14.2 How to Build and Run Applet Code

Steps to build and run applet code are :

(i) **Build an applet Code**

Example :

Write a simple applet "FirstApplet" as :

```
import java.awt.*;
import java.applet.*;
public class FirstApplet extends Applet
{
    public void paint (Graphics g)
    {
        g.drawString ("My First Applet", 120, 120);
    }
}
```

Note that **paint( )** method is defined by the AWT and this method is used to display the result of applet code on the screen. The output may be text, graphics or sound. The **paint( )** method, which requires a **Graphics** object as an argument is defined as

public void paint (Graphics g)

The statement

g.drawString ("My First Applet", 120, 120);

displays the string

My First Applet

on the screen at the position of 120, 120 (pixels) as shown in figure 14.4.

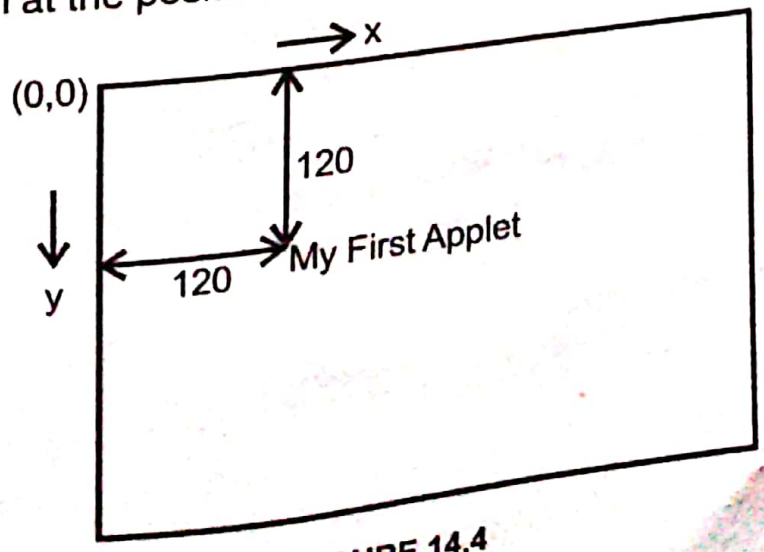


FIGURE 14.4

(ii) Create an Executable Apple (.class file)

Now compile the above applet code to create an executable applet. Therefore, write the following command in command prompt.

c : \>javac FirstApplet.java

This command will create **FirstApplet.class** file.

(iii) Write HTML Code to Embed Applet

Write the following HTML code to embed the applet in web page.

```
<HTML>
   <HEAD>
      <TITLE>First Applet </TITLE>
   </HEAD>
   <BODY>
      <HI> My First Applet Program </HI>
      <APPLET CODE = "FirstApplet.class"
         Width = 400
         Height = 180>
      </APPLET>
   </BODY>
</HTML>
```

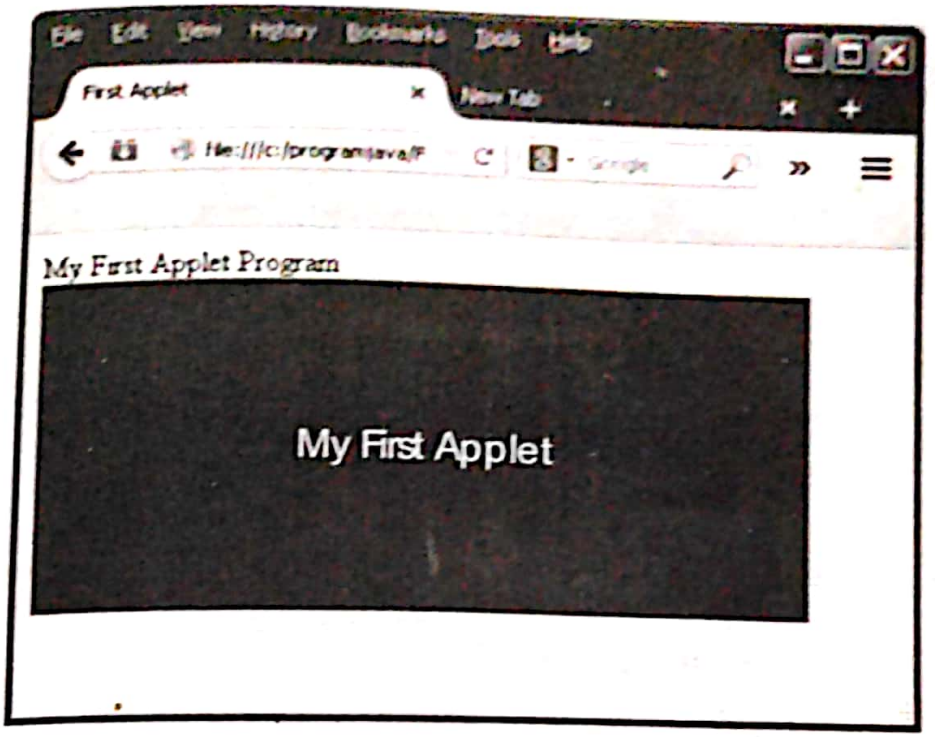
Note that <Applet> tag is used in the body section. The <APPLET> tag supplies the name of the applet to be loaded, width and height (in pixels) of the applet window on the web page (or width and height of applet viewer window). Save this file as **FirstApplet.HTML** and save it in the same directory as the compiled applet.

(iv) Runing the Applet

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

(a) Java-enabled web browser (such as Internet explorer or Netscape)

Open the FirstApplet.html in Internet explorer, we get the following output.

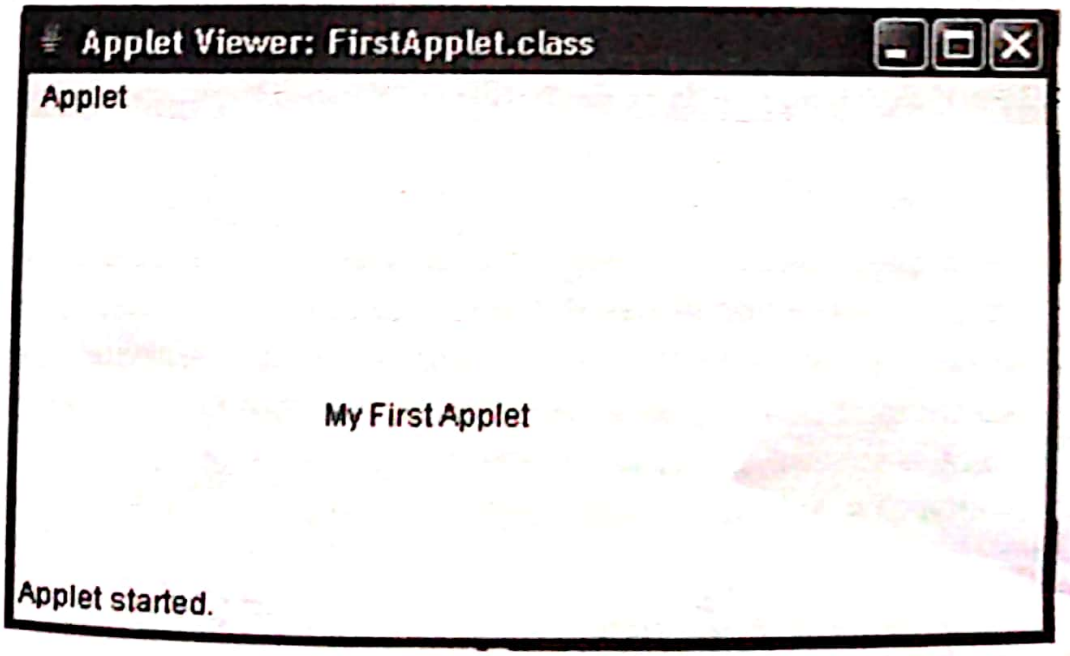


(b) Java AppletViewer

Write the following command to the command prompt

c:\>appletviewer FirstApplet.html

Output is



Note that, the **appletviewer** is not a full-fledged Web browser and therefore it ignores all of the HTML tags except the part pertaining to the running of the applet.

Note :

To Create and run an Applet program follow the steps :

1. Create a Java file containing Applet Code and Compile it.
2. Create a HTML file and embed the .Class File of the Java file created in the first step
3. Run Applet using either of the following methods
   - Open the HTML file in java enabled web browser
   - Use AppletViewer tool (used only for testing purpose)

14.3 Applet Architecture

An applet is a window-based program. Its architecture is different from the simple console-based programs. There are a few key concepts that the user must understand.

First, applets are **event driven**. An applet resembles a set of interrupt service routines. An applet waits until an event occurs. The AWT notifies the applet about an event by calling an event handler that has been provided by the applet. Once this is done, the applet must take an appropriate action and then quickly return control to the AWT. This is a very important point. The applet should not enter a "mode" of operation in which it maintains control for an extended period. Rather, it must perform some specific actions in response to the events and then return control to the AWT run-time system. In the cases where the applet needs to perform a repetitive task on its own, we must start an additional thread of execution.

Secondly, the user initiates interaction with an applet. The user interacts with the applet as and when required. These interactions are sent to the applet as events to which the applet must respond. Applets contain various controls, such as push buttons and check boxes. When the user interacts with one of these controls, an event is generated. While the architecture of an applet is not as easy to understand as that of a console-based program, Java's AWT makes it as simple as possible.