Chapter – 2

Applet as Java Application

INTRODUCTION TO APPLET:

An applet is a java program that runs in a web browser. Applet is a container class like frame. An applet is a java class that extends the java.applet.Applet class. Applets are designed to be embedded within HTML page. Whenever a user views in HTML page that contains an applet, the code for the applet is downloaded to the user machine. A JVM is required to view an applet. The JVM on the user machine creates an instance of the applet and invokes various method during applet life time.

In order to create a java applet we have to import a package called applet.Applet, for example:

```
import java.applet.Applet;
public class MyApplet extends Applet{
        statements;
}
```

<applet code = “MyApplet.class” height = 300 width = 300></applet>

Save this file as: MyApplet.html or MyApplet.txt or We can write the applet code in the java file as well but should provide in a comment section as below:

```
//<applet code = “MyApplet.class” height = 300 width = 300></applet>
import java.applet.Applet;
public class MyApplet extends Applet{
        statements;
}
```

JVM on the user machine creates an instance of the applet class and invokes various methods during the applet lifetime.

CREATING AN APPLET:

```
import java.awt.*;
import java.applet.Applet;
public class DemoApplet extends Applet{
        public void paint (Graphics g){
                g.drawString("First Applet Program",80,40);
        }
}
```

In order to run this java applet program, we must create a file that should contain a tag called <applet></applet> as mention below:

- <applet code = “DemoApplet.class” height = 300 width = 300></applet>
- File should be saved as DemoApplet.html or DemoApplet.txt

- To run the program, we have a tag called appletviewer such as appletviewer DemoApplet.html or appletviewer DemoApplet.txt.

BASIC METHODS OF APPLET CLASS:

1. void **init**()

This method is called once by the browser or applet viewer when the applet that it has been loaded into the system. It performs the initialization of an applet. Typical examples are initialization of instance variables and GUI components of the applet.

2. void **start**()

This method is called after the **init** method completes execution and every time the user of the browser returns to the HTML on which the applet resides (after browsing to another HTML page). Typical actions include starting an animation or other execution threads.

3. void **paint**(Graphics g)

This method is called automatically every time the applet needs to be repainted. For example, if the user of the browser covers the applet with another open window on the screen, then uncovers the applet, the **paint** method is called. (Inherited from Container)

4. void **stop**()

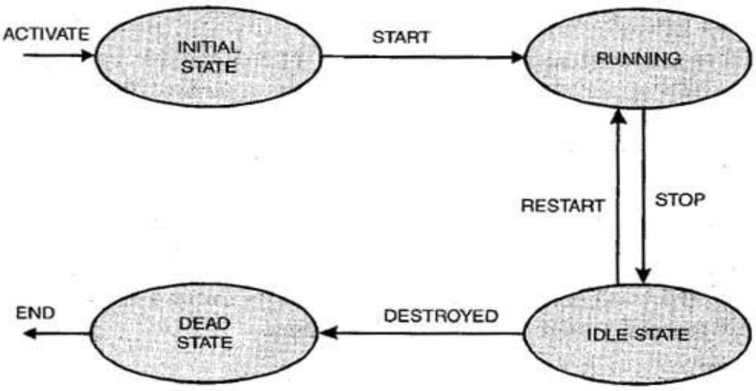
This method is called by the browser or applet viewer when the applet should stop its execution, normally when the user of the browser leaves the HTML page on which the applet resides. Typical actions performed here are to stop execution of animation and threads.

5. void **destroy**()

This method is called by the browser or applet viewer when the applet is being removed from memory, normally when the user of the browser exits the browsing session. This method performs any tasks that are required to destroy any resources that it has allocated.

EXPLAIN APPLET LIFE CYCLE:

Java applet inherits features from the class Applet. Thus, whenever an applet is created, it undergoes a series of changes from initialization to destruction. Various stages of an applet life cycle are depicted in the figure below:



Life Cycle of an Applet

1. **Initial State:**

When a new applet is born or created, it is activated by calling init () method. At this stage, new objects to the applet are created, initial values are set, images are loaded and the colors of the images are set. An applet is initialized only once in its lifetime. It's general form is:

```
public void init( ) {
        //Action to be performed
}
```

2. **Running State:**

An applet achieves the running state when the system calls the start () method. This occurs as soon as the applet is initialized. An applet may also start when it is in idle state. At that time, the start () method is overridden. It's general form is:

```
public void start( ) {
        //Action to be performed
}
```

3. **Idle State:**

An applet comes in idle state when its execution has been stopped either implicitly or explicitly. An applet is implicitly stopped when we leave the page containing the currently running applet. An applet is explicitly stopped when we call stop () method to stop its execution. It's general form is:

```
public void stope
{
        //Action to be performed
}
```

4. **Dead State:**

An applet is in dead state when it has been removed from the memory. This can be done by using destroy () method. It's general form is:

```
public void destroy( ) {
        //Action to be performed
    }
```

DIFFERENCE BETWEEN APPLET AND APPLICATION:

Applet	Application
❖ Small Program	❖ Large Program
❖ Used to run a program on client Browser	❖ Can be executed on standalone computer system
❖ Applet is portable and can be executed by any JAVA supported browser.	❖ Need JDK, JRE, JVM installed on client machine.
❖ Applet applications are executed in a Restricted Environment	❖ Application can access all the resources of the computer
❖ Applets are created by extending the java.applet.Applet	❖ Applications are created by writing public static void main (String [] s) method.
❖ Applet application has 5 methods which will be automatically invoked on occurrence of specific event	❖ Application has a single start point which is main method

❖ Example:
import java.awt.*;
import java.applet.*;
public class Myclass extends Applet
{
public void init() { }
public void start() { }
public void stop() {}
public void destroy() {}
public void paint(Graphics g) {}
}
	❖ Example:
	public class MyClass
	{
	public static void main(String args[]) {}
	}

EVENT HANDLING IN APPLET:

```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;
//<applet code = "EventApplet.java" height = 300 width = 300></applet>

public class EventApplet extends Applet implements ActionListener{
        Button b;
        TextField tf;
        public void init(){
                tf=new TextField();
                tf.setBounds(30,40,150,20);
                b=new Button("Click");
                b.setBounds(80,150,60,50);
                add(b);
                add(tf);
                b.addActionListener(this);
                setLayout(null);
        }
         public void actionPerformed(ActionEvent e){
                tf.setText("Welcome");
         }
}
```