

## CS320 Web and Internet Programming

### Introduction to Java Servlets

Chengyu Sun  
California State University, Los Angeles

## Overview

- ◆ HTTP Request and Response
- ◆ Form Handling
- ◆ Session Tracking

## Servlet Hello World

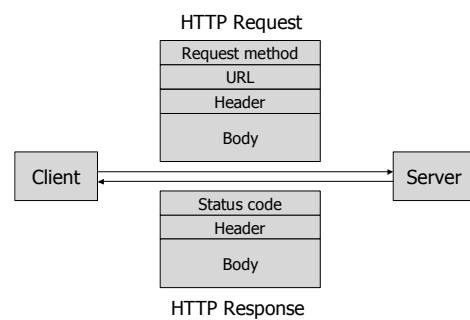
```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

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

## HttpServlet

- ◆ In `javax.servlet.http` package
- ◆ Implements `Servlet` interface
- ◆ `doGet()`

## HTTP Protocol



## HTTP Request Methods

- ◆ GET
- ◆ HEAD
- ◆ POST
- ◆ PUT
- ◆ DELETE
- ◆ OPTIONS
- ◆ TRACE

## HttpServlet Methods

	→	service()
◆ GET	→	◆ doGet()
◆ HEAD	→	◆ doHead()
◆ POST	→	◆ doPost()
◆ PUT	→	◆ doPut()
◆ DELETE	→	◆ doDelete()
◆ OPTIONS	→	◆ doOptions()
◆ TRACE	→	◆ doTrace()

## Overriding HttpServlet Methods

- ◆ doGet(), doPut(), doPost(), doDelete()
- ◆ Default implementations return a HTTP *Bad Request* error
- ◆ Browsers – GET, POST
- ◆ HTML Editors – PUT, DELETE

## HTTP Request Headers

- ◆ Connection
- ◆ User-agent
- ◆ Host
- ◆ Referrer
- ◆ Accept
- ◆ Accept-encoding
- ◆ Accept-language
- ◆ Accept-charset

## HTTP Request Headers

- ◆ Connection
  - ◆ User-agent
  - ◆ Host
  - ◆ Referrer
  - ◆ Accept
  - ◆ Accept-encoding
  - ◆ Accept-language
  - ◆ Accept-charset
- ◆ HttpServletRequest
    - getHeader()
    - getHeaders()
    - getHeaderNames()
    - getMethod()
    - getRequestURI()
    - getProtocol()
  - Cookies, Auth, Content

## HTTP Response

- ◆ Status code
    - 200 (OK)
    - 404 (NOT Found)
    - 500 (Internal Server Error)
  - ◆ Headers
    - Content-Type
    - Content-Encoding
    - Content-Language
  - ◆ Body
- ◆ HttpServletResponse
    - setStatus()
    - addHeader()
    - setHeader()
    - setContentType()
    - getWriter()
    - getOutputStream()

## Servlet Life Cycle

- ◆ init() – invoked once when the servlet object is created
- ◆ service(), doXxx() – invoked per request
- ◆ destroy() – invoked once when the servlet is unloaded

## Single Thread Mode

- ◆ The `SingleThreadModel` interface
- ◆ Guarantees no more than one request thread accessing the servlet object
- ◆ Thread safe
- ◆ Bad performance

## HTML HelloWorld

- ◆ `HelloServlet.java`
- ◆ Content type and DOCTYPE

## Development Environments

- ◆ Netbeans + Tomcat module
- ◆ J2SE + Tomcat
- ◆ Deploy
  - `cs.calstatela.edu`
  - `public_html/WEB-INF/classes`

## Handling Form Data

- ◆ `HttpServletRequest.getParameter()`
- ◆ `HttpServletRequest.getParameterValues()`

## The Need for Session Tracking

- ◆ E-commerce, personalization, automatic user authentication ...
  - Large intervals between requests
- ◆ HTTP is a "stateless" protocol
- ◆ Even persistent connections only last seconds

## Three Ways of Session Tracking

- ◆ Hidden form fields
- ◆ URL Rewriting
- ◆ Cookies

## A Simple Shopping Cart Example

- ◆ Cart0 – skeleton
- ◆ Cart1 – using hidden form
  - `<input type=hidden name=se ss value="">`
- ◆ Cart2 – using URL rewriting
- ◆ Cart3 – using cookies

## URL Rewriting

<http://cs.calstatela.edu:8280/~csun/servlet/Cart2>

Extra path info:

<http://cs.calstatela.edu:8280/~csun/servlet/Cart2/0>

Added parameter:

<http://cs.calstatela.edu:8280/~csun/servlet/Cart2?sess=0>

Custom change:

<http://cs.calstatela.edu:8280/~csun/servlet/Cart2;sess=0>

## Cookies

- ◆ Bits of textual information sent by server to a browser, then later returned by the browser when it visits the server again
- ◆ Generally considered the best session tracking method
- ◆ Security and privacy concerns

## Servlet Cookies API

- ◆ Cookie Object
  - `Cookie( String name, String value )`
  - `getName(), setName( String )`
  - `getValue(), setValue( String )`
  - `setMaxAge( int seconds )`
- ◆ `HttpServletRequest`
  - `Cookie[] getCookies()`
- ◆ `HttpServletResponse`
  - `addCookie( Cookie c )`

## Servlet Session Tracking API

- ◆ `HttpServletRequest`
  - `getSession( boolean create )`
- ◆ `HttpSession`
  - `isNew()`
  - `setAttribute( String name, Object value )`
  - `getAttribute( String name )`
  - `removeAttribute( String name )`
  - Enumeration `getAttributeNames()`

## Using Session Tracking API

- ◆ Examples in the textbook
  - Per-client access count
  - Shopping Cart

## Servlets vs. CGI

- ◆ Efficient
- ◆ Portable
- ◆ Secure
- ◆ Convenient, powerful, and inexpensive