**Aims and Background**

This project aims to give students hands-on experience in designing and implementing a Web application on their own.

In recent years, scientists have increasingly turned to the web for keeping up-to-date with the recent publications in their respective fields of research. Citation databases such as Citeseer and Pubmed, and search engines such as Google Scholar and Microsoft Academic Search have made this job easier by providing a one-stop shop where publications can be searched using author, year or title information. The task in this assignment is to design a **web application that enables a scientist to search for and select publications through a XML-based publication database using XML, Servlets and JSP**.

**Requirements**

In this assignment,you will implement an application for searching through the MEDLINE publication database. The data will be provided to you (see below). You **must** use this data and no other. You will associate data such as search preferences and a "Reading List" with a user using cookies and HTTP sessions. You should not use a login (user/password) for this purpose.

**Home Page**

The user starts at a home page with a search interface that features:

1. 4 Input Fields, one each for Author, Article Title, Journal Title and the Number of Results to be displayed in a page with the following conditions:
	1. The number of results to be displayed field - When the user first visits the site, this field displays the default value (10). The user can change the value to any integer. On subsequent visits by the same user, the new value must be displayed.
	2. Author field - The user provides a series of last names of authors of each publication, separated by commas. E.g.: "Smith, Jones" selects all publications that have at least 2 authors, one with the last name Smith AND another with last name Jones.
	3. Article title - Series of words appearing in the titles of publications, separated by spaces. E.g.: "web application engineering" selects all publications that have all the words "web", "application" and "engineering" appearing in any order.
	4. Journal title - Series of words appearing in the titles of journal, separated by spaces. Selection condition same as that for article title
	5. When two or three fields are provided, then the search results MUST satisfy the condition for each field.
2. a submit button
3. And, a link to a Reading List page.

 After completing input, the user clicks the submit button to start the search process.

**Search Results**

1. The search functions forwards the users to the results page. At the top of the page is a link to the user's Reading List
2. If the search has turned up empty, the results page must display "Sorry, no matching articles found!"
3. The search process returns with a list of publications that match the criteria in the manner described abovethat. The publications are displayed in this format.

 Line 1: Authors, ArticleTitle, Journal Title, Journal Volume(Journal Issue), Publication Month, Publication Year

 Line 2: AbstractText

The number of publications displayed must be restricted to that set by the user on the home page.

1. The user then selects via a checkbox which of the publications he/she would like to follow-up. These publications are added via a submit button to the user's Reading List.
2. At the bottom of the page are two navigation links Previous and Next that allow the user to navigate the results 1 page at a time. Enure that the Previous and Next links are not shown on the first page and the last page of results respectively.

**Reading List**

1. The Reading List page is organised in the same way as the Search Results page except that at the bottom of the page there are 2 buttons: Remove from Reading List and Clear Reading List
2. User can select articles via a checkbox next to each article. Clicking the Remove button removes these from the Reading List. The list is then reloaded.
3. User can choose to clear his/her Reading List by clicking the Clear button. The Reading List is reloaded.
4. If the Reading List page is opened but the reading list is empty, then the page must display "Reading List is Empty!", followed by a "Go Back" button. This button takes the user to the page from where he/she entered the Reading List page.
5. The previous and next links in the Reading List page behave in the same manner as described for the search results.

**MEDLINE data**

MEDLINE is a database of journal abstracts for biomedical literature. Further information about MEDLINE can be found at the [US National Library Medicine](http://www.nlm.nih.gov/bsd/pmresources.html). You only have to work with the publications in [this XML file](http://www.cse.unsw.edu.au/~cs9321/13s2/medsamp2013a.xml.gz)(contains 30,000 records). For testing purposes, you can use a [smaller sample of the XML](http://www.cse.unsw.edu.au/~cs9321/13s2/medsamp2013.xml). The DTD for the file is available [here](http://www.cse.unsw.edu.au/~cs9321/13s2/nlmmedlinecitationset_110101.dtd).

**Assignment Execution**

You have been provided with the skeleton files for the search application. You have to implement the functionality listed above in the jsps and the servlets that have been provided to you, using the MEDLINE database that has been provided as well.

The skeleton files are provided in this zip file. A few points to note

* Please read the comments in each file.
* Please DO NOT alter any existing code in the skeleton files. You can add your own lines but keep the existing structure of the program.

**Submission Requirements**

1. Generate a war file from your project. In Eclipse, this is Right-Click on project name --> Export --> WAR file. Make sure that the "Export Sources" checkbox is checked.
2. Run the command

give cs9321 Assign1 <war-file>

Note: It is extremely important that your war file can be loaded in any Tomcat 7 in the CSE environment. Make sure you test your assignment thoroughly.

**Evaluation and Marking**

Your assignment will be evaluated using Apache Jmeter, a testing tool for web applications. We will test the functionality of the search application as well as the handling of errors such as invalid parameters and operations. This is an individual assignment worth 10 marks. You will lose marks for any failed tests.

**Important**

The answers to all these questions is NO.

1. Could you extend the deadline ?
2. Can I use XPath/XQuery/XMLBeans ?
3. There are too many records. Can I show only a subset of the database ?
4. Can I use an RDBMS ?

Please use the message board for resolving other doubts.