

Lab3 - Express server

1- Provide REST API implementing CRUD operations on Employees

- Use a filesystem, same as Lab01.
- You must use middleware to validate inputs.
- You must return descriptive status codes in case of errors.
- Routes:
 - GET /employees
 - GET /employees/\${id}
 - POST /employees
 - DELETE /employees/\${id}
 - PATCH /employees/\${id}

2- Provide a Static files server

- Download this [image](#) and serve it using express static.

3- Serve SSR html page on route GET /

- Render the current employees list in styled components
- Serve css from separate file

Bonus:

- 1- Can filter employees using queryparam (one filter)
Ex: `/employees?name="ahmed"&salary=10000&...`

Must read before Day4:

- [Introduction to Mongoose for MongoDB](#)
- [REST API Best Practices – REST Endpoint Design Examples](#)
- [Authorization vs Authentication](#)
- [Hashing vs. Encryption vs. Encoding](#)
- [JWT](#)
- [CORS](#)