Redo your servlet from assignment 2 so that your Servlet integrates with JSPs in the style of MVC. Your Java Servlet methods: **doGet, doPost** and/or **processRequest** should do no IO.

**Delegate all IO to JSPs.**

You need three JSPs:

- **login.jsp** [publicly available] –
  - must contain the phrase **Login Service** and a button with name="submit"
  - must contain two textFields: name="userid" and name="passwd"
  - when returning to the page after a bad login attempt, must display the word: **again**

- **getDate.jsp** [in WEB-INF] -
  - must display the user’s name somewhere on the page
  - must contain the word **Date**
  - requires a **form**
    - button: with name="submit"
    - textField: name="key"

- **showResult.jsp** [in WEB-INF] –
  - display the text corresponding to the key
  - form:
    - button: name="more"
    - button: name="logout"
    - (if more, display getDate.jsp; if logout, remove session; display login.jsp)

Set up your servlet so that you support users logging in with userid and password.
- You must support a user with userid=0001 and password=0001 and name=Garcia
- You must not support a userid=0002
- Any other users and passwords are ok.

**Naming and Defining Your Servlet:**
- Define your war file using the prefix a3 and some variant of your name (e.g. a3jones.war)
- Name your servlet anything you want (e.g. dateLookup)

**Accessing Your Servlet:**
Users should be able to access your application (implemented as a servlet and three JSPs) using login.jsp or going directly to the servlet, as in:
- [http://neo.engr.smu.edu:8080/a3jones/login.jsp](http://neo.engr.smu.edu:8080/a3jones/login.jsp) OR
- [http://neo.engr.smu.edu:8080/a3jones](http://neo.engr.smu.edu:8080/a3jones)
**login.jsp**
- Captures login and password and passes to the servlet as parameters.
- If servlet sees that there are incoming parameter values for userid and password, it means a new user is logging in.

**Servlet responsibility [when userid and passwd parameters exist]**
- **DO:** invalidate any session information, before checking for valid user.
- If NOT valid user, redirect to login.jsp and include phrase: `invalid user` on the page somewhere.
- If valid user:
  - store userid and username in session object.
  - redirect to getDate.jsp [under WEB-INF].

**getDate.jsp**
- The `getDate.jsp` should display the user’s name somewhere on the page. The servlet will have stored the user’s name in the session object.
- Get the key (date) from the user with form with textfield name=“key” and pass to the same servlet.

**Returning to Servlet from getDate.jsp**
- Servlet will not find parameters userid and password. This means existing user
- Check session object for userid and username
- If not found, redirect to login.jsp
- If found in session object, lookup the key date and arrange for the corresponding phrase and the user name to be available to the showResponse.jsp
- Redirect to showResponse.jsp – which should display the text associated with the date and also display the user’s name somewhere on the page.

Track your users with a session object, so that if a user hits your servlet while their session is active, they do not see the login page but go directly to **getDate.jsp**

**Web Service -- XML Response:**

If a client hits your servlet using:

http://neo.engr.smu.edu:8080?userid=0001&passwd=0001&key=Jan+23&out=xml

Return an **XML document** just as you did in Assignment 2.
Application Flow

1. login.jsp
2. Process login
3. Ok?
   - no
   - yes
   - More

4. getdate.jsp
5. lookup
6. show Result.jsp

Assignment 3
Application Flow

logout
Servlet Application  Web pages (JSPs)

login.jsp

JSP Date Application Login Service

Please try your Login again

login.jsp --- after incorrect Login – contains word: again
**getDate.jsp** (following successful Login) – name of user should be displayed – your 0001/0001 user should be **Garcia**.

**showResult.jsp**

User presses more.
User presses logout