Servlet, Session & Database
Project grading criteria

- Complexity of business logic
  - if the logic is too simple …..
- MVC framework
  - isolate “model” from “view”
    - e.g. database access in the JSP is a BAD style
- Workload
- Misc, e.g.
  - transaction requirement
  - appropriate use of session and request
IDE

- Web programming may be a little different from traditional programming, especially the debugging
  - Eclipse + Tomcat plugin
  - NetBean (bundled with Tomcat)
  - Tomcat log is very useful

- install it today and play with it
what’s the difference between JSP and Servlet?
Multiplier example

- **tomcat\webapp\multiplier\WEB-INF\web.xml**

```xml
<servlet>
  <servlet-name>multiplier</servlet-name>
  <servlet-class>MyMultiplier</servlet-class>
</servlet>

<servlet-mapping>
  <servlet-name>multiplier</servlet-name>
  <url-pattern>servlet/MyMultiplier</url-pattern>
</servlet-mapping>
```

- **tomcat\webapp\multiplier\multiplier.html**

```html
<form method="GET" action="servlet/MyMultiplier">
  Provide the number to be multiplied:
  
  <input type="text" name="num"/>
  <p>
  <input type="submit" value="Click Here to Submit"/>
</form>
```

Accessing URLs calls the corresponding Servlets.
public class MyMultiplier extends HttpServlet {

public void doGet(HttpServletRequest req, HttpServletResponse res)
    throws ServletException, IOException {

    res.setContentType("text/html");
    PrintWriter out = res.getWriter();
    out.println("<HTML><HEAD><TITLE>Multiply times "+3 +
            "</TITLE></HEAD></HTML>");
    out.println("<BODY>");
    String parameter = req.getParameter("num");
    out.println(parameter + " * " + 3 + " = " +
            3 * (Integer.parseInt(parameter)));
    out.println("</BODY>");
    out.println("</HTML>");
}
Struts: a special Servlet

- **struts:**
  
  ```xml
  <servlet>
    <servlet-name>action</servlet-name>
    <servlet-class>org.apache.struts.action.ActionServlet</servlet-class>
  </servlet>
  <init-param>
    <param-name>/WEB-INF/struts-config.xml</param-name>
  </init-param>
  ...
  </servlet>

  <servlet-mapping>
    <servlet-name>action</servlet-name>
    <url-pattern>*.do</url-pattern>
  </servlet-mapping>

  <servlet>
    <servlet-name>multiplier</servlet-name>
    <servlet-class>MyMultiplier</servlet-class>
  </servlet>
  <servlet-mapping>
    <servlet-name>multiplier</servlet-name>
    <url-pattern>servlet/MyMultiplier</url-pattern>
  </servlet-mapping>
```
Session & request

- session: calls from same browser process
- request: each HTTP call
  - HTTP is stateless
- Usage of session & request
  - pass parameters back and forth!
    - e.g. retrieve data from database
session can be more complicated than you think

think of an application of bidding items
  - for each item, I can bid multiple times
  - also, I can bid multiple items
  - what is the scope of the session
    - biddings for a single item?
    - or across multiple items?
Database – web application perspective

- **Database**
  - data model
  - transactional storage (ACID)
    - application requirements, e.g. financial charge
  - computation

- Most web applications mainly take the DB as the transactional storage
  - many emerging systems, e.g. distributed systems/data center
  - different applications may need different level of consistency
    - ACID is a very strong requirement
  - performance & cost tradeoff