

Chapter 20

Applets

Slides prepared by Rose Williams,
Binghamton University

Kenrick Mock, University of Alaska
Anchorage

Introduction

- Java programs are divided into two main categories, *applets* and *applications*
- An application is an ordinary Java program
- An applet is a kind of Java program that can be run across the Internet

A Brief Introduction to HTML

- *HTML* stands for *Hypertext Markup Language*
 - *Hypertext* is text viewed on a browser that contains clickable entries called *links* or *hyperlinks*
 - When a link or hyperlink is clicked, the document specified by the link is displayed
- HTML is a language used to write *HTML documents* or *pages* that will be viewed on a Web browser

A Brief Introduction to HTML

- HTML is made up of a collection of simple commands that can be inserted into a text file
 - This converts the text file into a document meant to be viewed with a Web browser
- Some commands allow pictures and hyperlinks to be inserted
- Others are editing commands that specify the main heading, subheading, paragraph beginning, and so forth

A Brief Introduction to HTML

- Much of HTML is simply a language for formatting text
 - However, HTML is not a word processor
 - It is more like a very simple programming language
 - It is similar to the annotations used by copy editors to mark a manuscript before it is typeset for production
- HTML is not part of the Java language
 - There can be interaction between HTML and Java
 - HTML can be used to display a Java applet program

HTML Formatting Commands

- There are two basic kinds of HTML commands :
 - Those that mark the beginning and end of a section of text
 - Those that mark a single location in the text
- Commands that mark the beginning and end of a section of text have the form:

```
<Command>  
Some text  
</Command>
```

HTML Formatting Commands

- The following makes the phrase "World's Greatest Home Page" a level 1 heading
 - Level 1 is the largest standard heading
- ```
<h1>
World's Greatest Home Page
</h1>
```
- Smaller headings, Level 2 and level 3, are generated by the commands **h2** and **h3**, and so forth

# HTML Formatting Commands


- Commands that mark a single location in the text are not closed with the command of form **</Command>**
  - For example, the horizontal line command:  

```
<hr>
```
- Commands in HTML are relative commands, instead of being absolute commands that determine exact size or locations
  - The browser determines the exact sizes and locations

# HTML Formatting Commands

- The browser normally determines the location of line breaks in the displayed text
  - However, a line break can be forced by inserting a break command:  
`<br>`
- Some layout specifications can be made as well
  - Anything between the commands `<center>` and `</center>` will be centered on the page

# HTML Formatting Commands

- Matching pairs of commands may be nested inside one another, but they may not cross each other:  

- Unlike Java, HTML commands are not case sensitive
- An HTML file is a text file whose name should end with `.html`

# Outline of an HTML Document

- The entire HTML document should be enclosed in the pair `<html>` and `</html>` at the beginning and end of the document
- The head of the document is enclosed in `<head>` and `</head>`
  - The head is not displayed when the page is viewed
  - It records information that is used by a browser
- The head can contain a title, enclosed in `<title>` and `</title>`
  - The title is used as a name for the document

# Outline of an HTML Document

- The part of the document that is displayed on the screen is divided into two parts
- The body is the real content of the document
  - It is enclosed in `<body>` and `</body>`
- The other part should contain the e-mail address for contacting the document's maintainer, and the date that the document was last modified
  - It is enclosed in `<address>` and `</address>`

## Outline of a Simple HTML Document (Part 1 of 3)

### Outline of a Simple HTML Document

---

`<html>` ← Beginning of HTML document

`<head>` ← Beginning of document head

`<title>` ← Beginning of document title

Document Title

`</title>` ← End of document title

`</head>` ← End of document head

`<body>` ← Beginning of main text to appear on screen

`<h1>`

First of Largest Size Headings

`</h1>`

(continued)

## Outline of a Simple HTML Document (Part 2 of 3)

### Outline of a Simple HTML Document

---

Some text

`<h2>` ← *Most browsers do not require all the entries pointed to with blue arrows, but it is good to include them all.*

First Subheading

`</h2>`

Some text

`<h2>`

Second Subheading

`</h2>`

Some text.

`<h1>`

Second of Largest Size Headings

`</h1>`

More of the same

.

.

(continued)

## Outline of a Simple HTML Document (Part 3 of 3)

### Outline of a Simple HTML Document

---

`</body>` ← End of main text to appear on screen

`<address>` ← Beginning of address section

`<hr>` ← Horizontal line; nice, but not required

The e-mail address of the person maintaining the page.  
Also, the date of the last time the page was changed.  
(You can put in whatever you want here, but the e-mail address and date are what people expect.)

`</address>` ← End of address section

`</html>` ← End of HTML document

*Although this is not part of the main body of text, it does appear on the screen.*

## An HTML Document (Part 1 of 3)

### An HTML Document

---

`<html>`

`<head>`

`<title>`

Liars Club Home Page ← *This text does not appear on the screen but is used as the default name for any bookmark ("favorite") to this page.*

`</title>`

`</head>`

`<body>`

`<h1>`

`<center>`

Liars Club

`</center>`

`</h1>`

*Blank lines are ignored when the document is displayed, but they can make your HTML code easier to read.*

(continued)

# An HTML Document (Part 2 of 3)

## An HTML Document

```
<h2>
Club Goals
</h2>
<p>
The goal of the club is to take over the world.
We already have members in key government positions.
</p>
<p>
Another goal is to improve the image of liars.
To this end, we have infiltrated many advertising agencies.
</p>
<h2>
Meeting Times
</h2>
```

*Text may have different line breaks when displayed on your browser.*

*A new paragraph will always produce a line break and some space.*

(continued)

# An HTML Document (Part 3 of 3)

## An HTML Document

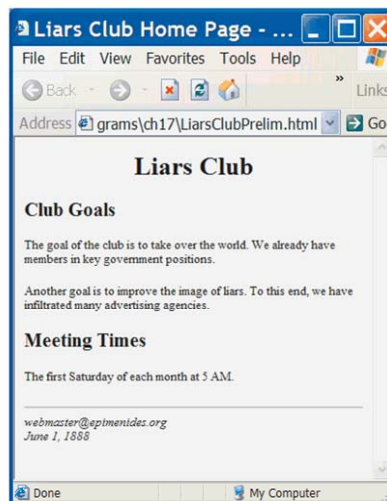
```
The first Saturday of each month at 5 AM.
<p> <!--To add some space.-->
</p>
</body>
<address>
<hr>
webmaster@epimenides.org

June 1, 1888
</address>
</html>
```

*A comment*

*This is the file LiarsClubPrelim.html.*

# Browser View of HTML Document



# URL

- A **URL** is the name of an HTML document on the Web
  - URL is an acronym for *Uniform Resource Locator*
- URLs often begin with **http**
  - This is the name of the protocol used to transfer and interpret the HTML document
  - Most browsers will fill in **http://** if it is omitted

# Hyperlinks

- Text can be marked as a hyperlink so that if a user clicks that text, the browser goes to another Web page specified by the link

```

TextToClick

```

- The *PathToDocument* can be a full or relative path name to an HTML file, or a URL to any place on the Web
- The *TextToClick* will be displayed and underlined by the browser

# Inserting a Picture

- A picture can also be inserted in an HTML document

```

```

- The *PathToPicture* can be a full or relative path name to a file with a digitally encoded picture
- Most commonly used picture-encoding formats are accepted, such as *.gif*, *.tiff*, and *.jpg*

## An HTML Document with a Hyperlink and a Picture (Part 1 of 3)

### An HTML Document with a Hyperlink and a Picture

---

```
<html>
<head>
<title>
Liars Club Home Page
</title>
</head>

<body>
<h1>
<center>
Liars Club
</center>
</h1>
<h2>
Club Goals
</h2>
```

*This is the file LiarsClub.html.*

(continued)

## An HTML Document with a Hyperlink and a Picture (Part 2 of 3)

### An HTML Document with a Hyperlink and a Picture

---

```
<p>
The goal of the club is to take over the world.
We already have members in key government positions.
</p>
<p>
Another goal is to improve the image of liars.
To this end, we have infiltrated many advertising agencies.
</p>

<h2>
Meeting Times
</h2>
The first Saturday of each month at 5 AM.
<h2>
```

*This is explained in the subsection entitled "Inserting a Picture."*

(continued)

## An HTML Document with a Hyperlink and a Picture (Part 3 of 3)

### An HTML Document with a Hyperlink and a Picture

```
Other Liar Organizations
</h2>

Click here for another kind of liar.

<p> <!--To add some space.-->
</p>
</body>

<address>
<hr>
webmaster@epimenides.org

June 1, 1888
</address>
</html>
```

*A hyperlink to another HTML document, in this case another Web page.*

## Browser View of an HTML Document with a Hyperlink and a Picture



## Pitfall: Not Using Your Browser's Refresh Command

- Browsers normally keep copies of the most recently viewed HTML pages
  - This helps the browser retrieve a page quickly when someone returns to that page
- This feature can be a problem when designing and debugging an HTML page
  - If a change is made to a page, and that page is viewed again, it may still look the same
  - This is because the copy is being viewed, not the new page
- Browsers have a command to reload a page, and thus get the most recent version of it
  - It is usually called "Refresh" or "Reload", and is a button or menu item

## Tip: Other Languages for Authoring Web Pages

- HTML is a low-level language for a Web browser much the same as assembly language is a low-level language for a computer
- Most Web page designers today use a high-level Web page design language that translates into HTML
  - For example Dreamweaver (Macromedia, Inc.), FrontPage (Microsoft Corporation), and GoLive (Adobe Systems Inc.)

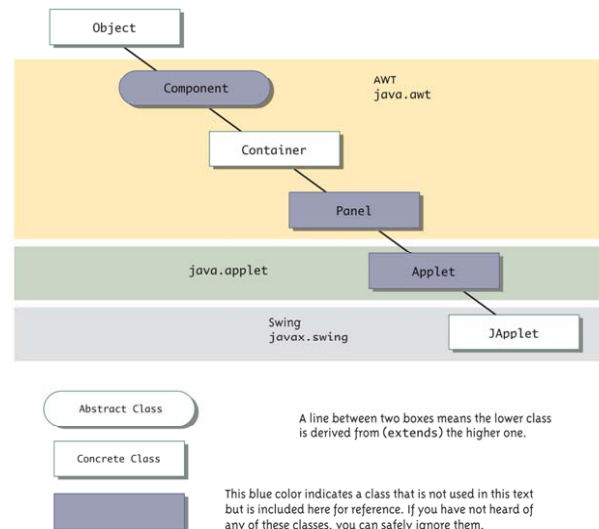
## Programming Applets

- The word *applet* is meant to suggest a small *application*
- Applets were intended to be small programs run over the Internet
  - However, there are no size constraints on applets
  - Applets can be viewed over the Internet, or without any connection to the internet
- An applet is similar to a Swing GUI
  - In fact, almost all of the Swing techniques can be used in applets

## Defining an Applet

- An applet class is normally defined as a derived class of the class **JApplet**
  - The class **JApplet** is in the package **javax.swing**
- There is also an older class, **Applet**, which has been superseded by the **JApplet** class

## Applets in the Class Hierarchy



## Designing an Applet

- An applet class can be designed as a derived class of **JApplet** in much the same way that regular Swing GUIs are defined as derived classes of **JFrame**
- However, an applet normally defines no constructors
  - The method **init** performs the initializations that would be performed in a constructor for a regular Swing GUI



# Designing an Applet

- Components can be added to an applet in the same way that a component is added to a **JFrame**
  - The method **add** is used to add components to an applet in the same way that components are added to a **JFrame**

# An Applet (Part 1 of 2)

## An Applet

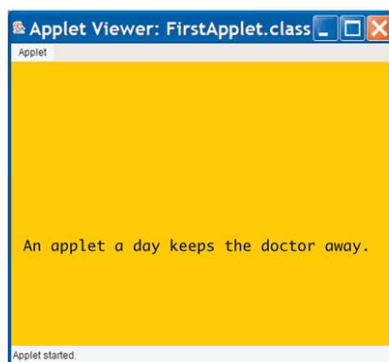
```
1 import javax.swing.JApplet;
2 import javax.swing.JLabel;
3 import java.awt.BorderLayout;
4 import java.awt.Color;
5
6 public class FirstApplet extends JApplet
7 {
8 public void init()
9 {
10 getContentPane().setBackground(Color.ORANGE);
11 setLayout(new BorderLayout());
12 JLabel aLabel =
13 new JLabel("An applet a day keeps the doctor away.");
14 add(aLabel, BorderLayout.CENTER);
15 }
```

(continued)

# An Applet (Part 2 of 2)

## An Applet

**RESULTING GUI** (Using an Applet Viewer)



*This close-window button and the other two buttons are part of the applet viewer, not part of the applet.*

# How Applets Differ from Swing GUIs

- Some of the items included in a Swing GUI are not included in an applet
- Applets do not contain a **main** or **setVisible** method
  - Applets are displayed automatically by a Web page or an applet viewer
- Applets do not have titles
  - Therefore, they do not use the **setTitle** method
  - They are normally embedded in an HTML document, and the HTML document can add any desired title

## How Applets Differ from Swing GUIs

- Applets do not use the `setSize` method
  - The HTML document takes care of sizing the applet
- Applets do not have a close-window button
  - Therefore, they do not have a `setDefaultCloseOperation` method
  - When the HTML document containing the applet is closed, then the applet is automatically closed

## Running an Applet

- An applet class is compiled in the same way as any other Java class
  - However, an applet is run differently from other Java programs
- The normal way to run an applet is to embed it in an HTML document
  - The applet is then run and viewed through a Web browser

## Running an Applet

- An applet can also be viewed using an *applet viewer*
  - An applet viewer is a program designed to run an applet as a stand-alone program
- The Java `appletviewer` can be used to run an applet:  
`appletviewer FirstApplet.html`
- It may be necessary, however, to create the HTML document, and place the applet in it

## Menus in a JApplet

- Menus are constructed and added to a `JApplet` as they are for a `JFrame`
  - `JApplet` has a method named `setJMenuBar` that behaves the same as the `setJMenuBar` method of a `JFrame`
  - `JApplet` can also have menu bars added to a `JApplet` or to a panel that is part of the `JApplet` using the `add` method

## Tip: Converting a Swing Application to an Applet

- The fastest and easiest way to explain how to define an applet, is to explain how to modify a Swing GUI to transform it into an applet
  1. Derive the class from the class `JApplet` instead of from the class `JFrame`
  2. Remove the `main` method
  3. Replace the constructor with a no-parameter method named `init`
    - The body of the `init` method can be the same as the body of the deleted constructor, but with some items removed

## Tip: Converting a Swing Application to an Applet

4. Delete any invocation of `super`
  5. Delete any method invocations that program the close-window button of a windowing GUI
  6. Delete any invocation of `setTitle`
  7. Delete any invocation of `setSize`
- The following applet was generated in this way

## An Applet Calculator (Part 1 of 9)

### An Applet Calculator

---

```
1 import javax.swing.JApplet;
2 import javax.swing.JTextField;
3 import javax.swing.JPanel;
4 import javax.swing.JLabel;
5 import javax.swing.JButton;
6 import java.awt.BorderLayout;
7 import java.awt.FlowLayout;
8 import java.awt.Color;
9 import java.awt.event.ActionListener;
10 import java.awt.event.ActionEvent;
```

(continued)

## An Applet Calculator (Part 2 of 9)

### An Applet Calculator

---

```
11 /**
12 A simplified calculator as an applet.
13 The only operations are addition and subtraction.
14 */
15 public class AppletCalculator extends JApplet
16 implements ActionListener
17 {
18 public static final int WIDTH = 400;
19 public static final int HEIGHT = 200;
20 public static final int NUMBER_OF_DIGITS = 30;
```

(continued)

## An Applet Calculator (Part 3 of 9)

### An Applet Calculator

```
21 private JTextField ioField;
22 private double result = 0.0;

23 public void init() We deleted the main method.
24 {
25 setLayout(new BorderLayout());
```

*We deleted invocations of setSize, setTitle, and setDefaultCloseOperation.*

(continued)

## An Applet Calculator (Part 4 of 9)

### An Applet Calculator

```
26 JPanel textPanel = new JPanel();
27 textPanel.setLayout(new BorderLayout());
28 ioField =
29 new JTextField("Enter numbers here.", NUMBER_OF_DIGITS);
30 ioField.setBackground(Color.WHITE);
31 textPanel.add(ioField);
32 add(textPanel, BorderLayout.NORTH);

33 JPanel buttonPanel = new JPanel();
34 buttonPanel.setBackground(Color.BLUE);
35 buttonPanel.setLayout(new FlowLayout());
```

(continued)

## An Applet Calculator (Part 5 of 9)

### An Applet Calculator

```
36 JButton addButton = new JButton("+");
37 addButton.addActionListener(this);
38 buttonPanel.add(addButton);
39 JButton subtractButton = new JButton("-");
40 subtractButton.addActionListener(this);
41 buttonPanel.add(subtractButton);
42 JButton resetButton = new JButton("Reset");
43 resetButton.addActionListener(this);
44 buttonPanel.add(resetButton);

45 add(buttonPanel, BorderLayout.CENTER);
46 }
```

(continued)

The method `actionPerformed` is identical to the one in Display 17.19

## An Applet Calculator (Part 6 of 9)

### An Applet Calculator

```
47 public void actionPerformed(ActionEvent e)
48 {
49 try
50 {
51 assumingCorrectNumberFormats(e);
52 }
53 catch (NumberFormatException e2)
54 {
55 ioField.setText("Error: Reenter Number.");
56 }
57 }

58 //Throws NumberFormatException.
59 public void assumingCorrectNumberFormats(ActionEvent e)
60 {
61 String actionCommand = e.getActionCommand();
```

The methods `assumingCorrectNumberFormats` and `stringToDouble` are identical to the ones in Display 17.19

(continued)

## An Applet Calculator (Part 7 of 9)

### An Applet Calculator

```
62 if (actionCommand.equals("+"))
63 {
64 result = result + stringToDouble(ioField.getText());
65 ioField.setText(Double.toString(result));
66 }
67 else if (actionCommand.equals("-"))
68 {
69 result = result - stringToDouble(ioField.getText());
70 ioField.setText(Double.toString(result));
71 }
72 else if (actionCommand.equals("Reset"))
73 {
74 result = 0.0;
75 ioField.setText("0.0");
76 }
```

(continued)

## An Applet Calculator (Part 8 of 9)

### An Applet Calculator

```
77 else
78 ioField.setText("Unexpected error.");
79 }

80 //Throws NumberFormatException.
81 private static double stringToDouble(String stringObject)
82 {
83 return Double.parseDouble(stringObject.trim());
84 }

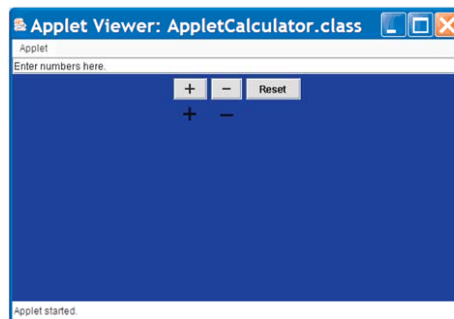
85 }
```

(continued)

## An Applet Calculator (Part 9 of 9)

### An Applet Calculator

RESULTING GUI (When started)



## Icons

- An icon is a picture
  - It is typically, but not always, a small picture
- An icon can be stored in a file of many different standard formats
  - Such as `.gif`, `.tiff`, or `.jpg`
- The class `ImageIcon` is used to convert a picture file to a Swing icon
  - Then it can be added as a component to any `Container` class, such as `JApplet`
  - The class `ImageIcon` is in the `javax.swing` package

```
ImageIcon NameOfImageIcon = new
 ImageIcon("PictureFileName");
```

# Adding Icons to an Applet

- The easiest way to display an icon in an applet is to place it in a **JLabel**
- The following three lines create a label, create an icon, and then add the icon to the label:

```
JLabel aLabel=new JLabel("Welcome to my applet.");
ImageIcon dukeIcon = new
 ImageIcon("duke_waving.gif");
aLabel.setIcon(dukeIcon);
```

- The character pictured in this icon is named *Duke*
  - He is Sun Microsystem's mascot for the Java language

# An Applet with an Icon (Part 1 of 3)

## An Applet with an Icon

```
1 import javax.swing.JApplet;
2 import javax.swing.JLabel;
3 import javax.swing.ImageIcon;
4 import java.awt.BorderLayout;
5 import java.awt.Color;

6 public class IconApplet extends JApplet
7 {
8 public void init()
9 {
10 getContentPane().setBackground(Color.YELLOW);
11 setLayout(new BorderLayout());
```

(continued)

# An Applet with an Icon (Part 2 of 3)

## An Applet with an Icon

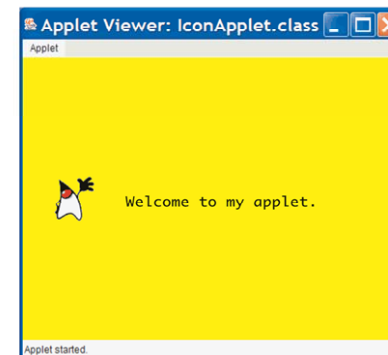
```
12 JLabel shift = new JLabel(" ");
13 JLabel aLabel = new JLabel("Welcome to my applet.");
14 ImageIcon dukeIcon = new ImageIcon("duke_waving.gif");
15 aLabel.setIcon(dukeIcon);
16 add(shift, BorderLayout.WEST);
17 add(aLabel, BorderLayout.CENTER);
18 }
19 }
```

(continued)

# An Applet with an Icon (Part 3 of 3)

## An Applet with an Icon

### RESULTING GUI



## Inserting an Applet in an HTML Document

- An applet can be placed in an HTML document with an *applet tag*:

```
<applet code="PathToApplet"
 width=Number1 height=Number2>
</applet>
```

- If given a **.class** file name only, then the HTML file and the applet file must be in the same directory
  - The *PathToApplet* can be a full or relative path name

## Inserting an Applet in an HTML Document

- Note that the name of the **.class** file, not the **.java** file, is given
- Note also that the width and height of the applet is given in this command, and not within the applet class definition
  - The width and height are in pixels
- The following code, when placed in an HTML document, will display the calculator applet in a browser as shown

```
<applet code="AppletCalculator.class"
 width=400 height=300>
</applet>
```

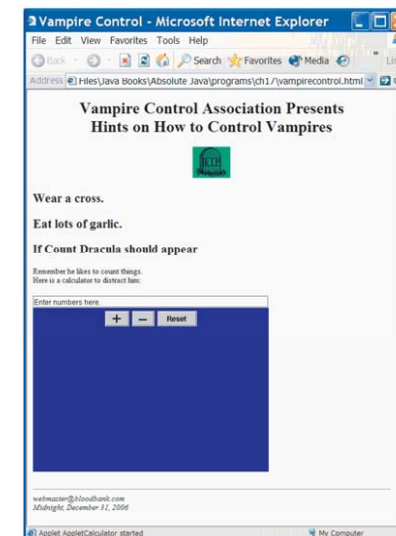
## An Applet in an HTML Document

```
<html>
<head>
<title>
Vampire Control
</title>
</head>
. . .

<applet code="AppletCalculator.class"
 width=400 height=300>
</applet>

. . .
</html>
```

## Browser View



## Pitfall: Using an Old Web Browser

- An old browser may not be able to run applets from an HTML document
  - Even if a java application runs correctly on the same system
- Web browsers do not use the same Java Virtual Machine used to run regular Java applications
  - An old browser will have an old Java Virtual Machine, or perhaps, no Java Virtual Machine
- However, an applet viewer will work, as long as a recent version of Java is installed

## Applets and Security

- An applet can be a program, written by someone else, that runs on your computer
- Whenever someone else's program runs on your computer, there are security questions you should ask:
  - Will it read information from your files?
  - Will it corrupt your operating system?

Applets are designed so that they cannot do any of these things (at least easily)