Backend Development Assignment
Dream journal (REST API)

Domain
<u>Dream</u> is entity which has a <u>title</u>, <u>description</u>, <u>date</u>, and <u>type.</u>
Dream <u>type</u> is an enum with following values: <u>happy</u>, <u>sad</u>, <u>exciting</u>, <u>scary</u>.

Tasks
1) Create an empty **Express.js** app.
2) Implement "**Get all dream types**" endpoint which should return an array of strings (values of the enum).
3) Implement **Dream CRUD** endpoints.
4) Implement **Dream search** (with pagination). Dreams can be queried by any of the following: **title**, **type** and **date** range.
   Example: Get all scary dreams between 1.2.2021 and 5.2.2021.
5) Implement **API logging** (minimalistic output within the CLI) using the external tool that is most fitting for the stack of your choice.

Tech stack
- You can chose between plain **Javascript** and **Typescript** (preferred).
- You can chose between **SQL (PostgreSQL)** and **NoSQL (MongoDB)** database.
- You can use any **ORM/ODM** such as **Mongoose**, **Sequelize**…
- You can assume that app should start in dev mode (no need for setting up multiple environments such as stage or prod).
- You can include any dependency (e.g. **moment** for date-time manipulation).

Deliverables
- **Public code repository** which should include:
  - **Source code**
  - **README.md** file with instruction on how to start the app
  - **ASSIGNMENT.pdf** (this very file)
  - **Postman collection** export with all implemented requests

Notes
- The assignment is a one-day activity. The solution should be submitted before midnight, and any commits to the repository made after that period will not be considered.
  Even though it can be a whole-day activity, we highly encourage working on the assignment within the 8 hour time scope, and pushing commits to remote repository in real time as you complete each task.

- You should not be discouraged if you do not finish all tasks - that is pretty much expected since the number of tasks and their complexity is not balanced with the timeframe allowed for the completion. The assignment is intentionally challenging to allow better understanding of your current capabilities and skills.
  The **quality** of completed tasks will be of <u>primary</u> concern, and the **number** of completed tasks of <u>secondary</u> concern.

- Each task should be a new commit. This will allow for partial completion of the assignment which will still be valuable.
  Example: Finishing task #2 and not finishing task #3 will still provide value. On the other hand, combining them and ultimately having it unfinished provides almost no value.

- If something is not clearly defined within this document, use common sense to deduce the correct approach which will produce a solution that matches constraints of the assignment the most.

- Use a public **Github** repository as remote.

- Access to the repository should be given to user **@deltakroneker**.

- If you are satisfied with the solution you provided, we highly recommend to keep (and improve if needed) your solution on your Github profile whatever the outcome of the position we are offering.