

Commerce-IQ-(Assignment)

Solution:

Application Language: Developed using **Java-Spring Boot**

JSON Data:

- Data stored in store.json that is initialized on the server in the following format:

```
{
  "authors": [
    {
      "firstName": "Elon",
      "lastName": "Smuck",
      "post": 2,
      "id": 1
    },
    {
      "firstName": "Nas",
      "lastName": "Medium",
      "post": 1,
      "id": 2
    }
  ],
  "posts": [
    {
      "reviews": 30,
      "author": "Elon",
      "id": 1,
      "title": "Post1",
      "views": 600
    },
    {
      "reviews": 50,
      "author": "Elon",
      "id": 2,
      "title": "Post2",
      "views": 500
    },
  ],
}
```

```
{
  "reviews": 60,
  "author": "Nas",
  "id": 3,
  "title": "Post3",
  "views": 100
}
]
```

Features Implemented:

- ❑ Every data set should have a parent identifier (entity type), which will be used in the GET APIs.
- ❑ Every data set should have an ID (Primary key)
- ❑ ID should be immutable, error needs to be thrown if ID is tried to be mutated.
- ❑ If you make POST, PUT, PATCH or DELETE requests, changes have to be automatically saved to store.json.
- ❑ The store.json file should support multiple entity types: Code changes required to GET/PUT new entities
- ❑ Sample APIs to be supported by the mock server on store.json file:
 - GET /posts
 - GET /posts/0
 - POST /posts
 - PUT /authors/1
 - PATCH /posts/1
 - DELETE /posts/1
- ❑ Enable filtering at entity level :
 - GET /posts?title=title1&author=CIQ (single entity filter and sort functionality code changes are not implemented yet)
- ❑ Treat store.json as an empty slate where you can add and retrieve any data.

Application Deployment:

- ❑ Application hosted on AWS EC2 instance for accessing API Requests
- API Access on:** <http://3.135.234.185:8080/>

GitHub Repository:

- ❑ <https://github.com/sanjana3011/json-api/tree/master>

Dev Testing and Endpoints Supported:

❑ Postman Collection API Link:

https://github.com/sanjana3011/json-api/blob/master/postman_collection.json

❑ Postman Documentation Link:

<https://documenter.getpostman.com/view/11826258/TzCQaRU1>