UNIT 10 : INTRODUCTION TO APPLETS

UNIT STRUCTURE

10.1 Learning Objectives
10.2 Introduction
10.3 Applets and the World Wide Web
10.4 The Applet Class
10.4.1 Basic Methods of Applet Class
10.5 Applets and HTML
10.6 The Life Cycle of an Applet
10.7 Responding to Events
10.8 Using Window Components
10.9 Adding Audio and Animation
10.10 Let Us Sum Up
10.11 Further Readings
10.12 Answers to Check Your Progress
10.13 Model Questions

10.1 LEARNING OBJECTIVES

After going through this unit, you will be able to:

- write applets in Java
- describe the life cycle of applet
- describe event handling
- add windows components to applets
- add audio file as well as animation to applets.

10.2 INTRODUCTION

In the previous unit, we discussed the **io** package as well as the different **stream** classes in Java. Besides these, we also presented how to read and write console input/output and handle files in Java.

Today internet has become an inevitable part of life. It is used for accessing libraries, getting information on latest happening, to transfer in-

formation, send email and communicate with others. Java has become a prime language today in making web pages interactive and user friendly. Java programs written to run on ***World Wide Web*** (WWW) are known as ***applets***.

In this unit we will learn how to write applets and how to include them in web pages. The life cycle of applet will also be discussed in this unit. It is possible to change the colors, font of the text, set different background colors for different applets and make them look more attractive. In the next unit, we will learn about concepts related to AWT.

10.3 APPLETS AND THE WORLD WIDE WEB

An applet is a Java program that we can embed in a web page. Java applications are run by using a Java interpreter. Applets can run on any browser that supports Java. Applet can also be tested using the applet viewer tool included in the Java Development Kit. In order to run an applet it must be included in a web page, using HTML tags. Since Java's bytecode is platform independent, Java applets can be executed by browsers for many platforms, including Windows, Unix, Mac OS and Linux. When a Java technology-enabled web browser views a page that contains an applet, the applet's code is transferred to the clients system and executed by the browser's Java Virtual Machine (JVM).

Advantages of Applet:

- Applets are cross platform and can run on Windows, Mac OS and Linux platform
- Applets can work all the version of Java Plug-in
- Applets are supported by most web browsers
- Applets are cached in most web browsers, so will be quick to load when returning to a web page.
- User can also have full access to the machine if user allows.

Disadvantages of Applet :

- Java plug-in is required to run applet.
- Java applet requires JVM so first time it takes significant startup time.
- If applet is not already cached in the machine, it will be downloaded from internet and will take time.

- It is difficult to design and build good user interface in applets compared to HTML technology.

10.4 THE APPLET CLASS

The java.applet package is the smallest package in the Java API. The Applet class is contained in the java.applet package. Applet contains several methods that gives us detailed control over the execution of the applet. In addition, java.applet also define three interfaces: AppletContext, AppletStub, and AudioClip.

All applets are subclasses of Applet. Thus, all applets must import java.applet. applets must also import java.awt. all applets run in a window, it is necessary to include support for that window. Applets are not executed by the console-based Java run-time interpreter, they are executed by either a Web Browser or an applet viewer.

The Applet class contains a single default parameterless constructor, which is generally not used. Applets are constructed by the runtime environment when they are loaded and do not have to be explicitly constructed.

Applet provides all necessary support for applet execution, such as starting and stopping. It also provides methods that load and display images, and methods that load and play audio clips.

10.4.1 Basic Methods of Applet Class

void init(): This method is called once by the browser or applet viewer when the applet is loaded into the system. It performs the initialization of an applet.

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).

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)

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.

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.

void play(URL url): If an audio clip is found at the location specified by url, the clip is played.

void resize(Dimension dim): Resizes the applet according to the dimensions specifiedby dim. Dimension is a class stored inside java.awt. It contains two integer fields: width and height.

10.5 APPLETS AND HTML

In order to run a Java applet, it is first necessary to have a web page that refer the applet. The <Applet> tag supplies the name of the applet to be loaded and tells the browser how much space the applet requires. Applet tag takes zero or more parameters.

Syntax :

<APPLET

CODE=”name of the class file that extends java.applet.Applet”

CODEBASE=”path of the class file” HEIGHT=”maximum height of the applet, in pixels”

WIDTH=”maximum width of the applet, in pixels”

VSPACE=”vertical space between the applet and the rest of the HTML”

HSPACE=”horizontal space between the applet and the rest of the HTML”

ALIGN=”alignment of the applet with respect to the rest of the web page”

ALT="alternate text to be displayed if the browser does not support applets">

<PARAMNAME="parameter_name" value="value_of_parameter">

</APPLET>

The most commonly used attributes of the Applet tag are CODE, HEIGHT, WIDTH, CODEBASE and ALT. Also you can send parameter to the applet using the PARAM tag. The PARAM tag must be written between <APPLET> and </APPLET> .

CODE : This is required attribute that gives the name of the file containing your applet's compiled in class file.

CODEBASE : This is an optional attribute, that specifies the base URL of the applet code.

HEIGHT AND WIDTH : These two are required attributes that gives the size in pixels of the applet display area.

VSPACE AND HSPACE : These attributes are optional. VSPACE specifies the space, in pixels, above and bellow the applet. HSPACE specifies the space, in pixels, on each side of the applet.

ALIGN : This is an optional attribute that specifies the alignment of the applet. The possible values are LEFT, RIGHT, TOP, BOTTOM, MIDDLE, BASELINE, TEXTTOP, ABSMIDDLE and ABSBOTTOM.

ALT : This is also an optional attribute used to specify a short text message that should be displayed if the browser can not run the Java Applet.

PARAM NAME AND VALUE : the PARAM tag allows to specify applet specific arguments in an HTML page.

//***Program 10.1 :*** Program for Simple Applet displaying **"HELLO WORLD"**

```
import java.awt.Graphics;
/*This imports our Graphics class, which is used for drawing lines, squares,
        circles, text.*/
public class SimpleJavaApplet extends java.applet.Applet;
/*This means that our custom HelloWorld class extends the Applet class.
That is , it is a subclass of Applet. Therefore we have access to all the
methods of Applet, and we can extend our custom class to do further
things.*/
```

```
        public void paint(Graphics g);
/*This method is our drawing method. This method draws anything that is
            in our applet to the screen. Note that we have passed an
            instance(g) of the Class Graphics to it.*/
        g.drawString("HELLO WORLD", 5, 25);
/*drawString is a method of the Graphics class. We have an instance of the
            Graphics class - g. drawString takes a String and an x
            and a y coordinate.*/
        }
    }
```

simpleapplet.html

```
<html>
<head>
<body>
<APPLET CODE="SimpleJavaApplet.class" HEIGHT=
250 WIDTH=250 >
</APPLET>
</body>
</html>
```

Both this file needs to stay in the same directory.

The HTML (*Hyper Text Marked up Language*) code tells the browser to load the compiled java applet SimpleJavaApplet.class that is in the same directory as this HTML file. It specifies the display area for the applet output as 250 pixels width and 250 pixels height.

To run an applet, we require one of the following tools:

a) Java Applet Viewer

b) Java enabled Web Brower (such as IE or Firefox)

Before running the program we need to compile the program. The next section demonstrate how to compile and run an applet in a graphical way. The example shows here are keep in a directory under c:\html.

a) Using Java Applet Viewer:

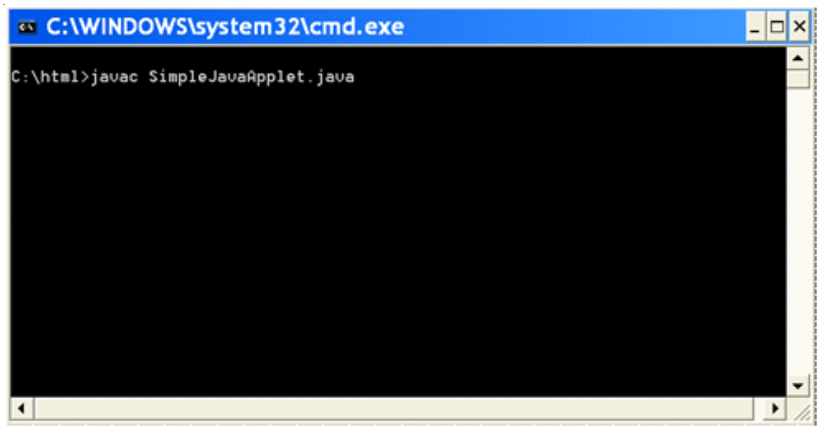


Figure 10.1 : Compiling the Java program SimpleJavaApplet.java

The applet viewer is available as a part of the JDK. We can now run our applet as follows:

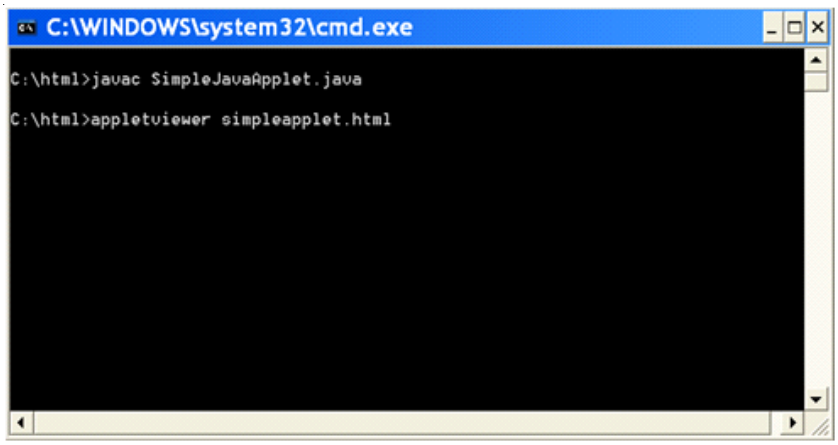


Figure 10.2 : Running applet using Applet ViewerOutput

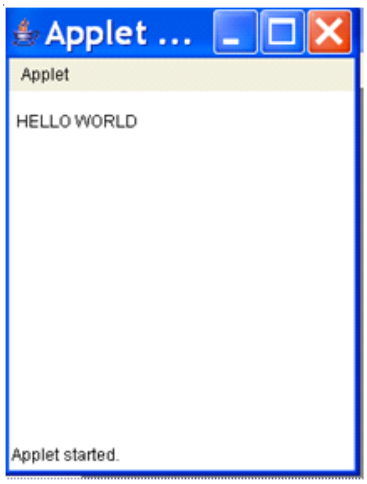


Figure 10.3: Applet view

When the browser encounters an <Applet> tag, it reserves a display area of the specified width and height for the class in the browser window, loads the bytecodes for the specified Applet subclass, creates an instance of the subclass and then calls for the instances init and start method.

b) Using Java enabled browser (Firefox) : Run the browser and then go to File->Open File , it will open a screen looks bellow. Select the file simpleapplet.htm and click on open.

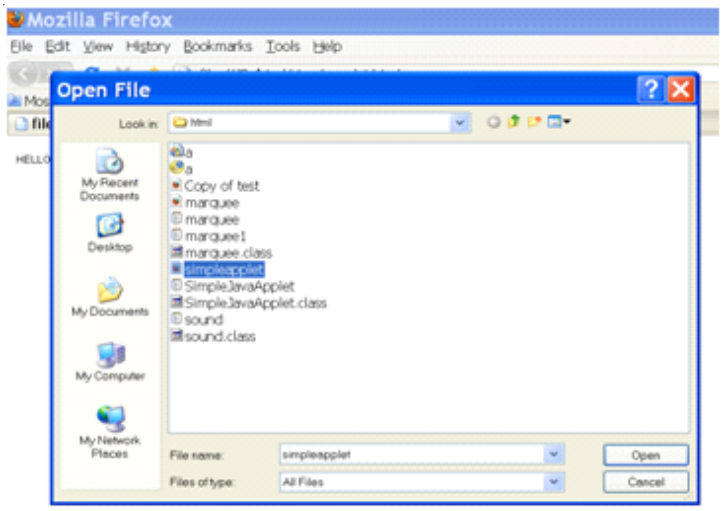


Figure 10.4 : Select the html file that included the applet class

The applet is run on the browser and displays the word “HELLO WORLD” as bellow.

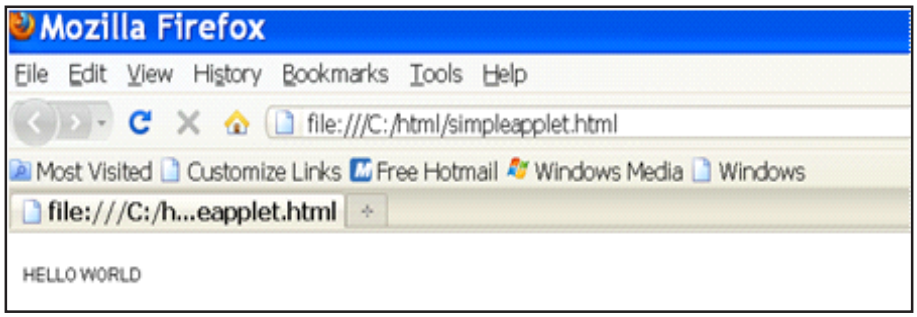


Figure 10.5 : Run the applet on the browser

We can embed applets into Web pages in two ways.

- One way is that we can write our own applets and embed them into web pages. An applet developed locally and stored in a local system is known as a local applet.
- Secondly, we can download an applet from a remote computer system and then embed it into a web page. A remote applet is that which is developed by someone else and stored on a remote computer (web server) connected to the Internet. In order to locate and load a remote applet, we must know the applets address on the web. This address is known as the **URL** and must be specified in the applet's HTML document as the value of the CODEBASE attribute.

Drawing Shapes Example : This program will see how to draw different types of shapes like line, circle and rectangle. There are different types of methods in the **Graphics** class of the *java.awt.*;* package which have been used to draw the appropriate shape.

Here is the java code of the program :

//**Program 10.2 :** Program to show drawing of different shapes

```
      import  java.applet.*;
      import  java.awt.*;
      public  class   CircleLine  extends  Applet
{
      int  x=300,y=100,r=50;
      public  void  paint(Graphics  g)
{
      g.drawLine(3,300,200,10);
// used to draw the line in the applet.
      g.drawString("Line",100,100);
 // draws the given string as the parameter
      g.drawOval(x-r,y-r,100,100);
// draws the circle
      g.drawString("Circle",275,100);
// draws the given string as parameter
```

```
        g.drawRect(400,50,200,100); // draws the rectangle
        g.drawString("Rectangel",450,100);
//draws the given string asparameter
            }
        }
```

Here is the HTML code of the program :

CircleLine.html :

```
<HTML>
<HEAD>
</HEAD>
<BODY>
<div align="center">
<APPLETCODE="CircleLine.class"
WIDTH="800"HEIGHT="500"></APPLET>
</BODY>
</HTML>
```

Output :

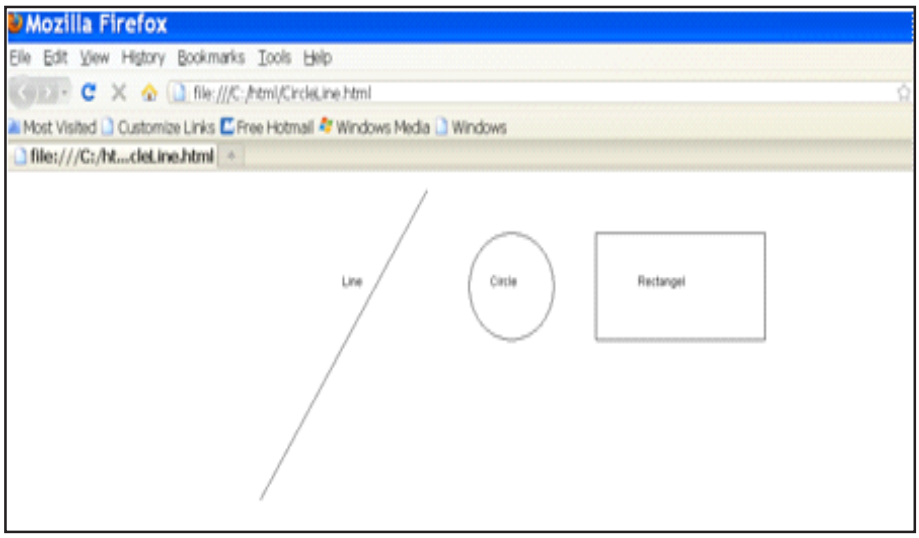


Figure 10.6 : Output of program 10.2



CHECK YOUR PROGRESS

Q1. Fill in the blanks :

i) The _____ packages is the smallest package in the Java API.

ii) _____ is used to execute an Applet.

iii) The Applet tag is written the _____ of an HTML document.

Q2. State whether the following statements are true or false:

a) Applet can be run using Applet Viewer.

b) Applet are not supported by web browser.

c) Applet can run on Linux platform.

10.6 THE LIFE CYCLE OF AN APPLET

In this section we will learn about the lifecycle of an applet. Applet runs in the browser and its lifecycle method are called by JVM when it is loaded and destroyed. The lifecycle of an Applet are:

- init(): This method is called to initialized an applet
- start(): This method is called after the initialization of the applet.
- stop(): This method can be called multiple times in the life cycle of an Applet.
- destroy(): This method is called only once in the life cycle of the applet when applet is destroyed.

init() : The life cycle of an applet is begin on that time when the applet is first loaded into the browser and called the init() method. The init() method is called only one time in the life cycle on an applet. The init() method is basically called to read the PARAM tag in the html file. The init () method retrieve the passed parameter through the PARAM tag of html file using get Parameter() method All the initialization such as initialization of variables and the objects like image, sound file are loaded in the init () method. After

the initialization of the init() method user can interact with the Applet and most applet contains the init() method.

start(): The start method of an applet is called after the initialization method init(). This method may be called multiples time when the Applet needs to be started or restarted. For example if the user wants to return to the Applet, in this situation the start() Method of an Applet will be called by the web browser and the user will be back on the applet. In the start method user can interact within the applet.

stop(): The stop() method can be called multiple times in the life cycle of applet like the start() method or should be called at least one time. There is only minor difference between the start() method and stop () method. For example the stop() method is called by the web browser at the time when the user leaves one applet to go another applet and the start() method is called at the time when the user wants to go back into the first program or Applet.

destroy(): The destroy() method is called only one time in the life cycle of Applet like init() method. This method is called only on that time when the browser needs to shutdown. Fig.10.7 depicts the life cycle of an applet :

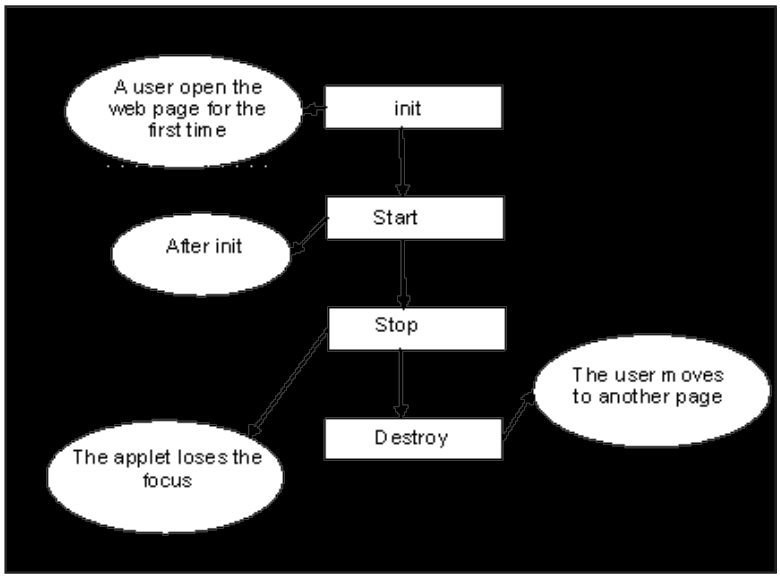


Figure 10.7 : Life cycle of applet

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, 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.

Quitting the browser : When the user quits the browser, the applet has the opportunity to stop itself and perform a final cleanup before the browser exits.



CHECK YOUR PROGRESS

Q3. Fill in the blanks :

i) The _____ method is called the first time an applet is loaded into the memory of a computer.

ii) The _____ method is called by the browser when the user moves to another page.

10.7 EVENT HANDLING

Event handling is essential to GUI programming. The program waits for a user to perform some action. The user controls the sequence of operations that the application executes through a GUI. This approach is called event driven programming.

- **Components of an event:** An event comprises of three components.
  - **Event Object:** When an user interacts with an application by clicking a mouse or pressing a key from the keyboard, an event is generated. The operating system keep track of this event and the data associated with it. For example the time at which the event occurred, the event type (like a mouse click).
  - In Java events are represented by objects that describes the events themselves. Java has a number of classes that describes and handle different categories of event.
  - **Event Source :** An event source ia an object that generates an event. For example if you click on a button, an ActionEvent object is generated. The object of the ActionEvent class contains information about the event.
  - **Event-handler :** An event handler is a method that understands the event and processes it. The event-handler method takes an Event object as a parameter.
- **Event Classes :** The EventObject class is at the top of the event class hierarchy. It belongs to the java.util package. Most other event classes are present in the java.util.event package.
- **Event Listener :** An object delegates the task of handling an event to an event listner. When an event occurs, an event object of the appropriate type is created. This object is passed to the listener. A listener must implement the interface that has the method for event-handling. A component can have multiple listeners.
- **Event-Handling :** When an event occurs, it is sent to the component from where the event originated. The component registers a listener, which contains event-handler. Event-handler receives and process events.

Every event has a corresponding listener interface that specifies the methods that are required to handle the event. Event object are reported to registered listener. To enable a component to handle events, you must register an appropriate listener for the components. Fig.10.8 depicts the event handling process.