

5G00DM03-3003 Basics of Web Development,

Exercise 7 (Max 20 XP)

Let us practice transferring data between frontend and backend sides.

Create a subdirectory with name **ex7** in your local Git working directory, and operate within it. Create files **ex7.html**, **ex7.js**, and **ex7.css**, and implement the task features for the (single) web page resulting as a combination of these files. The user should not need to run **npm install** (or anything like that), so make sure you either refrain from using external libraries altogether or load them from a content delivery network.

Task 1 (max 5 XP)

Make a GET request to <https://aws.random.cat/meow>, and make your page to show a hypertext link pointing to the (random cat picture) address you'll get within the response.

Task 2 (max 15 XP)

Copy the file **db.json** from Moodle to some directory (which does not need to be **ex7** or any git-controlled directory). To keep it simple for now, let us use the file as our database as was done during the class. (That is, run, e.g., `npx json-server -H localhost -p 3010 -w db.json` command in the directory in which **db.json** is located. This starts the backend server to make requests to. (Generally, the port could be any free port, but let us fix 3010 to be used now in order to facilitate checking the submissions.))

After the link of Task 1, implement these features:

- Add a button with the text “List all” (1 XP)
- Clicking this “List all” button dynamically fetches the necessary information and shows a table listing all the students in the “database,” one per row. The table content is updated every time the button is clicked. (3 XP)
- For each student row, three information columns are shown corresponding to the JSON properties *name*, *info*, and *exercise_points* (in this order). The columns should have appropriate headings. The exercise point values should be aligned to right in their column. Possible null values should be represented by “–” (n-dash character) (3 XP)
- The rows concerning students not known to have at least 50 exercise points (i.e., the value of the *exercise_points* property is less than 50 or null) should be highlighted by some reddish background color. (2 XP)
- Add appropriate input fields for the name, the informative text, and the number of exercise points concerning a new student, as well as a button with “Add” text. (2 XP)
- Clicking this “Add” button adds a student instance using the given input information into the “database” by making a POST request to the back end. Use suitable default values, if input values are missing. After adding a student, the student list should be shown/updated. (3 XP)
- Add clarifying headings and labels and decorate the page as you wish. Figure 1 depicts an example of a simple implementation. You are free to create something much more beautiful! (1 XP)

Exercise 7

A link to a random cat picture

https://purr.objects-us-east-1.dream.io/i/20171129_203407.jpg

Student list

List all

Name	Info	Points
Lennier		232
Sheridan		53
Mollari		–
Kosh	Did not participate	0
Zathras	Species: unknown	1
Student X	No info.	60

Add a new student

Name: Info: Points:

Figure 1: An example page after implementing the task features and clicking the “List all” button. (Student X has been added.)

Submission

Commit the files (db.json not needed) with commit message “Submit Exercise 7”. Push into the remote repository. Get the address to the commit by right clicking the name of it in GitLab and selecting “Copy Link Location” (or something like that). Paste this address as your answer to Moodle (Session 7 homework submission area), and submit. **N.B.: you should submit a single address only pointing to the commit, not several addresses pointing to, e.g., individual files.**