# Node.js task

1. First, make the environment work (you will get help for sure)
   a. database instructions and passwords, etc.: in Teams, DevopsLinux > Files > IdeaCase_NodeJS_projectTask
   b. Clone Repo: https://github.com/valju/idea-case-backend
      Read the installation instructions on readme.md and you will probably have to watch some of the recordings of the lecture.
2. Test the functionality of the environment, for example. /api/category
   a. should fetch all the categories from the database
3. a) Enable Postman and put a POST request at /api/category with such JSON data (body > raw > text=>json)

```
{
 "name": "Pre-Historic tours",
 "description": "Going to museum but also real evet places",
 "budgetLimit": 9999,
 "isActive": 1
}
```

Should succeed. Follow: 1. the return value (400?) and returned data in Postman 2. backendLog.log 3. backend console 4. database data, all of these.

b) Do it again. Should fail this time as the name (UNIQUE CONSTRAINT) already exists.

c) Change the name of the Category, the new addition should be successful again.

d) Change the budgetlimit e.g. to 9999999, should fail with validationError.

4. Ok. when all is well, choose from the following tasks to your liking, so that you use the usual MAX 7-9 hours before next session. Max 6 hours after lectures if you attended (long) lectures.

a. Change the routes in the member.js file to use the helpers of the responseHandlers/index.js file, such as category.js already uses. That is, shorten and tidy them up. Add validation to at least three routes, using the example from category.js POST/api/category and validationHandler/index.js
**OR**

b. Convert all the code in the project to TypeScript and test the operation. And add a new validation to two routes, e.g. PUT /api/category and other. Note that then id must come in inside the request. To know which Category should be updated. Unlike in POST, which will only receive its AUTO-INCREMENT id from the database server, after successful insert.

**Submission format:**
either a) repo link
or
b) one Word file (4-1 5 pages) saved as PDF. Where you have collected the most important parts you have edited in VS Code and the surrounding old code that helps to understand them as needed. As well as some screens from Postman. And to start with, you can tell in a couple of lines what you got done.

The task is extensive in terms of initial files and concepts , and it will be interesting what all the challenges will arise. For some, everything is likely to work as a train toilet.

Start now, take steps until postman experiments as soon as you can. That you get help with environment debugging issues.

Please note that here we are looking for a 5-9 hour effort. Lectures including max 8-9 hours. Write down what you do and how far you got to.