

Problem:01

Problem Statement: Java Environment Setup .

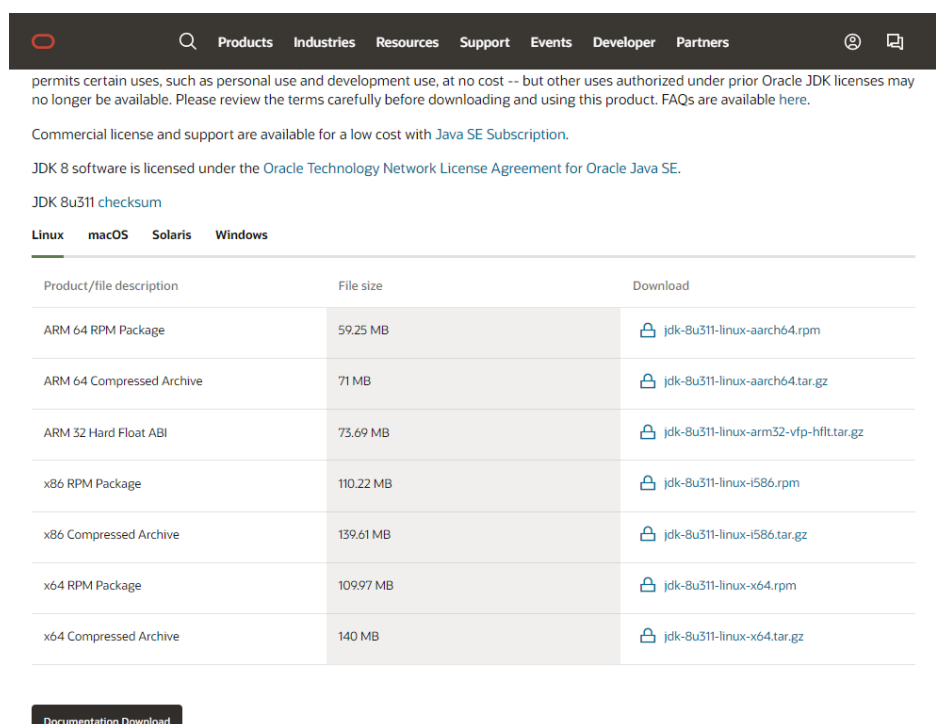
Problem Description: The JDK is a key platform component for building Java applications. At its heart is the Java compiler. The Java Development Kit (JDK) is one of three core technology packages used in Java programming along with the JVM (Java Virtual Machine) and the JRE (Java Runtime Environment). It's important to differentiate between these three technologies, as well as understanding how they're connected:

- The JVM is the Java platform component that executes programs.
- The JRE is the on-disk part of Java that creates the JVM.
- The JDK allows developers to create Java programs that can be executed and run by the JVM and JRE.

Installation Process:

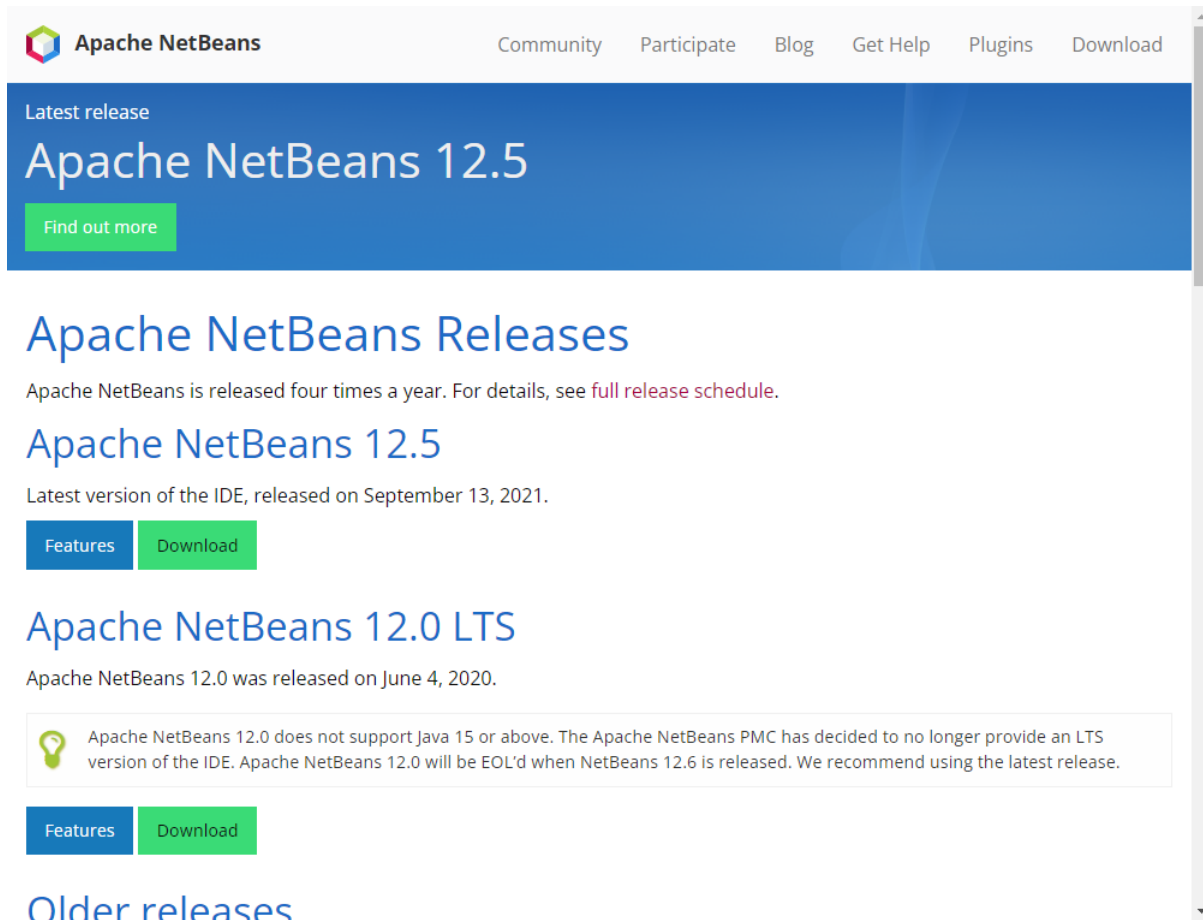
1. Java JDK is available at this link:

<https://www.oracle.com/java/technologies/downloads/#java8>.



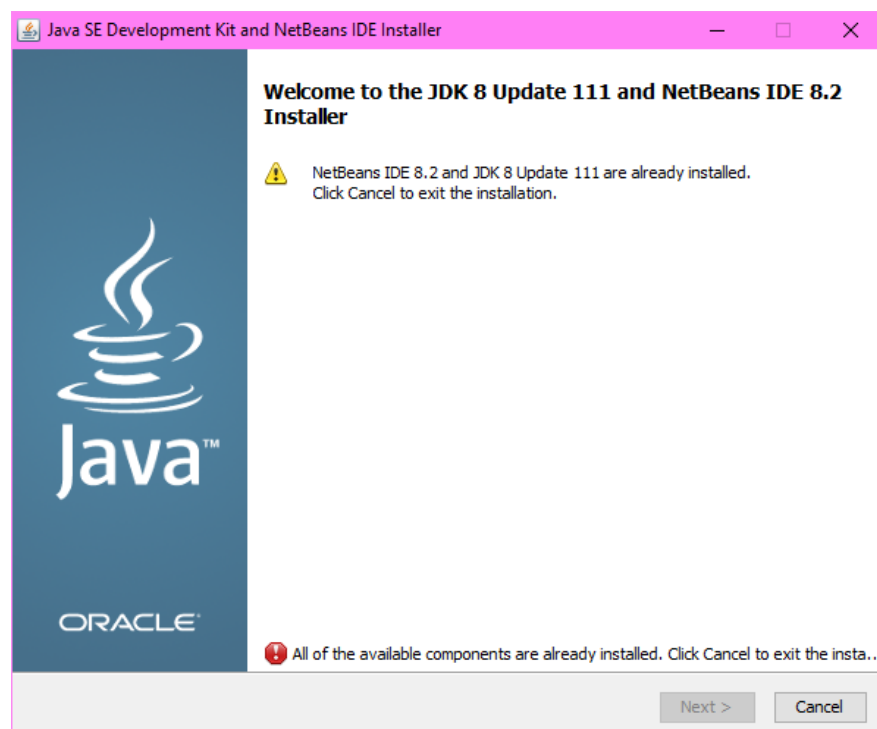
Product/file description	File size	Download
ARM 64 RPM Package	59.25 MB	jdk-8u311-linux-aarch64.rpm
ARM 64 Compressed Archive	71 MB	jdk-8u311-linux-aarch64.tar.gz
ARM 32 Hard Float ABI	73.69 MB	jdk-8u311-linux-arm32-vfp-hflt.tar.gz
x86 RPM Package	110.22 MB	jdk-8u311-linux-i586.rpm
x86 Compressed Archive	139.61 MB	jdk-8u311-linux-i586.tar.gz
x64 RPM Package	109.97 MB	jdk-8u311-linux-x64.rpm
x64 Compressed Archive	140 MB	jdk-8u311-linux-x64.tar.gz

2. After download install it And download NetBeans From this link:
<https://netbeans.apache.org/download/index.html>.

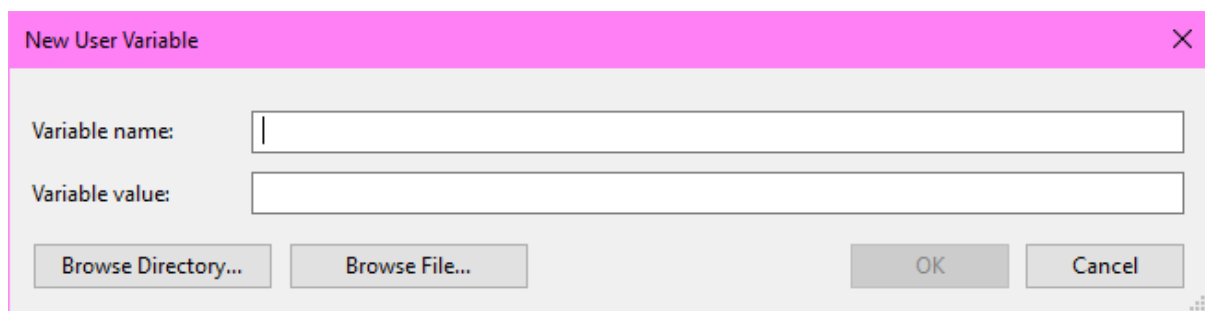
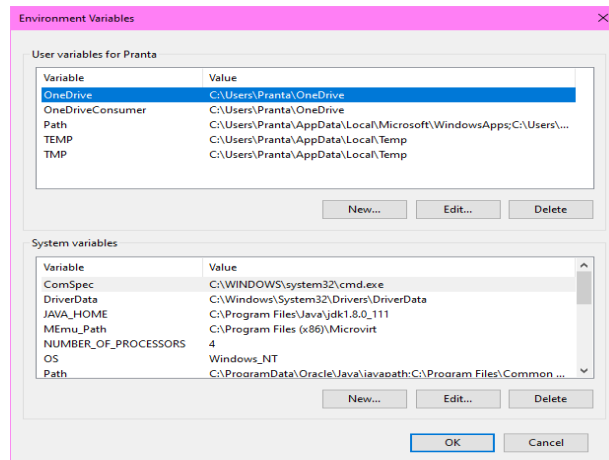
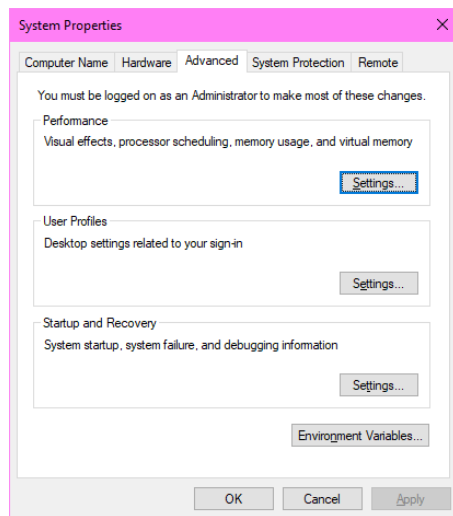


The screenshot shows the Apache NetBeans website. The header includes the Apache NetBeans logo and navigation links: Community, Participate, Blog, Get Help, Plugins, and Download. A blue banner highlights the 'Latest release' as 'Apache NetBeans 12.5' with a 'Find out more' button. Below this, the 'Apache NetBeans Releases' section states that the IDE is released four times a year and provides a link to the 'full release schedule'. It then details 'Apache NetBeans 12.5' as the latest version, released on September 13, 2021, with 'Features' and 'Download' buttons. A section for 'Apache NetBeans 12.0 LTS' notes its release on June 4, 2020, and includes a warning that it no longer supports Java 15 or above and will be EOL'd when NetBeans 12.6 is released. This section also has 'Features' and 'Download' buttons. The 'Older releases' section is partially visible at the bottom.

3.



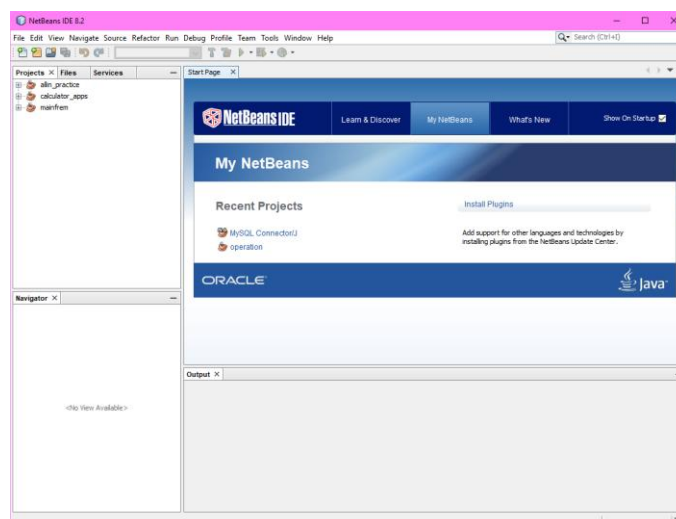
4. Then go to the advanced system settings and setup jdk. And setup new path. And follow the Pictures.



4. Setup Variable Name to JAVA_HOME.

5. Set Variable Value to C:\Program Files\Java\jdk1.8.0_111

6. Our NetBeans-



Problem:02

Problem Statement: Calculator Design & Implementation .

Problem Description: We can develop calculator in java with the help of AWT/Swing with event handling. Let's see the code of creating calculator in java.

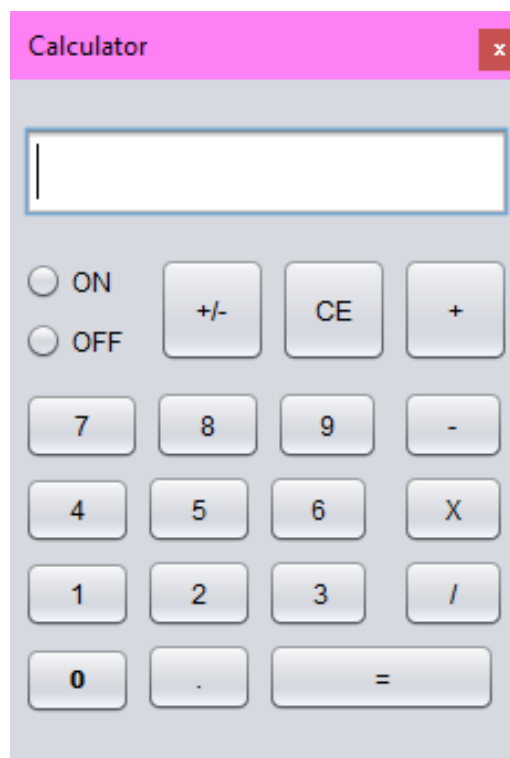
Code:

```
public class calculator extends javax.swing.JFrame {
    double num,ans;
    int calculation;
    public calculator() {
        initComponents();
    }
    public void arithmetic_operation()
    {
        switch(calculation)
        {
            case 1:
                ans=num+ Double.parseDouble(jTextField1.getText());
                jTextField1.setText(Double.toString(ans));
                break;
            case 2:
                ans=num- Double.parseDouble(jTextField1.getText());
                jTextField1.setText(Double.toString(ans));
                break;
            case 3:
                ans=num* Double.parseDouble(jTextField1.getText());
                jTextField1.setText(Double.toString(ans));
                break;
            case 4:
                ans=num/ Double.parseDouble(jTextField1.getText());
                jTextField1.setText(Double.toString(ans));
                break;
        }
    }
}
```

```
    }  
    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {  
        jTextField1.setText("Your calculator is off");  
    }  
    private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {  
        jTextField1.setText(jTextField1.getText() + "8");    // TODO add your  
        handling code here:  
    }  
    private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {  
        jTextField1.setText(jTextField1.getText() + "2");  
    }  
    private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {  
        jTextField1.setText(jTextField1.getText() + "5");  
    }  
    private void jButton12ActionPerformed(java.awt.event.ActionEvent evt) {  
        jTextField1.setText(jTextField1.getText() + ".");    // TODO add your  
        handling code here:  
    }  
    private void jButton14ActionPerformed(java.awt.event.ActionEvent evt) {  
        num=Double.parseDouble(jTextField1.getText());  
        calculation=1;  
        jTextField1.setText("");  
    }  
    private void jButton16ActionPerformed(java.awt.event.ActionEvent evt) {  
        jTextField1.setText("");  
    }  
    private void jButton17ActionPerformed(java.awt.event.ActionEvent evt) {  
        jTextField1.setText(jTextField1.getText() + "+/-"); // TODO add your  
        handling code here:  
    }  
    private void jButton11ActionPerformed(java.awt.event.ActionEvent evt) {  
        jTextField1.setText(jTextField1.getText() + "0");  
    }  
    private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {  
        jTextField1.setText(jTextField1.getText() + "3");    // TODO add your  
        handling code here:  
    }
```

```
private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {  
    jTextField1.setText(jTextField1.getText() +"1");    // TODO add your  
    handling code here:  
}  
private void jButton10ActionPerformed(java.awt.event.ActionEvent evt) {  
    jTextField1.setText(jTextField1.getText() +"4");    // TODO add your  
    handling code here:  
}  
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {  
    jTextField1.setText(jTextField1.getText() +"6");  
}  
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {  
    jTextField1.setText(jTextField1.getText() +"7");  
}  
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {  
    jTextField1.setText(jTextField1.getText() +"9");    // TODO add your  
    handling code here:  
}  
private void jButton19ActionPerformed(java.awt.event.ActionEvent evt) {  
    num=Double.parseDouble(jTextField1.getText());  
}
```

Output:



Problem:03

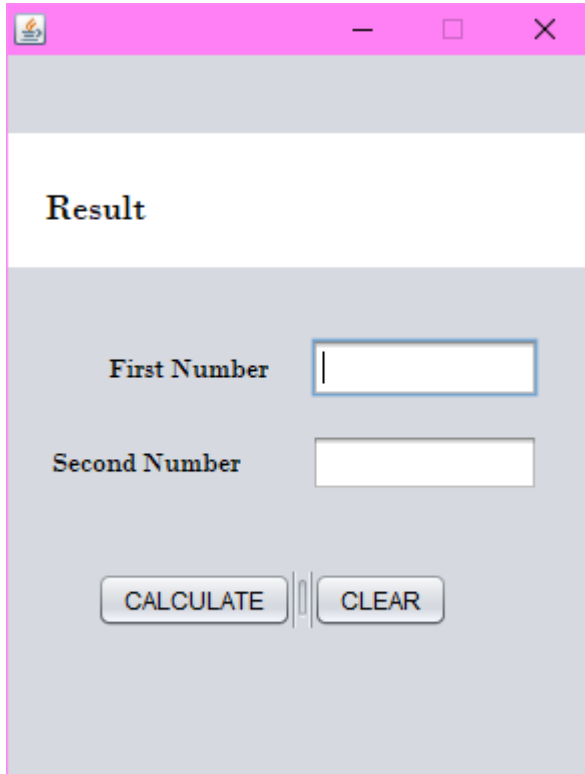
Problem Statement: Write a program that create a user interface to perform integer division.

Problem Description: Division is one of the four basic operations of arithmetic. When dividing two integers, Java uses integer division. In integer division, the result is also an integer. The result is truncated (the fractional part is thrown away) and not rounded to the closest integer. Here we use applet to solve this problem at first we import java swing and create text field and button to perform integer division.

Code:

```
public class main extends javax.swing.JFrame {
    public mainframe() {
        initComponents();
    }
    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        int a=Integer.parseInt(jTextField1.getText());
        int b=Integer.parseInt(jTextField2.getText());
        float result=(float)a/b;
        jLabel2.setText(""+result);
        if(result==0){
            jLabel2.setText("EXCEPTION OF NUMBER");}
        }
        private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
            jLabel2.setText("");
            jTextField1.setText("");
            jTextField2.setText("");
        }
    }
```

Output:

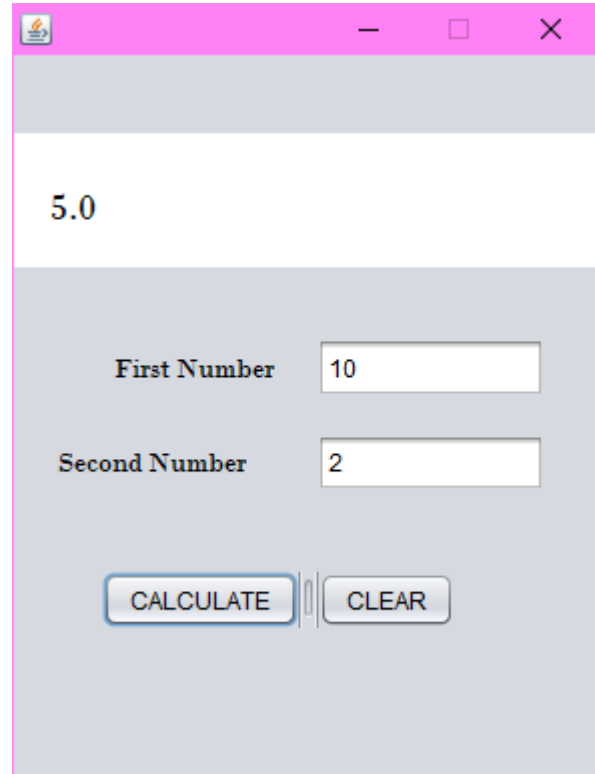


Result

First Number

Second Number

CALCULATE CLEAR

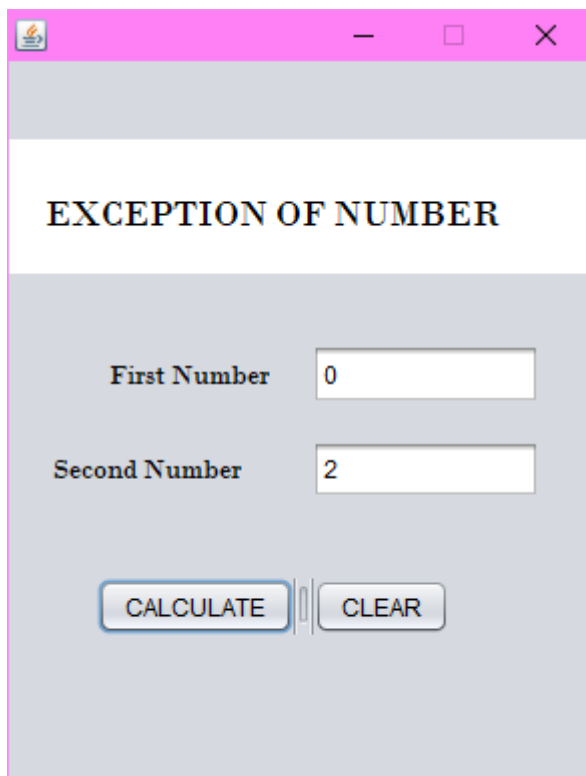


5.0

First Number

Second Number

CALCULATE CLEAR

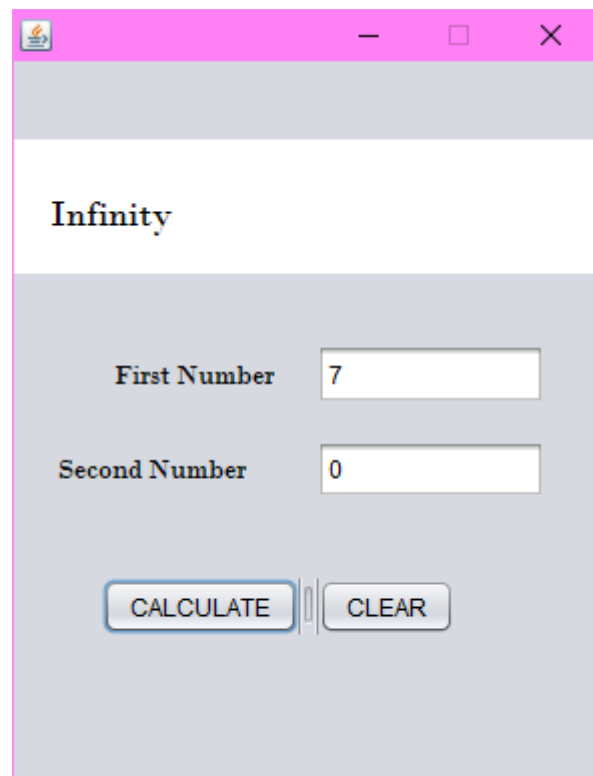


EXCEPTION OF NUMBER

First Number

Second Number

CALCULATE CLEAR



Infinity

First Number

Second Number

CALCULATE CLEAR

Problem:04

Problem Statement: Write A Applet program that print simple message in Java.

Problem Description: we are using a function named g.drawString() which is a method of Graphics class, here g is an object of Graphics class. drawString() is taking three parameters string to display. Here we import

```
import java.applet.*;
```

```
import java.awt.*;
```

Code:

```
package javaapplication14;
```

```
import java.applet.Applet;
```

```
import java.awt.Graphics;
```

```
public class javaapplication14 extends Applet  
{
```

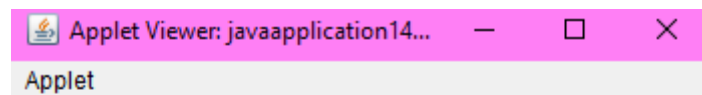
```
    public void paint (Graphics g)
```

```
    {
```

```
        g.drawString ("Welcome To Java ", 100, 100);
```

```
    }
```

```
}
```

Output:

Welcome To Java

Applet started.

Problem:05

Problem Statement: Develop An Digital Clock By Using Applet in Java.

Problem Description: Digital clock can be created by using the Calendar and SimpleDateFormat class. In that problem, getX() and getY() method of MouseEvent is used to get the current x-axis and y-axis. The getGraphics() method of Component class returns the object of Graphics.

```
Code: import java.applet.*;
import java.awt.*;
import java.util.*;
import java.text.*;
```

```
public class DigitalClock extends Applet implements Runnable {
```

```
    Thread t = null;
    int hours=0, minutes=0, seconds=0;
    String timeString = "";
```

```
    public void init() {
        setBackground( Color.green);
    }
```

```
    public void start() {
        t = new Thread( this );
        t.start();
    }
```

```
    public void run() {
        try {            while (true) {
```

```
            Calendar cal = Calendar.getInstance();
            hours = cal.get( Calendar.HOUR_OF_DAY );
            if ( hours > 12 ) hours -= 12;
            minutes = cal.get( Calendar.MINUTE );
            seconds = cal.get( Calendar.SECOND );
```

```
SimpleDateFormat formatter = new SimpleDateFormat("hh:mm:ss");
    Date date = cal.getTime();
    timeString = formatter.format( date );

    repaint();
    t.sleep( 1000 ); // interval given in milliseconds
}
}
catch (Exception e) { }
}
public void paint( Graphics g ) {
    g.setColor( Color.blue );
    g.drawString( timeString, 50, 50 );
}
}
```

Output: