<http://18.232.148.34/syntaxapi/swagger/editor/#/>

End point: Address wheree API is hosted on the Server.HTTP methods which are commonly used to communicate with Rest API's areGET, POST, PUT, and DELETE

GET- The GET method is used to extract information from the given server using a given URI.

While using GET request, it should only extract data and should have no other effect on the data. No Payload/Body requiredHow to send input data in GET?  
Ans: Using Query Parameters

POST- A POST request is used to send data to the server, for example, customer information, file upload, etc. using HTML forms.How to send input data in POST?  
Ans: Using Form Parameters /Body Payload

PUT- Replaces all current representations of the target resource with the uploaded content.

DELETE- Removes all current representations of the target resource given by a URI.Resources:

Resources represent API/Collection which can be accessed from the ServerGoogle.com/maps  
google.com/search  
google.com/imagesPath Parameters:  
Path parameters are variable parts of a URL path. They are typically used to point to a specific resource within a collection, such as a user identified by ID<https://www.google.com/Images/1123343>  
<https://www.google.com/docs/1123343>  
<https://amazon.com/orders/112><https://www.google.com/search?q=newyork&oq=newyork&aqs=chrome..69i57j0l7.2501j0j7&sourceid=chrome&ie=UTF-8>

Query Parameters:

Query Parameter is used to sort/filter the resources.Query Parameters are identified with?""<https://amazon.com/orders?sort_by=2/20/2020>  
Headers/Cookies:Headers represent the meta-data associated with the API request and response. In layman terms, we were sending Additional details to API to process our request.  
Example : Authorization detailsEnd Point Request URL can be constructed as below  
Base URL/resource/(Query/Path)Parameters

API class 2 notes in case you dont have them

[6:06](https://syntaxtechnologies-hq.slack.com/archives/GTKT019G9/p1592863581045400)

What is API  
Application Programming Interface  
3 tier client-server architecturemost of the websites are 3 tier nowadays.  
Presentation layer(UI/GUI),

Application layer(logic/code)

Data layer(also called Server layer,Collection of tables which cocnsist of records and every record is a row)JSON-Java Script Object Notation- subset of JavaScript,  
JSON files are a format that uses readable text by both humans and systems  
{  
"key" : "value"-if there is other pair,we use comma","  
}  
how does a request get procces from Presentation layer(consumer)? It needs to hit the application layer. request from consumer(presentation layer),goes to API, translated to JSON format.once the API translates it to JSON format.API transmits it.once the request goes to application layer,application layer directly communicates the data layer(data server/producer).once it come back to application layer,retrive data sent back to the API. API converts it to the JSON format and respons back to the comsumer(customer,presentation layer)

API accesible over the internet is considered Webservice. But not all API'S are webservices.

All web service are API's.  
API's can work without internet on your local machine but they are not considered as"Webservice". Only if it the are accesible they are considered"Webservices"the only way you will be able to access a web service is with the help of internet connection. To be spesific,you would have to establish HTTP protocol as shown below.Before we start using Postman, there are certain rules that we need to follow to make call to these webservices.

When we make a request to a Webservice,we call it"MAKING CALLS". For example; let s make a call to the API.URL:Uniform Resoruce Locator  
URI:Uniform Resource Identifier

C-Create data-- we want to create a user whose name is John- createUser();  
R-Retrieve data-we want to retrieve a user-getUser();  
U-Update data—we want to update a user-updateUser();  
D-Delete data-- we want to delete a user-deleteUser():HTTP

Methods;POST,GET,PUT,DELETE,PATCHCreate-POST  
Retrieve-GET  
Update-PUT  
Delete-DELETE

on order to make calls to a webservices,we need a web service URL and we can only make calls/request to the web service using HTTP methods because we are accessing the webservice over HTTP protocolWhen we want to make a POST call

==============POST=============================

first we need a URI (not URL)  
URI-Uniform Resource Identifier  
URI=BaseURL/DomainURL+webserviceURL(Resource/endpoint)BaseURL(Domain URL can be something like;[http://google.com](http://google.com/) or [http://amazon.com](http://amazon.com/) or [http://xyz.com](http://xyz.com/)Our Domain url(base url=[http://xyz.com](http://xyz.com/)  
webservice URL(Resource URL,endpointURL)= /createUserURI=<http://xyz.com/createUser>  
Next thing we need is a PayLoad(body) To send the information to the server

1.way is JSON:{"user":"Lok",  
"age":"20",  
"job title":"QA engineer"  
}

2ND WAY XML:  
<user>Lok</user>  
<age>20</age>  
<jobtitle></job title>

3TH WAY STRING:  
"user=Lok,age=20,job title=QA engineer"Next thing we need are Headers: in headers what we are basically specifying what our request will contain. It might be 3 different types

Content – Type:application/text  
Content -Type:application/json  
Content -Type:application/xmlNext is we need to authenticate our self in order to make a call. We cant make a call without authentication  
we would neeD some sort of username or password or we would need some Auth key: ksjdkjdgd3743876  
or we would need a JSON WebToken Key which is called  
(JWT)  
OR You may not need to authenticate yourself at all because of the website.some website provides us authentication key.

================GET============================

First we need a URI  
URI  
OUR domain URL/BaseUrl=[http://xyz.com](http://xyz.com/)  
Webservice URL(endpoint,resoruce)is=/getUser.and our URI Will be;  
URI=<http://xyz.com/getUser>We dont need a payload(Body) with a GET call because we are not creating anything.  
Next thing is we need to specify our web service URI.  
(getUser but what user? Will talk about parameters now)  
<http://xyz.com/getUser/firstnames(lets> say I wanna get the firstnames of these users. If you see a slash after endpoint,This is called "PATH PARAMETER")  
/firstnames=path parameter<http://xyz.com/getUser/firstname>s[? firstname=gulsum](http://xyz.com/getUser/firstname?firstname=gulsum)  
anything after a question mark is considered"QUERY PARAMETERS"lets say we want to add another query parameters,then we use "&" sign do add another query parameters;  
<http://xyz.com/getUser/firstnames?firstname=gulsum&lastname=ince&jobtitle=testlead>Path parametermain goal of these parameters is to narrow down our search. Path parameter will narrow down our search but NOT spesifically like a query parameter. For broad search we use Path parameter,to narrow down our search we use Query parameter. Query parameter cocnsist of key and value pairs i.e= firstname=gulsum  
Parameters are not mandatory you will inly use them if you need to for Testing.  
Domain URL(Base URL)=[http://xyz.com](http://xyz.com/)  
Web service ULR(endpoint,resource=/getUsers  
path parameter=/jobtitles  
query parameters=/jobtitles=qaengineersURI=<http://xyz.com/getUsers/jobtitles?jobtitles=qaengineers>Next thing we will need are headersContent=type=application/json  
authentication=JWT(Most companies will have this)

================PUT(UPDATE)===================  
First we need;  
1.URI  
2.PayLoad/body  
3.headers-Content-type,Authentication  
PUT is mainly used to update exisiting data.  
Put request is gonna behave as POST call. But depending on how developers have developed the web service,at times they will allow a PUT request to behave as a POST reuqest if the data that is being updated doesnt exist.  
Developers can restrict PUT request from creating new data.  
POST is the parent to PUT and DELETE  
WE are updating following;  
JSON:  
{"user":"Lok",  
"age":"20",  
"job title":"QA engineer"  
}  
With a put request,we will have to send it as a same format. As shown above,this PUT request will be in JSON format  
JSON:  
{  
"employee ID":"12345",  
"user":"Lok",  
"age":"22",  
"job title":"SDET"}

===================PATCH==========================  
PATCH is used to update only a spesific part or parts of existing data without having to send a full payload.patch  
allows us to update partial existing data without having to send a full payload  
we need;  
1.URI  
2.Headers  
3.Payload/body  
lets say we only wanted to update the age;  
{"employee ID":"12345",  
"age":"25"  
}==================DELETE==========================11.URI  
2.we may or may not need payload.if we send a payload,we will need to specify the information again in(json,xml or string format as we did before)  
3.Headers  
Depends on the company you work at,you may not be able to delete anything at all. Most probably DBA-Database Administrators will acces them only because of the sensitive information they store in Data base.API /web service documentation  
This documentation is most places it should contain information that you will need to make calls to the available web servicesBase URL/Domain URL  
endpoints/resources/webservice URL'syou will not be provided with a URI- you must make it yourself  
 if there are any steps we need to take to generate a token or authentication key,that information will also be displayed for usthis documentation will have information of the expected responses plus the expected status codesThis documantation is increadibly important,it needs to be up to date because this is what you will be needed to testThis documantation is provided by your lead,developers,lead architects,test managers,PMDomain URL is actually is our IP adress  
if you see a website starts with numbers like"166.62.36.207"  
this means,there is no register domain name for this application.