Dept. of Information Technology
Jadavpur University, Salt Lake Campus,
Block-LB/8, Sector-III, Kolkata-700106

Web Technology Lab.

Important Instructions

1. Please sign twice (when you enter and leave the lab) in the appropriate places of the attendance sheet that will be placed on the lab instructor's table
2. All assignments given must be completed individually in the laboratory.
3. **Talking inside the lab is STRICTLY prohibited. No discussion with your classmates is allowed. Get your issues (if any) resolved by the lab instructors only.**
4. Do keep your mobile phones in silent mode.
5. Use login credential (username name: user, password: user) to login to the desktop system in windows OS
6. It is recommended to use tomcat as the web server. For other web servers, concerned student has to take sole responsibility from installing to configuration and maintenance.
7. A tomcat web server is already installed (and configured) in every computer in the lab. However, a student should install and configure a tomcat of his/her own.
8. Since JavaScript does not provide diagnostic errors messages→write a few lines in your program and test. This debugging process easier.
9. In your solution of the assignments, write only those instructions that you understand clearly. Instructions present in your program but remains unexplained will be considered as 'copied'.
10. Do write appropriate comments throughout your programs.
11. Save all files regularly during the lab period to avoid losing your unsaved programs/data. Create a single web application (containing a directory for every assignment) and put it in the tomcat's document root.
12. Use appropriate names of your program files for seamless identification of the programs corresponding to the assignments.
13. Backup your programs as it is not guaranteed that you will get same computer everyday.

To work with MySQL database(your login credential will be provided later), use either of the following methods:

Method 1:

Visit the URL   **http://172.16.4.50:8080/sqlclient1**
Create a new source. Provide necessary information. Sample information for be2001 is given below:

Source: be2001@172.16.4.50
JDBC Url: jdbc:mysql://172.16.4.50:3306/test
Driver Name: Select MySQL Driver
User: be2001

A source will appear under "Sources" tab in the "Sources, Drivers, Active Connections" window. Double click the source. Provide password. If everything is fine, a connection will be established. You will then have a GUI based environment to work with the MySQL database.

Method 2:

Use following command in the command prompt:

mysql -u <username> -h 172.16.4.50 -p

It will ask for the password. Provide your password. The MySQL command prompt will appear.

Assignment sheet

1. Download apache tomcat (.zip) and install(unzip) it. Read documentation and configure tomcat. Configuration typically involves defining an environmental variable JAVA_HOME whose value should be the home directory of JDK. Check the port it uses. Start the web server. Check if it is working or not. Now, write a simple HTML file (say first.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.[CO1]

2. Using telnet application and HTTP, obtain the file first.html from a remote machine.[CO1]

3. Now, write Java socket program to obtain the file first.html from a remote machine.(optional assignment)[CO1]

4. Design a web page containing one text box (for login name) and one password field (for password) and a submit button. Label these items.[CO1]

5. Write a necessary JavaScript functions to verify the content of text box and password field in previous assignments. If there is any error, show appropriate message in red color in a label. Use following rules:
   A login name can only have alphanumeric characters and '_'(underscore)
   Number of characters in the login must not exceed 10
   A password must have 6 to 10 characters containing at least one digit and one uppercase letter.[CO1]

6. Write a web page home.html that loads another page (say logout.html) if user is ideal for 10 seconds[CO1].

7. Write a simplified web page for online bill payment. The page typically contains a group of two radio buttons labeled "pay now" and "pay later". When the radio button labeled "pay later" is selected, an additional text box (to hold the date of payment) appears. [CO2]

8. Write a web page that 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. When a particular year and semester is selected, a list of subjects taught in that specific year and semester should appear. [CO2]

9. Write a web page that contains two drop-down lists captioned states and districts(initially empty). The first one contains the names of states in India. When a specific state is selected, the second will have names of districts of the selected state. When a specific state and district is selected, show the information of that district in very brief. [CO2]

10. Write an applet to display the message "Hello World".(Optional assignment) [CO2]

11. 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. [CO2]

12. Use HTML5 canvas and <SVG> tag to create following picture. Create one and use transformation to create others. [CO2]
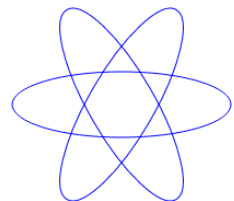
13. Consider a web page that locally stores (using IndexDB) the details (such as name, manufacturer, price etc) of different components(e.g. HDD, Monitor etc) of the computer. The web page contains buttons captioned "add", "show", "update" and "delete". When "add" button is clicked, the pages shows a dialog box to get the details of the item to be added and inserts it to the IndexDB. Implement the functionality of the other buttons. Consider (name, manufacturer) as the primary key.[CO3]

14. Write a simple web page in a web application running under tomcat. The web page displays one image and refers to one JavaScript source file. Use HTML5 application cache to cache all these in the web browser. Check if they are really cached or not. Now shutdown the tomcat and test the page is working or not. (Optional assignment) [CO3]

15. Write a web page that generates a factorial table using a web worker. The main page sends an integer number (say 5) and the worker returns the factorial table (1 to 5) and the main page finally displays it. [CO3]

16. Write a servlet to display the message "Hello World".[CO4]

17. Develop a simple chat web page for using WebSocket. [CO4]

18. Develop a simple web page (use SSE) showing price of at least two stocks.

19. Create a web page with appropriate forms, and servlets, to perform each of the following actions: [CO4]

    a. Form 1: Accepts a string, and displays a list of students whose name contains the given string, their roll numbers, dept_name etc.

b. Form 2: Shows a drop down menu containing the names of all departments (generate this from the database), and when a particular department is selected, a list of all students in the department is displayed (as in the previous question).

20. Write a JSP page to display the message "Hello World". [CO4]

21. Write an application as follows: For the first request, the server generates an HTML file to display the integer 0, with two buttons captioned "prev" and "next". If the user clicks on the *next* button, the server sends the integer next to the one displayed. Similarly, when *prev* button is clicked, the server sends the integer previous to the one displayed. Track the session this using, Hidden fields, URL Rewriting, Cookies, and Session API [CO4]

22. Design a simple page to open an account. The page should contain one text box for login name, one password box and other information. Use AJAX technology to inform if the login name typed is available or not. Write a target JSP page to insert data supplied in the form into the database. [CO4]

23. Write a web page that contains two drop-down lists captioned states and districts(initially empty). The first one contains the names of states in India. When a specific state is selected, the second one should be populated by the values obtained from server using AJAX technology. Create suitable tables in the database. When a specific state and district is selected, get the information using AJAX and show it. [CO4]

24. Design a simple login page. Use AJAX technology to print appropriate error message in case of a login failure. [CO4]

25. Write a Java class as follows:
    a. The constructor takes the name of an XML file containing object type questions with answer.
    b. The insert() method inserts the questions together with the options and answer into the database table.
    c. Now write a page that allows user to upload an XML file and use previous class to insert questions into the database [CO4]

26. Write a JSP page to retrieve all the questions together with options and generate a HTML file as follows: [CO4]
    a. Each question will have query text and a set of radio buttons one for each option.
    b. HTML page contains a submit button.
    c. The visitor answers the questions by selecting appropriate radio buttons and finally presses submit button.
    d. Also write a JSP page to collect the answers and to inform how many of them are correct after consulting the database.
27. Use Ajax Technology to change the password of a user. The page usually contains three password boxes and one button. The user will type the old password in the first field and new one in the rest two. [CO4]

28. Write a XSLT file to transform a XML question file into a HTML file. Test if it is working or not. [CO4]

29. Create a database table to store the user names and the corresponding passwords. Also write a JSP page to verify the login name and password supplied by the users through the page designed in 15 against the values stored in the database tables. [CO4]

30. Create a database table to contain price, manufacturer and model of different components (such as HDD, Motherboard, processor, RAM, Monitor, CD/DVD R/W etc.) of a computer system. Create an html form where user will select model and manufacturer of different components and asks for the total price. Write a JSP page to display the total price of the system for the given configuration after obtaining the information from the database. [CO4]

31. Create a database to store the marks of all students in all semester in all subjects. Write a JSP page to select the semester, roll number and subject. JSP page will in turn find the corresponding marks and display it. You can make your own assumptions depending upon your needs. [CO4]

32. Design a HTML page to create an account. The page should contain one text box for login name chosen, two password boxes (for double verification), one for email address, one for your name and one for your contact number and others (if you think necessary) . When a visitor provide these data, validate them using JavaScript code. If all data are correct, send it to JSP program (say create.jsp) which inserts the data into database tables. Also write pages to check whether a user can now login or not. [CO4]

A few CGI assignments: [CO5]