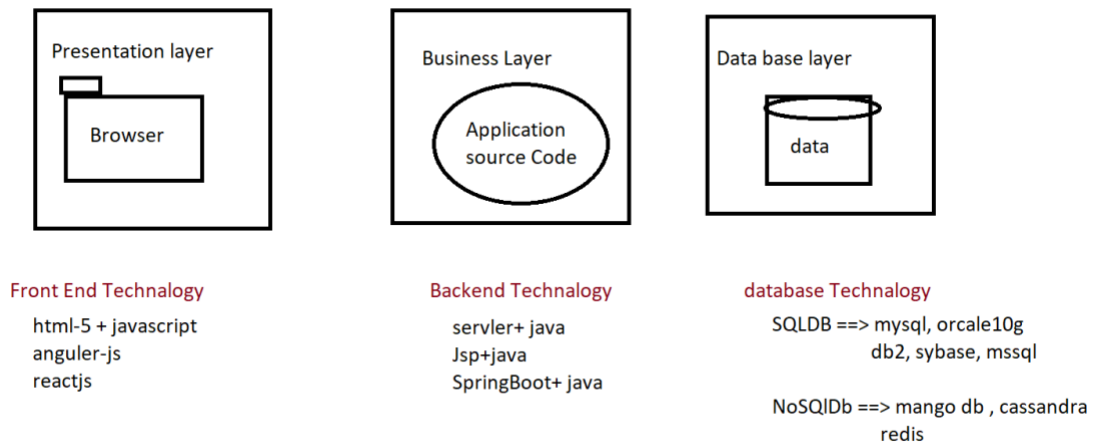


## 1. What technology your application was developed?



## 2. Difference between Unit testing & API testing

Unit Testing	API testing(web service)
Done by development team	Done by testing team or SDET team
Should have knowledge on source code	Should have knowledge on expectations and agreements of API (Api-Functional Spec)
Test each and every source code of the application	Test the data flow between 2 application
It is a white box testing	It is a grey box testing
Tools: eclipse+testNG	Tools: SOAP UI, POSTMAN, RESTCLIENT, RESTASSURED

## 3. Difference between JSON & XML language

JSON	XML
JSON object has a type	XML data is type less
JSON types: string, number, array, Boolean, null, Object, arrayOfObject	All XML data should be string
Data is readily accessible as JSON objects	XML data needs to be parsed.
JSON is supported by most browsers.	Cross-browser XML parsing can be tricky
Retrieving value is easy	Retrieving value is difficult
A fully automated way of DE serializing/serializing JavaScript.	Developers have to write JavaScript code to serialize/de-serialize from XML
Native support for object.	The object has to be express by conventions - mostly missed use of attributes and elements.
It supports only UTF-8 encoding.	It supports various encoding.
Structure is simple	Structure is complex
It doesn't support comments.	It supports comments.
JSON files are easy to read as compared to XML.	XML documents are relatively more difficult to read and interpret.

#### 4. Difference between SOAP & Rest WebService

SOAP Web Service	Rest Web Service
SOAP is a protocol	REST is Architectural style
Simple Object Access Protocol	Representational State Transfer
SOAP can't use REST Web services, it is a protocol	REST can use xml as well Web Services since it is concept
SOAP expose the services (Business Logic) via .WSDL file	REST expose the service (Business Logic) via URI
SOAP designed using too much standards	REST does not define too much standards
SOAP permits XML data format only	REST permits different data formats-
	Plain text, html, JSON, etc
SOAP requires more bandwidth and resources	REST requires less bandwidth and resource than SOAP
SOAP defines its own security	Inherits security measures from underlined transport Protocol (ouath-1.0 outh-2.0 , Bearer token)
Less preferred than REST	REST more preferred than SOAP

#### 5. Difference between Frontend & Backend language

Frontend testing	BackEnd testing
Front-end is always performed on GUI	Back-end testing is done on Business layer and should have knowledge on Data base and business logic of software
Testing becomes slower coz it have to wait for browser rendering time	It is faster and saves time.
Performance testing Is not easy in front-end	Performance testing is easy and faster
Adoc, usability and compatibility testing is possible	Adoc, usability and compatibility testing is not possible
Manual & selenium automation will be done at front end(BBT)	whiteBox , Soap API & rest API teting will be done in the backEnd(WBT. GBT)
End to end testing is possible	End to end testing is not possible
Tools: selenium, QTP, test complete etc	SOAP UI, POSTMAN, RESTCLIENT, RESTASSURED, testNG , JUNIT

#### 6. Which API testing Tool you have used?

**PostMan & RestAssured** , RestClient , testOptimize , read API

#### 7. Which type of Webservice testing you worked on

Rest WebServciec testing

#### 8. Types of API testing ?

- A. Soap WebServcie API testing
- B. REST WebServcie API testing

#### 9. What is API ?

Application programming Interface, testing interface between two application, or  
Testing the application in business layer without browser is called API testing

EG : Rest webservice is API testing

Soap Webservice is API testing

#### 10. What are the advantages of the webservice API testing?

- a. API testing very Faster (because no to wait for Browser rendering time)
- b. testing the functionality with out GUI(Browser)
- c. testing the functionality early stages (We can start API testing in Sprint-1)
- d. find defect in early stages
- e. Time effective & fast to release
- f. whenever API provider develop an API , every api should be tested the Functionality, performance security, reliability before exposing those API to consumer

#### 6. what are the disadvantages API testing?

- a. can't perform negative testing
- b. End to end workflow testing is difficult
- c. testing should have knowledge on Programming & data base (Backend)

#### 11. What is Unit testing ?

Testing the source of the application using another program is called unit API testing

#### 12. Why JSON is very Popular?

- a. Java Script Object Notation
- b. It's a Programming language to exchange information between 2 application
- c. The JSON format is syntactically identical to the code for creating JavaScript objects.
- d. Because of this similarity, a JavaScript program can easily convert JSON data into native JavaScript objects.
- e. JSON is a lightweight format for storing and transporting data
- f. JSON is often used when data is sent from a server to a web page
- g. JSON is "self-describing" and easy to understand
- h. All Browser & Mobile UI can easily consume json language
- i. NO SQL databases can directly store the data in the form of JSON (MongoDb, Casendra )

j. Platform independent

13. What is Soap webservice testing

- When two application exchange information via **soap** protocol, which is based on xml based , testing those request and response is called Soap webservice testing
- SOAP Webserive exposed via .WSDL file
- EG : SOAP API

<http://www.dneonline.com/calculator.asmx?WSDL>

<http://map.google/api/soap/service.wsdl>

14. What is Rest Webservice testing ?

- When two application exchange information via **http** protocol , which is based json/xml/text/html/js , testing those request and response is called Rest webservice testing
- Rest API Webserice are exposed via URI
- EG : Rest API

<http://map.google/api/getlocation>

<http://map.google/api/gettraffice?src='value'&dest=value>

<http://map.google/api/gettraffice>

15. How many years of Experience in API project?

2 years

16. Where did you used API in your Previous project & Purpose of the API?

Product , contacts , Quote , Case

17. How many you have tested

20 RestFull API

18. Can you Write or list Out few API in your Previous Project

Product

Get <Http://autodesk-test-env.com:8084/products>

Post <Http://autodesk-test-env.com:8084/products/addproduct>

Put patch ,delete <Http://autodesk-test-env.com:8084/product/{productID}>

Get <Http://autodesk-test-env.com:8084/products?name=hplabtaop>

19. Write on POST API request in your previous Project

EG:

Post <Http://autodesk-test-env.com:8084/products/addproduct>

```
{
  "ProductNAme" : "name"
  "SerailNum" : 123
  "Manfauting date" : ""
}
```

```
"unitPrice" : 2000INR
"Qty" : 10

}
```

## 20. Write one API response in your previous Project

```
{
  "name": "name",
  "msg": "successfully added",
  "id": "625",
  "createdAt": "2020-11-02T08:05:40.255Z"
}
```

EG : get all Products

Get [Http://autodesk-test-env.com:8084/products](http://autodesk-test-env.com:8084/products)

```
{
  "page": 2,
  "per_page": 6,
  "total": 12,
  "total_pages": 2,
  "data": [
    {
      "id": 7,
      "product": "michael.lawson@reqres.in",
      "first_name": "Michael",
      "last_name": "Lawson",
      "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/follettkyle/128.jpg"
    },
    {
      "id": 8,
      "email": "lindsay.ferguson@reqres.in",
      "first_name": "Lindsay",
      "last_name": "Ferguson",
      "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/araa3185/128.jpg"
    },
    {
      "id": 9,
```

```

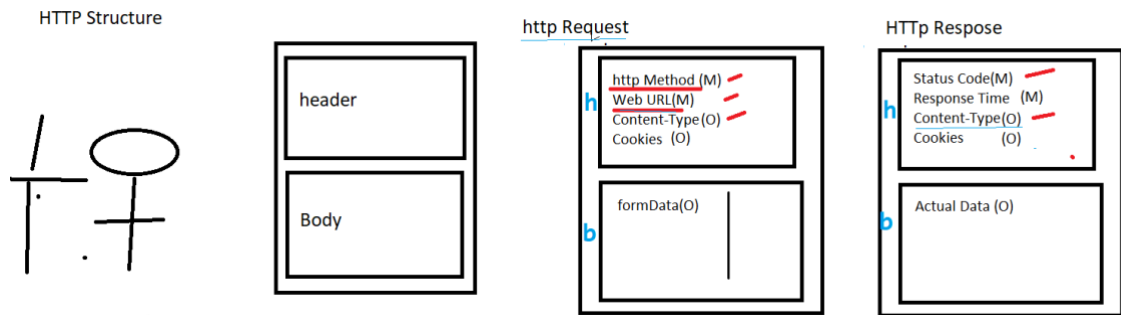
    "email": "tobias.funke@reqres.in",
    "first_name": "Tobias",
    "last_name": "Funke",
    "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/vivekprvr/128.jpg"
  },
  {
    "id": 10,
    "email": "byron.fields@reqres.in",
    "first_name": "Byron",
    "last_name": "Fields",
    "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/russoedu/128.jpg"
  },
  {
    "id": 11,
    "email": "george.edwards@reqres.in",
    "first_name": "George",
    "last_name": "Edwards",
    "avatar": "https://s3.amazonaws.com/uifaces/faces/twitter/mrmoiree/128.jpg"
  },
  {
    "id": 12,
    "email": "rachel.howell@reqres.in",
    "first_name": "Rachel",
    "last_name": "Howell",
    "avatar":
"https://s3.amazonaws.com/uifaces/faces/twitter/hebertialmeida/128.jpg"
  }
],
"ad": {
  "company": "StatusCode Weekly",
  "url": "http://statuscode.org/",
  "text": "A weekly newsletter focusing on software development, infrastructure, the
server, performance, and the stack end of things."
}
}

```

## 21. How many API test case you written

20 API → 100 tc

## 22. Explain httpRequest / httpResponse structure



http method : its a mandatory information present in the header of the request which is used specify the type of the request client is sending to server

1. Get (used to read information/resource from the Server)
  2. POST (used to create resource inside the Server)
  3. PUT (used to completely update the resource inside the server)
  4. Patch (used to partial update the resource inside the server)
  5. Delete (used to delete the existing resource available inside the server)
  6. HEAD
  7. Connect
  8. Option
  9. Trace
- C  
R  
U  
D

Status Code

its a mandatory information present header of the response, its will tell you the status of the Request

- 100 : Continue
- 200 : Successfull
- 300 : Redirect
- 400 : Client Side Error
- 500 : Server Side Error

Web URL : its is the header of the Request, used to uniquely identify specific WebResource inside the Server

Syntax :

http://<domainName>:portNum/ResourcePath?QueryParameter

http: // Ty-Deepak-Tr:8080/StudentApp/Login.html?username=admin& password=manager

Base URI

End Point

https://www.google.com/Search?q=jobs&key=hp&key=value

Content-Type:

Its is present in the header of the Response .its Specify the Type of the Content available in body of the Response

text/html : html                      MIME TYPE  
application/xls : eXcel  
application/pdf : PDF  
application/xml : XML  
application/json : Json  
image/png : image

Form Data :

Data collected from client to server is called form data, form data will be always present HTML Request Body  
Data collected using htmlForm is called formData

- Size of the form will vary form to form
- Get -> will not have form data
- Post-> will have form data

Actual Data

1. Its an optional information present in the response body
2. Actual data responded by server, if in case request is successful. other wise server provide error message in the body of the request

HttpRequest :

httpMtd : specify the type of the request

Web URL : uniquely identify the WebResource inside the Web Application

ContentType : specify the type of content available in body the Request/Response

formData : data collected in html Form / or request body

HttpResponse

Status Code : specify the status of the Request

ContentType

Actual Data : data available in response body

### 23. Explain the http method which you used In your previous project

Get => read resource from the sever  
Post => create resource inside the sever  
Put=> create resource/ complete update resource inside the sever  
Patch => partial update resource inside the sever  
Delete => delete resource inside the sever

### 24. Explain the status code which you encounter in your previous project

#### 100 Continue

#### -----SucessFull-----

**200 (OK)** : able to read resource from the server  
201 (Created) : able to create **resource inside the server**  
202 (Accepted) : able to accept the permission request inside the server  
204 (No Content) : got response from the server but no content

#### -----Redirected-----

300 Redirected : indicates that further action needs to be taken by the user agent in order to fulfill the request. http method is not supported

#### -----Client Side Error-----

400 (Bad Request) : The request could not be understood by the server due to malformed syntax  
401 (Unauthorized) : request is not authorized to access the resource inside the server  
403 (Forbidden) : server understood the request but not authorized person to access  
404 (Not Found) : not able to find the resource inside the server which you requested  
405 Method Not Allowed

#### -----Server Side Error-----

500 Internal Server Error : problem from server side  
502 Bad Gateway : proxy / firewall (network) issue in server side  
503 Service Unavailable : The server is currently unable to handle the request  
504 Gateway Timeout : server not response even waiting for long time

### 25. What is Crud operation

Create read update delete



## 26. What Authorization & Authentication



### Authorization

What you can do



### Authentication

Who you are

Authentication to check whether you are authenticated client or not ?

Authorization to check what resource your accesses

## 27. Challenges you faced in API Testing

- a. End to End Scenario testing is challenging because we have to do API Chaining

EG: Scenario → Search Product + ADD to cart + Billing + Logistic

API : API -1 for Search Product

API-2 for ADD to cart

API-3 for Billing

API-4 for Logistic

- b. API Document is not clear, API testing challenging
- c. Negative testing is challenging because in api document will not have complete requirement
- d. Validation of Complex response is challenging
- e. Deriving api test scenario is challenging ,

## 28. Prerequisite for APi testing

- Knowledge of client server architecture of the application
- Knowledge of HTTP request and HTTP response properties;
- Knowledge of JSON or XML language
- API documentation / API Functional spec (Swagger Document) from Development team
- Get the list of URI (BASEURL + ENDPOINT → URI)
- Knowledge of CRUD
- Write test case for every API that include CRUD operation
- Required tool for API TESTING
  - Postman → GUI based API testing tool
  - RestAssured → Headless Java libraries for REST API testing
- Collect API authentication if required (like Basic Auth , Bearer Auth , Oauth-1.0, Oauth-2.0)

### 29. How to verify the complex response in API testing?

Will write jsonPath to navigate to specific data inside the response , then will go for "JSoN Value checker "Assertions in postMan

#### Complex response

```
{
  "name" : "ram"
  "wife" : {
    "name" : "rani"
    "father" : {
      "name" = "king"
    }
    "mobile" : [ 9999, 8888]
  }
}
```

#### JSON-XPath / JsonPath

```
jsondata.nama
jsondata.wife.name
jsondata.wife.father.name
jsondata.wife.mobile[1]
```

### 30. How many ways we can post a request in POSTMAN

1. Post a request body by JSONOBJECT
2. Post a request body by Key:value pair (hashMAp)
3. Post a request body by JSON File

### 31. How to post Complex request in postman

We can post json File Itself in postman → body→binary→ select File

**Note:** file extension should be file.json , if content Type in JSON

### 32. Explain the validation available in POSTMAN APi testing

1. ResponseBody : Verify status Code : used to verify the response status Code
2. ResponseBody : time is less than --- : used to verify the Response time
3. ResponseBody : json value Check : used to navigate to particular key
4. ResponseBody : contain string : used to check the particular String is available in entire response body
5. ResponseBody : equal to String
6. **sendRequest** : we can send another request with in a request

### 33. What is Parameter & types of Parameter?

To achieve data driven testing we go for Parameter

1. Path parameter: it's the Part the ENDPoint, execute same request with different data we go for Path Parameter

GET ▼ https://api.github.com/repos/ownerName/param Send ▼

Params ● Authorization Headers (7) Body Pre-request Script Tests ● Settings

KEY	VALUE	DESCRIPTION
Key	Value	Description

Path Variables

KEY	VALUE	DESCRIPTION
ownerName	qspidersseleniumoar	
param	SDET_Project	Description

2. Query parameter : it's part of the Query String, execute same request with diff filter /search criteria

GET ▼ https://reqres.in/api/users?page=1 Send ▼

Params ● Authorization Headers (7) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> page	1	
Key	Value	Description

3. form Parameter : send complete request body in the form of Parameter

GET ▼ https://reqres.in/api/users Send ▼

Params ● Authorization Headers (9) **Body** ● Pre-request Script Tests Settings

☐ none ☒ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

KEY	VALUE	DESCRIPTION
<input checked="" type="checkbox"/> author	deepak	
<input checked="" type="checkbox"/> title	selenium	
<input checked="" type="checkbox"/> id	123	
Key	Value	Description

### 34. What Authentication & Authorization

Authentication: used to check whether you are valid user or not?

Authorization : used to check your permission / accesses to the resource

### 35. Types of API Authentication & which Authentication you have used in your previous project

1. Basic Auth (send a request using username/ password)
2. Bearer Token (send a request using tokenID, but token is fixed)
3. OAuth-1.0(older)
4. OAuth-2.0 (send a request using tokenID , but token ID is dynamic created via Outh-cleintAPP) or (Gmail app , allow grant permission to skillRaray app without sharing your username/password via Oauth-2 Protocol )

### 36. What is Request chaining?

Capturing data from one response using `jsonXPath()` , & pass same data in to another request is called Request chaining .

In below example, capture “id” value from the first api request & pass same data in another URL

EG :

In below example, capture “id” value from the first api request & pass same data in another URL within the same api execution

GET ▼ https://reqres.in/api/users?page=2 Send

Params ● Authorization Headers (7) Body Pre-request Script **Tests** ● Settings

```
1 var data;
2 pm.test("Your test name", function () {
3     var jsonData = pm.response.json();
4     data = jsonData.data[0].id
5     console.log ("=====>" + data)
6 });
7
8 pm.sendRequest("https://reqres.in/api/users/"+data, function (err, response) {
9     console.log(response.json());
10 });
```

Test scripts are written in JavaScript, run after the response is received. [Learn more about tests scripts](#)

SNIPPETS

- Clear an environment variable
- Clear a global variable
- Send a request
- Status code: Code is 200

EG : 2 → Send data from one request to Another request

RMGYantra / createUser Save ...

POST ▼ http://localhost:8085/employees ... Send

Params Authorization Headers (10) Body ● Pre-request Script **Tests** ● Settings

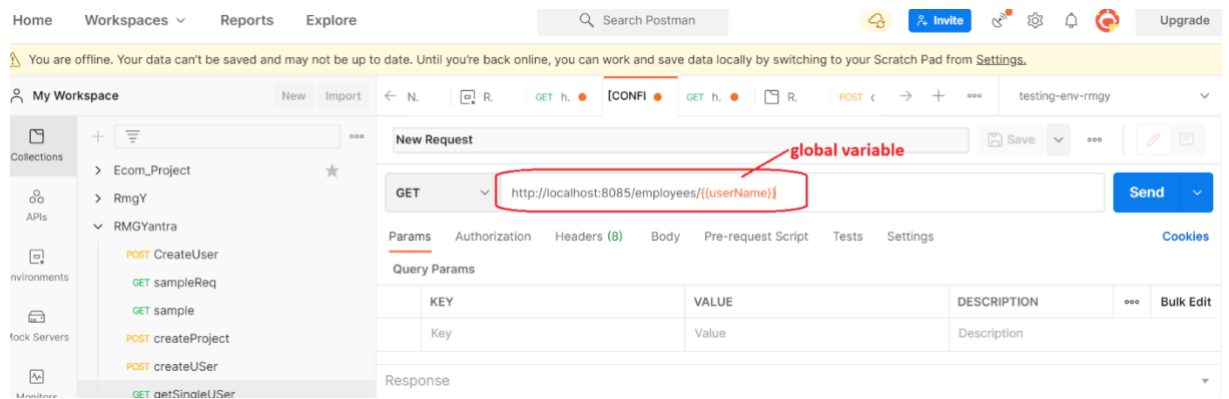
```
1 var empID;
2 pm.test("Your test name", function () {
3     var jsonData = pm.response.json();
4     empID = jsonData.employeeId
5 });
6
7
8 pm.globals.set("userName", empID);
```

Test scripts are written in JavaScript, are run after the response is received. [Learn more about tests scripts](#)

SNIPPETS

- Get a collection variable
- Set an environment variable
- Set a global variable

Send “userName” data to another request



### 37. What Is Collection in PostMan

Collection of API requests is called collection or suite

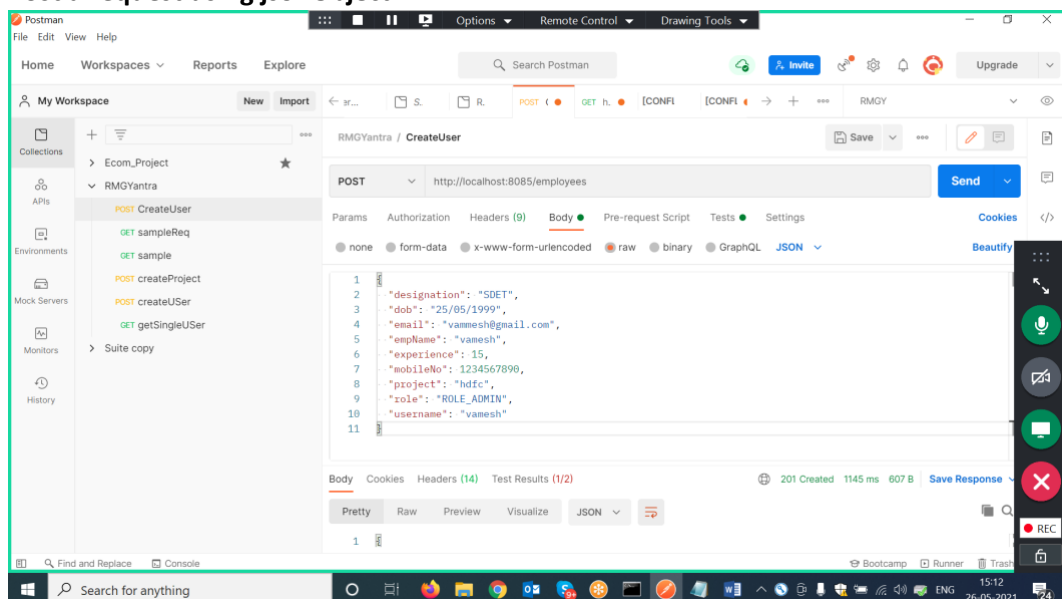
### 38. How to execute API collection in command line

1. Export Collection from POSTMAN
2. Download Collection in .json format
3. Go to Commandline
4. Install newMAN  
cmd> npm install -g new man
5. Execute collection in CMD  
Cmd> newman run path

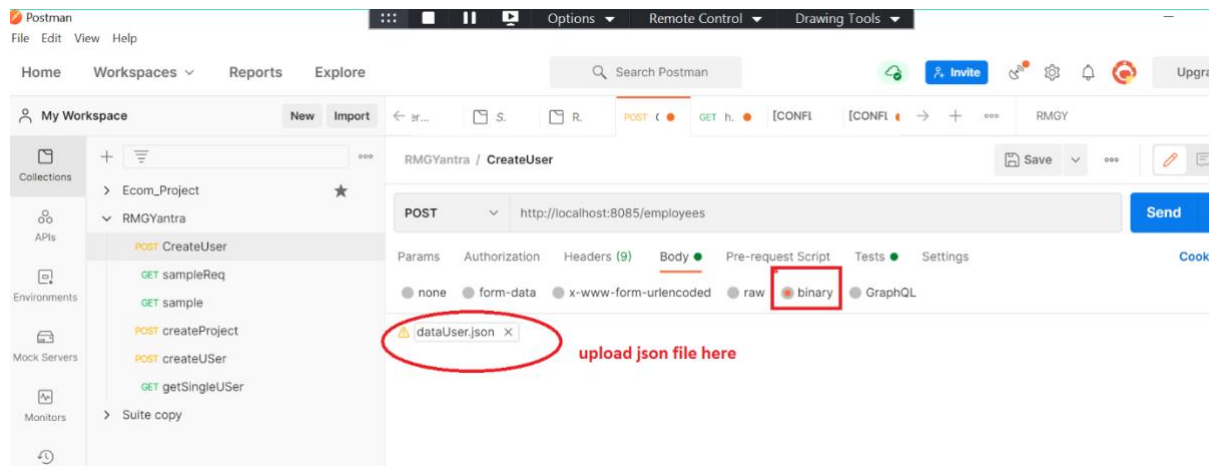
### 39. How many ways we can post request in JSon in PostMan

In postman we can post request in 3 ways

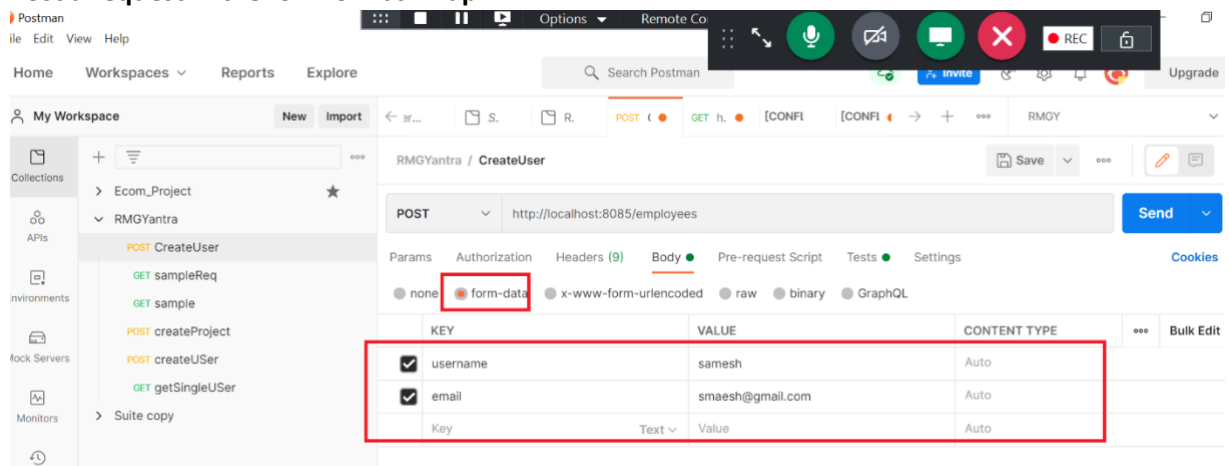
1. Post a request using jsonObject



2. Post a request using Json File



### 3. Post a request in the form of hashmap



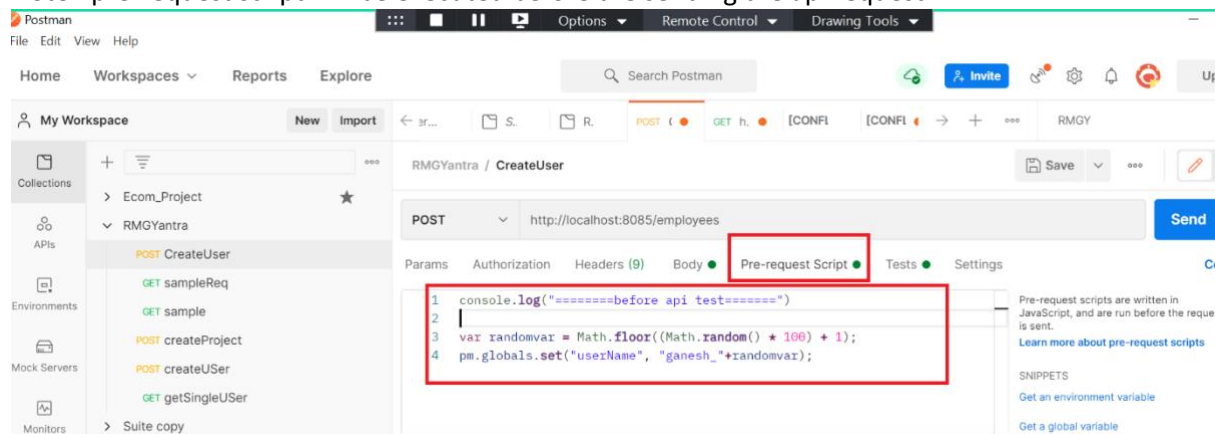
### 36 How to execute same request with dynamic request body

We can post same request with dynamic data without any changes in the request body,

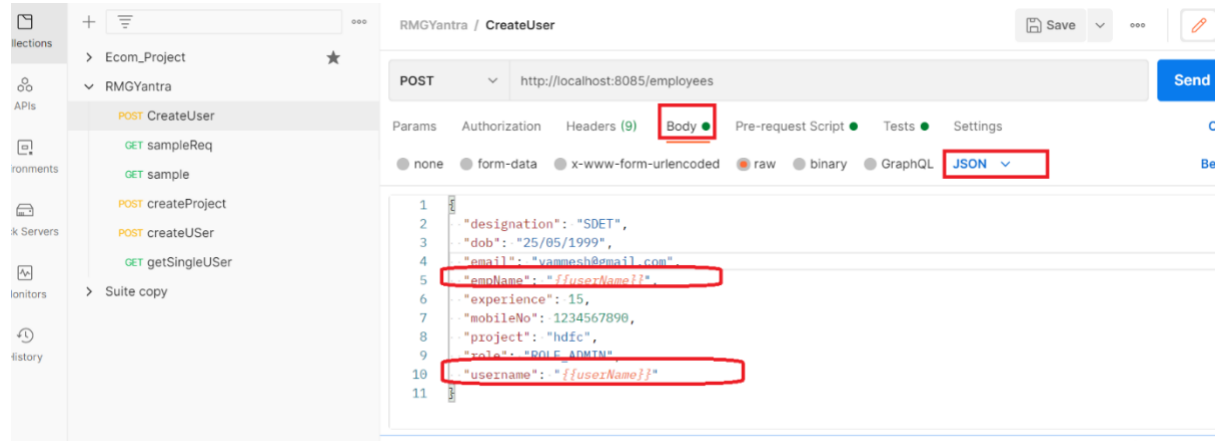
in order to achieve this follow below steps

1. should write random math function in "pre-request-script", then store the random data in global variable

Note : pre-request script will be executed before the sending the api request



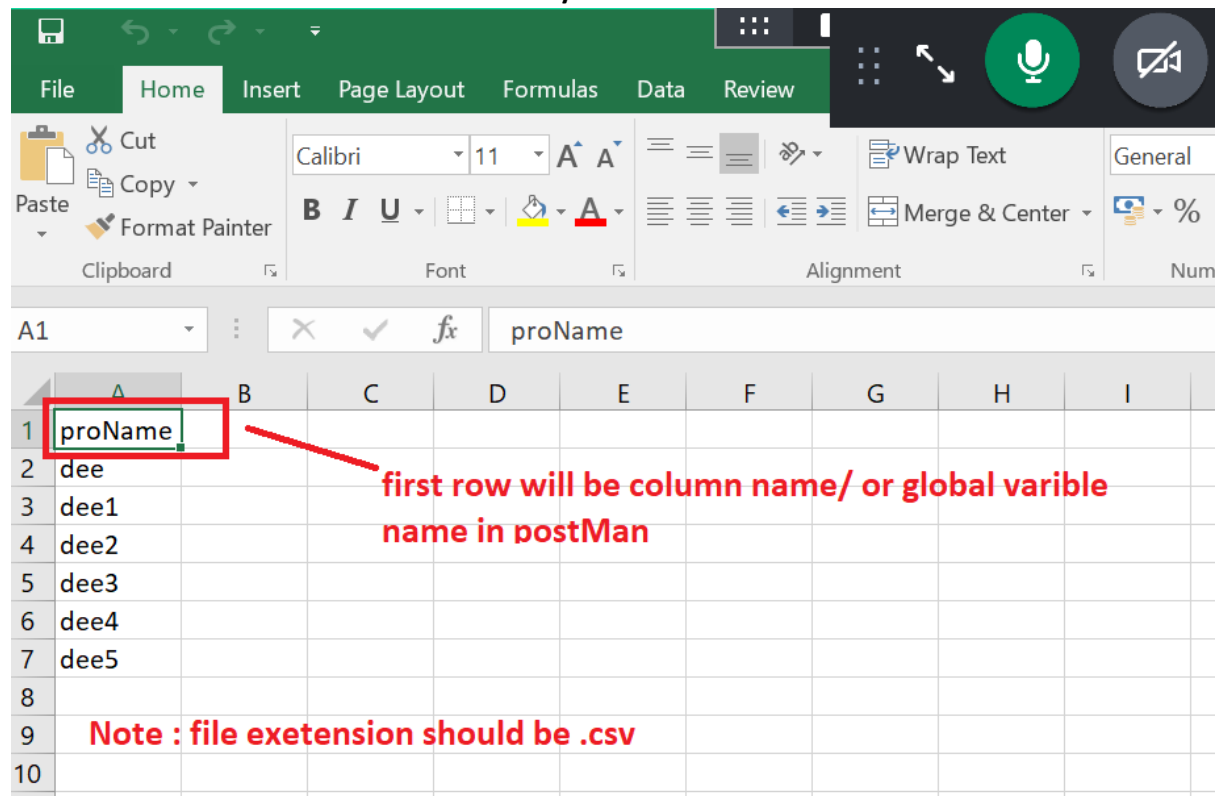
2. use the same random variable in request body



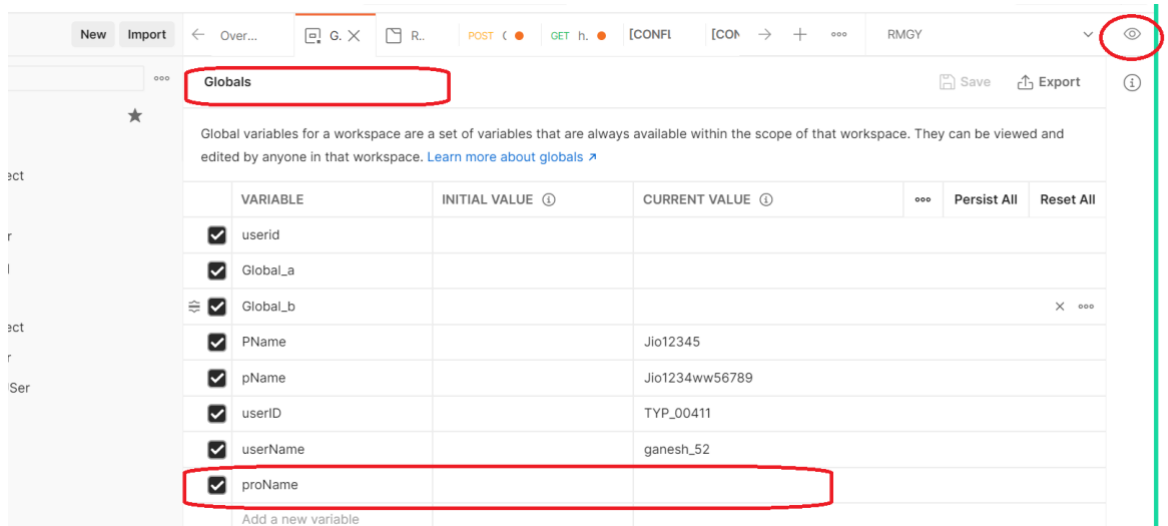
### 37 How to execute same request multiple time with different data

We can execute same api with multiple data using .csv file , to achieve follow the below steps

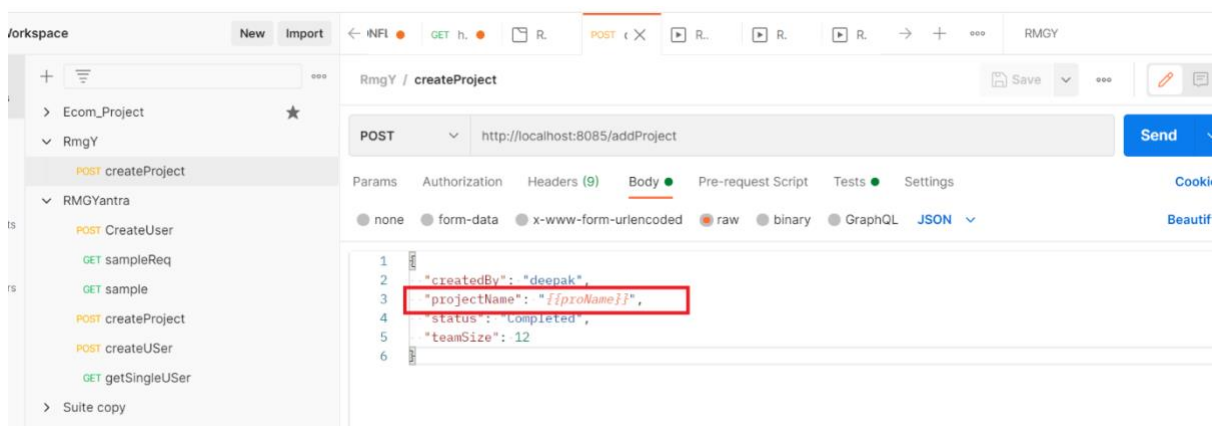
1. Create .csv file with column name followed by data



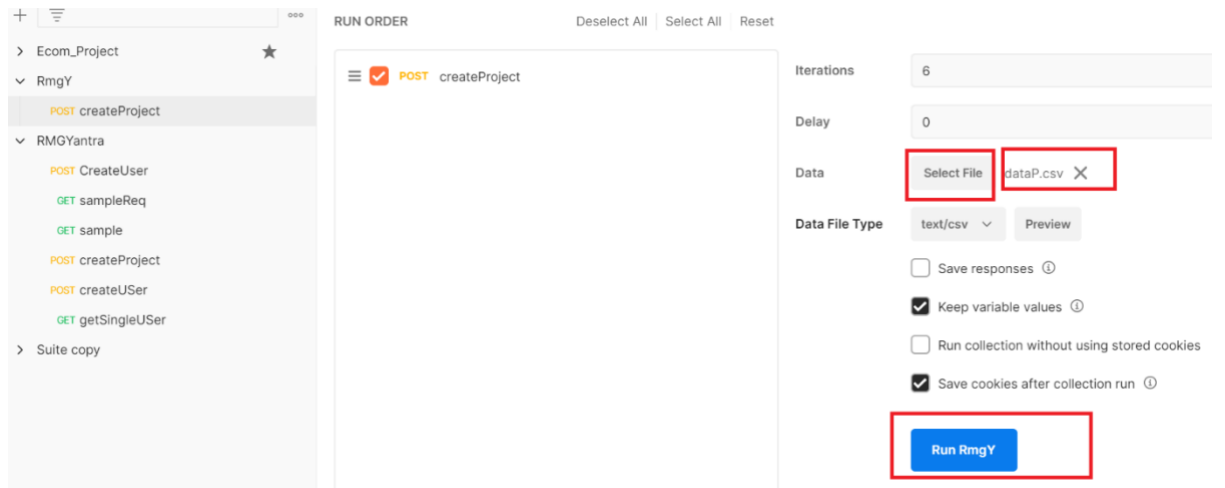
2. Create on one global variable in postName , name should be same as column name in .csv file



### 3. Set the global variable in request body



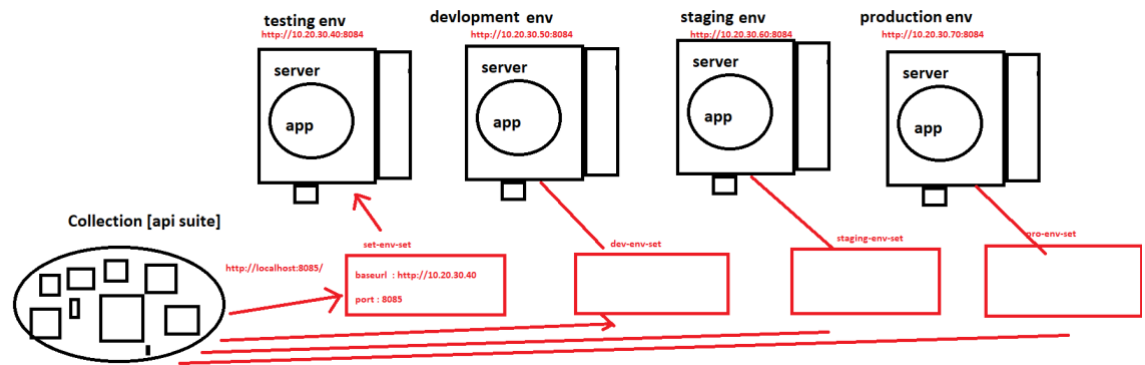
### 4. Run the collection with .csv file



## 38. how to execute API collection with different Environment like testing env , development, staging , production

In order to run same collection with different environment, take a help environment variable available in postman





EG : API URI :

Get → {{baseuri}}/{{port}}/endpoint

### 39. Explain the variables available in postman

1. local variable: (scope: it's a data within the api request) can be used with in a request
2. Global variable: (scope: it's a data across the collection) can be used between the api request, outside the collection also
3. Environment variable: (scope with in the collection) it's a configuration data to run the collection

### 39. What is the difference between Oauth1.0 Oauth2.0

Oauth1.0	Oauth2.0
older version of protocol for authentication	latest version of protocol for authentication
Two level authentication required for every api	One level authentication required for every api
complex authentication approach	Simple authentication approach compare to Oath1.0
To get bearer token , should pass consumerID , consumer secret & accessID , access secret	To get bearer token , should pass Client & Client Secret