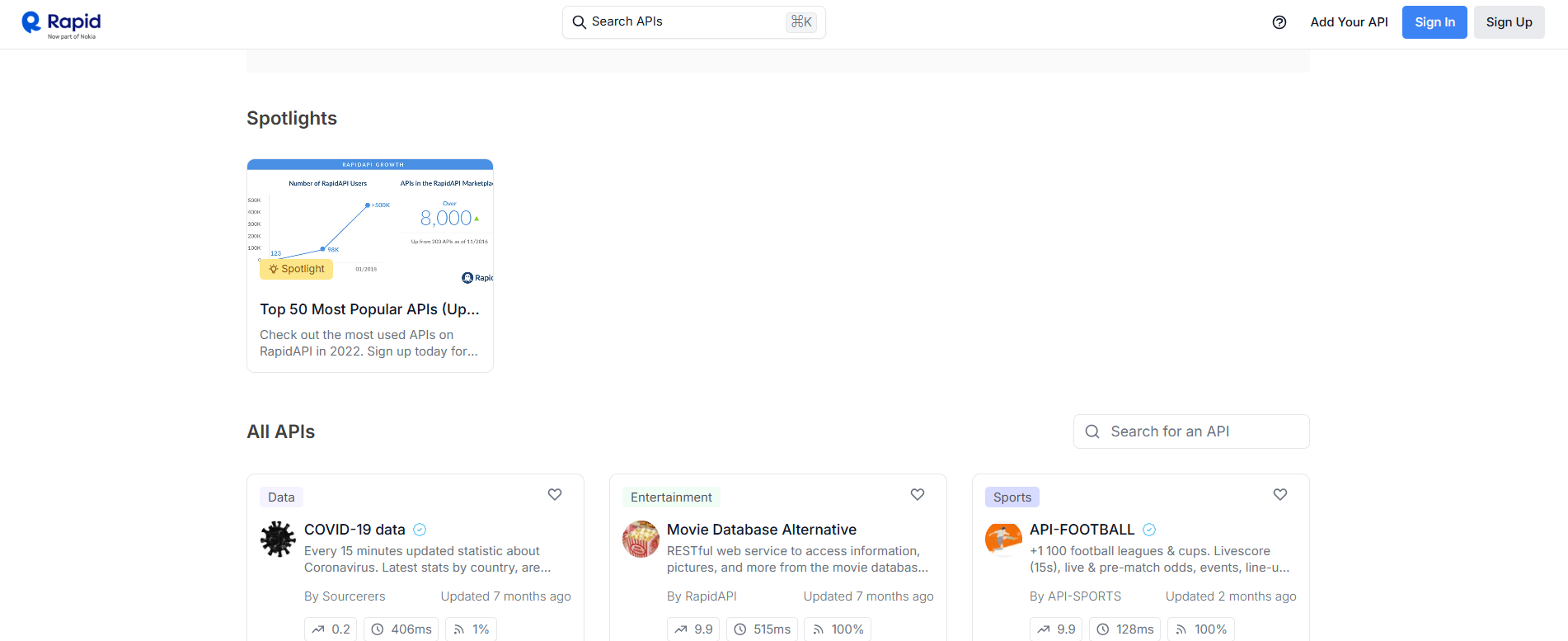
**Building our own API:**

* We use API’s because they help in development process faster.
* Rapid API has free APIs for us developers to learn and play along on it’s working. This helps us complete our work faster.



* Some APIs are private but some are public which can be used.

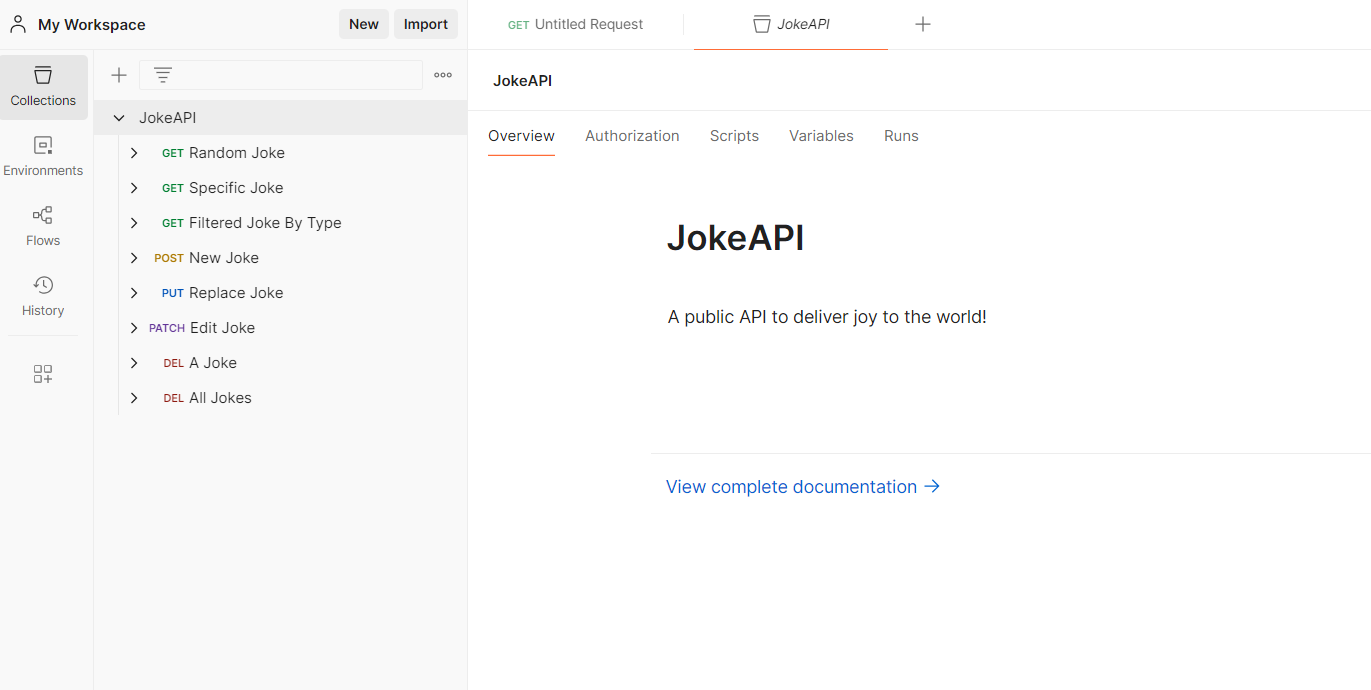
**Monetized API’s:**

* Data Collection ( recipie app, or weather data)
* Algorithm/ Service ( to find distance between 2 algorithms or ChatGPT)
* Simplified Interface (Ordering food or google maps where the codes are processed and we see it simplified)

**Creating GET Routes:**

* In the classes we are going through, to learn on how to create GET Routes, they have provided a file called JokerAPI.postman-collection.
* To open that in postman we need to go to **=>**

Workspaces< My workspace< open folder/drag and drop the folder you want to open it like in the picture given below.



* The final output is to be provided as given in the link below

<https://documenter.getpostman.com/view/6048123/2s9XxsTv8Y>

* We can publish the new GET routes created if we want to. We can go through the above link and see how it works and copy those localhosts and paste it in postman to see the files.
* We might not be able to see the specified jokes output files properly. But if we download the diy files provided, we could open them in Visualstudiocode and see output later on.

**POST, PUT and PATCH Methods:**

* Post,
* It is used to create a new resource.
* Endpoint: /resource
* Example: (javascript)

app.post('/resource', (req, res) => {

const newResource = req.body;

// Save newResource to the database

res.status(201).send(newResource);

});

* Put,
  + It is used to replace or update an existing resource entirely.
  + Endpoint: /resource/:id
  + Example: (javascript)

app.put('/resource/:id', (req, res) => {

const updatedResource = req.body;

// Replace resource with id in database

res.status(200). send(updatedResource);

});

* Patch,
* Used to update/modify a part of a resource.
* Endpoint: /resource/:id
* Example: (javascript)

app.patch('/resource/:id', (req, res) => {

const partialUpdate = req.body;

// Update specific fields in resource

res.status(200). send(partialUpdate);

});

Note: You need Express.js and Node.js for these to work.

**DELETE Method:**

* It is used to delete a resource.
* Endpoint: /resource/:id
* *Example*: (javascript)

app.delete('/resource/:id', (req, res) => {

const { id } = req.params;

res.status(200).send({ message: ‘Resource with id ${id} deleted’ }); });