King Fahd University of Petroleum & Minerals
College of Computing and Mathematics

SWE 363: Web Engineering and Development (Term 232)

Homework Assignment #3

Submission Guidelines

- It is an individual assignment.

1. Use your work on Assignment 2 and add JavaScript (as a separate file and link it to the HTML) to produce an interactive website like the one in this video.
2. Install node on your system then type the following commands (in VsCode you can open the integrated terminal to run these commands):
   a. `npm install -g json-server json-server-auth`
   b. create a folder named backend in your code and save the attached db.json file in it.
   c. navigate to the backend folder then run the following command to start a server:
      i. `cd backend`
      ii. `json-server-auth -w db.json`
3. Create a file named `forntendJS.js` to hold all your JS code.
   a. Create a function named `toggleForms` to handle when the user clicks on the register button on auth.html. it should hide the login form and show the register form.
   b. In index.html, make the form for asking a new question hidden by default then create a function named `openNewQuestionDialog` to popup a modal containing the form when the user clicks on the "Ask New Question" button. The function should close the modal when the user clicks on the "X" button.
   c. Remove the hardcoded list of questions and create a function named `getQuestions` to retrieve the list of questions from the server by sending a GET request to `'http://localhost:3000/questions?_sort=createdAt&_order=desc'` using fetch. On successful response call another function that you will implement named `showQuestions`. The `showQuestions` function loops through the retrieved questions and creates the necessary HTML to display the questions on the page. Make sure to call `getQuestions` when index.html loads. Hint: create a global variable to hold the array of questions as you will need it in other places.
   d. Create a function named `sortQuestions(criteria)` to handle when the user selects a sorting criterion from the dropdown menu at the top of the page. The function should sort the array of questions using the criteria and call `showQuestions` to update the page.

e. Create a function named `filterQuestions(criteria)` to handle when the user selects a filtering criterion from the dropdown menu at the top of the page. The function should filter the displayed questions using the selected.
f. Create a function named `validateLoginForm` to handle when the user submits the login form in auth.html. the function validates that all fields are required then it sends the form as a POST request to 'http://localhost:3000/login'. If the response is unsuccessful, alert the user to use the correct email and password (use "ahmed@example.com" and "abc123"). On successful response, store the response in `localStorage` with `user` key, hide the login link in the header, and navigate to index.html.
g. Add an event listener for page load to check the localStorage when the page loads. If the user key exists hide the login link in the header.
h. Create a function named `validateRegisterForm` to handle when the user submits the registration form in auth.html. the function validates that all fields are required except the profile picture then it sends the form as a POST request to 'http://localhost:3000/register'. On successful response, store the response in `localStorage with user key`, hide the login link in the header, and navigate to index.html.
i. Create a function named validateNewQuestionForm to handle when the user submits the new question form in index.html. the function validates that all fields are required, convert the tags to an array, then it sends the form as a POST request to 'http://localhost:3000/questions'. The function should read the user information from the LocalStorage if it exists otherwise use id=1 and username=Anonymous. Similarly, the function should populate the fields that are not part of the form like: answers, answered, and createdAt.
j. Make sure to use the same names and structures of the data exchanged with the server as given in the db.json file. You don't need to keep track of the id values or pass it when creating new resources, the server will handle that.

Deliverables:

- A compressed file containing all the HTML, CSS, and JS

Useful Links

- https://www.w3schools.com/jsref/prop_win_localstorage.asp
- https://developer.mozilla.org/en-US/docs/Web/API/Window/DOMContentLoaded_event