**APPLET**

**Definition**

**An applet is a program written in the Java programming language that can be included in an HTML page, much in the same way an image is included in a page. When you use a Java technology-enabled browser to view a page that contains an applet, the applet's code is transferred to your system and executed by the browser's Java Virtual Machine (JVM).**

**Life Cycle of an Applet**

An applet can react to major events in the following ways:

* It can *initialize* itself.
* It can *start* running.
* It can *stop* running.
* It can perform a *final cleanup*, in preparation for being unloaded.

This section introduces a new applet, Simple, that uses all of these methods. Unlike Java applications, applets do *not* need to implement a mainmethod.

Here is the Simple applet.

**Note:** If you don't see the example running, you might need to enable the JavaScript interpreter in your browser so that the Deployment Toolkit script can function properly.

The following is the source code for the Simple applet. This applet displays a descriptive string whenever it encounters a major milestone in its life, such as when the user first visits the page the applet is on.

import java.applet.Applet;

import java.awt.Graphics;

//No need to extend JApplet, since we don't add any components;

//we just paint.

public class Simple extends Applet {

StringBuffer buffer;

public void init() {

buffer = new StringBuffer();

addItem("initializing... ");

}

public void start() {

addItem("starting... ");

}

public void stop() {

addItem("stopping... ");

}

public void destroy() {

addItem("preparing for unloading...");

}

private void addItem(String newWord) {

System.out.println(newWord);

buffer.append(newWord);

repaint();

}

public void paint(Graphics g) {

//Draw a Rectangle around the applet's display area.

g.drawRect(0, 0,

getWidth() - 1,

getHeight() - 1);

//Draw the current string inside the rectangle.

g.drawString(buffer.toString(), 5, 15);

}

}

**Note:** In this example, the Applet class is extended, not the Swing JApplet class, as Swing components do not need to be added to this applet.

**Loading the Applet**

As a result of the applet being loaded, you should see the text "initializing... starting...". When an applet is loaded, here's what happens:

* An instance of the applet's controlling class (an Applet subclass) is created.
* The applet initializes itself.
* The applet starts running.

**Leaving and Returning to the Applet's Page**

When the user leaves the page, for example, to go to another page, the browser stops and destroys the applet. The state of the applet is not preserved. When the user returns to the page, the browser intializes and starts a new instance of the applet.

**Reloading the Applet**

When you refresh or reload a browser page, the current instance of the applet is stopped and destroyed and a new instance is created.