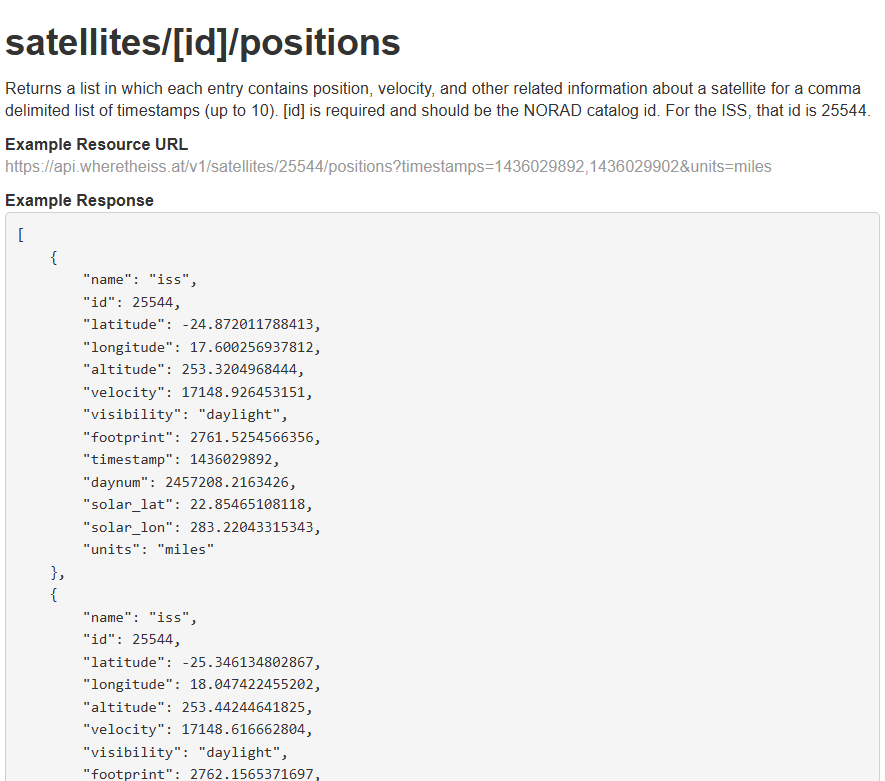
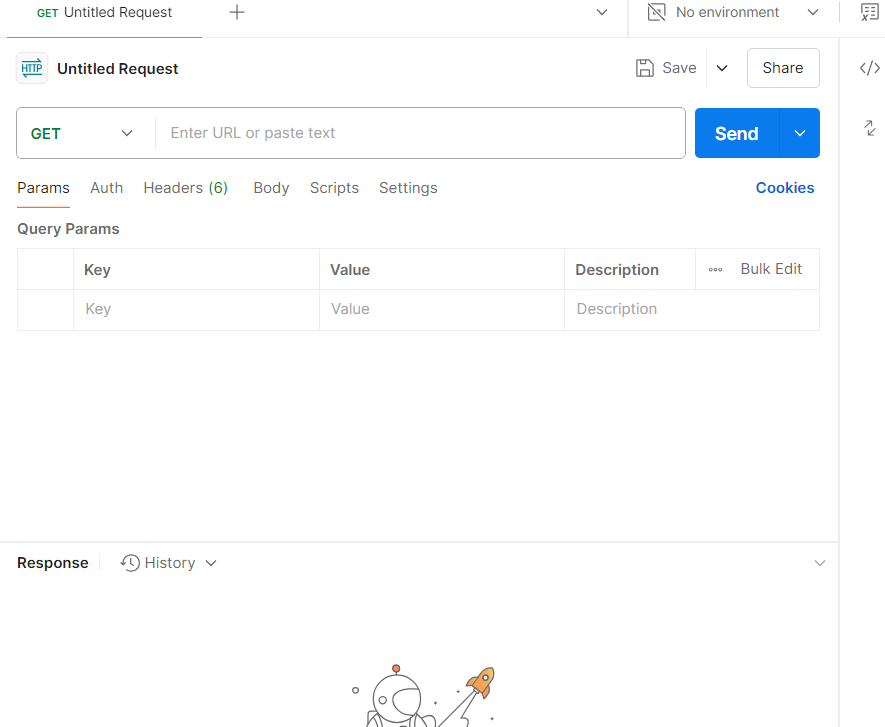
**Application Programming Interface (API):**

* It bridges the communication between the two pieces of software.
* ( <https://wheretheiss.at/w/developer> ) This website has an example of how an API works.
*  The above pasted pic is an example of what you would see when u go through the link I have given.
* For example, take the latitudes and longitudes specified under the subtopic, ‘satellite’. And write them or copy paste them in Google maps to see the exact location of the satellites.
* You need to use postman app to make GET requests (Remember to download it).
* 
* The above is the view of the postman page. Here you would have to enter the website and it would give you the important latitudes and longitudes. They can be pasted in the google maps to check the exact location of satellite.

**Formatting API Requests:**

When sending a request, we need to make sure that the server understands it perfectly. If not, we wouldn’t get what we want.

Methods of formatting API requests with HTTP:

* **GET**: Retrieve data
* **POST**: Create data
* **PUT**: Update data
* **DELETE**: Delete data

**JSON:**

* It is structured after a js object.
* We use keys in Js while in JSON we use strings. The structure and everything else is same apart from keys being used as strings here.
* JSON makes the code feel minimal. It basically just reduces the space occupied comparatively.
* If you don't understand the JSON codes you could use JSON visualizer.
* To convert JS Object to JSON:

const jsonData = JSON.stringify(data);

* To convert JSON to JS Object:

const data = JSON.parse(jsonData);

**API Authentication:**

* No Authentication:

No authentication is required to access the API. Anyone can make requests to the API.

* Basic Authentication:

Involves sending a username and password with each request. The credentials are sent in the ‘Authorization’ header as a base64-encoded string

* API Key Authorization:

An API key is a unique identifier that the client includes in the request, either as a query parameter or in the headers.

* Example:

GET /data?apiKey=your\_api\_key

Authorization: Bearer your\_api\_key

* Token Based Authentication:

More secure and flexible than API keys. Users authenticate using credentials to receive a token(e.g., JWT or OAuth token). The token is included in each request for authentication.

**REST APIs:**

* REST (Representational State Transfer) APIs allow clients to interact with resources on a server using standard HTTP methods, while ensuring statelessness, flexibility, and scalability.
* Cacheability: Responses can be cached to improve performance, especially for frequently requested data.
* API Versioning: REST APIs can be versioned (e.g., /v1/users) to ensure compatibility with older clients.
* Look into axios-http.com on how REST APIs work.