

1. HTTP Overview & Developer Tools:

1.1. HTTP Methods:

- GET
- POST
- PUT
- DELETE
- PATCH - update resource partially
- HEAD - retrieve the resource's headers
- OPTIONS - returns the HTTP methods that the server supports for the specified URL

1.2. Content-type and Disposition:

- The Content-Type/Content-Disposition headers specify how the HTTP request/response body should be processed

2. REST and RESTful Services

- REST = Representational State Transfer
- Architecture for client-server communication over HTTP
- nested resources

3. Accessing the GitHub API

4. NPM and package.json:

4.1. NPM:

- A package manager for JS and node.js projects
- simplifies installing, managing, and sharing libraries and tools in web development
- bundled with Node.js, it offers a command-line interface for various tasks
- facilitates integrating third-party packages to reuse existing code and speed up development
- helps manage project dependencies, ensuring required packages are available
- allows defining and running scripts through the project's package.json file

4.2 Install packages from the package.json:

- Navigate to Project:

```
npm init -y (in order to create package.json)
```

```
npm start
```

```
npm i is (== npm install is)
```

```
-- declare -- const is = require('is');
```

```
npm install
```

5. CRUD Operations to the NodeJS REACT API