C340 Concurrency: Programming in Java

Wolfgang Emmerich

Mark Levene

What is Java?

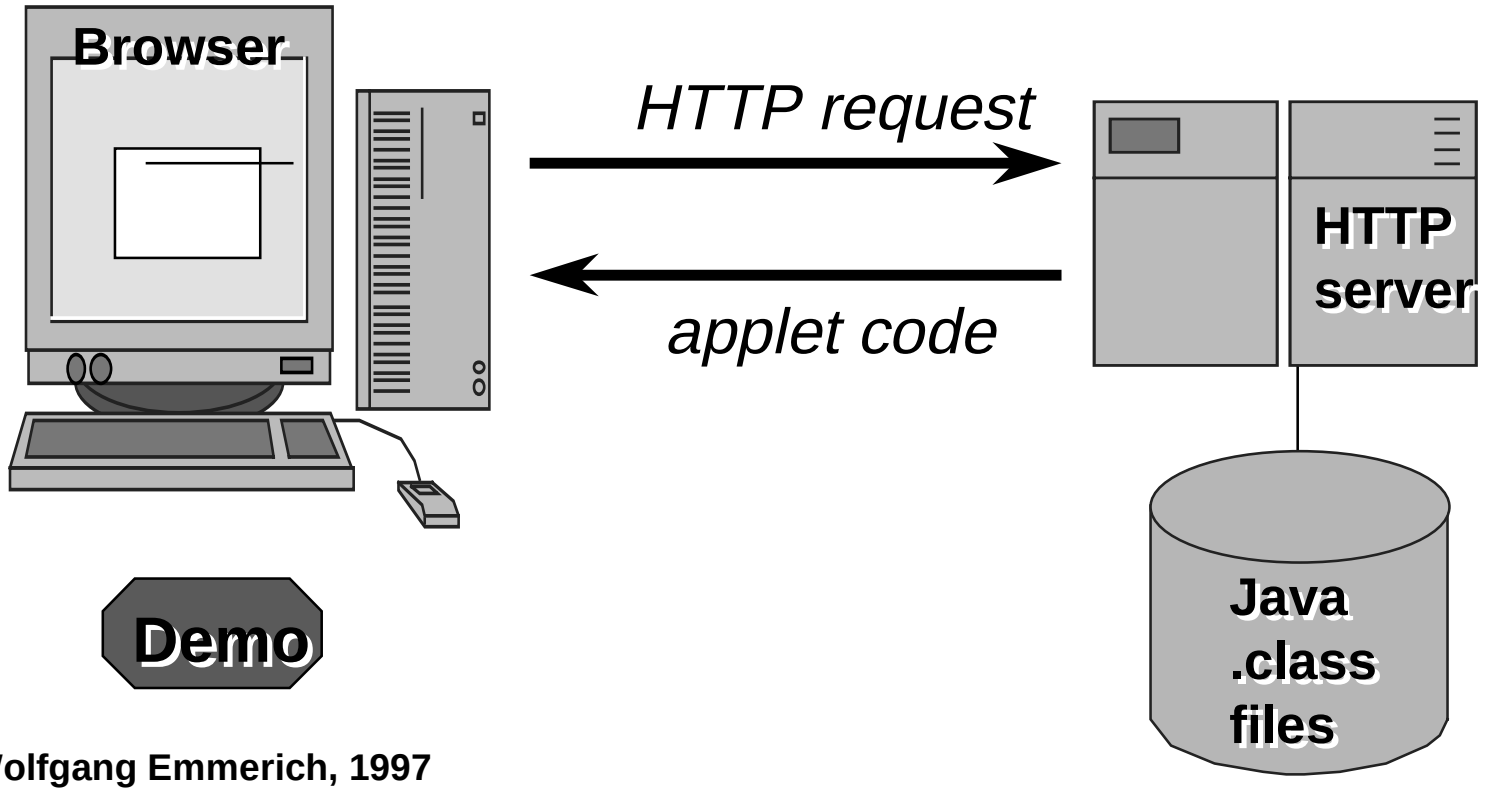
- OO Programming Language
- C/C++ control statements & expressions
- Automatic garbage collection
- Single inheritance + interfaces
- Concurrent Threads + Monitors
- Standard Packages
  - Graphical User Interface support.
  - Network support.
  - Multi-media support - animation, sound
  - ADTs - Vector, Hashtable, Dictionary etc

Why is Java interesting?

- **Portability**
  - Java programs are compiled into bytecode and are executed by interpreter
  - Efficiency through JIT bytecode compilers
  - Java Virtual Machine implementations available for all common platforms (CPU + OS)
  - Package `java.awt`: portable interface to window systems e.g X11, Windows, Mac OS.
- **Accessibility**
  - Compiled Java programs can be transmitted to and executed on remote computers.
  - Web browsers execute Java code - Applets.

What is an Applet?

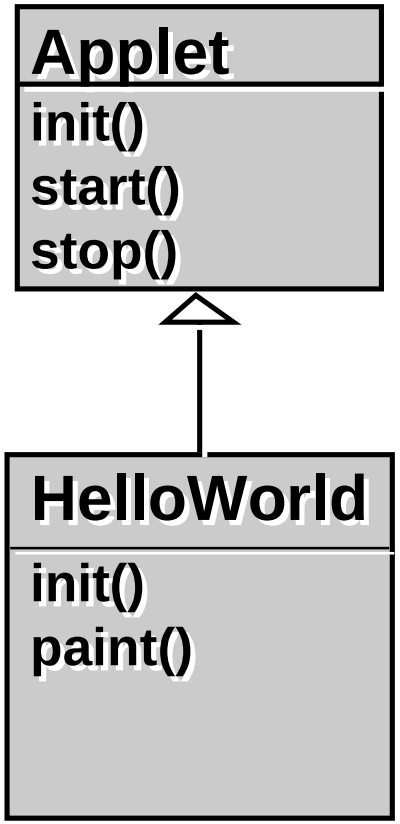
- Java class derived from application window toolkit (AWT) class Applet
- Executable by a Web browser.



A simple Applet - Hello World

```
import java.awt.*;                    //windows toolkit
import java.applet.*;                 //applet support
public class HelloWorld extends Applet {
   private Font f1;
   public HelloWorld() {
      f1 = new Font("Helvetica",Font.BOLD,36);
   }
   public void paint(Graphics g) {
      g.setFont(f1);
      g.drawString("Hello World",25,50);
   }
}
```

Hello World - Class diagram



Embedding Applets in HTML

```
<HTML>
<HEAD>
<TITLE> A simple program </TITLE></HEAD>
<BODY>
<H1 ALIGN=CENTER>Hello World</H1>
<CENTER><P>
<APPLET CODE=”Hello World.class"
        WIDTH=250
        HEIGHT=150>
</APPLET>
</P></CENTER>
</BODY>
</HTML>
```

How does Java differ from C/C++?

- ***Primitive Datatypes***
  - **`boolean`,`char`,`byte`,`short`,`int`,`long`,`float`, `double`.**
  - **`boolean` *is a 1 bit value (*`true`, `false`*)***
  - **`char` *is a 16 bit Unicode character***
  - ***constants are declared like:***
    ```
    public final static PI = 3.14159;
    ```
- ***Derived Types***
  - ***classes & arrays***
  - ***Instances of derived types handled by reference***
  - ***primitive types handled by value.***

Java Classes

- *Single inheritance hierarchy*
- *Rooted in class Object.*

```
class Counter extends Object{
    private int count;
    Counter(int i)
        {count = i;}
    void increment()
        { ++count; }
    void decrement()
        { --count; }
    int value()
        { return count;}
}
```

Java Interfaces

- ***Do not contain any method code***
- ***Similar to C++ classes with pure virtual member functions*** `(int a()=0;)`
- ***Implemented by other classes that declare*** `implements` ***relationship***
- ***Used to implement callbacks***

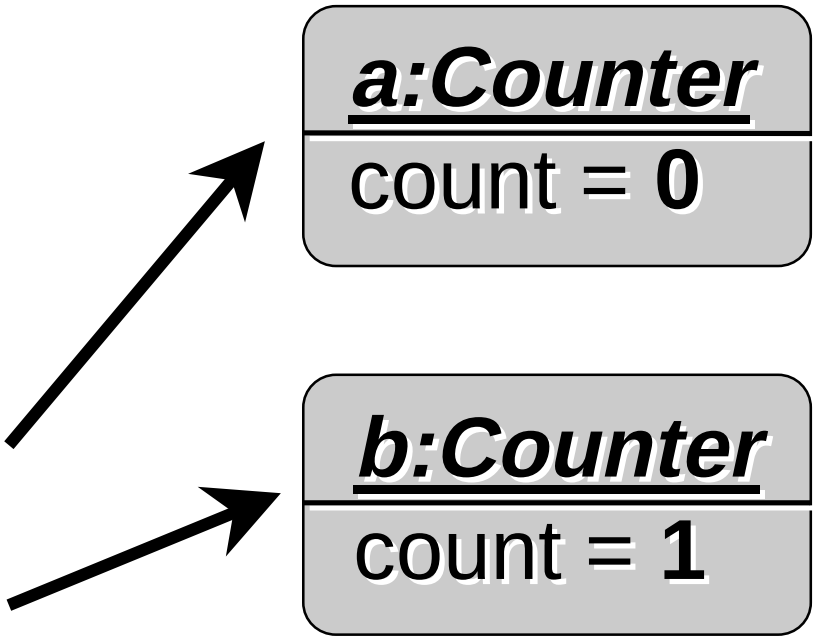
Objects & Object References

- *Declaring objects:*

```
Counter a;
Counter b;
```

- *Creating objects:*

```
a=new Counter(0);
b=new Counter(1);
```



- *Method invocation:*

```
a.increment();
b.decrement();
```

Objects & Object References (cont'd)

- *Identity vs. equality:*

```
(a == b)        // identity test
(a.equals(b)) // equality test
```

- *Automatic garbage collection:*

```
a = b;     // if a is last ref
a = null; // garbage collection
          // deletes object after
          // these assignments
```

Arrays

- *Arrays are created similarly to objects:*

```
int table[]= new table[128];
int lookup[] = {1,2,4,8,16,32};
```

- *Arrays have a length field:*

```
for(int i =0; i<table.length; i++)
   table[i] *= 2; // double
```

Strings

- *Strings are not null terminated arrays of characters:*

```
String a = “Hello”;
String b = “World”;
String message = a + “, ” + b;
```

- `A String` *is immutable*
- *Use* `StringBuffer` *to modify the contents of a* `String`

Further Information

- *Java in a Nutshell by David Flanagan O'Reilly & Associates Inc. 1996*
- *The Java Tutorial by Mary Campione and Kathy Walrath Addison-Wesley, 1996. http://www.javasoft.com/*
- *Java API Documentation http://www.javasoft.com/*
- *Concurrency Course using Java http://www-dse.doc.ic.ac.uk/~jnm*

Summary

- *Advantages of Java*
- *Applets vs. Applications*
- *Differences between Java and C++*
- *Classes and Inheritance in Java*
- *Interfaces in Java*
- *Objects and Object References*
- *Arrays*
- *Strings*
- *Next Lecture: Concurrency in Java*