Java applets

SwIG
Jing He
Outline

- What is Java?
- Java Applications
- Java Applets
- Java Applets Securities
- Summary
What is Java?

- Java was conceived by James Gosling at Sun Microsystems Inc. in 1991
- Java is platform independent language
- Java programming is a/an object-oriented programming.
Object oriented language

The world around us consists of objects.

e.g. the ATM
Object oriented language

The world around us consists of objects.

Let the program consist of objects.
The program consist of objects.

Objects of the same kind form a class.

E.g. class ATM or class Money.
**Object oriented language**

The program consist of **objects**.

Objects of the same kind form a **class**.

Each object has some **methods**.

Money `withdrawMoney(ATMCard card, int amount)`

(Objects in the same class have the same methods.)
A method of the ATM class:

```java
Money withdrawMoney(ATMCard card, int amount);
```

myPurse.addMoney(theATM.withdrawMoney(myATMCard, 1000));
Object oriented language

more ideas borrowed from the real world:

**encapsulation** – you do not need to know how the ATM works inside.

**inheritance** – you can easily create class ATMWithClocks extending class ATM. The new class inherits the methods of the ATM class.
Java Architecture

source code

Compiler

byte code

JVM

Computer

portability

security

speed

programmer

user
Why Java?

- simple
- portable
- secure
- free

- slow
So What’s Java Good For?

Web applications!

Java Applet

Java Applet

Server
Learning Java

- language
- libraries

book, lectures

documentation
http://java.sun.com/docs/

examples on the web
(problem – often old version of Java)
How are Java Applications written?

- HelloWorld.java:

  ```java
  public class HelloWorld {
      public static void main (String[] args) {
          System.out.println("Hello, World");
      }
  }
  ```

- Compile HelloWorld.java

  ```bash
  javac HelloWorld.java
  Output: HelloWorld.class
  ```

- Run

  ```bash
  java HelloWorld
  Output: Hello, World
  ```
Building JAVA Application

- Prepare the file `HelloWorld.java` using an editor
- Invoke the compiler:
  - `javac HelloWorld.java`
- This creates `HelloWorld.class`
- Run the java interpreter:
  - `java HelloWorld`
What is an applet?
APPLE  APPLET
An applet is a small Java program that is embedded and ran in some other Java interpreter program such as
- a Java technology-enabled browser
- Sun’s applet viewer program called appletviewer
Applet

Client

Internet

TCP/IP (socket)

TCP/IP (HTTP)

TCP/IP (socket)

TCP/IP (HTTP)

TCP/IP (HTTP)

Netscape (JVM)

Loads HTML file

Loads Java Applet

Server

HTML files (Applet Tags)

Applets

Web Server

Netscape (JVM)

Loads HTML file

Loads Java Applet

TCP/IP

(socket)

TCP/IP

(HTTP)

TCP/IP

(socket)

TCP/IP

(HTTP)
Applets, web page, client, server

server host

web server

myWebPage.html

...<applet code=HelloWorld.class</applet>
...HelloWorld.class

browser host

browser

request for myWebPage.html

myWebPage.html

request for HelloWorld.class

HelloWorld.class

HelloWorld.class
An applet program is a written as a inheritance of the java.Applet class

There is no main() method in an Applet.

An applet uses AWT for graphics
**Applet Execution - 2**

- **Life Cycle of an Applet:**
  - **init:** This method is intended for whatever initialization is needed for an applet.
  - **start:** This method is automatically called after init method. It is also called whenever user returns to the page containing the applet after visiting other pages.
  - **stop:** This method is automatically called whenever the user moves away from the page containing applets. This method can be used to stop an animation.
  - **destroy:** This method is only called when the browser shuts down normally.

The applet is running and rendered on the web page.

Every Applet needs to implement one or more of the init(), the start() and the paint() methods.

At the end of the execution, the stop() method is invoked, followed by the destroy() method to deallocate the applet’s resources.
Applet life cycle

- browser visits page containing an applet
  - browser calls init on that applet, once
  - browser calls start on that applet

- browser goes away from that page
  - browser calls stop on that applet

- browser comes back to that page
  - browser calls start again on that applet

- browser shuts down
  - browser calls destroy on the applet, once
<APPLET
  // the beginning of the HTML applet code
  CODE="demoxx.class"
  // the actual name of the applet (usually a 'class' file)
  CODEBASE="demos/"
  // the location of the applet (relative as here, or a full URL)
  NAME="SWE622"
  // the name of the instance of the applet on this page
  WIDTH="100"
  // the physical width of the applet on the page
  HEIGHT="50"
  // the physical height of the applet on the page
  ALIGN="Top"
  // align the applet within its page space (top, bottom, center)
<APPLET CODE="test.class" CODEBASE="example/"
       WIDTH=460 HEIGHT=160
       NAME="buddy">
<PARAM NAME="imageSource" VALUE="images/Beans">
<PARAM NAME="backgroundColor" VALUE="0xc0c0c0">
<PARAM NAME="endImage" VALUE=10>
</APPLET>
## Applet's Attributes

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Explanation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>Name of class file</td>
<td>Code=“applet0.class”</td>
</tr>
<tr>
<td>Width</td>
<td>Width of applet</td>
<td>Width=300</td>
</tr>
<tr>
<td>height</td>
<td>Height of applet</td>
<td>Height=60</td>
</tr>
<tr>
<td>Codebase</td>
<td>Applet’s Directory</td>
<td>Codebase=“/applets”</td>
</tr>
<tr>
<td>alt</td>
<td>Alternate text if applet not available</td>
<td>Alt=“menu applet”</td>
</tr>
<tr>
<td>name</td>
<td>Name of the applet</td>
<td>Name=“appletExam”</td>
</tr>
<tr>
<td>Align(top,left,right,bottom)</td>
<td>Justify the applet with text</td>
<td>Align=“right”</td>
</tr>
</tbody>
</table>
<html>
  <head>
    <title>Hello World Applet</title>
  </head>
  <body>
    <applet code="HelloWorld.class"
           width=300 height=200>
    </applet>
  </body>
</html>
Your Turn!

- You first applet
- “Hello World”
History of an Applet

- Edit java source code & html
  - notepad Hello.java
  - notepad Hello.html

- Compile source to ByteCodes
  - javac Hello.java
  - produces Hello.class

- View applet (Java Virtual Machine)
  - appletviewer Hello.html
  - browser Hello.html
<html>
<body>

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

</body>
</html>
<html>
<body>

<applet code="Hello.class"
 width=300 height=300>

</applet>

</body>
</html>

Save as Hello.html
```java
import java.applet.Applet;
import java.awt.*;

public class Hello extends Applet {
    public void init() {
        repaint();
    }

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

what to draw

where to draw it

Save as Hello.java
History of an Applet

- Edit java source code & html
  - notepad Hello.java
  - notepad Hello.html

- Compile source to ByteCodes
  - javac Hello.java
  - produces Hello.class

- View applet (Java Virtual Machine)
  - appletviewer Hello.html
  - browser Hello.html
History of an Applet

- Edit java source code & html
  - notepad Hello.java
  - notepad Hello.html

- Compile source to ByteCodes
  - javac Hello.java
  - produces Hello.class

- View applet (Java Virtual Machine)
  - appletviewer Hello.html
  - browser Hello.html
The coordinate system

The applet is drawn in a rectangle, which consists of pixels.
The coordinate system

Each pixel has a coordinate \((x,y)\)
Sample Graphics methods

- A Graphics is something you can paint on.

```java
g.drawString("Hello", 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);
g.drawString("Hello", 20, 20);
```
Applet Security

For security reasons, applets that are loaded over the network have several restrictions.

- an applet cannot ordinarily read or write files on the computer that it's executing on.
- an applet cannot make network connections except to the host that it came from.

Ref: http://java.sun.com/docs/books/tutorial/deployment/applet/index.html/
What are the disadvantages of applets?

- Applets can’t run any local executable programs
- Applets can’t with any host other than the originating server
- Applets can’t read/write to local computer’s file system
What are the disadvantages of applets? (Cont’d)

- Applets can’t find any information about the local computer

- All java-created pop-up windows carry a warning message

- Stability depends on stability of the client’s web server

- Performance directly depend on client’s machine
What are the advantages of applets?

- Automatically integrated with HTML; hence, resolved virtually all installation issues.

- Can be accessed from various platforms and various java-enabled web browsers.

- Can provide dynamic, graphics capabilities and visualizations

- Implemented in Java, an easy-to-learn OO programming language
What are the advantages of applets? (Cont’d)

- Alternative to HTML GUI design

- Safe! Because of the security built into the core Java language and the applet structure, you don’t have to worry about bad code causing damage to someone’s system

- Can be launched as a standalone web application independent of the host web server
Summary

- An **applet** is a Java class
- Its code is **downloaded** from a web server
- It is invoked by a browser when it scans a web page and encounters a class specified with the **APPLET** tag
- For security reasons, the execution of an applet is normally subject to restrictions:
  - Applets cannot access files in the file system on the client host
  - Applets cannot make network connection exception to the server host from which it originated
Resources

- http://java.sun.com/
  - Java[tm] 2 Platform, Standard Edition v1.6.1
    - java, javac, jar, jre, etc.
    - Any platform... FREE!
  - Online documentation and tutorials

- http://www.eclipse.org/
  - Integrated development environment (IDE) for nothing in particular
  - Java[tm] development tools (JDT) (comes with Eclips)
    - Project management
    - Editor
    - Incremental compiler
    - CVS support
  - C/C++ extension in progress
  - AspectJ support
  - Windows, Linux, and Mac.... FREE!