Task 1
Learn About Web APIs

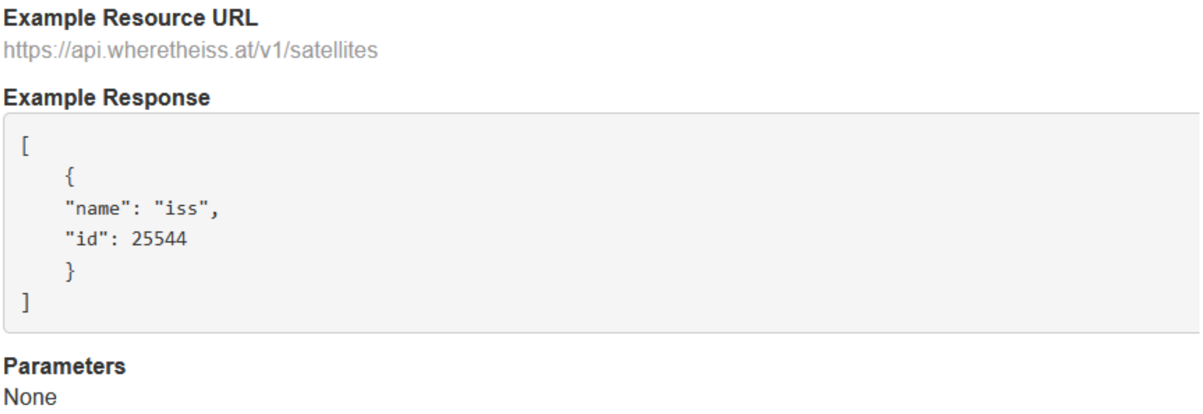
What are APIs?

APIs are just a set of rules and definitions that allow different pieces of software to communicate with each other. Think of it like the menu in a restaurant, it tells you what you can order, and how much it costs. But it doesn't make the order for you, it just tells you how to do it.

An example

This task will specifically focus on web apis. We will focus on this api for this task https://wheretheiss.at/w/developer.

Scroll down to endpoints, it lists a bunch of URLs and something called a response and some parameters.



Example Resource URL

https://api.wheretheiss.at/v1/satellites

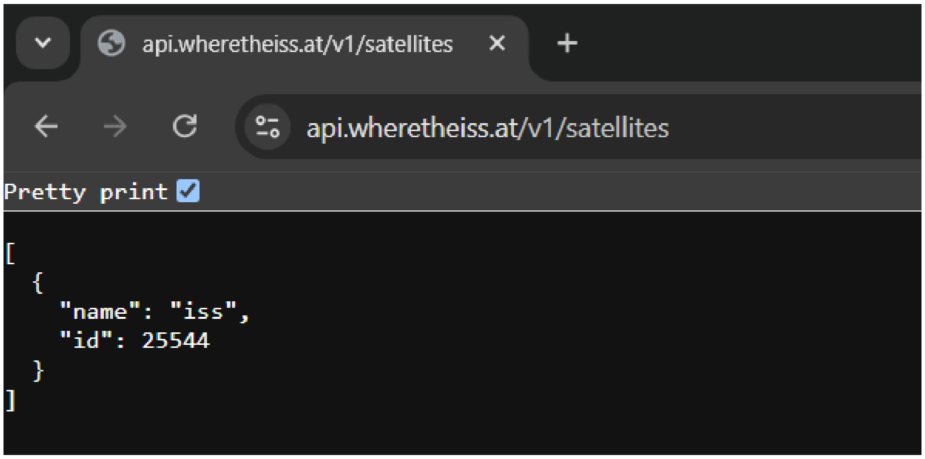
Example Response

```
[
    {
    "name": "iss",
    "id": 25544
    }
]
```

Parameters

None

So what happens if you visit this URL?
(It may look different on a different browser, this is chrome)



api.wheretheiss.at/v1/satellites
api.wheretheiss.at/v1/satellites

Pretty print ☑

```
[
  {
    "name": "iss",
    "id": 25544
  }
]
```

Well the URL is not a website, instead it's some data in a format called JSON. This data is the response.

To see what this is we read the documentation about the endpoint.

> This endpoint returns a list of satellites that this API has information about, including a common name and NORAD catalog id. Currently, there is only one, the International Space Station. But in the future, we plan to provide more.

So, we accessed the endpoint by going to that url, and it gave us a response of all the satellites it knows about. It gave us the name, and some sort of id.

Take a look at the other endpoints of the api, enter the URLs in the browser and see what you get, modify the URLs and see what happens. Think of the endpoints as the different types of info that the API will give you.

What if we need to pass some parameters to the API? We just modify the URL, look at the other endpoints for more details. (There are other ways to do this too, but for this task you need not worry about that.)

How do we use APIs?

Now we have been accessing the api through the browser, but this is highly unusual. Normally apis are accessed using programs. For this task you can use either go, python or NodeJS. We recommend you to use go. Look up "How to do HTTP get requests in X language".

For example: for python you will need the requests module.

Write programs to do the following (separate program for each):

1) Figure out where the ISS is, output the location in latitude and longitude
2) Do 1) again But output the location's country code and time zone instead of latitude and longitude.

3) Make a program that takes in a date (DD/MM/YYYY format) and time (Hours:Minutes:Seconds) and outputs the location of the ISS at that time.

4) Bonus: Making 3 programs that do very similar things might seem too easy and scattered. Combine the programs from all the three questions above into one program, and use command-line-arguments to switch between the 3 options. Use any argument format, but explain how to use it as a separate *-h* or *–help* flag. If you do this question, you need not do the other 3 separately.

We will circulate a google form for the responses. You need not attempt the bonus (but it is encouraged). The deadline is the end of Oasis (27/10/2024 EOD).

Further things to try (Not graded but will help you learn) :

- Try to misuse the api, modify the URL in ways that are not specified by the API. See what happens. When writing backend code, why do we have to think about the possibility of users misusing the api?

- If you mistyped any of the URLs or parameters you would have got some error codes in response, What are these and what do they mean and what purpose do they serve?