<WA1/>
<AW1/>
2021

JavaScript Exercises

Examples and Exercises

Fulvio Corno
Luigi De Russis
Enrico Masala

Improve my scores

- Develop a small JS program to manage the scores of your exams that you got in your Bachelor degree. You should:
  - Define an array with all your scores, in chronological order
    - For the time being, embed the scores directly in the source code
    - For the time being, ignore the course name, CFUs and date
    - For the time being, ignore the 30L
  - Eliminate the two lowest-ranking scores
  - Add two new scores, at the end, equal to the (rounded) average of the existing scores
  - Print both arrays, comparing the scores before and after the “improvement”, and showing the averages in both cases

My courses list

- Develop a small JS program to manage a list of your courses.
  - Define the names of your courses, as a comma-separated list
    - E.g.: “Web Applications I, Computer Architectures, Data Science and Database Technology, Computer network technologies and services, Information systems security, Software engineering, System and device programming”
  - Parse the string and create an array containing the names, one name per array position.
    - Ensure that there are no extra spaces.
  - Create a second array by computing the *acronyms* of the courses (the initial letters of the name)
    - E.g., Computer Architectures -> CA
    - Acronyms should be all-capital letters
  - Print the resulting list of acronyms and full names
    - Extra: in alphabetical order of Acronym

Courses and scores, together

- Using JS objects, merge the two previous exercises, and manage a list of objects that will include information about the exams:
  - Course Code, Course name, CFU
  - Attained score (number between 18 and 30, plus a Boolean for the *laude*)
  - Date
- Define a constructor function **Exam** to create a new object
- Define a constructor function **ExamList**, with the following methods: → →
  - add(exam) // pass a fully-constructed Exam object
  - find(course_code) // returns the matching Exam
  - afterDate(date) // returns an ExamList with the subset of Exams after the given date
  - listByDate() // returns an array of Exams, sorted by increasing date
  - listByScore() // idem, by decreasing score
  - average() // return the average value

License

- These slides are distributed under a Creative Commons license “**Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)**”
- **You are free to:**
  - **Share** — copy and redistribute the material in any medium or format
  - **Adapt** — remix, transform, and build upon the material
  - The licensor cannot revoke these freedoms as long as you follow the license terms.
- **Under the following terms:**
  - **Attribution** — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
  - **NonCommercial** — You may not use the material for commercial purposes.
  - **ShareAlike** — If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
  - **No additional restrictions** — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.
- https://creativecommons.org/licenses/by-nc-sa/4.0/