Web Application Lab

Assignment 1

1. Download apache tomcat (.zip) and install(unzip) it. Read documentation and configure tomcat. Check the port it uses. Start the web server. Check if it is working or not. Now, write a simple HTML file (say New\_XX.html) containing your name and put it in the web server’s ROOT directory. Now check if it is available through the web browser and appropriate URL. Change the port of the web server to 8002 and restart the web server. Check if it is working or not. (XX will be your two digit roll number).
2. Using telnet application and HTTP, obtain the file New\_XX.html from a remote machine.
3. Design a web page containing one text box (for login user id) and one password field (for password) and a submit button. Label these items. After pressing the submit button will check the password; for correct password, it will load another page (say Welcome.html). Otherwise it will show an incorrect password message. [Create a database table to store userID and password using Java Database Connectivity.

NB: Information use connection url: jdbc:oracle:thin:@172.16.4.121:1521:ibmora

Download the necessary driver for Oracle version. ]

1. Write a simplified web page for online fees payment. The page contains two drop-down lists (i.e. combo box) captioned year and semester. The first one contains values 1st, 2nd, 3rd and 4th and second one contains values 1st and 2nd. The page also contains a group of two radio buttons labeled “pay fees now” and “pay fees later”. When a particular year and semester will be selected, the fees amount will be display. If the radio button labeled “pay fees later” will be selected, an additional text box (to hold the last date for payment) appears, otherwise a submit button will appear.
2. Write a question paper as an XML file (question.xml say). The DTD is given below:

<!ELEMENT question-paper (question\*)>

<!ELEMENT question (text, optionA, optionB, optionC, optionD, answer)>

<!ELEMENT text (#PCDATA)>

<!ELEMENT optionA (#PCDATA)>

<!ELEMENT optionB (#PCDATA)>

<!ELEMENT optionC (#PCDATA)>

<!ELEMENT optionD (#PCDATA)>

<!ELEMENT answer (#PCDATA)>

<!ATTLIST question no CDATA #REQUIRED>

<!ATTLIST answer value (optionA|optionB|optionC|optionD) #REQUIRED>

Now, validate your XML file against this DTD.

1. Consider a web page that locally stores (using IndexDB) the details (such as Roll no, name, year, semester, Mobile no etc) of different students of your Department. The web page contains buttons captioned “add”, “show”, “update” and “delete”. When “add” button is clicked, the pages shows a dialog box to get the details of the student to be added and inserts it to the IndexDB. Implement the functionality of the other buttons. Consider (Roll no) as the primary key.