Exercise7: ANGULAR Js

Objectives:

To learn to use angular js framework

Work with your group (or by yourself). Each group is to upload only one submission.

1   Run UwAmp server

UwAmp is a WAMP server (Windows Apache MySQL, PHP). It does not need administrator privileges to install/run. You can download it by using the following steps. You can also run it off a USB stick.

1.1   Download UwAmp

1)   go to http://www.uwamp.com/en/?page=download

2)   download .zip file .and extract the contents into a folder.

3)   change the port number to 8080

4)   start UwAmp

1.2   Check to make sure it is working ok

1)   Run UwAmp

2)   Select the first two check boxes you get.

3)   Make sure that ALL the code you want to run is in a folder INSIDE www folder of UwAmp

2. Warm Up: Try Some Examples

1.   First, open blackboard, go to Course Contents, and then download exercise11.zip file into your workspace (U:\workspace or something like that!). Then, unzip.

2.   Copy all the contents of the assignment to www folder of Uwamp.

3.   Play with each of the given examples (in examples directory). Open them using a text editor of your choice and modify parts of the html or js files.

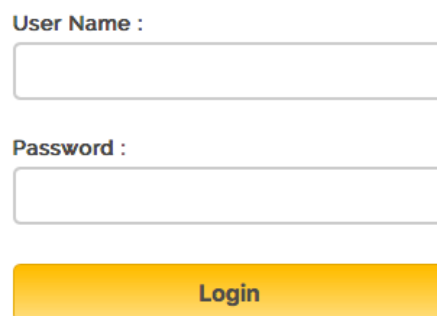IT IS REQUIRED THAT YOU TRY EACH EXAMPLE.

2   Library

The goal of this exercise is to design and implement a books library <mark>using angular Js framework</mark>.
Use angular js views, controllers and routing to design your books library.

1. Assume there are 25 books present in the Library.

2. There are four categories of books : Art, Science, Sport, literature.

3. Assume we have the following types of book in the Library.

    - 5 Reference books
    - 20 ordinary books.

4. Each book should have an attribute of borrowedBy which showed the user name of the student who borrowed that book. At the first none of the books are borrowed by students.

5. Each book should have an attribute of presence which shows the presence (1) or borrowed(0) situation of the book. At the first all books are present. Use local storage in order to save presence attribute and borrowedBy attribute for each book.

6. <mark>Note that Reference books cannot be checked out.</mark>

2.1   The library System Operations

2.1.1   Login page

1.   User Interface: Should consist of Username and password fields and the login button.

User Name :

Password :

Login

2.   Functionality:
   1.   If the Username and password are equal to admin, login as a librarian.

   2.   If the username starts with U, login as an Undergraduate student.
   3.   Otherwise, Show an alert that this is not a correct username and password.

2.1.2 Librarian view

1. User interface: Show library with current shelves and books (HTML table is generated on the fly where each column is a shelf. Each cell is a book)

Shelf Literature	Shelf Science	Shelf Sport	Shelf Art
R1	R3	B6	B13
B1	B11	B7	B14
R2	B12	B8	B15
B2	B4	B9	B16
B3	B5	B10	B17
R5	R4	B19	B18
B20			

2. Functionality:
    a. Ability to add a book to the specific shelf. Create a text box for the name of the book, a dropbox for selecting shelf, a checkbox which indicates the reference book and an add button. When the librarian press add button, the book should be added. Also, set presence characteristics to 1 and show on the table.
    b. Clicking on the cell gives book's detail which consists of BookName, BookType, BorrowedBy, Present .

2.1.3 Undergraduate View
1. UserInterface: The undergraduate student should have the similar view as a librarian.

2. Functionality:
    a. An undergraduate student can borrow at most two ordinary books (remember that reference book should not be able to checked out) each time. So, a student can select at most two ordinary books and change the color of that cell to red (i.e. the book is borrowed, so change the borrowedBy attribute to the username).
    b. The student should be able to return the book which he/she borrowed and change the color of borrowed book back to white.

MAKE SURE TO:

- [ ] Use Angular views and controllers. **THIS MUST BE A SINGLE PAGE APP**.

- [ ] It should contain minimum **three views** (one each for Login, librarian, and student).

- [ ] Use **routing** to navigate to different views.

- [ ] Make sure to have the **Librarian, Student, and Login object** perform the required operations.

- [ ] Make sure to use **ng-repeat** to draw table.

- [ ] Write a Client HTML code that includes the angular ng-app and use ng-views

- [ ] **You will need to submit all your controller files, app.js, and html files**

2.1.4 Submission:

Make sure your solutions work on Chrome (which is what TAs will use to grade the assignment). Zip your html, js files, and participation file (i.e. who worked on which part or if you worked together). Then, submit this zip file on black board. Remember there is only one submission per group. Make sure to include all the files that are needed in order to run your program.

Participation file is a simple txt file, which clarifies the specific participation of two members.