

**Objective:**

The goal of this assignment is to demonstrate your ability to integrate external APIs into a custom backend application using NestJS. You will search for a free public API, understand its documentation, and build a backend system that integrates with it.

**GIT:** <https://github.com/JapsSwain/Assignment-OpenAPI>

**GET:** <http://localhost:3000/anime/search?q=naruto>

The screenshot shows a web browser interface with a dark theme. At the top, a green status bar indicates a GET request to `http://localhost:3000/anime/search?q=naruto`. Below this, a dropdown menu shows the method `GET` and the URL. The `Params` tab is selected, showing a table of query parameters:

Key	Value
q	naruto

The `Body` tab is also visible, showing the JSON response:

```
{
  "pagination": {
    "last_visible_page": 1119,
    "has_next_page": true,
    "current_page": 1,
    "items": {
      "count": 25,
      "total": 27952,
      "per_page": 25
    }
  },
  "data": [
    {
      "mal_id": 1,
      "url": "https://myanimelist.net/anime/1/Cowboy_Bebop",
      "images": 5
    }
  ]
}
```

At the bottom left, there is a small notification that says "ing great 🍌".

GET : <http://localhost:3000/anime/search?query=Solo%20Levling>

The screenshot shows a web browser with the address bar displaying `http://localhost:3000/anime/search?query=Solo%20Levling`. The browser's developer tools are open, showing the 'Params' tab with a single query parameter: `query=Solo%20Levling`. The 'Body' tab is selected, showing a JSON response. The response is a JSON object with the following structure:

```
{  "approved": true,  "titles": [    {      "type": "Default",      "title": "Ore dake Level Up na Ken"    },    {      "type": "Synonym",      "title": "Na Honjaman Level Up"    },    {      "type": "Synonym",      "title": "나 혼자만 레벨업"    },    {      "type": "Synonym",      "title": "..."    }  ]}
```

GET : <http://localhost:3000/anime/search?query=Bleach>

The screenshot shows a web browser with the address bar displaying `http://localhost:3000/anime/search?query=Bleach`. The browser's developer tools are open, showing the 'Params' tab with a single query parameter: `query=Bleach`. The 'Body' tab is selected, showing a JSON response. The response is a JSON object with the following structure:

```
{  "approved": true,  "titles": [    {      "type": "Default",      "title": "Bleach"    },    {      "type": "Japanese",      "title": "BLEACH - ブリーチ -"    },    {      "type": "English",      "title": "Bleach"    }  ],  "title": "Bleach",  "title_english": "Bleach",  "title_japanese": "BLEACH - ブリーチ -",  "title_synonyms": []}
```

GET: <https://img.youtube.com/vi/AwaHrHMgz5Q/hqdefault.jpg>

HomeWorkspacesAPI Network

Search Postman

Invite

Upgrade

OverviewPATCH http://k...GET http://loc...GET https://cc...GET https://lm...GET https://w...GET http://loc...GET http://loc...GET https://cc...GET https://lr...

No environment

https://img.youtube.com/vi/AwaHrHMgz5Q/hqdefault.jpg

SaveShare

GEThttps://img.youtube.com/vi/AwaHrHMgz5Q/hqdefault.jpg

Send

ParamsAuthorizationHeaders (6)BodyScriptsSettings

Cookies

Query Params

Key	Value	Description
Key	Value	Description

BodyCookiesHeaders (17)Test Results

200 OK218 ms30.11 KB

HexPreviewVisualize

You're doing great

OnlineFind and replaceConsole

PostbotRunnerStart ProxyCookiesVaultTrash

10:43 PM