Web Applications [2023/2024]

Lab 6: JavaScript in the Browser

In this lab, you will add dynamic behaviors to the FilmLibrary interface that you have implemented in the last lab.

1. Populate the Film Library

Starting from the outcome of the previous lab, replace the hard-coded films displayed in the main content area with a dynamically populated list of films. For this purpose, you can use the same **FilmLibrary** object that you created in previous labs, containing the films with their corresponding data. Remember that this **FilmLibrary** object must be declared in a JavaScript file that you should import into the HTML file via the `<script>` tag.

Then, modify the GUI to display the films in the list dynamically. To ensure the correct integration of the **FilmLibrary** object, if you change the content in its JS Films array and reload the page, the list in the main content area must update accordingly.

2. Let's make the filters work!

Implement the filtering functionality by associating the items in the sidebar to the following filters:

- **All**: retrieves and display *all* the films in the Film Library.
- **Favorite**: retrieves and display only films marked as *favorite.*
- **Best Rated**: retrieves and display only films whose score is **five** out of five.
- **Seen Last Month**: retrieves and display the films watched between today and the last 30 days.

Use the DOM manipulation methods to **update** the list of films displayed in the webpage based on the applied filter. When a filter is applied, you must update the specific HTML where the films are being displayed.

Hint: it is easier to completely delete the list of displayed films and then create a new one rather than trying to update the displayed list of films.

Take also into account that the filtering function should not modify the original set of Films in the JavaScript data structure, it should just return a subset of them. Finally, add some suitable entries to the **FilmLibrary** to test all the filters.

Note: For simplicity, only one filter may be active at a time.

"**All**" is the active filter by default when the page is loaded. The interface should always highlight the currently applied filter. Additionally, the interface must display, at the top of the main content area, the name of the current applied filter. Checkboxes for the favorite attribute does not need to work now.

3. Optional: Add the functionality to delete the Films

Enable users to delete specific films. Add a trash icon (🗑) at the end of the row for each Film and link it to a JS function that removes the corresponding Film from the **FilmLibrary** array. When a Film is deleted, the list of films should update automatically, maintaining the current filter activated.

Note: a deleted film cannot be restored. If you want to go back to the full list of films you need to re-run the JS code from the start, i.e., reload the page.

Hints:

1. **Use the Bootstrap framework** (v. 5.3) and its components to implement the static webpage: https://getbootstrap.com/docs/5.3/getting-started/introduction/
2. Use the methods for DOM manipulation to dynamically populate the list of films defined in the JavaScript data structure (e.g., `getElementById`, `createElement`). Thus, add the `id` attribute wherever you deem it useful in the HTML.
3. You can include the `dayjs` library in your HTML page by inserting the following code in the HTML head section.

```
<script defer src="https://unpkg.com/dayjs@1.11.10/dayjs.min.js"
  integrity="sha384-DpVxUeeBWjUvUV1czyIHJAjh+jYUZFu2lLakbdua5vbwOrBGi1UgaKCHjTC+x3Ky"
  crossorigin="anonymous">
</script>
```

   Note: remember to use the `defer` attribute to postpone the loading of the script.

4. **Reminder**: you **must not** execute your JavaScript application using Node.js. You must include the JavaScript inside the HTML page using the `<script>` tag. To check the JavaScript execution you can still use the `console.log()` method: printing will happen inside browser's developer console (typically, you can open it pressing F12 inside your browser, or search the developer tools in the browser menu).
5. You can use the solution of the HTML/CSS lab as starting point.