

CS320 Web and Internet Programming

Introduction to Java Servlets

Chengyu Sun
California State University, Los Angeles

Servlet Hello World

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class HelloWorld extends HttpServlet {
    public void doGet( HttpServletRequest request,
                     HttpServletResponse response )
        throws ServletException, IOException
    {
        PrintWriter out = response.getWriter();
        out.println( "Hello World" );
    }
}
```

Some Simple Observations

- ◆ Inherits from `HttpServlet`
- ◆ There's no `main()` method
 - Who "runs" `HelloWorld`?
- ◆ `doGet()`
 - `HttpServletRequest`
 - `HttpServletResponse` → sent back to the client browser

Generate HTML

- ◆ Set content type to "text/html"
- ◆ Output a HTML page using `println()`
 - `<html>`
 - `<head>`, `<title>`
 - `<body>`
- ◆ Validate
 - `validator.w3c.org`
 - Sometimes useful for debugging

Package

- ◆ The advantages of using packages
- ◆ The necessity of using packages
 - A Class without a package qualifier cannot be used in a class that belongs to a package (enforced in JDK 1.4)
 - All servlets generated from JSPs are in packages

Compile Servlet The Hard Way

- ◆ `javac HelloWorld.java`
 - Where is the servlet library?
 - How to include it in compilation?
- ◆ Copy `HelloWorld.class` to the right directory
- ◆ Modify **web.xml**
 - `<servlet>` and `<servlet-mapping>`

Using Eclipse

- ◆ New → Project → Dynamic Web Project
 - Target Runtime
 - Apache Tomcat 6.0 (or 5.5)
- ◆ New → Web → Servlet
- ◆ Right click on the servlet → Run As → Run on Server

Leave everything to default; and understand the correspondence between your local Eclipse setup and the CS3 server.

About Eclipse

- ◆ <http://www.eclipse.org/>
- ◆ Arguably the most popular Java IDE
- ◆ Download Eclipse with WTP 2.0 - <http://download.eclipse.org/webtools/downloads/drops/R2.0/R-2.0-200706260303/>

Servlet Life Cycle

- ◆ Loaded for the first time – `init()`
- ◆ Per request – `service()`
 - dispatch to `doXxx()`
- ◆ Unloaded – `destroy()`
 - When requested by the manager user
 - When the context is *reloadable* and a new version of the servlet is found

I/O Re-examined

- | | |
|---|-------------|
| ◆ Java application | ◆ Servlet |
| ■ Input <ul style="list-style-type: none">• Console, GUI• File• CLP, Resource | ■ Input ?? |
| ■ Output <ul style="list-style-type: none">• Console, GUI• File | ■ Output ?? |

File I/O Example

- ◆ Read `quotes.txt`
- ◆ Display the stocks which have gone up in a HTML table
 - `<table>`
 - `<tr>`
 - `<td>`

More About Files

- ◆ Paths
 - Absolute path
 - Relative path
 - *File path on the CS 3server??*
- ◆ Permissions
 - Who owns the file?
 - Who can read/write the file?

Servlet Debugging Tips

- ◆ Debugging servlets could be rather hard
- ◆ Section 3.8
 - Use print statements
 - Invest some time in a good IDE
 - Netbeans - <http://www.netbeans.org/>
 - Eclipse (WTP) - <http://www.eclipse.org/>
 - Follow good programming practice
 - Check out HTML source
 - Restart server