# JavaScript

* JavaScript is Scripting, programming, Server language

->JavaScript is a scripting or programming language that allows you to implement complex features on web pages

**.intro---**

4dec 1995 launched by Brendan Eich.

.Name given ---

* Mocha
* LiveScript
* javaScript
* ECMA Script

.HTML -> Js (we can used js with HTML that’s called internal js )

.JavaScript can be used by two method

1.internal…



2.external ….



.We have to link js file in HTML file for external js

Like….

****

.console, document- is an object

.log, write, alert- is a function

To declare variable ..

we can use ---

1. Const- it can not re-declare and re assign .

2. var-can re declare and re assign the variable.

3. let- can only be re assign

.used parseInt(prompt(“ enter any number”))

-used to take integer from user ,that will show pop in the document page (browser).

-parseInt convert string to integer.

Operator -- 3type

1. unary operator ( increment, decrement , negation,)
2. Binary operator(Arithmetic, Relation, Logical, Assignment =, +=, -=, /= )
3. Ternary operator ( Conditional Statement ? :)



Loops

1. do while loop (where we should run program at least on times)
2. While loop (where terminating value is not known)
3. For loop(where terminating value is known)
4. For in loop
5. For of loop ( to take one by one element from array)



1. For each loop

Function

- function is a block of code or set of instruction

- re-usability is the property of function

- function must have…

1. function declaration
2. Function defination
3. Function calling

--- function is two type…

1. pre defined
2. User defined

De-structuring of array :-

- it is used to access of data from array using variable at the place of indexing.



Spread opreator (…)

-use to merge element or data of two array.

- it will fetch only element from array.



Object

- it is a datatype ,which store multiple data

- assign using {}

- store data in unordered (that position of data is not fixed) .

- store data in key value pair.

Nested object : we can use object inside object .

. We can use array inside object .

.we can use function inside object .and we can also pass argument inside function or we can use all to type of using function



.this(function) : refer to the data of (same) that

Object.



---> it will print the name (manish).

for in Loop: used to fetch key of object or index of array.



---> it will print name, age,contact

. We can also fetch values of object using ob[variable].



---> it will print manish, 21, 532545.

**Array of object:**

-- it is used to store multiple object in single variable.

- to store data of multiple user in one variable



**Arrow function:**

-- made in 2015.

--- used for code concise.

-- one liner function .

-- return will always used with curly braces{}

Syntax are:-

1.() =>{return}



.Argument-ed arrow function:



2.() =>()

3.() =>

4. => ()

5. =>

.callback function :

-- when one function is used in argument of other function.

.map function : fetch data from one array and store in other array.

. It accept function in argument.

. If we use operator in map function it will give boolean function .

.filter function :

- filter out the data at any specific condition

- it also accept function in argument.

- if we use operator it will give numerical values.



.setInterval function :

-it will give some pop up at some interval

-it will accept 2 argument

1. Function
2. Interval time (in mili sec)

.clearInterval : to stop that pop up executed by setInterval.



.setTimeout function :

- pr-defined function.

- it will used to take a pop up at after some time for only one time .



local Storage:

- it store data in string type.

- string store in key and value pair format.

- key is always unique. If we set agin store item it will override.

- local storage store data until we delete that data.

--3 things important

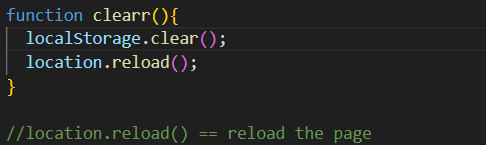
1. Protocol -- http, https
2. Domain -- www.google.com
3. Port -- 5500(number written on the browser at the end)

. Four main function in local storage:

1. setItem() -- store data in local storage.
2. getItem() -- retrive data from local storage.
3. removeItem() -- remove pericular data from local storage.
4. clear() -- clear all data from local storage.

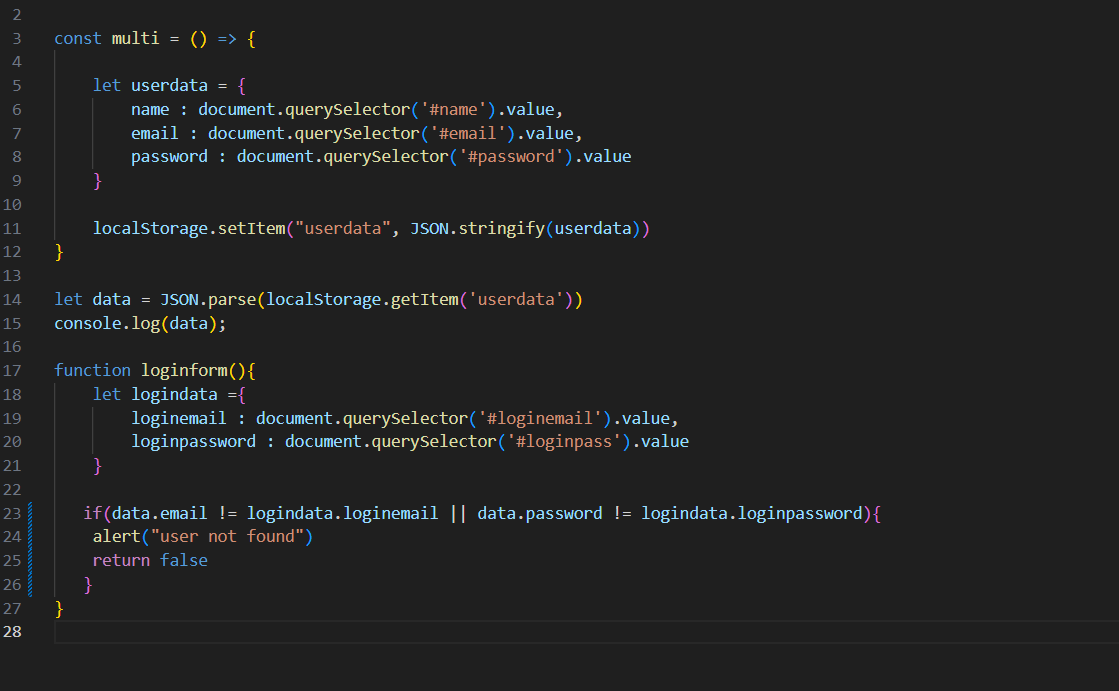


.location.reload() -- is used to reload the page.



.JSON.Stringify() -- it will convert data into string .

.JSON.parse() -- will convert the string data in its previous format.



.Authontication: -

- it means that user is authentic (real).

- sign-up id and login id is same.

.API- Application programming interface.

- pre-defined service that we use in our website like payment gatway.

1.fatch(url,{method})

-- it will use to fetch api

. It will use methods --> get, post(insert), delete, put/patch(update) --> Http method/request

- fetch function return promise:-

-- promise has 3 stage.

- promise is a object .

1. Reject,
2. Pending
3. Fulfil

-- to handle promise -> async and await is used

-- both async and await will be take simultaneously.

.JavaScript - synchronous, lightweight

.Synchronous

-It will execute code line by line.

-every line of code wait for his turn.

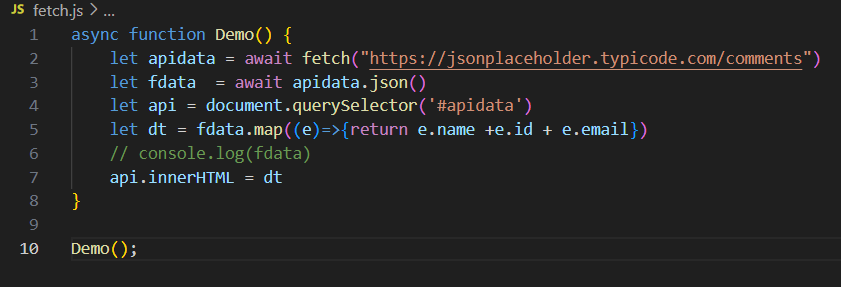
. Asynchronous--

-it will execute that line of code which take less time

.if server is slow synchronous will be work as asynchronous.

. For taking fack api we will go on jsonplaceholere website.

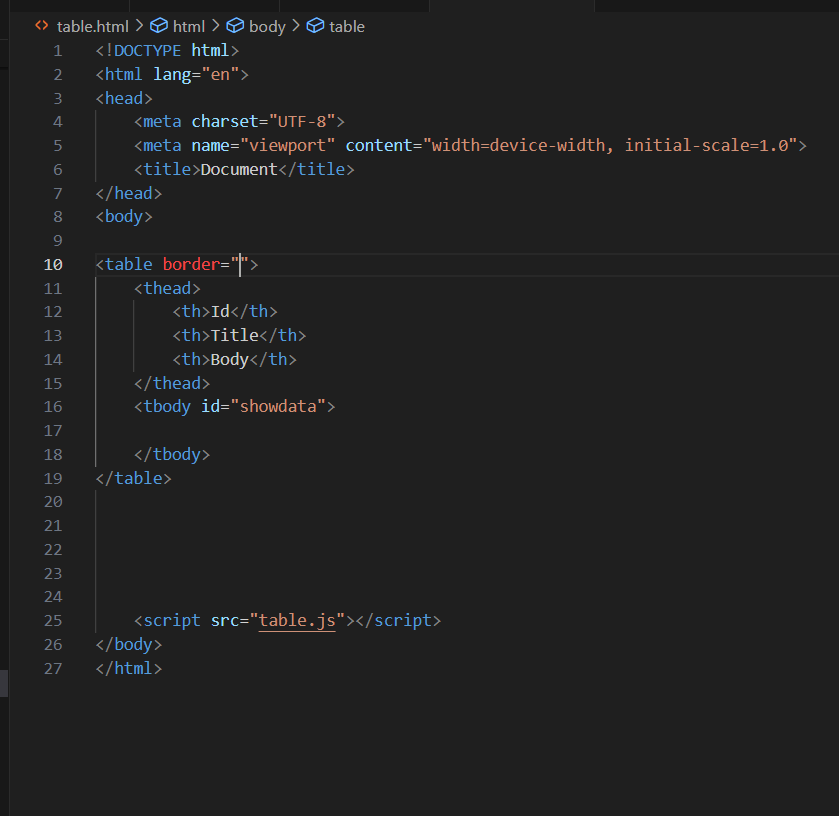
- .json() - it will used to convert



Q1. how two fetch data from api and insert into table?

Ans..

.html



.js



- .join(“”) -- it is used to add , or more like that .

. ` ` - back-tag or

Q1. what is eventbubbling.

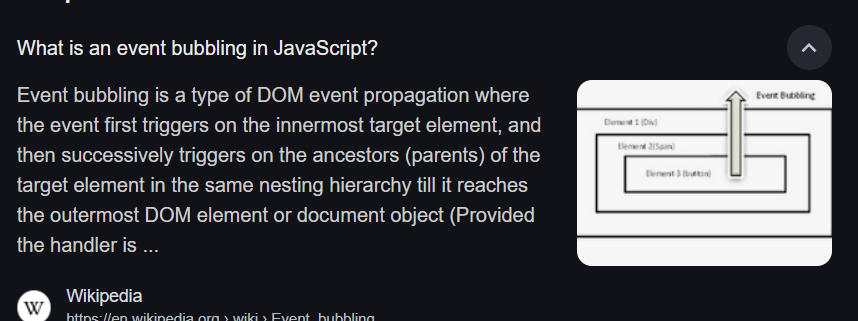
//Q2. what is event deligation .

**Q1. what is event-bubbling?**

- it is process when an event happens on an element ,it first runs the handlers on it, then on its parent, then all the way up on other ancestors.

**Q2. what is event capturing ?**

- Event capturing is opposite to the event bubbling. In event capturing the flow goes from outermost element to the target element.



**Q2. what is event delegation ?**

-Event delegation is a technique in JavaScript where a single event listener is attached to a parent element instead of attaching event listeners to multiple child elements.

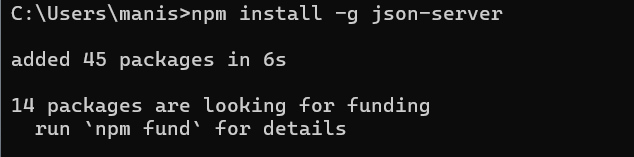
**1.- npm i -g npm@latest** (command to update npm)

.node js -- run time environment

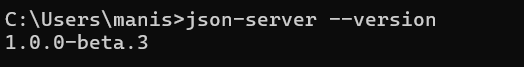
.npm - node package manager -- package manager of node, with help of this we can down

**.json - JavaScript object notation**

( it will be use as database), ( as a api), ( json will run on server),

**2.npm install -g json-server** (command to install json server)

**3.json-server --version** (command to check json server)

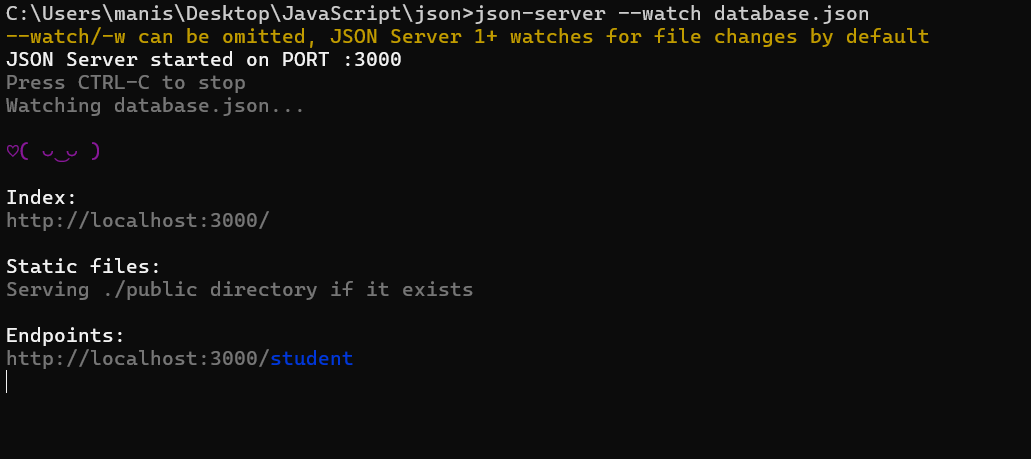


**4.node -v** (to check node is installed)

**5.npm -v** (to check npm )

.json will run only when json server open.

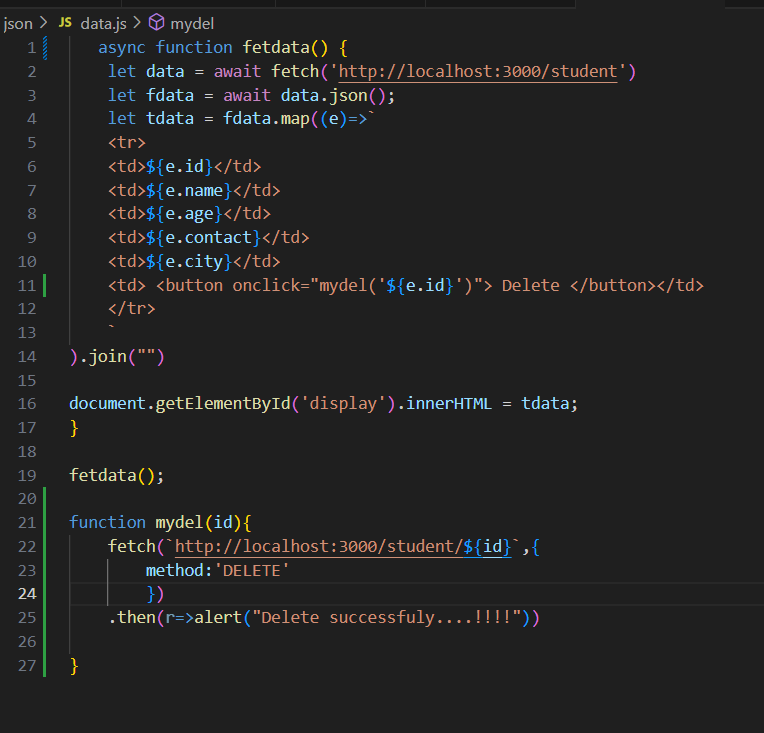
**6.json-server --watch jsonfile.json**



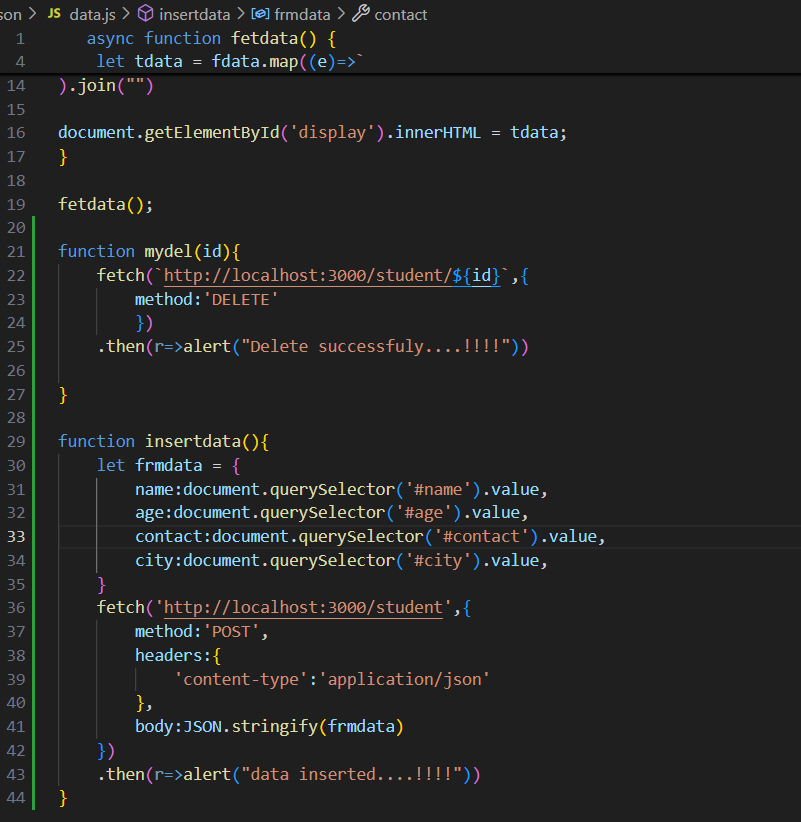
.when we wanted to use JavaScript and html together then we use back tag. (``).

**.To delete :**

-to delete data from json file using



.To insert data:



Html:

