5 A) Aim :

Write a Java program that creates a user interface to perform integer division.
The user enters two numbers in the text fields, Num1 and Num2.
The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked.
If Num1 and Num2 were not integers, the program would throw a Number Format Exception.
If Num2 were zero, the program would throwan Arithmetic Exception Display the exception in a message dialog box

Description:

- The AWT supports a rich assortment of graphics methods.
- All graphics are drawn relative to a window. this can the main window of an applet, a child window of an applet, or a stand alone application window.
- The origin of each window is at the top-left corner and is 0,0cordinates are specified in pixels.
- All output to a window takes place through a graphics context.

Program:

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

public class Division extends Applet implements ActionListener
{

        TextField       t1,t2,t3;

        Button b;

        Label   L1,L2,L3,L4;

        String s;

        Division e;


        public void init()
        {
                e=this;

                t1=new TextField(10);

                t2=new TextField(10);

                t3=new TextField(10);


                L1=new Label("enter num1");

                L2=new Label("enter num2");

                L3=new Label("Result is");

                L4=new Label("Division of 2 numbers");


                b=new Button("Divide");
                add(L4);
                add(L1);
                add(t1);
                add(L2);
                add(t2);
                add(L3);
                add(t3);
                add(b);

                b.addActionListener(this);

        }

        public void actionPerformed(ActionEvent ae)
        {
                try
                {
                    int num1=Integer.parseInt(t1.getText());
                    int num2=Integer.parseInt(t2.getText());
                    s=""+(num1/num2);

                    t3.setText(s);
                }
                catch(ArithmeticException a)
                {

                    JOptionPane.showMessageDialog(null,"Divide by zero");

                }
                catch(NumberFormatException b)
                {

                    JOptionPane.showMessageDialog(null,"NumberFormateException");
                }
        }
}
```

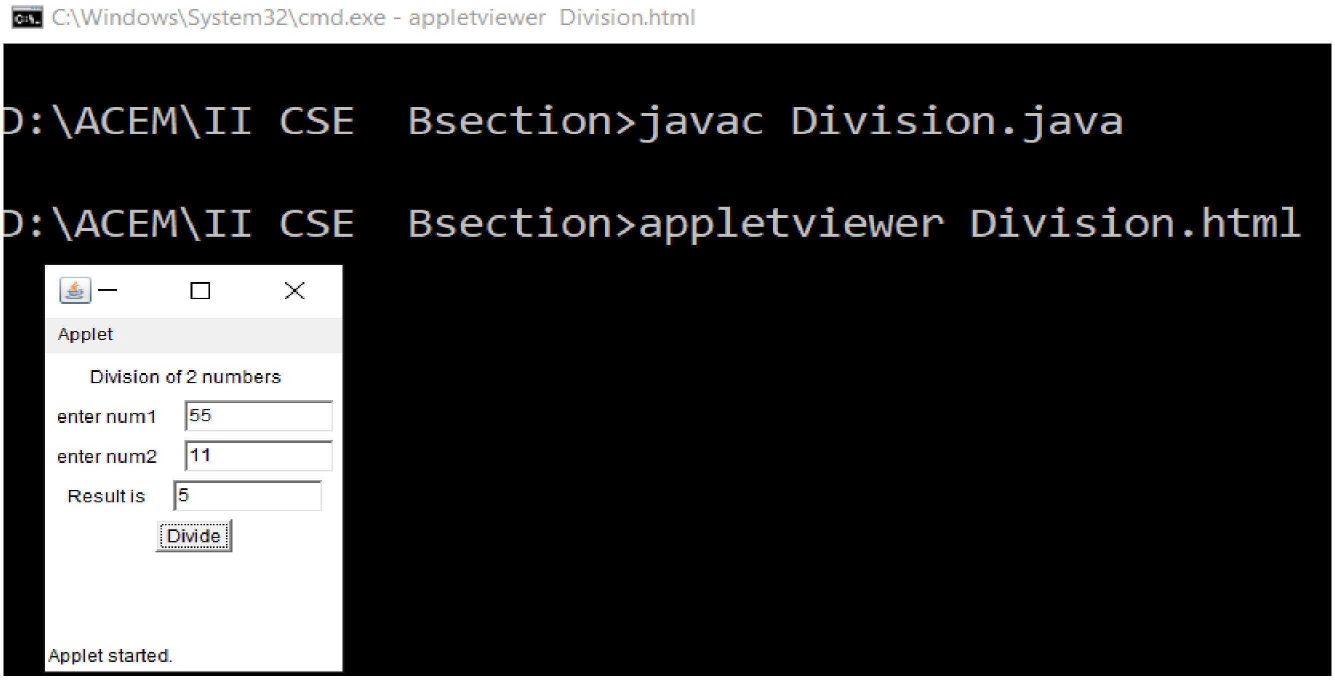
Open →notepad→ type below code and save with file name.html   and open with appletviewer.html

```
<html>

    <applet code="Division.class" height=200 width=320>

    </applet>
 </html>
```

Output:



```
C:\Windows\System32\cmd.exe - appletviewer  Division.html

D:\ACEM\II CSE  Bsection>javac Division.java

D:\ACEM\II CSE  Bsection>appletviewer Division.html
```

Division of 2 numbers

enter num1 55

enter num2 11

Result is 5

Applet started.

5 b) Aim:

Write a Java program that creates three threads. First thread displays Good Morning for every one second, the second thread displays Hello for every two seconds and the third thread displays Welcome for every three seconds.

Description:

- The first thread displays Good Morning every one second, the second thread displays Hello every two seconds and the third thread displays Welcome every three seconds.
- The technique used to implement threads here is by extending the Thread class.
- While executing a thread, if any interrupt occurs then there arises a java exception.
- In such cases, to handle these exceptions thread statements are kept in try-catch blocks.

Program:

```
class A extends Thread
{
        public void run()
        {
                try
                {
                         for(int i=0;i<5;i++)
                        {
                                sleep(1000);
                                System.out.println("good morning");
                        }
                }
                catch(InterruptedException e)
                {
                        System.out.println(e);
                }
    }
}
```

```
class B extends Thread
{
        public void run()
          {
                try
                {
                     for(int i=0;i<5;i++)
                     {
                        sleep(2000);
                        System.out.println("hello");
                      }
                }
                catch(InterruptedException e)
                {
                        System.out.println(e);
                }
    }
}
class C extends Thread
{
        public void run()
        {
                try
                 {
                         for(int i=0;i<5;i++)
                        {
                                sleep(3000);
                                        System.out.println("welcome");
                        }
                }
                catch(InterruptedException e)
                {
                        System.out.println(e);
                  }
         }
}
class MultiThreadDemo
{
        public static void main(String args[])
        {
                 A t1=new A();
                 B t2=new B();
                 C t3=new C();

                 t1.start();
                 t2.start();
                 t3.start();
        }
}
```

Output :

```
D:\ACEM\II CSE  Bsection>javac  MultiThreadDemo.java
D:\ACEM\II CSE  Bsection>java  MultiThreadDemo
good morning
good morning
hello
welcome
good morning
hello
good morning
good morning
welcome
hello
hello
welcome
hello
welcome
welcome      */
```