DAY10

Friday, May 12, 2023 11:45 AM

Http Request

Structure of Http Request

Start line	http-method URL http-version
Http Request Header	Key : Value pairs specifying **info to be sent to the server**
	host : IP:port of the server
	user-agent : info about the browser
	accept : MIME TYPES that are acceptable to the browser
	content-type : MIME type of the data that is sent to the server
	Other keys are accept-language, accept-encoding , cookies, timestamp etc
Empty line	
Http Request Body	The form elements and values can be put here OR **data** to be uploaded to the server can be put here (images / files etc)

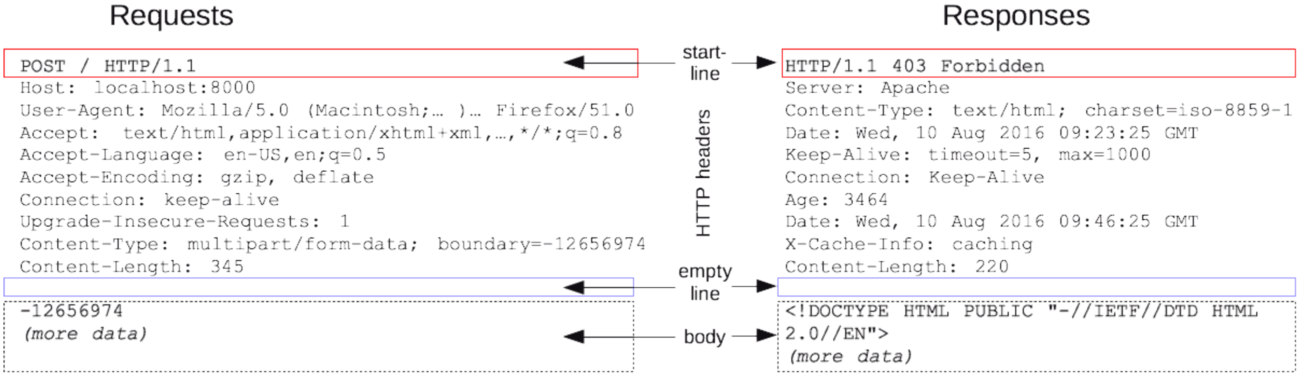
http-method	GET , POST, PUT , DELETE, PATCH ,HEAD ,TRACE ,…….
URL	uniform resource locator = address of the resource to which request is passed/send
	URL format = protocol : //IP :port/AppName/resourceName http://localhost : 5000/kyaboltatu
Http version	It could be 1.1 or 2

Http Response

Start line	Http-version status_code status_message
Http-Response_Header	Key value pairs info about the server to the browser
	server : info about the server that is sending the response
	Content-type : mime type of response data sent to browser
	Content-length : length of content sent
	Date : date timestamp
	Other than that cookies, connection etc
Empty line	
Http Respose body	The data that is sent back to the client as a response

Status_code	What happened to the request is informed to client using some PREDEFINED status codes
	200 OK ====================>Status messages 404 Page Not found 405 method not found 500 internal server error

	…..



```
POST / HTTP/1.1
Host: localhost:8000
User-Agent: Mozilla/5.0 (Macintosh;… )… Firefox/51.0
Accept: text/html,application/xhtml+xml,…,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Content-Type: multipart/form-data; boundary=-12656974
Content-Length: 345

-12656974
(more data)
```

```
HTTP/1.1 403 Forbidden
Server: Apache
Content-Type: text/html; charset=iso-8859-1
Date: Wed, 10 Aug 2016 09:23:25 GMT
Keep-Alive: timeout=5, max=1000
Connection: Keep-Alive
Age: 3464
Date: Wed, 10 Aug 2016 09:46:25 GMT
X-Cache-Info: caching
Content-Length: 220

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
(more data)
```

Http Methods

GET method	When we submit --- data is passed to server as QUERY PARAMETERS	http://localhost:5000?Fname=xx
POST method	When we submit --- data is passed to server in REQUEST BODY	

If we fire any URL from the browser	Then the DEFAULT http method is **GET** method

WE will need a Http Client	POSTMAN	To fire different HTTP methods for URLS
Download the Desktop APP for POSTMAN	https://www.postman.com	

We Wrote REST API on server side !!!!!
We called the REST API using POSTMAN

REST	Rest is a web service based on http
Web Service ?	It is a set of functions/API/service on the server side that gives some functionality to the user
	The REST API or the webservice will run on the SERVER side WHY ? 1. Hard disk and processor resources required to run the API is available on server 2. For security reasons the processing logic and data must always be on server 3. Client does not want to bother about how the answer is generated

CLIENT of REST API	Make a function call to the REST API
	Function is defined on server side and it will run on server side
	The result is returned and transported VIA HTTP to client side

RESTFUL WEB SERVICE ----------------

CLIENT		SERVER
func call	-------HTTP METHOD -----HTTP REQUEST ----->	Func definition
		Func executes
Get the return value Use it	<-------------HTTP RESPONSE-----------return value----	

Basically a WEB service is a REMOTE function call

Local function call	Where func call and func definition is in same process
Remote function call	Where func call and func definition are in different processes

GENERAL UNDERSTANDING - CONVENTION

GET	used by client to GET some data from server side (SELECT)
POST	Used by client to POST new data to server side (INSERT into DB)
PUT	Used by client to CHANGE some data on server side (UPDATE on DB)
DELETE	Used by client to remove some data on server side(DELETE on DB)

From client side we want To PASS the data to server side

1. Query parameters
2. Path Parameters
3. Request Body

Query parameters -------------

Ex1 - pass a uname and password to the REST API method=post url =/login

Pass it as a query parameter

On the server side check if uname=iet and password=123 then return result welcome uname else return result wrong user

Path parameters --------------

Ex1 - pass a number to the REST API method= GET url =/factorial

Pass it as a Path parameter

On the server side receive the number and calculate its factorial and return the result

Request Body -----------

Ex1 ---- pass a json object(name and date of expiry of a product) to the REST API

Method=delete url=/should_we_throw

Response --- if the expiry has already occurred then send a message - throw it

Otherwise send a message you can still use it

Parser ??? === extracts or interpretes data in a certain format .

We will activate the JSON parser of the express using the following ---

```
app.use( express.json() )
```

Routes in express JS ------ to organize different routes as per modules or topics etc in different files client -----------Request ------------->**server2**.js ------observe the url_path and redirect to ---->planerouter.js