DOM

DEADLINE: 12/04/2021

FOLDER STRUCTURE

```
FL1_HW13/ [*]                                          [*]  - required
    ├── task [**]                                      [**] - restricted (no need to submit)
    ├── homework/ [*]
            ├── material-icons/ [*]
            ├── index.html [*]
            ├── styles.css [*]
            ├── app.js [*]
            └── .eslintrc.js [*]
```
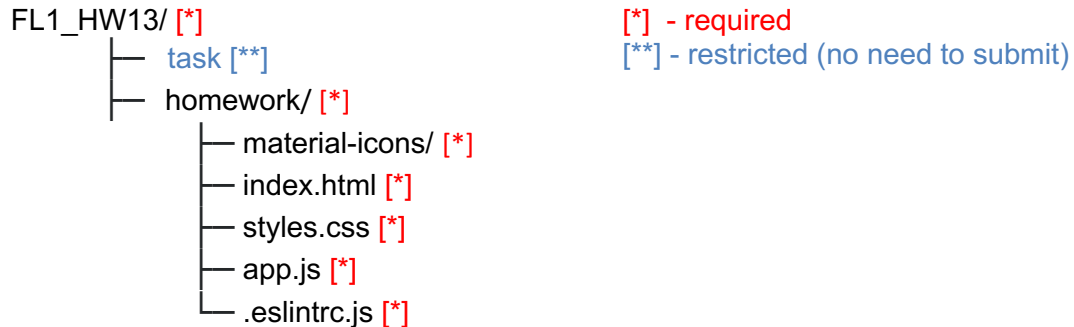
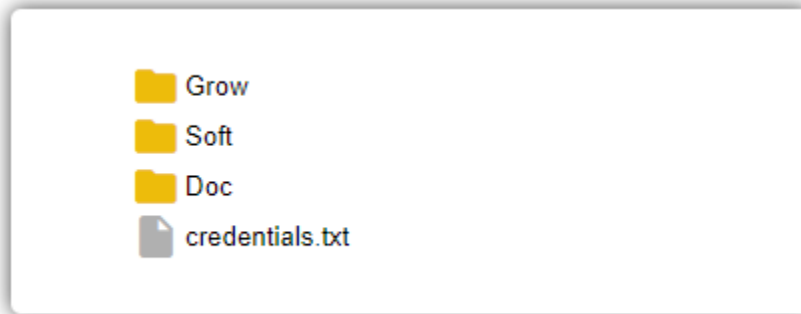## TASK

- Using the standard JavaScript functions, create a file tree (see Figure 1) based on the data in structure variable.
- Requirements:
- By default all folders are closed
- When you click by left mouse button on a closed folder, folder opens. Otherwise close.
- When you click on file by left mouse button - nothing happen
- If the top-level folder closes when the subfolder is open, then after reopening the top-level folder, the subfolder remains in the same state.
- Use the appropriate icon if the folder is open or closed.
- Folder/File may be clickable out of name
- The folder / file is clickable outside the title. (It means you can open folder by clicking on row instead title directly)
- Create a custom context menu, which opens with the right-click
- Menu has two options: rename and delete item
- When context menu is open appropriate item become focused
- If context menu open outside of the item, rename and delete options are disabled
- If you rename the file, the file name without it's extension should be selected
- If you delete the last item in the folder, the "Folder is empty" text should appear.
- Context menu can be hidden by left-click outside it's container
- Icons, to the left of file/folder names can be found at the link - https://material.io/icons/ (they are already included). Use icons like this

`<i class="material-icons">done</i>` will display icon ->

- Figure 1 - Folder tree (To see folder tree in the action, open - /animations/HM_14.gif)

RESTRICTIONS

- Editing index.html is forbidden.
- Adding **task/** folder is forbidden. Do not push it to repository. (Only **homework/** and **material-icons/** folders should be pushed)

BEFORE SUBMIT

- Code should be clean, without comments, readable, and tested
- Make sure your GitLab folder structure meets folder structure from this document (without **task** folder)

SUBMIT

- The **FL1_HW13** folder without **task** folder should be uploaded to GitLab repository "**FL-HU1**" into **master** branch.

USEFUL LINKS

- https://developer.mozilla.org/uk/docs/Web/API/Document
- https://learn.javascript.ru/traversing-dom