**HOMEWORK 2**

**PREREQUISITES:**

1. Install nodemon and express dependencies (if not installed already)

2. **Optional –** Try installing express-generator and check if it is properly installed by running express -h command. (this should show few available options)

3. Create a repo for your homework tasks on **Github** (https://github.com/) or **git.epam.com**.

4. Provide your mentor with the link to the repo and add read access permissions.

5. Create package.json by running the following commands npm init or npm init -y.

**Task 1:**

Implement an express server that runs on 3000 port and configure the below:

1. Enable the server to parse incoming requests with JSON payloads
2. Enable the server to serve static files (use built-in middleware)
3. Enable the server to parse incoming requests with URL encoded
4. Change the server settings to accept maximum request body size to 10 MB (this applies for JSON and URL encoded)
5. Enable router config on express

**Task 2:**

Modify the above server implementation to meet below needs:

We are trying to build APIs for a bookstore. Each book should have valid properties as mentioned below

* **Use fs module wherever necessary**

Book name - string

ISBN – International Standard Book Number - string

Publisher Name - string

Author Name - string

Author ID - number

Price - number

Published year - number

Reviewers – Array

Number of pages: number

Country: string

Create a JSON file with 10 books and call it as books.json

Implement the below endpoints – all must have proper HTTP error codes, exception handling

**/books** – this is a GET call to retrieve all books information

**/books/:id** – this should retrieve a single book info from the book database (**id** should refer **ISBN in schema)**

**/books?year=2019 –** this should retrieve all books published in 2019

**/books?pricestart=<number>&priceend=<number> -** this should retrieve all books between the specified price

**Task 3:**

Modify your server implementation to POST new books to books.json (use fs module)

1. Use custom middleware with express JSON parser and urlencoded to add a new book from POSTMAN
2. Perform schema validation to check if the provided data is of correct format – use 3rd party schema validators like JOI or AJV etc) – two books can’t share same ISBN & title
3. Update new book data in books.json file

***Task 4:***

Modify your server implementation to do below 2 changes:

1. Update a book author to “Adam Freeman” whose ISBN number is 9781484249789 (use appropriate HTTP verb)
2. /books/:ISBN – this should delete the book from books.json

**Task 5:**

Implement an endpoint to download an image from static server – try your own image and PDFs in a static folder (Hint: use **express.usestatic**)

/download/img – this should download a image

/download/pdf – this should download a PDF