MIT WORLD PEACE UNIVERSITY

Object Oriented Programming with Java and C++
Second Year B. Tech, Semester 1

APPLET USING JAVA AND HTML

PRACTICAL REPORT
ASSIGNMENT 9

Prepared By

Krishnaraj Thadesar
Cyber Security and Forensics
Batch A1, PA 20

November 28, 2022

Contents

1 Aim and Objectives

Aim

Develop an applet that displays a simple message in centre of the screen

Objectives

1. To understand concept of Java // Krishnaraj Thadesar // Batch A1, PA20 // OOPCJ Assignment 9Applets

2. To explore features of applets to develop web applications

2 Problem Statements

Write a Java applet program that displays a simple message in centre of the screen.

3 Theory

3.1 Introduction to an Applet and its features

Java applets were small applications written in the Java programming language, or another programming language that compiles to Java bytecode, and delivered to users in the form of Java bytecode. The user launched the Java applet from a web page, and the applet was then executed within a Java virtual machine (JVM) in a process separate from the web browser itself. A Java applet could appear in a frame of the web page, a new application window, Sun's AppletViewer, or a stand-alone tool for testing applets.

Java applets were introduced in the first version of the Java language, which was released in 1995. Beginning in 2013, major web browsers began to phase out support for the underlying technology applets used to run, with applets becoming completely unable to be run by 2015-2017. Java applets were deprecated by Java 9 in 2017

An applet is a Java program that can be embedded into a web page. It runs inside the web browser and works at client side. An applet is embedded in an HTML page using the APPLET or OBJECT tag and hosted on a web server. Applets are used to make the website more dynamic and entertaining.

Here are some important points:

- All applets are sub-classes (either directly or indirectly) of java.applet.Applet class.

- Applets are not stand-alone programs. Instead, they run within either a web browser or an applet viewer. JDK provides a standard applet viewer tool called applet viewer.

- In general, execution of an applet does not begin at main() method.

- Output of an applet window is not performed by System.out.println(). Rather it is handled with various AWT methods, such as drawString().

3.2 The Lifecycle of an Applet

Five methods in the Applet class gives you the framework on which you build any serious applet.

1. *init* - This method is intended for whatever initialization is needed for your applet. It is called after the param tags inside the applet tag have been processed.

2. *start* - This method is automatically called after the browser calls the init method. It is also called whenever the user returns to the page containing the applet after having gone off to other pages.

3. *stop* - This method is automatically called when the user moves off the page on which the applet sits. It can, therefore, be called repeatedly in the same applet.

4. *destroy* - This method is only called when the browser shuts down normally. Because applets are meant to live on an HTML page, you should not normally leave resources behind after a user leaves the page that contains the applet.

5. *paint* - Invoked immediately after the start() method, and also any time the applet needs to repaint itself in the browser. The paint() method is actually inherited from the java.awt.

3.3 Discuss about applet tag and its importance

An applet may be invoked by embedding directives in an HTML file and viewing the file through an applet viewer or Java-enabled browser. The <applet> tag is the basis for embedding an applet in an HTML file.

```html
<!DOCTYPE html>
<html>
   <head>
    <title>HTML applet Tag</title>
   </head>
   <body>
    <applet code = "newClass.class" width = "300" height = "200"></applet>
   </body>
</html>
```

It is now only supported on Internet Exporer, Firefox, and Safari. Here is an example using the above attributes in the applet tag.

```html
<!DOCTYPE html>
<html>
  <head>
    <title>HTML applet Tag</title>
  </head>
  <body>
    <applet
      code="Demo.class"
      width="300"
      height="200"
      alt="It is a class"
      align="left"
      name="New Applet"
      title="A Title for the Applet"
      vspace="30px"
    ></applet>
  </body>
</html>
```

3.4 Explain various methods of Applet class with necessary examples.

Attribute	Value	Description
align	URL	*Deprecated - Defines the text alignment around the applet*
alt	URL	Alternate text to be displayed in case browser does not support applet
archive	URL	Applet path when it is stored in a Java Archive ie. jar file
code	URL	A URL that points to the class of the applet
codebase	URL	Indicates the base URL of the applet if the code attribute is relative
height	pixels	Height to display the applet
hspace	pixels	*Deprecated - Defines the left and right spacing around the applet*
name	name	Defines a unique name for the applet
object	name	Specifies the resource that contains a serialized representation of the applet's state.
title	test	Additional information to be displayed in tool tip of the mouse
vspace	pixels	*Deprecated - Amount of white space to be inserted above and below the object.*
width	pixels	Width to display the applet.

4 Platform

Operating System: Arch Linux x86-64
IDEs or Text Editors Used: Visual Studio Code
Compilers : g++ and gcc on linux for C++, and javac, with JDK 18.0.2 for Java

5 Output

The Applet with some text written on it being displayed on the screen.
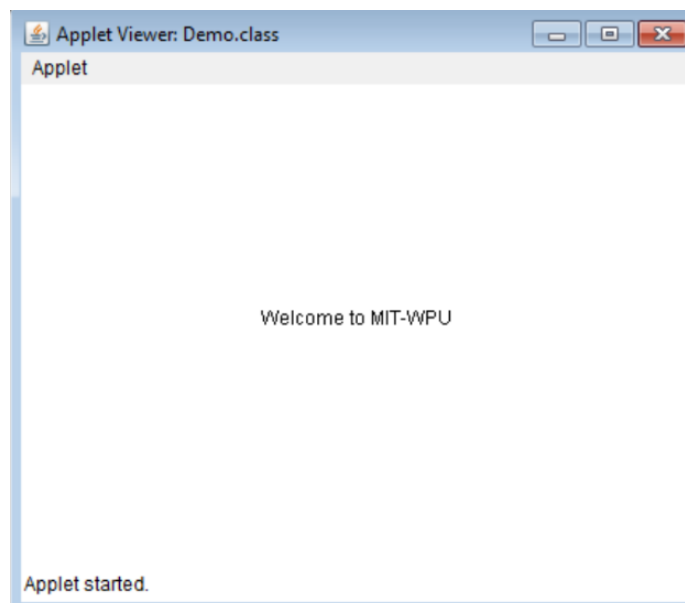

Figure 1:

6   Code

```
1 // Krishnaraj Thadesar
2 // Batch A1, PA20
3 // OOPCJ Assignment 9
4 // Write a Java applet program that displays a simple message in centre of the
      screen.
5
6
7 import java.applet.*;
8 import java.awt.*;
9
10 public class Assignment_9 extends Applet {
11     public void print(Graphics g) {
12         g.drawString("Welcome to Java Applets in Assignment 9", 150, 150);
13     }
14 }
```

Listing 1: applet.java

```
1 <!-- <!DOCTYPE html> -->
2 <html>
3   <head>
4     <title>Assignment 9</title>
5   </head>
6   <body>
7     <applet code="Assignment_9.class" width="300" height="300"></applet>
8   </body>
9 </html>
```

Listing 2: applet.html

7   Conclusion

Thus, developed an applet that displays a simple message in centre of the screen.

8   FAQs

1. *What are the restrictions imposed on Java applets?* Mostly due to security reasons, the following restrictions are imposed on Java applets:

   (a) An applet cannot load libraries or define native methods.

   (b) An applet cannot ordinarily read or write files on the execution host.

   (c) An applet cannot read certain system properties.

   (d) An applet cannot make network connections except to the host that it came from.

   (e) An applet cannot start any program on the host that's executing it.

   (f)

2. *What is the applet class loader, and what does it provide?*

   The class loader is the means by which Java classes and resources are loaded into the JRE. It controls the policies ranging from where to load class definitions to the data format of the class definitions.

A system class loader responsible for loading in the Java runtime, the application, and classes and resources in the application's classpath. An applet class loader is responsible for loading the applets and their related classes and resources, possibly over the network by communicating with a Web serve

When an applet is loaded over the internet, the applet is loaded by the applet classloader. The class loader enforces the Java name space hierarchy. Also, the class loader guarantees that a unique namespace exists for classes that come from the local file system, and that a unique namespace exists for each network source.

When a browser loads an applet over the net, that applet's classes are placed in a private namespace associated with the applet's origin. Then, those classes loaded by the class loader are passed through the verifier.

3. *What is the applet security manager, and what does it provide?* A security manager is an object that defines a security policy for an application. This policy specifies actions that are unsafe or sensitive. Any actions not allowed by the security policy cause a SecurityException to be thrown. An application can also query its security manager to discover which actions are allowed.

Typically, a web applet runs with a security manager provided by the browser or Java Web Start plugin. Other kinds of applications normally run without a security manager, unless the application itself defines one. If no security manager is present, the application has no security policy and acts without restrictions.

4. *Explain the following with suitable examples*

   (a) **Creating an applet**

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

public class Assignment_9 extends Applet {
    public void print(Graphics g) {
        g.drawString("Welcome to Java Applets in Assignment 9", 150, 150)
    ;
    }
}
```

   (b) **Passing parameters to applets**

```java
import java.applet.Applet;
import java.awt.Graphics;

public class UseParam extends Applet{

  public void paint(Graphics g){
  String str=getParameter("msg");
  g.drawString(str,50, 50);
  }

}
```

```html
1  <html>
2  <body>
3  <applet code="UseParam.class" width="300" height="300">
4  <param name="msg" value="Welcome to applet">
5  </applet>
6  </body>
7  </html>
```

(c) Adding graphics and colors to applets.

```java
1  import java.applet.Applet;
2  import java.awt.*;
3
4  public class GraphicsDemo extends Applet{
5
6    public void paint(Graphics g){
7      g.setColor(Color.red);
8      g.drawString("Welcome",50, 50);
9      g.drawLine(20,30,20,300);
10     g.drawRect(70,100,30,30);
11     g.fillRect(170,100,30,30);
12     g.drawOval(70,200,30,30);
13
14     g.setColor(Color.pink);
15     g.fillOval(170,200,30,30);
16     g.drawArc(90,150,30,30,30,270);
17     g.fillArc(270,150,30,30,0,180);
18   }
19 }
20
```

```html
1  <html>
2  <body>
3  <applet code="GraphicsDemo.class" width="300" height="300">
4  </applet>
5  </body>
6  </html>
7
```