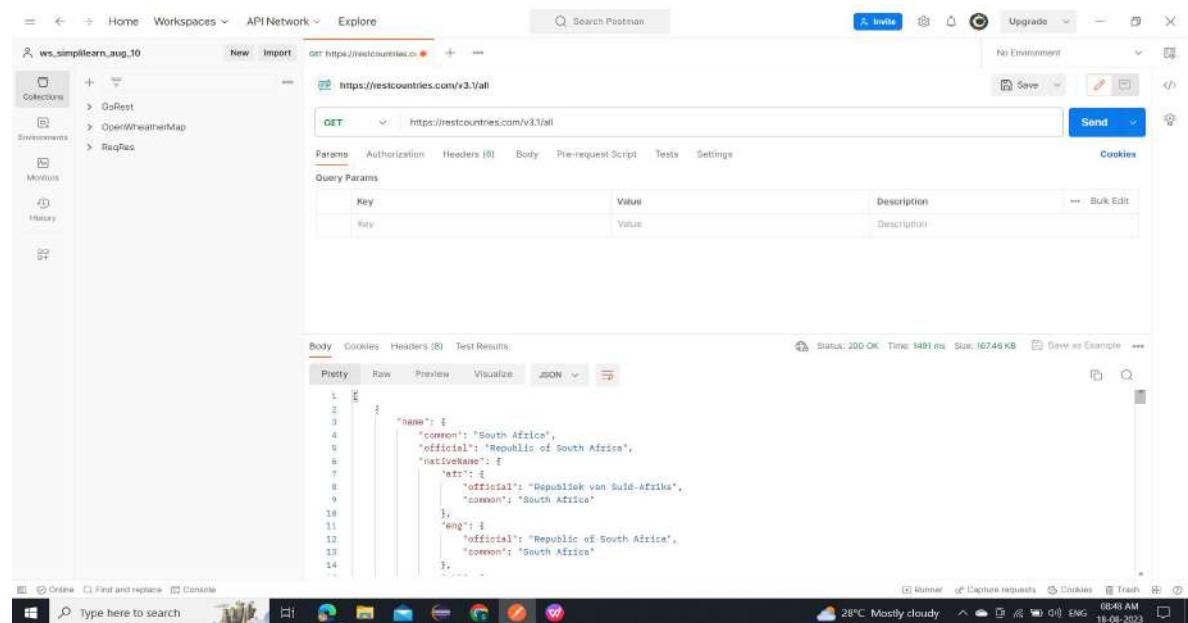


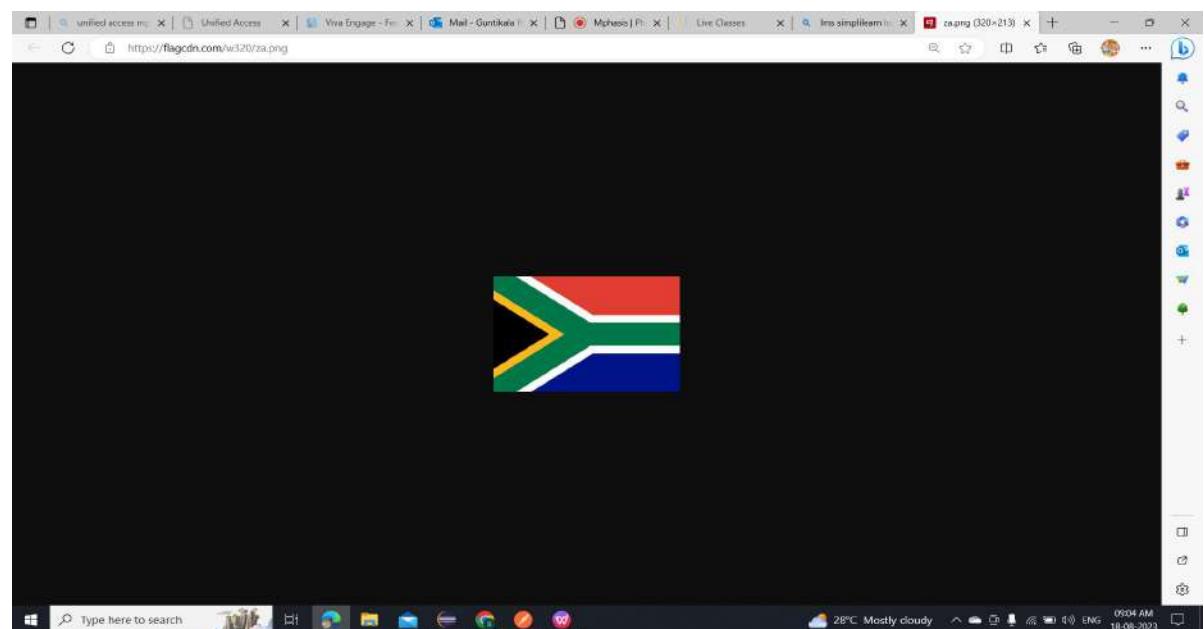
PRACTICE PROJECTS OF API TESTING

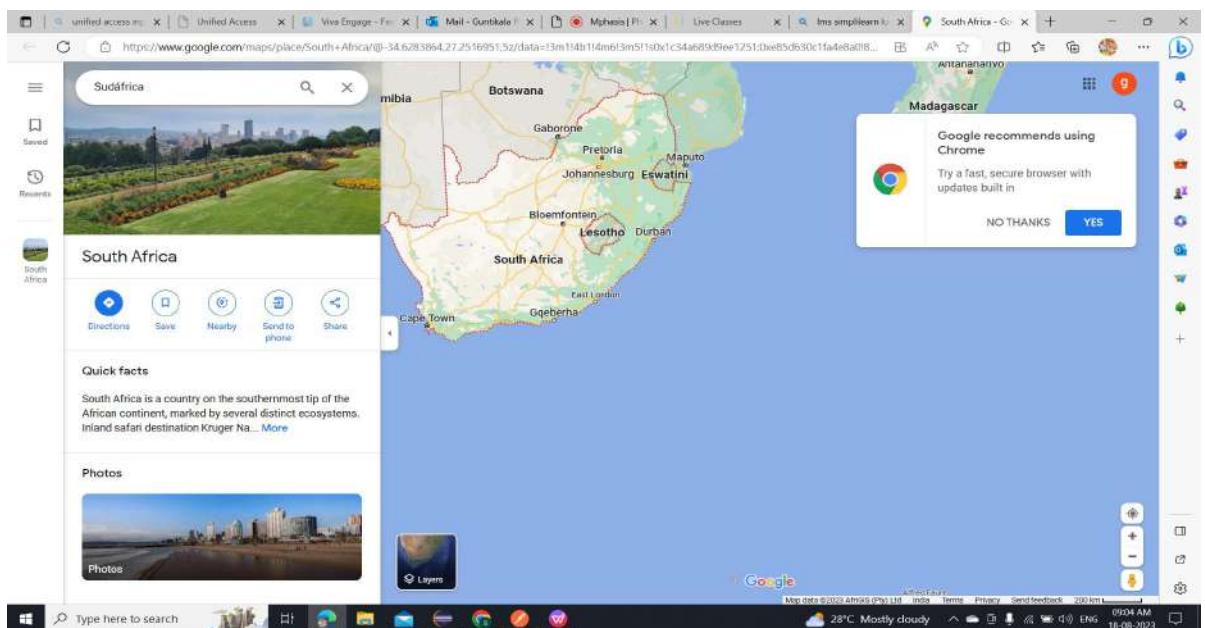
GET REQUEST:



The screenshot shows the Postman application interface. A GET request is made to <https://restcountries.com/v3.1/all>. The response status is 200 OK, time 1491 ms, size 16746 KB. The response body is displayed in JSON format, showing data for South Africa:

```
1: {
2:   "name": {
3:     "common": "South Africa",
4:     "official": "Republic of South Africa",
5:     "nativeName": {
6:       "afrikaans": {
7:         "official": "Die Republiek van Suid-Afrika",
8:         "common": "South Africa"
9:       },
10:      "eng": {
11:        "official": "Republic of South Africa",
12:        "common": "South Africa"
13:      }
14:    }
15:  }
16: }
```



A screenshot of the OpenStreetMap website showing a specific geographical feature. The main view is a map of southern Africa with a red boundary highlighting the area of South Africa. To the left, a detailed "Relation: South Africa (87565)" page is displayed, showing version information and a list of tags. The tags table includes:

ISO3166-1	ZA
ISO3166-1alpha2	ZA
ISO3166-1alpha3	ZAF
ISO3166-1numeric	710
admin_level	2
alt_name:eo	Sudafriko
alt_name:gl	África do Sur
alt_name:in	Afrika ya Südi
alt_name:wo	Afrig gu Béj-gánnhaar
boundary	administrative

At the bottom of the page, there is a note about a missing Shiselweni-Zombodze border edit.

POST REQUEST:

The screenshot shows the Postman interface with a collection named "ws_simpleteam.aug.10". A POST request is selected for the "Create User" endpoint. The URL is set to <https://orest.co.in/public/v2/users>. The request body is a JSON object:

```
[{"name": "Avinand", "email": "avinand9534@gmail.com", "gender": "male", "status": "active"}]
```

The response status is 201 Created, with a time of 100 ms and a size of 1.17 KB.

PARAMETRIZED REQUEST:

The screenshot shows the Postman interface with a collection named "ws_simpleteam.aug.10". A GET request is selected for the "Current Weather" endpoint. The URL is set to <https://api.openweathermap.org/data/2.5/weather?q=Bangalore&appid=6a02f39bcaa891ba278064ee12215cd9>. The request includes query parameters:

Key	Value	Description
q	Bangalore	
appid	6a02f39bcaa891ba278064ee12215cd9	

The response area shows a small cartoon character holding a map, with a placeholder message "Click Send to get a response".

COLLECTIONS:

The screenshot shows the Postman interface with a 'New Collection' dialog open. The left sidebar lists collections like 'GoRest' and 'OpenWeatherMap'. The main area displays a 'New Collection' template with sections for 'Overview', 'Authorization', 'Pre-request Script', 'Tests', 'Variables', and 'Runs'. A note says 'Speed up your work with collection templates'. The right side shows 'Created by You' and 'Created on 18 Aug 2023, 4:47 PM'. At the bottom, there's a link to 'View complete documentation'.

The screenshot shows the Postman interface with a 'Runner' configuration dialog open. The left sidebar is the same as the previous screenshot. The main area shows a 'Run order' section with checkboxes for 'POST Create User', 'GET Verify User', 'PUT Update User', and 'DELETE Delete User', all of which are checked. Below this is a 'Functional' tab with sections for 'Choose how to run your collection' (with options for 'Run manually', 'Schedule runs', and 'Automate runs via CLI'), 'Run configuration' (with 'Iterations' set to 1, 'Delay' set to 0 ms, and a 'Select File' button), and a checkbox for 'Persist responses for a session'. At the bottom is a large orange 'Run GoRest' button.

ENVIRONMENT:

The screenshot shows the Postman interface with the following details:

- Left Sidebar:** Shows collections like 'GoRest' and 'OpenWeatherMap', and environments.
- Top Bar:** Includes 'New', 'Import', 'Logout', and search bar.
- Request Panel:** A GET request to `https://qa.regres.in/api/users?page=2`. It has parameters: 'page' set to 2. The 'Body' tab is selected.
- Variables Panel:** Shows 'Initial value' and 'Current value' for 'url' (changed from https://qa.regres.in to https://qa.regres.in) and 'id' (changed from null to 4485254).
- Global Variables Panel:** Shows 'Initial value' and 'Current value' for 'id, response.id' (null) and 'id' (4485254).
- Bottom Status Bar:** Shows system status (32°C, Mostly cloudy), battery level (94%), and date/time (18-09-2023).

TESTS SCRIPTS:

The screenshot shows the Postman interface with the following details:

- Left Sidebar:** Shows collections like 'GoRest' and 'OpenWeatherMap', and environments.
- Top Bar:** Includes 'New', 'Import', 'Logout', and search bar.
- Request Panel:** A POST request to `https://gorest.co.in/public/v2/users`.
- Tests Tab:** Contains a detailed JavaScript test script:

```

1 var responseData = pm.response.json();
2 pm.globals.set("id", responseData.id);
3
4 pm.test("Response status code is 201", function () {
5   pm.response.to.have.status(201);
6 });
7
8 pm.test("Id is a positive integer", function () {
9   const responseData = pm.response.json();
10
11   pm.expect(responseData.id).to.be.a("number");
12   pm.expect(responseData.id).to.be.above(0);
13 });
14
15
16 pm.test("Email is in a valid format", function () {
17   const responseData = pm.response.json();
18
19   pm.expect(responseData.email).to.match(/^\w+([-]\w+)*@\w+([.-]\w+)*(\.\w{2,3})+$/);
20   pm.expect(responseData.email).to.not.be.empty();
21 });
22
23
24 pm.test("Gender is either 'male' or 'female'", function () {
25   const responseData = pm.response.json();
26
27   pm.expect(responseData.gender).to.exist.and.to.be.oneOf(['male', 'female']);
28 });

```
- Right Sidebar:** Shows snippets for common actions like 'Get an environment variable', 'Get a global variable', etc.
- Bottom Status Bar:** Shows system status (32°C, Mostly cloudy), battery level (94%), and date/time (18-09-2023).

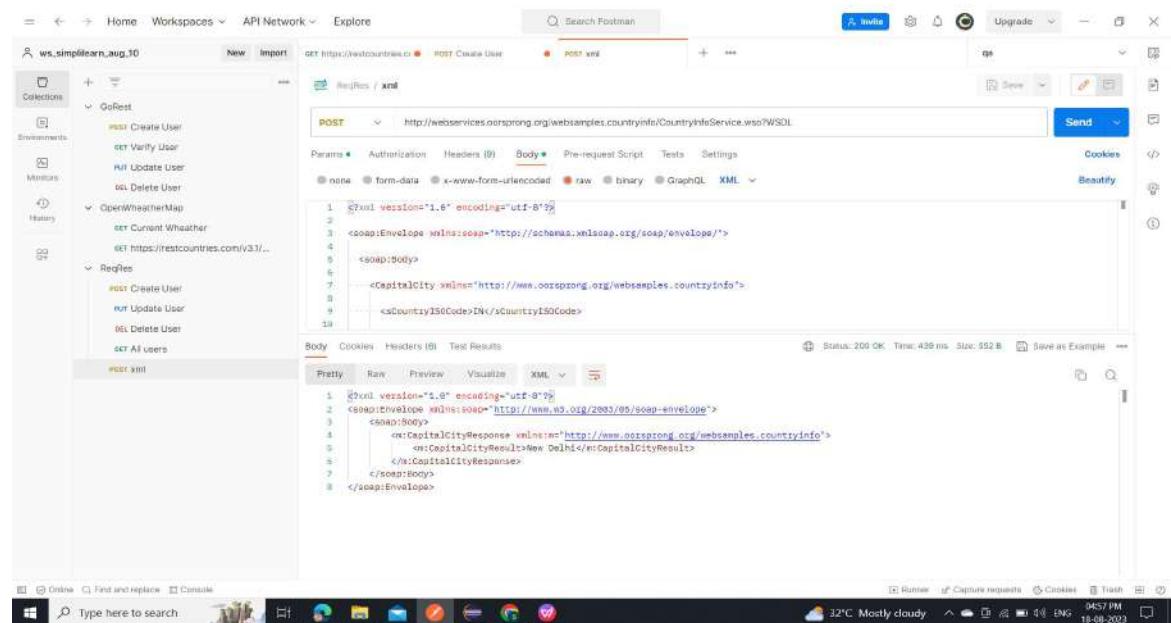
API DOCUMENTATION :

The screenshot shows the Postman application interface. On the left, there's a sidebar with 'Collections' (GoRest), 'Environments', 'Monitors', and 'History'. Under 'GoRest', there are several requests: 'POST Create User', 'GET Verify User', 'PUT Update User', 'DELETE Delete User', 'GET Current Weather', and 'GET All users'. The 'GET All users' request is selected. The main panel shows a 'GET' request to 'https://restcountries.com/v3.1/...'. The 'Params' tab has 'page' set to '2' and 'sortBy' set to 'Value'. Below the request, there's a 'Response' section with a small cartoon character and a 'Click Send to get a response' button.

BEARER TOKEN:

The screenshot shows the Postman application interface. The sidebar is identical to the previous one. The 'PUT GoRest / Update User' request is selected. The 'Authorization' tab shows 'Type: Bearer Token' with a dropdown and a 'Tokens' field containing a long token string. The 'Response' section is visible below, along with the Windows taskbar at the bottom.

SOAP API REQUEST:



The screenshot shows the Postman application interface. A collection named "ws_simplilearn_aug_10" is selected. Inside, there's a "Reqest" folder containing several requests: "POST Create User", "POST Verify User", "PUT Update User", "DELETE Delete User", "GET Current Weather", "GET https://restcountries.com/v3/..", and "POST Create User". The "POST Create User" request is currently selected and expanded. The "Body" tab is active, showing the XML code for the SOAP message:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<ns1:CapitalCity xmlns="http://www.oorsprong.org/websamples.countryinfo">
<s:CountryISOCode>IN

The "Test Results" tab shows a successful response with status 200 OK, time 439 ms, and size 552 B. The bottom of the screen shows the Windows taskbar with various pinned icons.


```