

# Java Applets



# Introduction to Java Applet Programs

- Applications are stand alone programs
  - executed with Java interpreter
- Applet is a small program
  - can be placed on a web page
  - will be executed by the web browser
  - give web pages “dynamic content”

# Java Applets

- Built using one of general definitions of applets
  - **Applet** class
  - **JApplet** class
- Java applets are usually graphical
  - Draw graphics in a defined screen area
  - Enable user interaction with GUI elements

# Java Applet Classes

- Abstract Windowing Toolkit AWT
  - Earlier versions of Java
  - **Applet** class is one of the AWT components
- Java Foundation Classes JFC
  - Extension to Java in 1997
  - Has a collection of Swing components for enhanced GUIs
  - Swing component classes begin with **J**

# Java Applets

- Applets are Java programs that can be embedded in HTML documents
  - To run an applet you must create a .html file which references the applet
  - *Ready to Program* also will run an applet
- When browser loads Web page containing applet
  - Applet downloads into Web browser
  - begins execution
- Can be tested using appletviewer program

# Contrast Application with Applet

Application	Applet
<ul style="list-style-type: none"><li>• Object class extended</li><li>• Class not declared <b>public</b></li><li>• Has a <b>main ()</b></li><li>• <b>static</b> keyword used</li><li>• Uses <b>System.exit(1)</b></li></ul>	<ul style="list-style-type: none"><li>• <b>JApplet</b> class extended</li><li>• class declared to be <b>public</b></li><li>• <b>init()</b> instead of <b>main ()</b></li><li>• <b>init()</b> not declared with static keyword</li></ul>

# Applet Declaration

- Syntax (note difference from application declaration)

```
public class ClassName extends JApplet
```

*ClassName* is an object that will be a subclass of JApplet

# Body of an Applet

- Note there is no `main()` method in an applet
  - `JApplet` class provides other methods instead of a `main` method
- First method executed is the `init()` method

# Applets

## ➤ Applet

- Program that runs in
  - **appletviewer** (test utility for applets)
  - Web browser (IE, Communicator)
- Executes when HTML (Hypertext Markup Language) document containing applet is opened
- Applications run in command windows

## ➤ Notes

- Focus on fundamental programming concepts first
  - Explanations will come later

# Applets and Web Pages – HTML

- Applets embedded in a web page
  - Executed when web page loaded by browser
- Web pages structured with **HTML** codes
  - HyperText **Mark-up Language**

## ➤ Syntax

`<command>`

...

`</command>`

Turns format on

Turns the format off

# Applets and Web Pages – HTML

- Embedding Java applets
  - Insert applet tags

```
<APPLET>
</APPLET>
```
- Call the specific applet by its file name

```
<APPLET CODE = "Whatever.class"
          WIDTH = nnn HEIGHT = mmm>
<\APPLET>
```

Where **nnn** and **mmm** are specific pixel sizes

# Applets and Web Pages – HTML

- Create the web page code using a text editor
- Save it with an .html suffix
- Open this file with appletviewer or with a web browser that supports Java
- Java Plug-in must be installed (part of J2SDK 1.4.1 from Sun)

```
<HTML>
<HEAD>
</HEAD>
<BODY>
<APPLET CODE = . . . >
</APPLET>
</BODY>
</HTML>
```

# Applets and Web Pages – HTML

- Client Web browser anywhere can access this web page from its host server
- Embedded Java applet runs on client browser (of any type platform)
- This means a client anywhere on any type of platform can run a piece of software developed on any other type of platform

Platform Independence

# Thinking About Objects

- Java an object-oriented language
  - However, Java has constructs from structured programming
- Object orientation
  - Natural way to think about world and writing computer programs
    - Object-oriented programming models the real world
  - Attributes - properties of objects
    - Size, shape, color, weight, etc.
  - Behaviors - actions that objects can perform
    - A ball rolls, bounces, inflates and deflates

# Thinking About Objects

- Object orientation (continued)
  - Inheritance
    - New classes of objects absorb characteristics of existing classes
  - Information hiding
    - Objects usually do not know how other objects are implemented
    - We can drive cars without knowing how every part works internally

# Thinking About Objects

Class - unit of Java programming

- Java focuses on nouns (classes)
  - C focuses on verbs and is action oriented
- Contain methods
  - Implement behaviors
- Contain data
  - Implement attributes
- Classes are reusable
  - Standardized, interchangeable parts

# A Simple Java Applet: Drawing a String

- Figure 3.6 – a welcome message applet
- The .html code to run the applet in a browser

```
<html>
<applet code = "WelcomeApplet.class" width = "300" height = "45">
</applet>
</html>
```

- The program output shown in the Applet Viewer

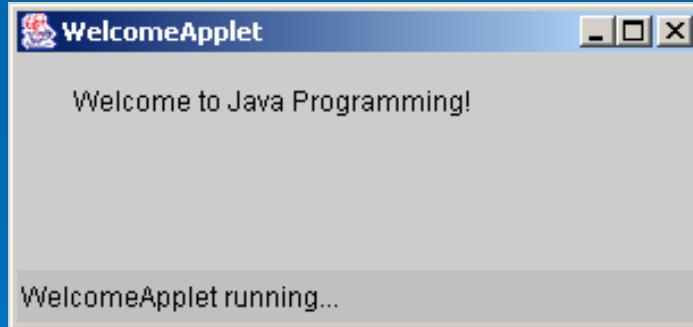


# Running the Applet

- Compile
  - Use *Ready to Program*
  - If no errors, bytecodes stored in `WelcomeApplet.class`
- We must create an HTML file
  - Loads the applet into `appletviewer` or a browser
  - Ends in `.htm` or `.html`
- To execute an applet
  - Create an HTML file indicating which applet the browser (or `appletviewer`) should load and execute

# Running the Applet - Alternatively

- Run from within *Ready to Program*
- Prompt for applet window size appears
- Applet window runs



C:\classes\csc2103\Java\WelcomeApplet.java\* - Ready to Program

File Edit Search Mark Run Debugger Help

Run Pause Open Save Indent Print

```
// Fig. 3.6: WelcomeApplet.java
// A first applet in Java.

// Java core packages
import java.awt.Graphics; // import class Graphics

// Java extension packages
import javax.swing.JApplet; // import class JApplet

public class WelcomeApplet extends JApplet {
    // draw text on applets background
    public void paint( Graphics g )
    {
        // call inherited version of method paint
        super.paint( g );

        // draw a String at x-coordinate 25 and y-
        g.drawString( "Welcome to Java Programming" );
    } // end method paint
} // end class WelcomeApplet
```

Execution Finished Line 5 of 23 Col 20

**import** allows us to use predefined classes (allowing us to use applets and graphics, in this case).

**extends** allows us to inherit the capabilities of class **JApplet**.

Method **paint** is guaranteed to be called in all applets. Its first line must be defined as above.



# Running An Applet

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

public class HelloApplet extends Applet {
    public void paint (Graphics g)
    {
        g.drawString ("Hello. Welcome to",25,25);
        g.drawString ("Java Programming",25,40);
    }
}
```

- Enter this text into your *Ready to Program* editor
- Compile the Java code

# Running An Applet

- Now create an .html file to run the applet

```
<html>
<applet code = "HelloApplet.class" width=275, height = 100>
</applet>
</html>
```

- Save it as **HelloApplet.html**
- Make sure you save it in the same directory as the **.java** file