

20

Introduction to Java Applets



Observe due measure, for right timing is in all things the most important factor.

— Georg Wilhelm Friedrich Hegel

Painting is only a bridge linking the painter's mind with that of the viewer.

— Eugene Delacroix

The direction in which education starts a man will determine his future in life.

— Plato



OBJECTIVES

In this chapter you will learn:

- To differentiate between applets and applications.
- To observe some of Java's exciting capabilities through the JDK's demonstration applets.
- To write simple applets.
- To write a simple HyperText Markup Language (HTML) document to load an applet into an applet container and execute the applet.
- Five methods that are called automatically by an applet container during an applet's life cycle.



- 20.1 Introduction**
- 20.2 Sample Applets Provided with the JDK**
- 20.3 Simple Java Applet: Drawing a String**
 - 20.3.1 Executing an Applet in the appletviewer**
 - 20.3.2 Executing an Applet in a Web Browser**
- 20.4 Applet Life-Cycle Methods**
- 20.5 Initializing an Instance Variable with Method `init`**
- 20.6 Sandbox Security Model**
- 20.7 Internet and Web Resources**
- 20.8 Wrap-Up**



20.1 Introduction

- **Applets**
 - **Java programs that can be embedded in HyperText Markup Language (HTML) documents**
 - **The browser that executes an applet is generically known as the applet container**



20.2 Sample Applets Provided with the JDK

- **Demonstration applets provided with the JDK**
 - **Demonstration programs are located in directory demo**
 - **Default location in Windows:**
C:\Program Files\Java\jdk1.5.0\demo
 - **Default location in UNIX/Linux/Mac OS X:**
**the directory in which you install the JDK followed by
jdk1.5.0/demo**
 - **JDK and the demos can be downloaded from the Sun Microsystems Java Web site**
 - **java.sun.com/j2se/5.0/**



20.3 Simple Java Applet: Drawing a String

- **Creating the applet class**
 - An applet container can create only objects of classes that are **public** and extend **JApplet**
 - An applet container expects every Java applet class to have methods named **init**, **start**, **paint**, **stop** and **destroy**
 - These methods are inherited from class **JApplet** and can be overridden
 - When an applet container loads an applet class, the container creates an object of the class then calls methods **init**, **start** and **paint**



Outline

```

1 // Fig. 20.6: welcomeApplet.java
2 // A first applet in Java.
3 import java.awt.Graphics; // program uses class Graphics
4 import javax.swing.JApplet; // program uses class JApplet
5
6 public class welcomeApplet extends JApplet
7 {
8     // draw text on applet's background
9     public void paint( Graphics g )
10    {
11        // call superclass version of method paint
12        super.paint( g );
13
14        // draw a String at x-coordinate 25 and y-coordinate 25
15        g.drawString( "Welcome to Java Programming!", 25, 25 );
16    } // end method paint
17 } // end class welcomeApplet

```

Import **Graphics** and **JApplet**

welcomeApplet
.java

Class **WelcomeApplet**
extends class **JApplet**

Call the superclass version of
method **paint**

Use **Graphics** method **drawString** to draw
Welcome to Java Programming!



welcomeApplet executing in the appletviewer



welcomeApplet executing in Microsoft Internet Explorer

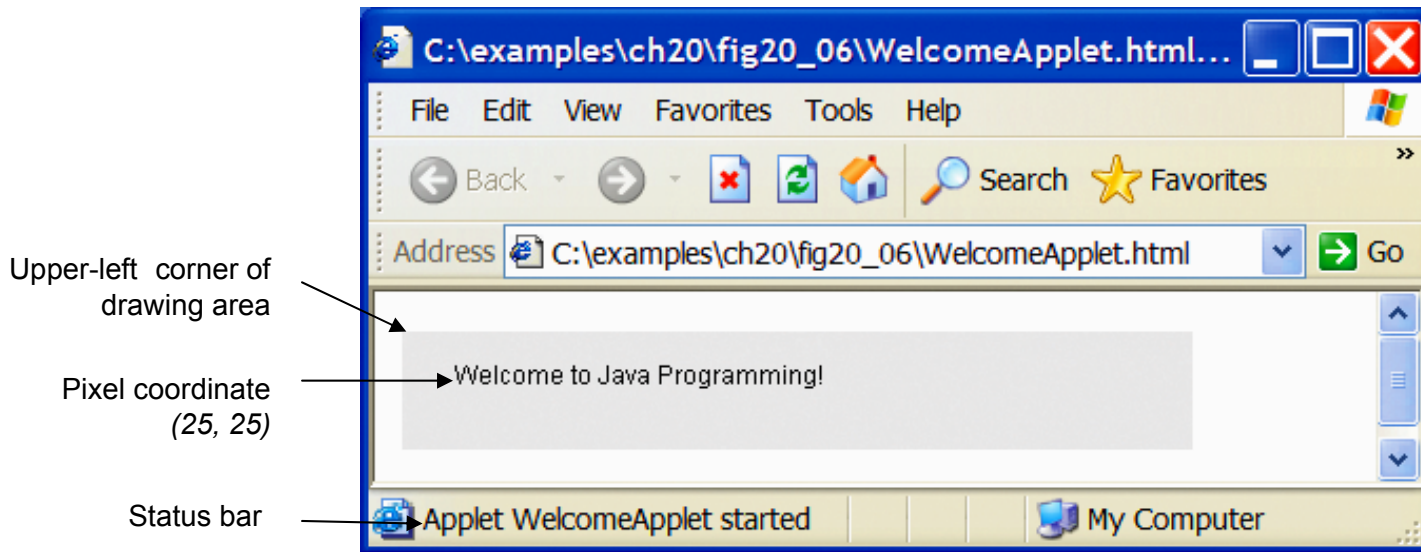


Fig. 20.7 | Sample outputs of the welcomeApplet in Fig. 20.6.



20.3 Simple Java Applet: Drawing a String (Cont.)

- **Overriding method `paint` for drawing**
 - The applet container calls method `paint` with a `Graphics` object as an argument to tell the applet when to draw



20.3.1 Executing an Applet in the appletviewer

- **Applets are embedded in Web pages for execution in an applet container**
 - **Before executing the applet, you must create an HTML document that specifies which applet to execute**
 - **HTML documents typically end with an “.html” or “.htm” file-name extension**
 - **Most HTML elements are delimited by pairs of tags**
 - **All HTML tags begin with a left angle bracket, <, and end with a right angle bracket, >**
 - **Execute WelcomeApplet in the appletviewer**
 - **In the directory containing your applet and HTML document, type `appletviewer WelcomeApplet.html`**
 - **The appletviewer understands only the `<applet>` and `</applet>` HTML tags and ignores all other tags**



Outline

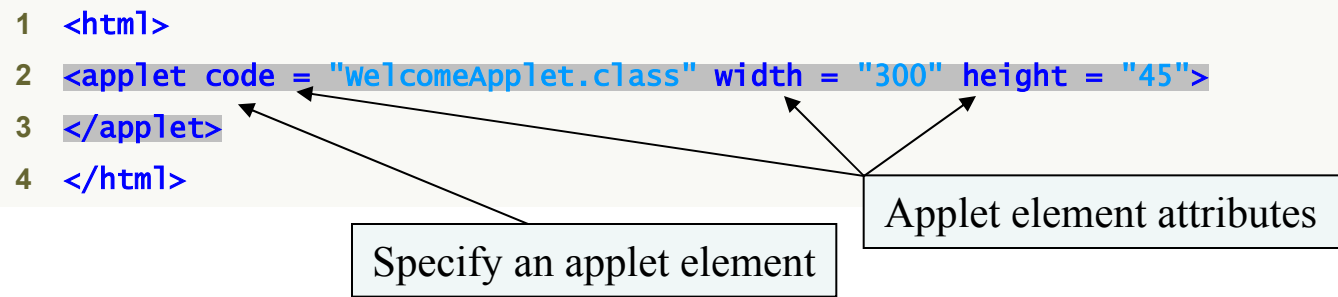


Fig. 20.8 | `welcomeApplet.html` loads `welcomeApplet` (Fig. 20.6) into an applet container.



Error-Prevention Tip 20.1

If you receive a `MissingResourceException` error message when loading an applet into the appletviewer or a browser, check the `<applet>` tag in the HTML document carefully for syntax errors, such as commas (,) between the attributes.



Error-Prevention Tip 20.2

Test your applets in the appletviewer applet container before executing them in a Web browser. Browsers often save a copy of an applet in memory until all the browser's windows are closed. If you change an applet, recompile it, then reload it in your browser, the browser may still execute the original version of the applet. Close all your browser windows to remove the old applet from memory. Open a new browser window and load the applet to see your changes.



Error-Prevention Tip 20.3

Test your applets in every Web browser in which they will execute to ensure that they operate correctly.



20.3.2 Executing an Applet in a Web Browser (Cont.)

- **If your applet executes in the appletviewer but not in your Web browser**
 - **Java may not be installed and configured for your browser**
 - **Visit the Web site java.com and click the **Get It Now** button to install Java for your browser**
 - **You may need to manually configure Internet Explorer to use J2SE 5.0**
 - **Click the **Tools** menu**
 - **Select **Internet Options...****
 - **Click the **Advanced** tab**
 - **Check the “**Use JRE v1.5.0 for <applet>** (requires restart)” option**
 - **Click **OK****
 - **Close all browser windows before attempting to execute another applet in the browser**



Method When the method is called and its purpose

`public void init()`

Called once by the applet container when an applet is loaded for execution. This method initializes an applet. Typical actions performed here are initializing fields, creating GUI components, loading sounds to play, loading images to display (see Chapter 20, *Multimedia: Applets and Applications*) and creating threads (see Chapter 23, *Multithreading*).

`public void start()`

Called by the applet container after method `init` completes execution. In addition, if the user browses to another Web site and later returns to the applet's HTML page, method `start` is called again. The method performs any tasks that must be completed when the applet is loaded for the first time and that must be performed every time the applet's HTML page is revisited. Actions performed here might include starting an animation (see Chapter 21) or starting other threads of execution (see Chapter 23).

Fig. 20.9 | JApplet life cycle methods that are called by an applet container during an applet's execution. (Part 1 of 3.)



Method When the method is called and its purpose

public void paint(Graphics g)

Called by the applet container after methods `init` and `start`. Method `paint` is also called when the applet needs to be repainted. For example, if the user covers the applet with another open window on the screen and later uncovers the applet, the `paint` method is called. Typical actions performed here involve drawing with the `Graphics` object `g` that is passed to the `paint` method by the applet container.

public void stop()

This method is called by the applet container when the user leaves the applet's Web page by browsing to another Web page. Since it is possible that the user might return to the Web page containing the applet, method `stop` performs tasks that might be required to suspend the applet's execution, so that the applet does not use computer processing time when it is not displayed on the screen. Typical actions performed here would stop the execution of animations and threads.

Fig. 20.9 | JApplet life cycle methods that are called by an applet container during an applet's execution. (Part 2 of 3.)



Method When the method is called and its purpose

`public void destroy()`

This method is called by the applet container when the applet is being removed from memory. This occurs when the user exits the browsing session by closing all the browser windows and may also occur at the browser's discretion when the user has browsed to other Web pages. The method performs any tasks that are required to clean up resources allocated to the applet.

Fig. 20.9 | JApplet life cycle methods that are called by an applet container during an applet's execution. (Part 3 of 3.)



Common Programming Error 20.2

Declaring methods `init`, `start`, `paint`, `stop` or `destroy` with method headers that differ from those shown in Figure 20.9 results in methods that will not be called by the applet container. The code specified in your versions of the methods will not execute.



20.5 Initializing an Instance Variable with Method `init`

- **Applet `AdditionApplet`**
 - computes the sum of two values input by the user and displays the result by drawing a `String` inside a rectangle on the applet
 - The sum is stored in an instance variable of class `AdditionApplet`
 - So it can be used in both method `init` and method `paint`



Outline

```
1 // Fig. 20.10: AdditionApplet.java
2 // Adding two floating-point numbers.
3 import java.awt.Graphics; // program uses class Graphics
4 import javax.swing.JApplet; // program uses class JApplet
5 import javax.swing.JOptionPane; // program uses class JOptionPane
```

```
6
7 public class AdditionApplet extends JApplet
8 {
9     private double sum; // sum of values entered by user
```

Declare instance variable
sum of type **double**

AdditionApplet
.java

(1 of 3)

```
11 // initialize applet by obtaining values from user
12 public void init()
13 {
14     String firstNumber; // first string entered by user
15     String secondNumber; // second string entered by user
16
17     double number1; // first number to add
18     double number2; // second number to add
19
20     // obtain first number from user
21     firstNumber = JOptionPane.showInputDialog(
22         "Enter first floating-point value" );
23
24     // obtain second number from user
25     secondNumber = JOptionPane.showInputDialog(
26         "Enter second floating-point value" );
27
```

init method called once when the
container loads this applet



Outline

AdditionApplet .java

(2 of 3)

```
28 // convert numbers from type String to type double
29 number1 = Double.parseDouble( firstNumber );
30 number2 = Double.parseDouble( secondNumber );
31
32 sum = number1 + number2; // add numbers
33 } // end method init
34
35 // draw results in a rectangle on applet's background
36 public void paint( Graphics g )
37 {
38     super.paint( g ); // call superclass version of method paint
39
40     // draw rectangle starting from (15, 10) that is 270
41     // pixels wide and 20 pixels tall
42     g.drawRect( 15, 10, 270, 20 );
43
44     // draw results as a String at (25, 25)
45     g.drawString( "The sum is " + sum, 25, 25 );
46 } // end method paint
47 } // end class AdditionApplet
```

Sum the values and assign the result to instance variable **sum**

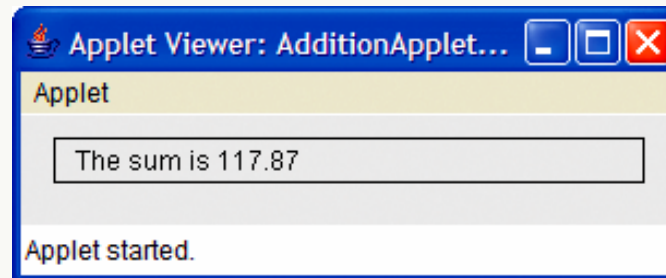
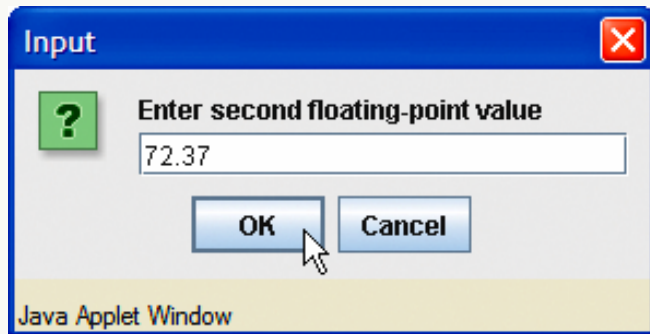
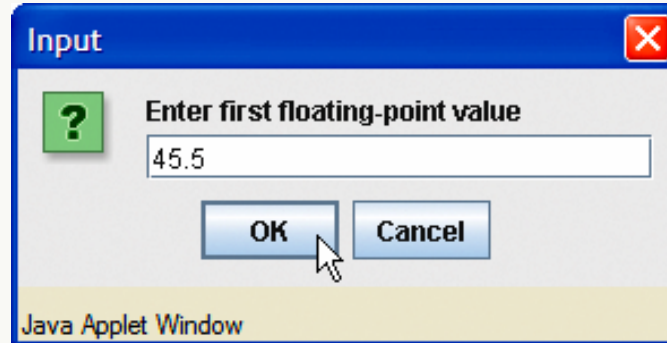
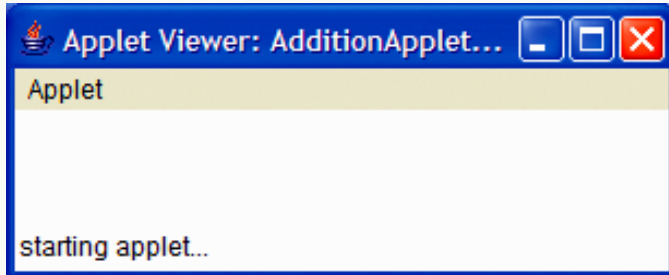
Call **drawString** to display **sum**



Outline

AdditionApplet
.java

(3 of 3)




```
1 <html>
2 <applet code = "AdditionApplet.class" width = "300" height = "65">
3 </applet>
4 </html>
```



Fig. 20.11 | AdditionApplet.html loads class AdditionApplet of Fig. 20.10 into an applet container.



Software Engineering Observation 20.1

The only statements that should be placed in an applet's `init` method are those that should execute only once when the applet is initialized.



20.6 Sandbox Security Model

- **Sandbox security model**

- Code executing in the “sandbox” is not allowed to “play outside the sandbox”
- Used by the Java platform to prevent code that is downloaded to your local computer from accessing local system resources, such as files
- For information on security and applets
 - developer.java.sun.com/developer/technicalArticles/Security/Signed
- For information on the Java 2 Platform security model
 - java.sun.com/j2se/5.0/docs/guide/security/spec/security-spec.doc1.html



20.7 Internet and Web Resources

- **Sun Microsystems Java Web site**
 - java.sun.com
 - Java applet resources
 - java.sun.com/applets
 - Free online tutorials
 - java.sun.com/learning
- **To install and configure Java for your browser**
 - Visit java.com
 - Click the **Get It Now** button

