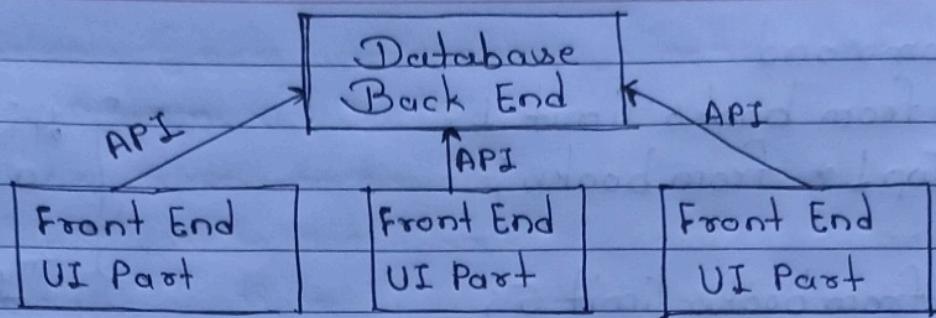


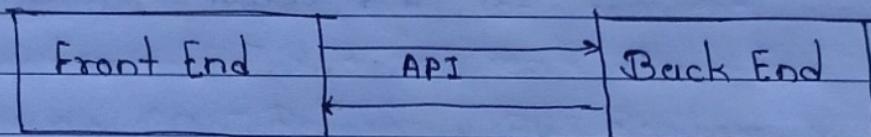
Day-1
Tuesday

* API Testing *

regres.in
Dummy website for assessment of API
'Apisetu.gov.in'
Date 18/1/22
Page



- * 'API' is a medium which connects 'Front End UI Part' & 'Database Back End'.
- * 'API' stands 'Application Program Interface'.



- * API is known as 'Middle layer' which helps to communicate betw UI and Backend.
- * App Develop करताना DataBase Create केला जातो यांतर API द्वारा केल्या जातात आणी या नंतर ते UI वरीलिंक केले जाते.
- * HTTPS - Hyper text transfer protocol.
- * JSON - Java script object Notation

Q. What is Postman?

→ It is an API Client used to develop, test, share, document API's

* CRUD Operation -

- ① Create - Post Request
- ② Read - GET request
- ③ Update - PUT request
- ④ Delete - DELETE request.

Day-2
Wednesday

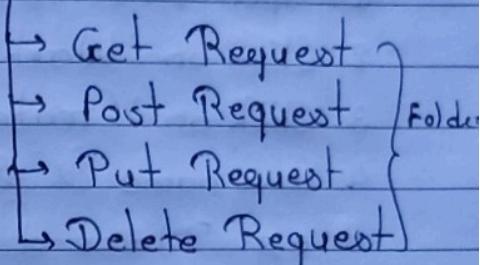
Flow of Creation

* Workspaces → Collections → December Batch → Get request

Date: 19/1/22

- * Developer provides API
- * In some organizations 'Swagger' is used.
- * URL is divided into two parts: (which is provided for testing)
 - Base URL & END Point URL.
(e.g. `reqres.in`)
- * Workspaces

Collections (December Batch)



Workspaces

Collections v December Batch

v Get Request ... Add request

GET New Request

* In GET → '200 OK' is getting for proper request hit.

Before the Question mark → Path ↓ ? After the question mark → Query Parameter
It is an [↑] API

* रखादी Base URL आपत्याला Continuously वापसायाची जिसेल तर व्याला आपां Variable ठेऊन Declare करू शकता.
या मध्ये आपत्याला तीन Choices आले. तीन locations गो Add करू शकता.
(1) Collect Level (2) Global Level (3) Environment Level.

Variable names {{ }} { } }

↑ Environments or
↑ Globals Right Side Symbol

Imp Interview
Question

Q. What is the scope of the variable?

* 'Console' in 'postman' shows the history of different API's hit by user. Also checks the response [Generalise or Overview]

* User Interface of 'Postman Tool' *

* Steps to Create Workspace, Collection & Folder & Request:

1. Open POSTMAN Tool.
2. Click on 'Workspaces'.
3. Click on 'Create Workspace'.
4. Give name & Select Visibility, Summary is optional.
e.g. Name of Workspace: 'December 2021' & Visibility: Team.
5. Click on '+' right side of 'collections'.
6. Give name to collection e.g. 'December Batch'.
7. Click on 3 Dots on that collection name & select 'Add Folder'.
8. Give name to Folder, e.g. 'GET Request'.
9. Repeat steps 7 & 8 to create additional 3 request folders
name as 'POST Request', 'PUT Request' & 'DELETE Request'.
10. Click on 3 dots on Folder name, e.g. 'GET Request' & select
'Add Request'.
11. Select the request as a 'GET' from dropdown list & copy the
'API URL' from different sources like 'regres.in', 'apisetu.gov.in'
or other live websites & paste the URL in box & click on 'Send'.
12. Check the status, time, body, code & console.
13. Save the request with name, e.g. 'List Users'.
14. Try different GET request for practice.

* Steps to Create Variable Declaration in Different Levels:-

1. Collection Level:-

- 1. To create variables click on Collection name e.g. 'DecemberBatch'
- 2. Click on 'Variables' & click on 'add a new variable' as a
'Base URL' & give an initial value as a base URL name.
e.g. 'https://regres.in/' & click on 'Save'.
- 3. Click on specific request & type the Variable Name in box
as a '{BaseURL}' & put rest of the part as it is & click on 'Send'.
- 4. Move the cursor on Variable name, it shows the scope as
'Collection'.

- * What is the meaning of API Testing?
- 1. It is a type of Software testing that involves testing 'API's' directly.
2. As a part of integration testing to determine if they meet expectations for functionality, Reliability, Performance & Security.
3. API Testing effectively validates the logic of the build architecture within a short amount of time.

2. Environment Level :-

- 1. To create variables click on Eye Symbol locate at right side & below the upgrade button.
- 2. Click on add then click on 'add a new variable' as a 'Base URL' & give an initial value as a base URL name.
e.g. 'https://regress.in/' & Click on save.
- 3. Give the name to Environment, e.g. 'DEV' or 'QA'.
- 4. Click on specific request & type the variable name in box as a '{{BaseURL}}' & put rest of the part as it is & click on send.
- 5. Select the 'Environment' from dropdown list locate near Eye symbol.
- 6. Move the cursor on Variable name, It shows the scope as 'Environment'.
- 7. If you select the 'No Environment' from dropdown list locate near Eye Symbol then moving the cursor on Variable Name, it shows the scope as 'Collection'.

3. Global Level :-

- 1. To create variables click on Eye Symbol locate at right side & below the upgrade button.
- 2. Click on add then Click on 'add a new variable' as a 'URL1' & give an initial value as a base URL name.
e.g. 'https://regress.in/' & click on save.
- 3. For working of Global Variable you need to give the unique name, i.e. different from the collect & Environment variable name.
- 4. Click on specific request & type the variable name in box as a '{{URL1}}' and put rest of the part as it is & click on send.
- 5. Move the cursor on Variable name, it shows the scope as 'Global'.

Day - 3
Thursday

To create Environment you are having 3 Options

[variable gave नियंत्रित तर तो
तुम्हाला Unresolved Variable classmate
महान दाखवेल.]

Date 20/1/22
Page

Q. What is the scope of variable.

- 1st Scope to Environment
- 2nd Scope to Collection
- 3rd Scope to Global.

* Collect ~~the~~ Runner :-

एक Collections मध्ये जेवढ्या Requests आदेते त्या संगण किंवा तजाविक Request Run करायल्या आदेते ते आपण फूटून कुरली.

& Delete

- * Get type ची Request असेल तर Body ची गरज नसते.
- * Post, type → Body ची गरज असेल.
& Put

(Q.) * Post And Put Difference :-

Created ← Create the Data → Updated Status 200 OK

status 201 * Post & Put :-

Give the data in Body → Select raw & JSON format.
copy from reqres.in

* Delete → Data Deleted → Status : 204 No Content

In Variable

* Persist :- If value given in current value then it takes in initial value.

* Reset :-

* Snippets :-

→ In tests → Snippets

→ Post request Script → Snippets.

* In post, we create the data, In put we updated the data which is previously created by post.

* Console is used to Debugging, History of API hitting.

Snippets मध्यून Variable Create करायेअसील तर
 e.g. → set a collection variable. Variable Key ^{Variable} _{ASSMATE value.}
 → pm.collectionvariables.set("POSTMAN", "API Testing Tool");
 → Save → Send → Check this is variable. Page

* After Run Collection we get the following response.

Request Name	Request Body	Response Body
1) GET	Unavailable	Available
2) POST	Available	Available
3) PUT	Available	Available
4) DELETE	UnAvailable	Unavailable

Day-14
Friday

21/01/2022

* Pre-request Script :-

आपल्याला Variable ~~create~~ मंजुरीली Create करावे लागत दिने रे Directly variable ~~create~~ Get, Delete, put किंवा Post करा शकतो.

* Tests :-

Tests मध्ये आपला Validation लावू शकता. e.g. आपला (status, string or other) गोष्टी दिवून त्याचा Result बघू शकता तो आपल्याला Test Results मध्ये बदायवा मिळेल.

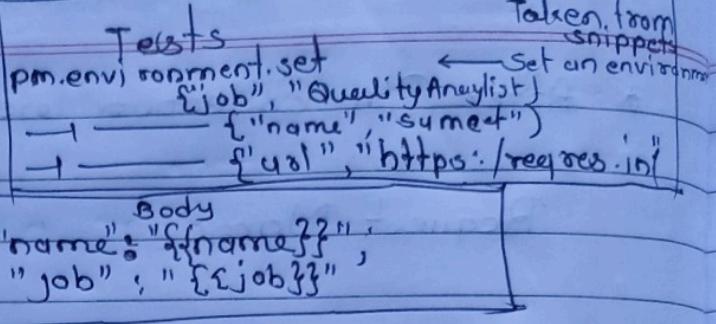
* Snippets आपला 'Pre-request Script' आणि 'Tests' यांची Use करू शकता.

* जी आपल्याला Response मिळालाय तो आपल्याला Check करावा लागते e.g. Status, Time, Userid or other things मिळतात तर? ते By using आपल 'Tests' ने करा शकतो जे आपला Snippets मध्यून घेता.

* Snippets आपला Edit दी करा शकतो किंवा As it is use करा शकता.

* Pre-request Script is perform before the API hit & post-request script (Tests) is perform after API hitting.

- Request chaining.
- then go for → list of user
- First go for → Post Create user



* Request Chaining :-

It is a linking between the two requests e.g. Get & POST. If we individually hit the API's it's not working, it shows error. But we use Run Collection it works, ~~but~~ It shows response

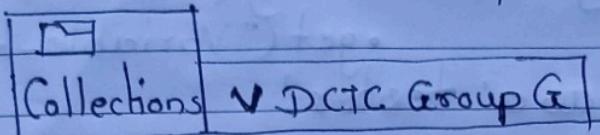
- * Snippets का Use करना आपा Pre-request script में different levels (Collection, Environment, Global) पर Variables create कर सकते हैं तो Delete ही कर सकते, क्योंकि Snippets में Variable fetch (Get) कर सकते.
- * Snippets are nothing but the 'script templates'.

- * Variables आपा आलील Levels पर Create कर सकते एवं Create के बावजूद तीव्र रूप से बद्ध नहीं होते.
 1. Collection Level
 2. Global Level
 3. Environment Level.

- * Pre-request Script एवं Post-Request Scripts (Tests) आपा आलील लेवलों पर Get, Set तो Clear कर सकते.

1. Collection level ← हावाह (लेवल) Run कराव लागत.
2. Folder Level ← → ←
3. Environment level ← → ←

* How to create 'Pre-request Script' at Collection Level.



1. Click on collection name e.g., DCTC Group G.
2. Click on Pre-request script.
3. Select respective snippet from Snippets.
4. e.g. I am taking 'Set a Collection variable' from snippets list.
 - Click on 'Set a Collection variable'.
 - It displays `pm.collectionVariables.get("variable-key");`
 - Give the name instead of "Variable.Key" as
 - It displays `pm.collectionVariables.set("variable-key", "variable-value");`
 - Give the name to 'variable-key' & 'variable-value' as 'Tester', 'Automation'.
 - Just save it & Run it at collection level.
 - It added in variable list.
- If you want to clear the collection variable.
 - Click on 'clear a collection variable'.
 - It displays `pm.collectionVariables.unset("variable-key");`
 - Give the name to 'variable-key' as a 'tester'.
 - Just save it & Run it.
 - It removed from variable list.
- If you want to get (fetch/reoid) the collection variable.
 - Click on 'Get a collection variable'.
 - It displays `pm.collectionVariables.get("variable-key");`

* Case-sensitive as well as no changes accepted
in terms giving the variable key name at level ^{variable}
collection & if use in get you puted as ^{uppercase}
or space in between it shows error like ^{Page} Undefined!

- modify this by using `console.log(pm.collectionVariables.get("Variable-key"))`
- Give the ^{name to} 'Variable-key' as a 'tester'.
- Just save it & run it.
- It displays response in console.

* Similar process is for global & environment level variables in Pre-request Scripts & also for 'Tests'

* If we execute 'pre-request script' we get the response in 'console' as.

"Automation" ← Variable value
► GET <https://regress.in/api/users?page=2> ← url of API

Sequence is, It displays first variable name & then url of API.

* If we execute 'tests' we get the response in 'console' as

► GET <https://regress.in/api/users?page=2> ← url of API
"QA" ← VariableValue

Sequence is, It displays first url of API & then name of Variable.

* If we execute 'pre-request script' & 'tests' at a time then we get the response in 'console' as.

"Automation"
► GET <https://regress.in/api/users?page=2> ← Url of API
"QA"

Variable
Values