**Programming expectations and conditions**

We expect to receive your exercise via a link to VCS platform (GitHub, GitLab, BitBucket, ...)

As a general rule, we want to assess your skills in realistic conditions as you would have on the job. Therefore:  
  
·        You have **internet** use it! Consult JS references, troubleshooting resources, etc.  
  
·        While the scope of the exercise is small you need to consider this functionality is part of a bigger codebase with a lot of developers expected to understand and augment your code as easily as possible:  
  
o   **Readability**: Think about how easy it will be for other developers to take over your code.  
  
o   **Flexibility/Generalisation**: Create a component that can be integrated easily in any application  
  
o   **Robustness**, things do not always go as expected, right?  
  
o   **Reutilisation**:The component should be reused to display other kinds of data  
  
**As a general rule, we prefer to see a solid foundation start rather than a rushed working solution which do not meet above conditions.**  
**The Exercise**  
The goal of the exercise is to create a list of test-takers (test-takers can be seen as users).

The homework has to be done in JavaScript (no TypeScript or other languages)

Create a component that enables you to load the data from the API and display the names of the test takers. Here is the link of the API: [http://hr.oat.taocloud.org/api/](https://slack-redir.net/link?url=http%3A%2F%2Fhr.oat.taocloud.org%2Fapi%2F&v=3)

* The component should contain one row per test takers, the number of rows can eventually be limited.
* The rows can be sorted
* The component let’s the user see the test taker details
* The source code must be written in plain JavaScript, CSS and HTML only. No library nor framework is allowed however you can use any tooling library. For example, you can’t use Angular, React or jQuery but you can use webpack, browserify, eslint.
* The component should work on the last versions Firefox, Chrome and Safari (including for mobile).