Java

Applets

Applets

- An applet is a Panel that allows interaction with a Java program.
- A applet is typically embedded in a Web page and can be run from a browser.
- You need special HTML in the Web page to tell the browser about the applet.
- Applets run in a sandbox: they have no access to the client's file system.

Applet Support

- Netscape claims to support Java 1.1, but has serious omissions.
- MS Internet Explorer supports most of 1.1.
- The best support isn't a browser, but the standalone program `appletviewer`.
- In general you want to write applets that can be run with any browser

What an applet is

- You write an applet by extending the class `Applet`.
- `Applet` is just a class like any other; you can even use it in applications if you want.
- When you write an applet, you are only writing *part* of a program.
- The browser supplies the `main` program.

The genealogy of Applet

```
java.lang.Object
  |
  +----java.awt.Component
         |
         +----java.awt.Container
                |
                +----java.awt.Panel
                       |
                       +----java.applet.Applet
```

The simplest possible applet

TrivialApplet.java

```
import java.applet.Applet;
public class TrivialApplet extends Applet { }
```

TrivialApplet.html

```
<applet
   code="TrivialApplet.class"
   width=150 height=100>
</applet>
```

The simplest reasonable applet

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

public class HelloWorld extends Applet {
   public void paint( Graphics g ) {
      g.drawString( "Hello World!", 30, 30 );
   }
}
```

Applet methods

- public void init ()
- public void start ()
- public void stop ()
- public void destroy ()
- public void paint (Graphics g)

Why an applet works

- You write an applet by *extending* the class `Applet`.
- Applet defines methods `init( )`, `start( )`, `stop( )`, `paint(Graphics)`, `destroy( )`
- These methods do nothing--they are stubs.
- You make the applet do something by overriding these methods.

public void `init ( )`

- This is the first method to execute
- It is an ideal place to initialize variables
- It is the best place to define and use buttons, text fields, sliders, layouts, etc.
- Almost every applet you ever write will have an `init( )` method

public void `start ( )`

- Not always needed
- Called after `init( )`
- Called each time the page is loaded and restarted
- Used mostly in conjunction with `stop( )`

public void `stop( )`

- Not always needed
- Called when the browser leaves the page
- Called just before `destroy( )`
- Use `stop( )` if the applet is doing heavy computation that you don't want to continue when the browser is on some other page
- Used mostly in conjunction with `start()`

public void `destroy( )`

- Seldom needed
- Called after `stop( )`
- Use to explicitly release system resources (like threads)
- System resources are usually released automatically

Applet flow of control

public void `paint(Graphics g)`

- Almost always needed
- Any painting you want to do should be done here, or in a method you call from here
- Painting that you do in other methods may or may not happen
- *Don't call this method.* It's called automatically.
- Call `repaint( )` instead.

Sample `Graphics` *methods*

- A `Graphics` is something you can paint on.
- `g.drawString("Hello, World", 20, 20);`
- `g.drawRect(x, y, width, height);`
- `g.fillRect(x, y, width, height);`
- `g.drawOval(x, y, width, height);`
- `g.fillOval(x, y, width, height);`
  `g.setColor(Color.red);`

repaint( )

- Call repaint( ) when you have changed something and want your changes to show up on the screen
- repaint( ) is a *request*--it might not happen.
- When you call repaint( ), Java schedules a call to update(Graphics g).

update( )

- When you call repaint( ), Java schedules a call to update(Graphics g)
- Here's what update does:

```
public void update(Graphics g) {
    // Fill applet with background color
    paint(g);
}
```

Other useful Applet methods

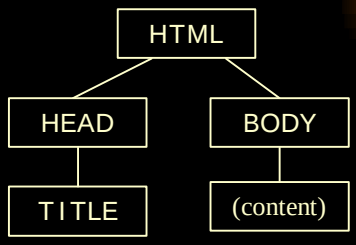
- System.out.println(String s) still works.
- Automatically opens an output window.

- showStatus(String) displays the String in the applet's status line.
- Each call overwrites the previous call.
- You have to allow time to read the line!

Applets are not magic!

- Anything you can do in an applet, you can do in an application.
- You can do some things in an application that you can't do in an applet.
- If you want to access files from an applet, it must be a "trusted" applet.
- Trusted applets are beyond the scope of this course.

Structure of an HTML page



- Most HTML tags are containers.
- A container is <tag> to </tag>

HTML

```
<html>
  <head>
    <title> Hi World Applet </title>
  </head>

  <body>
    <applet code="HiWorld.class"
      width=300 height=200>
        <param name=arraysize value=10>
    </applet>
  </body>
</html>
```

<param name=arraysize value=10>

- public String getParameter(String name)
- String s = getParameter("arraysize");
- try { size = Integer.parseInt (s) }
  catch (NumberFormatException) {… }

The End