

Лабораторная работа 14: Работа с архитектурой REST API

В этой работе мы поработаем с REST-подходом

Упражнение №14.1

1. Перейдите в консоли в папку **14-lab-rest/my-app1** и установите <https://www.npmjs.com/package/json-server>
2. Создайте JSON-файл **db.json** для хранения данных:

```
{
  "books": [
    {
      "id": 1,
      "title": "Book1",
      "price": 1000
    }
  ],
  "comments": [
    {
      "id": 1,
      "body": "some comment",
      "bookId": 1
    }
  ],
  "info": { "param": "value" }
}
```

3. Запустите сервер

```
json-server --watch db.json
```

4. Обратитесь к приложению по адресу <http://localhost:3000/books/1> должен показаться JSON описывающий книгу с номер 1:

```
{
  "id": 1,
  "title": "Book1",
  "price": 1000
}
```

5. Установите в браузере расширение, которое умеет работать с REST (например [Advanced REST client](#) в Google Chrome) и выполните GET, POST, PATCH и DELETE запросы к маршруту `http://localhost:3000/books` .
6. Убедитесь, что содержимое файла **db.json** меняется

Упражнение №14.2

1. В папку **my-app2** установите приложение из репозитория <https://github.com/docker/getting-started/tree/master>. *Примечание: если знакомы с Docker, то выполняйте упражнение в нём*
2. Перейдите в папку **my-app2/app** и установите зависимости
3. Запустите приложение `npm run dev`
4. Познакомьтесь в работой приложения по адресу `http://localhost:3000/` , посмотрите код приложения