

Java

Applets

Applet

- There are two common ways to write a Java program
 - Application
 - Applet.
- A Java application is a stand alone Java program that can be run by using just the Java interpreter from a command line. A Java class is an application if it contains a "main" method.
- To write a Java applet means to define a class which extends Applet or JApplet.

Running Applet

- An applet runs from inside a Web browser.
- You need special HTML in the Web page to tell the browser about the applet.
- A reference to an applet is embedded in a Web page using a special HTML tag "applet", with parameters specifying the name of the *.class file, as well as the width/height of the display area.
- When the page is loaded into a browser, the applet is also downloaded, and is executed in the local system.

Differences

- Java applets can be downloaded from any site on the WWW and run on a client's system.
- Java-enabled browsers and tools limit what applets can do
 - can't read/write to the reader's file system.
 - can't run any programs on the reader's system.
 - can't communicate with any network server other than the one that had originally stored the applet.
- Java applications have none of these restrictions.

A Simple Applet

- How to run :
 - ***javac*** SimpleApplet.java
 - ***appletviewer*** SimpleApplet.java
- ```
/*
<applet code="SimpleApplet" width=300
height=100>
</applet>
*/
```
- **Example:** SimpleApplet.java, TestApplet.java



# ***Applet Methods***

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

# *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.

# ***start***

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



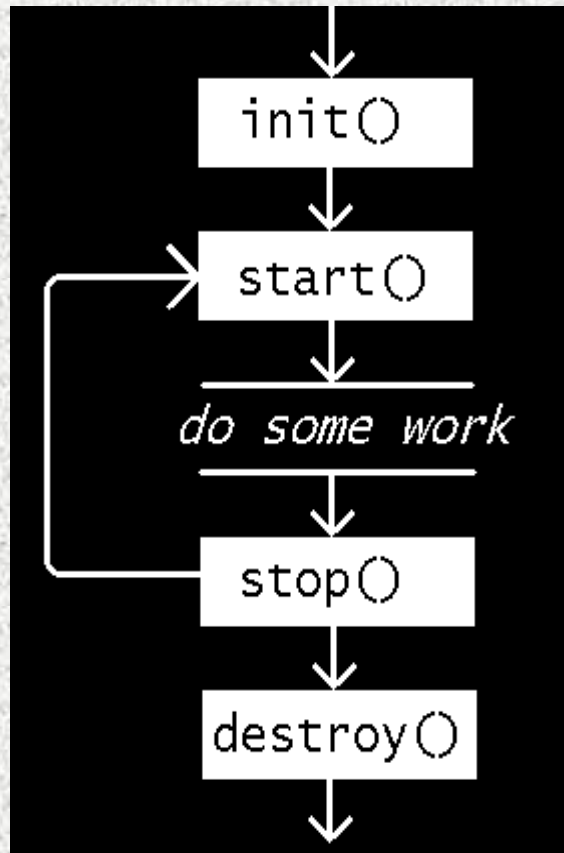
# ***stop***

- Not always needed.
- Called when the browser leaves the page.
- Called just before `destroy( )`.
- Used mostly in conjunction with `start()`.

# ***destroy***

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

# ***Flow of control***



# ***paint***

- Almost always needed.
- Any painting you want to do should be done here, or in a method you call from here.
- Don't call this method. It's called automatically.
- Call `repaint( )` instead.
- Some Graphics methods:
  - *`g.drawString("Hello, World", 20, 20)`*
  - *`g.drawRect/fillRect/drawOval/fillOval(x, y, width, height)`*
  - *`g.setColor(Color.red)`*

# *Sample HTML*

```
<html>
 <head>
 <title>Test Applet</title>
 </head>
 <body bgcolor="000000">
 <center>
 <applet
 code = "TestApplet.class"
 width = "500"
 height = "300"
 >
 </applet>
 </center>
</body>
</html>
```



# ***More Examples***

- *JApplet with JTextField and JButton*
- **Example:** MoreApplet.java
- *JApplet with parameters*
- **Example:** ParamApplet.java

***End***