline:");
      discip=s.nextLine();
   }
   public void biodata()
   {
      System.out.println("Personal Information:"+perinfo);
      System.out.println("Result:"+result);
      System.out.println("Discipline:"+discip);


   }
}
```

```
public class InterfaceDemo {
   public static void main(String[] args){
      Teacher t1=new Teacher();
      t1.readdata();
      t1.biodata();
      Student s1=new Student();
      s1.readata();
      s1.biodata();
   }
}
```

7.multithread

```
import java.util.Random;
import java.util.Scanner;

class Firstthread extends Thread
{
   public void run()
   {
      Random ran= new Random();
      while(true)
      {
         int random= ran.nextInt(20);
         System.out.println("Random number generated:"+random);
         try {
            Thread.sleep(1000);
         }
         catch (InterruptedException e)
         {
            System.out.println(e);
         }
         Secondthread st = new Secondthread(random);
         st.start();
         Thirdthread tt=new Thirdthread(random);
         tt.start();
      }

   }
}
class Secondthread extends Thread
      {
         int num;
         public Secondthread(int n)
         {
            num=n;
         }
         public void run()
         {
            int square=num*num;
            System.out.println("The square of the number "+num+"is:"+square);
         }


      }
      class Thirdthread extends Thread
      {
```

```
        int num;
        public Thirdthread(int n)
        {
           num=n;
        }
        public void run()
        {
           int cube=num*num*num;
           System.out.println("The cube of the number "+num+"is:"+cube);
        }
     }
     public class MultithreadDemo {
  public static void main(String[] args)
  {
     Scanner sc=new Scanner(System.in);
     Firstthread fs=new Firstthread();
     fs.start();
  }
}
```

8.ExceptionDemo

```
import java.util.InputMismatchException;
import java.util.Scanner;

class Generatingexp
{
   int[] myarray=new int[5];
   public void readelements()
   {
      Scanner sc=new Scanner(System.in);
      System.out.println("Enter the array elements");
      for(int i=0;i<myarray.length;i++)
      {
         myarray[i]=sc.nextInt();
      }
   }
   public void divide()
   {
      int small = getSmallestElement();
      for(int i=0;i<myarray.length;i++)
      {
         double result=myarray[i]/small;


      }
      System.out.println("division successfull");
   }
   int getSmallestElement()
   {
      int small=myarray[0];
      for(int i=0;i<myarray.length;i++)
      {
         if(myarray[i]<small)
         {
            small=myarray[i];
         }
```

```
        }
        return small;
    }
    public void displayElements()
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the indx you want to access:");
        int idx=sc.nextInt();
        int val=myarray[idx];
        System.out.println("Value is:"+val);


    }
    public void compute()
    {
        readelements();
        divide();
        displayElements();

    }
}
public class ExceptionDemo {
    public static void main(String[] args)
    {
        Generatingexp ge=new Generatingexp();
        try
        {
            ge.compute();
        }
        catch (ArithmeticException e)
        {
            System.out.println(e);
        }
        catch (InputMismatchException e)
        {
            System.out.println(e);
        }
        catch (ArrayIndexOutOfBoundsException e)
        {
            System.out.println(e);
        }
    }
}
```

9. Vowels

```
import java.util.Scanner;

class MyString{
    void VowelOccurences(String userstr) {


        int acount = 0, ecount = 0, icount = 0, ocount = 0, ucount = 0;
        for(int i=0;i<userstr.length();i++)
        {
            char ch=userstr.charAt(i);
```

```
            if(ch=='a'||ch=='A')
                acount++;
            else if(ch=='e'||ch=='E')
                ecount++;
            else if(ch=='i'||ch=='I')
                icount++;
            else if(ch=='o'||ch=='O')
                ocount++;
            else if(ch=='u'||ch=='U')
                ucount++;

        }
        System.out.println("No.of Occurences of a:"+acount);
        System.out.println("of e:"+ecount);
        System.out.println("of i:"+icount);
        System.out.println("of o:"+ocount);
        System.out.println("of u:"+ucount);
    }
    int noDigits(String userstr)
    {
        int nodigits=0;
        for(int i=0;i<userstr.length();i++)
        {
            char ch=userstr.charAt(i);
            if(ch>='0'&&ch<='9')
                nodigits++;
        }

        return nodigits;
    }
    int noCaps(String userstr)
    {
        int nocaps=0;
        for(int i=0;i<userstr.length();i++)
        {
            char ch=userstr.charAt(i);
            if(ch>='A'&&ch<='Z')
            {
                nocaps++;


            }
            System.out.println("position of "+ch+ "is" + i);
        }
        return nocaps;
    }


}
public class VowelsDemo {
    public static void  main(String[] args) {


        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the String:");
        String s = sc.nextLine();
```

```
        MyString ms = new MyString();
        int out = ms.noCaps(s);
        System.out.println("no of capital letters:" + out);
        int digits = ms.noDigits(s);
        System.out.println("no of digits:" + digits);
        ms.VowelOccurences(s);
    }
}
```

10.count the number of lines in file

```
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

class MyFile{
    public void processFile()
    {
        Scanner read=new Scanner(System.in);
        System.out.println("enter the file name:");
        String fname=read.nextLine();
        File file=new File(fname);
        int linecount=0;
        int wordcount=0;
        int charcount=0;
        try
        {
            Scanner sc=new Scanner(file);
            while(sc.hasNext())
            {
                String line=sc.nextLine();
                linecount++;
                String[] sarr=line.split(" ");
                wordcount+=sarr.length;
                charcount+=line.length();

            }
        }
        catch (FileNotFoundException e)
        {
            System.out.println(e);
        }
        System.out.println("no of line count:"+linecount);
        System.out.println("no of char count:"+charcount);
        System.out.println("no of word count:"+wordcount);

    }
}
public class FileDemo {
    public static void main(String[] args)
    {
        MyFile hehe=new MyFile();
        hehe.processFile();
    }


}
```

Part-B

11.ValidateUSNException

```
import java.util.Scanner;

class ValidateUSNException extends Exception
{
   String msg=" ";
   public ValidateUSNException(String msg)
   {
      this.msg=msg;
   }
   public String toString()
   {
      return msg;

   }
}

public class USNExceptionDemo {
   public static void ValidateUSN(String usn) throws ValidateUSNException
   {
      int len=usn.length();
      if(len!=10)
         throw new ValidateUSNException("Invalid Exception");
      boolean firstThree=usn.startsWith("NNM");
      if(firstThree==false)
         throw new ValidateUSNException("Invalid first three letters");
      String year=usn.substring(3,5);
      int nnyear=Integer.parseInt(year);
      if(nnyear>23)
         throw new ValidateUSNException("Invalid Year");
      String branch=usn.substring(5,7);
      String[] barr={"CS","DS","IS","ME","CC","EC","EE","RE","BT","AI","CV"};
      boolean branchval=false;
      for(String str:barr)
      {
         if(branch.equals(str))
         {
            branchval=true;
            break;

         }
      }
      if(branchval==false)
         throw new ValidateUSNException("USN with Invalid Branch");
      String numstr=usn.substring(7,10);
      try
      {
         int nnum=Integer.parseInt(numstr);

      }
      catch (NumberFormatException e)
      {
         throw  new ValidateUSNException("Last three numbers are invalid");
      }
```

```
    }
    public static void main(String[] args)
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the USN:");
        String str= sc.nextLine();
        try
        {
            ValidateUSN(str);
        }
        catch (ValidateUSNException e)
        {
            System.out.println(e);
        }
        System.out.println("Validation Completed");
    }

}
```

12.Synchronization

```
class callme
{
    synchronized static void call(String msg)
    {
        System.out.print("["+msg);
        try
        {
            Thread.sleep(1000);

        }
        catch (InterruptedException e)
        {
            System.out.println(e);
        }
        System.out.print("]");
    }
}
class caller extends Thread
{
    String msg;
    public caller(String s)
    {
        msg=s;
    }
    public void run()
    {
        callme.call(msg);
    }

}
public class Synch {
    public static void main(String[] args)
```

```
    {
        caller c1=new caller("Learn");
        caller c2=new caller("Java");
        caller c3=new caller("Programming");
        c1.start();
        try
        {
            c1.join();
        }
        catch (InterruptedException e)
        {
            System.out.println(e);
        }
        c2.start();
        c3.start();
    }
}
```

13.FileMethods

```
import java.io.File;
import java.util.Scanner;

public class FileMethods {
    public static void main(String[] args)
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter the file name");
        String filename=s.nextLine();
        File f=new File(filename);
        System.out.println(" file/directory exists"+f.exists());
        if(f.exists())
        {
            System.out.println("filename is:"+f.getName());
            System.out.println("file path:"+f.getPath());
            System.out.println("File absolute path:"+f.getAbsolutePath());
            System.out.println("length of the file is:"+f.length()+"bytes");
            System.out.println("is file readable:"+f.canRead());
            System.out.println("is file Writeable"+f.canWrite());
            try
            {
                String[] a=f.list();
                System.out.println("the contents are:");
                for(int i=0;i<a.length;i++)
                {
                    System.out.println(""+a[i]);
                }

            }
            catch (NullPointerException e)
            {
                System.out.println("it is not a directory so no content to shoe");
            }
            System.out.println("file deleted"+f.delete());
```

```
        }
        System.out.println("enter the directory name");
        String dir=s.nextLine();
        File file=new File(dir);
        if(!file.exists())
        {
           file.mkdir();
           System.out.println("Directory is created");
        }
        else {
           System.out.println("Directory already existed");
        }

    }
}
```

14.KeyboradListener

```
import java.awt.*;
import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;

class MyKeyListener implements
        KeyListener
{
    @Override
    public void keyTyped(KeyEvent e) {
        char keyChar=e.getKeyChar();
        System.out.println("Key Typed:"+keyChar);
    }
    public void keyPressed(KeyEvent e)
    {
        int keyCode=e.getKeyCode();
        System.out.println("Key Pressed:"+KeyEvent.getKeyText(keyCode));
    }
    public void keyReleased(KeyEvent e)
    {
        int keyCode=e.getKeyCode();
        System.out.println("Key Released:"+KeyEvent.getKeyText(keyCode));
    }
}
public class KeyEventDemo
{
    public static void main(String[] args)
    {
        Frame frame=new Frame("Key Events Demo");
        TextField tf=new TextField();
        tf.addKeyListener(new MyKeyListener());
        frame.add(tf);
        frame.setVisible(true);
        frame.setSize(300,200);

    }
}
```

15.MouseListenerExample

```
import java.awt.*;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;

public class MouseListenerExample extends Frame implements MouseListener
{
   Label l;
   MouseListenerExample()
   {
      addMouseListener(this);
      l=new Label();
      l.setBounds(20,50,100,20);
      add(l);
      setSize(300,300);
      setLayout(null);
      setVisible(true);
   }
   public void mouseClicked(MouseEvent e)
   {
      l.setText("Mouse Clicked");
   }
   public void mouseReleased(MouseEvent e)
   {
      l.setText("Mouse Released");
   }
   public void mouseEntered(MouseEvent e)
   {
      l.setText("Mouse Entered");
   }
   public void mouseExited(MouseEvent e)
   {
      l.setText("Mouse Exited");
   }
   public void mousePressed(MouseEvent e)
   {
      l.setText("Mouse Pressed");
   }
   public static void  main(String[] args)
   {
      new MouseListenerExample();
   }

}
```