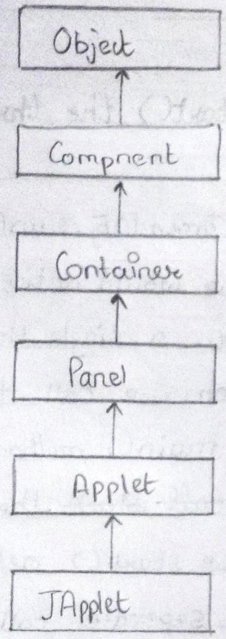
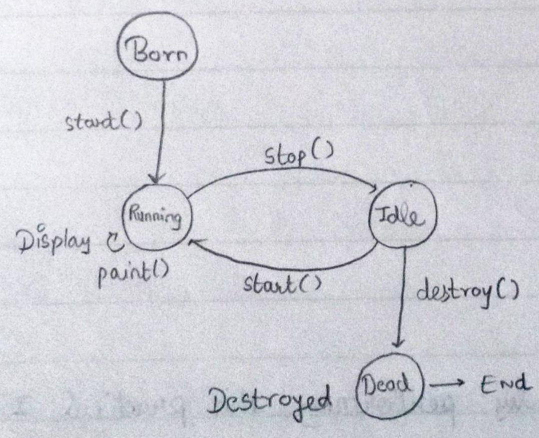
Practical No. 15

Aim: Create, debug and run Java programs based on Applets.

Diagram:



15.1 Hierarchy of Applet.



15.2 Applet Life Cycle

Practical No. 15

Aim: Create, debug and run Java programs based on Applets.

Theory:

What is a java applet?

Applet is a ~~specific~~ special type of program that is embedded in the webpage to generate the dynamic content. It runs inside the browser and works at client side.

What are the advantages of Applet?

- It works at client side, so less response time.
- Can be used by browsers running under various platforms, including Linux, windows, Mac OS X.

What is the drawback of applets?

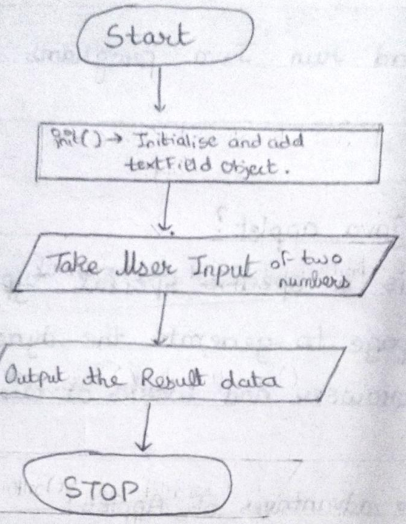
A major drawback of applets is that a plugin is required at client browser to ~~exeut~~ execute applet.

What are the stages of Applet life cycle?

The four states in an applet life cycle are:

1) Born or initialization state
2) Running state
3) Idle state
4) Dead or destroyed state

Flowchart:


Start
init() → Initialise and add textField object.
Take User Input of two numbers
Output the Result data
STOP

Conclusion:

Hence, by performing this practical, I created, debugged and executed java programs based on Applets.

Code:

Input.java

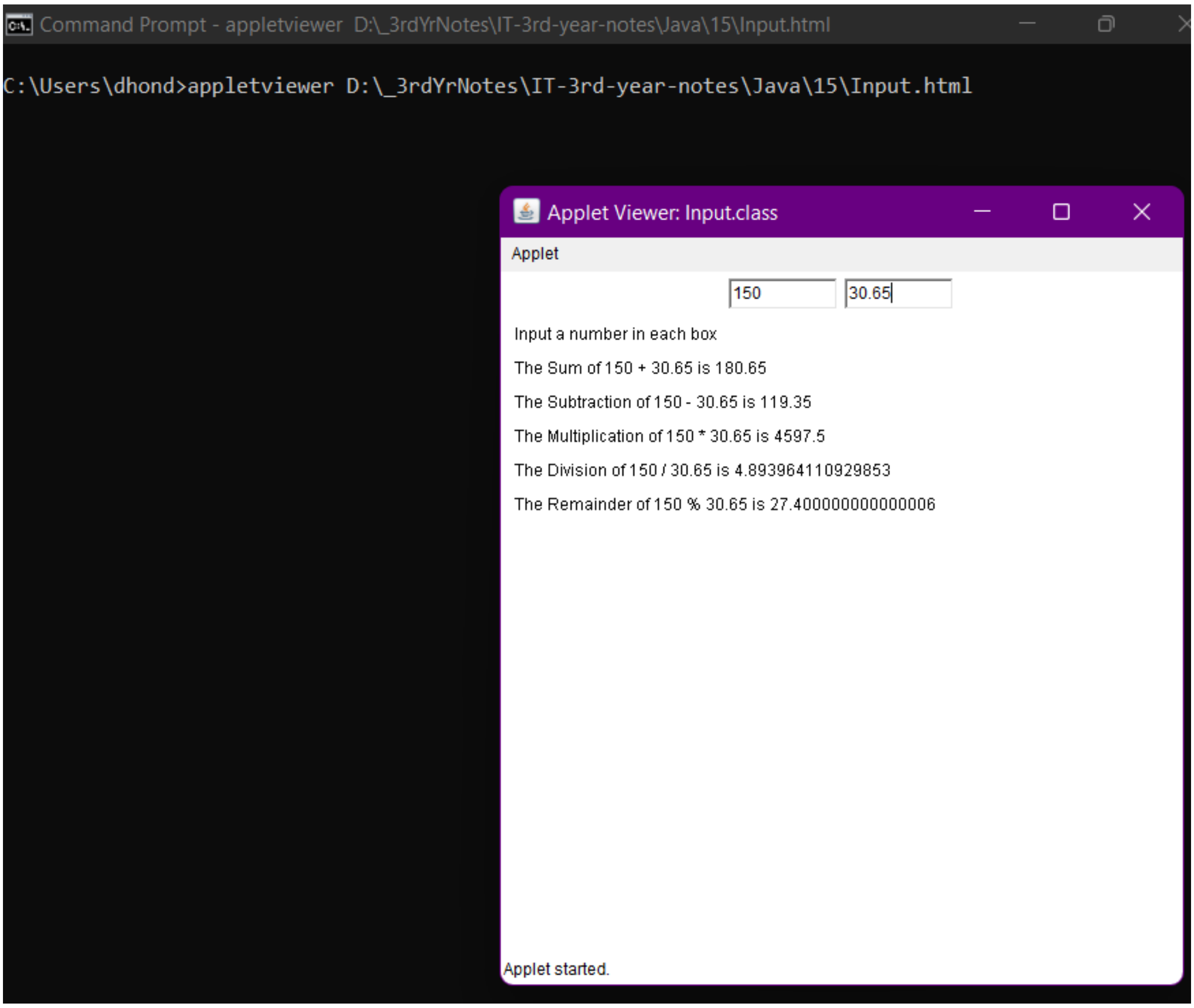
```
import java.awt.*;
import java.applet.*;

/*<applet code="Input.class" height="500" width="500">*/

public class Input extends Applet{
  TextField text1, text2;

  public void init(){
    text1 = new TextField(8);
    text2 = new TextField(8);
    add(text1);
    add(text2);
    text1.setText("0");
    text2.setText("0");
  }

  public void paint(Graphics g){
    double x = 0;
    double y = 0;
    double sum = 0, mul=0, div = 0, remainder = 0, sub = 0;
    String s1="",s2="",s3="";

    g.drawString("Input a number in each box",10,50);
    try{
      s1 = text1.getText();
      x = Double.parseDouble(s1);
      s2 = text2.getText();
      y = Double.parseDouble(s2);
    }catch(Exception e){
      System.out.println("Exception Occured : " + e);
    }finally{
    sum = x + y;
    sub = x - y;
    mul = x * y;
    div = x / y;
    remainder = x % y;

    s3 = String.valueOf(sum);
    String temp = "The Sum of " + s1 + " + " + s2 + " is " + s3;
    g.drawString(temp,10,75);
    temp = "The Subtraction of " + s1 + " - " + s2 + " is " +
String.valueOf(sub);
```

```
    g.drawString(temp,10,100);
    temp = "The Multiplication of " + s1 + " * " + s2 + " is " +
String.valueOf(mul);
    g.drawString(temp,10,125);
    temp = "The Division of " + s1 + " / " + s2 + " is " +
String.valueOf(div);
    g.drawString(temp,10,150);
    temp = "The Remainder of " + s1 + " % " + s2 + " is " +
String.valueOf(remainder);
    g.drawString(temp,10,175);
    }

  }

  public boolean action(Event event, Object object){
    repaint();
    return true;
  }

}
```

Input.html

```
<html>
<applet code="Input.class" height="500" width="500"> </applet>
</html>
```

Output:



Command Prompt - appletviewer D:_3rdYrNotes\IT-3rd-year-notes\Java\15\Input.html
Applet Viewer: Input.class
Applet

```
C:\Users\dhond>appletviewer D:\_3rdYrNotes\IT-3rd-year-notes\Java\15\Input.html
```

```
150    30.65

Input a number in each box
The Sum of 150 + 30.65 is 180.65
The Subtraction of 150 - 30.65 is 119.35
The Multiplication of 150 * 30.65 is 4597.5
The Division of 150 / 30.65 is 4.893964110929853
The Remainder of 150 % 30.65 is 27.400000000000006

Applet started.
```

Applet To Add Two Numbers

Conclusion:

Hence, by performing this practical, I created, debugged and executed Java programs based on Applets.