Part 1 : SWAPI

Base URL: https://swapi.dev/api/

1. the height of Darth Vader

[**https://swapi.dev/api/people/4**](https://swapi.dev/api/people/4)

Graphical user interface, text

Description automatically generated

1. the population of the planet Alderaan

**https://swapi.dev/api/planets/2**

A screenshot of a computer

Description automatically generated with medium confidence

1. the name of the manufacturer of the Millennium Falcon

**https://swapi.dev/api/starships/**

A screenshot of a computer

Description automatically generated with medium confidence

1. the name of the species that C-3PO belongs to (multiple URLs)

**https://swapi.dev/api/people/**

**https://swapi.dev/api/species/2/**

A screenshot of a computer

Description automatically generated with medium confidence

Text

Description automatically generated

1. the title of each film that Obi-Wan Kenobi is in (multiple URLs)

[**https://swapi.dev/api/people/10/**](https://swapi.dev/api/people/10/)

**Films**

**"https://swapi.dev/api/films/1/",**

**"https://swapi.dev/api/films/2/",**

**"https://swapi.dev/api/films/3/",**

**"https://swapi.dev/api/films/4/",**

**"https://swapi.dev/api/films/5/",**

**"https://swapi.dev/api/films/6/"**

**"title": "A New Hope",**

**"title": "The Empire Strikes Back",**

**"title": "Return of the Jedi",**

**"title": "The Phantom Menace",**

**"title": "Attack of the Clones",**

**"title": "Revenge of the Sith",**

1. use the search query (the how to on the search query is at the bottom of the Getting Started section of the documentation) to get the information about the Millennium Falcon, it’s a starship

**https://swapi.dev/api/starships/?search=Millennium Falcon**

{

    "count": 1,

    "next": **null**,

    "previous": **null**,

    "results": [

        {

            "name": "Millennium Falcon",

            "model": "YT-1300 light freighter",

            "manufacturer": "Corellian Engineering Corporation",

            "cost\_in\_credits": "100000",

            "length": "34.37",

            "max\_atmosphering\_speed": "1050",

            "crew": "4",

            "passengers": "6",

            "cargo\_capacity": "100000",

            "consumables": "2 months",

            "hyperdrive\_rating": "0.5",

            "MGLT": "75",

            "starship\_class": "Light freighter",

            "pilots": [

                "https://swapi.dev/api/people/13/",

                "https://swapi.dev/api/people/14/",

                "https://swapi.dev/api/people/25/",

                "https://swapi.dev/api/people/31/"

            ],

            "films": [

                "https://swapi.dev/api/films/1/",

                "https://swapi.dev/api/films/2/",

                "https://swapi.dev/api/films/3/"

            ],

            "created": "2014-12-10T16:59:45.094000Z",

            "edited": "2014-12-20T21:23:49.880000Z",

            "url": "https://swapi.dev/api/starships/10/"

        }

    ]

}

**Part 2: Social Mountain**

The base URL of your requests is: https://practiceapi.devmountain.com/api (make sure to have the “s” in “https”)

* Check if the POST request accept params, queries, and/or a body. Which one(s) and what information is it expecting to be sent?

Body.

{

"text" : "Social Mountain"

}

* What data type does the GET request return?

Array of all posts.

<https://practiceapi.devmountain.com/api/posts>

[

    {

        "id": 47,

        "text": "Elizabeth",

        "date": "13 Jul 2022"

    },

    {

        "id": 46,

        "text": "The text used here will appear in the new post object",

        "date": "13 Jul 2022"

    },

    {

        "id": 45,

        "text": "Social Mountain",

        "date": "13 Jul 2022"

    },

]

* What would the URL look like for deleting the post with the id 555? (This post does not exist anymore, but the syntax is the same for existing posts, )

**https://practiceapi.devmountain.com/api/posts?id=555**

* List the possible response codes from the GET request at ‘/posts/filter’

200 Returns an array of filtered posts

409 Request query is missing required text property.

* Create a post whose text is your name, record the URL and body here:

[**https://practiceapi.devmountain.com/api/posts**](https://practiceapi.devmountain.com/api/posts)

{

"text" : "Archana"

}

* What would the URL and body object be to update the post you just made to contain your faovrite color instead of your name?

<https://practiceapi.devmountain.com/api/posts?id=66>

{

"text" : "Orange"

}

* What is the URL to get posts that contain the text “blue”?

**https://practiceapi.devmountain.com/api/posts/filter?text=blue**

* Make a request to GET all the posts. What are the content type and charset of the response? (Hint: look on the Headers)

Application/json; charset=utf-8

* What would cause a PUT request to return a 409 status?

Request was missing req.query.id or req.body.text

When ID or Text Missing

* What happens if you try to send a query in the GET request URL? Why do you get that response?

Its not showing query related Particular ID or Text.

It is showing all the posts

# Part 3: Front End (Advanced