**JAVASCRIPT**

<https://www.programiz.com/javascript/online-compiler/#google_vignette>

variables

How to setup node js in the system and environment variables

Node js is mandatory for JS..

**After installing node js u will get some errors in the command prompt. Do the below.**

It looks like you have entered the [node REPL](https://nodejs.org/api/repl.html) and then typed node -v

The good news is that this means node is working!

To check the version from here you can type process.version to get the node version. Alternatively type process.exit() to exit the node REPL and get back to bash (or equivalent terminal).

From here node -v should just output the node version.

**Variables**

Var, let, const

Var- Global scope

Let- local scope

Const- if set, we cant change it

When defined inside a {}

Html AND CSS and javascript for interactive.

For validation of forms- on performing some actions in the web application js is used.

Js code-> js engine> web rendering in the browser.

Engine- will proerply render the code and display it in the web.

Context is proper because of js engine.

Context is DOM.

Js engine and it needs a browser page context.

V8 engine- is written C++ code.

Chrome browser has a v8 engine for rendering the code

It converts and render the code to display in the web browser.

Node--------

It has an v8 engine inside it.

It doenst need html or css.

Directly it can use js alone,

It can directly run on the server.

Html- css will run on the client side.

Node js is not a programming language

Not a framework.

It is JS runtime.

It has some built in apis---

It has some built in modules.

We can connect js to OS/server/file system/read and write.

And interact with above.

Running on node js.

JS can interact with node other modules.

It will interact to the other modules through the apis.

Client side is the browser

Js can used to access server through the node js.

It has pre written modules- wich we can use while run time.

Operators

**Operators:**

Incemrent and decrement operators

Post increment and pre increment operators:

I++ i--

--I, ++i

var a =10

b=a--

c= 20

d=c++

console.log(b)

console.log(d)

the value of a is not incremented until it meets a in the console again.

In the post increment values.

Where as in case of pre increment

Var a =10

++a;

++a, it has already has increment before it and then it sees the value a- so it will be displayed as 11.

**Data types:**

Type of is an imp operator type of- number and string

console.log(typeof("a"))

console.log(typeof(10))

**Boolean- true or false**

**Null - null**

**Undefined- if there is no value**

**Functions**

**For running the script in the terminal**

**Use node jsfilename.js in the terminal**

**PS C:\Users\Samidurai\Desktop\Samidurai 2024 Leanring\JavaScript> node first.js**

Start with function’

Function is the keyword

To call the function --- function name()

We can call this function n number of times and anywhere

function aaa(){

console.log("Samidurai")

}

aaa()

function with paramters inside

NAN- not a number

function pp(a, b) {

    console.log(a + b)

}

pp(10, 17)

**Return type and return keywords**

Function

What is the use of return function

function nam(){

var a=10

var b=17

return a+b

}

var namea=nam()

console.log(namea)

whatever we define in the return

that will be called when calling that function

when calling the function call the name along with the variables inside.

var total=news(9,3)

function news(c,d){

    return c\*d

}

var total=news(9,3)

console.log(total)

if else condition

if(){}  
else{  
}

if the condition **met is true** first set of code will work

if the condition is false then second set of code will work.

var rain=false

if (rain) {

    console.log("take umbrella take")

} else {

    console.log("sunshining")

}

Logical and or not operators

&& and OR ||

Both conditions must be true

If any one statement in And is false- then total is false

If any one statement in OR is true then totally it will be true

console.log(true && true)

console.log(true||false)

var score =50

if (score >=50){

    console.log("good score")

}

if (score>80) {

    console.log("outstanding")

}

if (score<=50) {

    console.log("need to improve")

}

Else if will have another condition in it

var score =50

if (score >=50){

    console.log("good score")

}

else if (score>80) {

    console.log("outstanding")

}

else if (score<=50) {

    console.log("need to improve")

}

**For loop statement:**

Initialize, condition, iteration- increment or decrement

for (let index = 1; index <=10; index++) {

            console.log(index+"X7="+index\*7)

}

Arrays and objects

JS dom manipulation

DOM

Document object modelling

html when going to browser- it goes like Document object

This DO will be in tree structure.

Fist html then it has head and then body

In the treet structure

Document

Root

Html

Head body

Title

The above is DOM

DOM manupilation

I have to go through the document ot change the dom values

For css- style

For javaacript- script

Select the value by id

code language not supported or defined. visual studio code html

while running the js code in html

<script>

console.log(document.getElementById("head"))

var hhh=document.getElementById("head")

console.log(hhh.textContent)

hhh.textContent="Samidurai"

</script>

Here for calling the file in html

First start with script

<>

And then inside the script tag for callng the html tag.

Html tag should have a name, class, id

Call it by id and

By

First Document.getElementById(“id name”)

Or else set it under the variable

And for getting the content of the variable use the gettextsontent.

Then

Var a = document.getelement byid(“idname”)

Then a.textcontent =”change the value.”

Variablename.textContent=”changed values”

 <button onclick="changes()">

        buttins

    </button>

Events and event handlers

Example

First write the code in the function for changing the value.

Hhh value is changed but function still not called

  function changes() {

        hhh.textContent = "changed values"

    }

To do it

I have to call it in the button event

Button click is an event.

Mouse moving, scrolling,. Moving, scrolling.

Any events around the browser is called event.

And functions handling the events are called as event handlers

<button onClick=”funcitonName()”> button <button>

5.10

What are we doing first

1. First create the input fields with ids or classes
2. Button fields
3. Result values in p in the html file

Now to call these values in the js

Add below the script file

1. Create a variable and call the value that was defined in the html file using the ids.
2. Now create a function for adding the values
3. And store the result in the variable

Now call the result function in the button in the html file

1. Button on click (function name())

<html>

<input id="num1", value="00">

<input id="num2", value="00">

<button onclick="ress()">Add</button>

<p id="re">Results</p>

</html>

<script>

    var n1 = document.getElementById("num1")

    var n2 = document.getElementById("num2")

    var res = document.getElementById("re")

    function ress() {

        var nn1 = Number(n1.value)

        var nn2 = Number(n2.value)

        var total = nn1 + nn2

        res.textContent = total

    }

</script>

Write all the infoin the function. So when callimng the function, total code inside will b ecevcuted.

--

==

Random numbers

While getting the or the setting the variable types.

Make sure to handle the Variable types.

var randonNum = Math.floor(Math.random() \* 10 + 1)

math.random() will get u numbers from 0.1 to 0.9 with some decimals upto 10

math.floor will round off to the maximum number

<h1>Enter the number</h1>

<input id=num>

<button onclick="checks()">Check</button>

<p id="res">The answer is Right/Wrong</p>

<p id="score">Your score </p>

//first declare the elements of the web page

<script>

    // then in the script call those variables using their ids, class names

    var snum = document.getElementById("num")

    var sp = document.getElementById("res")

    var randonNum = Math.floor(Math.random() \* 10 + 1)

    var score = document.getElementById("score")

    console.log(randonNum)

    var totalscore = 10

    //then create a function to do all the changes.

    // and then call it in the html buttons

    function checks() {

        var currentNum = snum.value

        if (currentNum == randonNum) {

            sp.textContent = "Right"

            alert("yeah")

            totalscore = totalscore + 1

            score.textContent = "score:+" + totalscore;

            console.log("if")

        }

        else {

            console.log("asd")

            sp.textContent = "wrong"

            totalscore = totalscore - 1

            score.textContent = "score:+" + totalscore;

        }

    }

</script>

CSS and JAVAscript

<script>

    var divs = document.getElementById("box")

    function change() {

        box.style.backgroundColor = "blue"

    }

</script>

For calling the style in the css

First define the variable and get the element in.

Then add those changes in the function.

And then for adding the styles.

Use

Variable name.style.options.

We can define values in the style tag

 .widthmax{

        width: 500px;

    }

And use this anywhere as an attribute to change or update something.

U can call it in the script

By using box. setAttribute(“class”, “class name”)

Transmission

transition: 5s;

for making the changes slowly

<p>paragraph <span style="color: red">Is normal</span> Paragraph</p>

<

Span is mainly for inline changes within an attribute.

<html lang="en">

<style>

    div {

        background-color: beige;

        width: 100px;

        height: 100px;

        border: solid;

        border-color: brown;

        color: rgb(46, 178, 46);

        transition: 5s;

    }

    .widthmax {

        width: 500px;

    }

</style>

<div id="box">Samidurai</div>

<button onclick="change()">Change color</button>

<p>paragraph <span style="color: red">Is normal</span> Paragraph</p>

<script>

    var divs = document.getElementById("box")

    function change() {

        // box.style.backgroundColor = "blue"

        box.setAttribute("class", "widthmax")

    }

</script>

Whenever input is used, to get the value. We have to use value.

And not textcontent.

**First declare the web elements in the html content**

**Then declare the variables in the script and get the elemetns by id**

**Then create a function to create activities and then call the function in the html elements.**

<input id="in">

<button onclick="text()">getText</button>

<h1 id="head"></h1>

<script>

    var inputs = document.getElementById("in")

    var heading = document.getElementById("head")

    function text() {

        heading.textContent = inputs.value

    }

</script>

<input id="in"onkeyup="text()">

<!-- <button onclick="text()">getText</button> -->

<h1 id="head"></h1>

<script>

    var inputs = document.getElementById("in")

    var heading = document.getElementById("head")

    function text() {

        heading.textContent = inputs.value

    }

</script>

For calling the function in the attribute.

We can use the function directly in the html tags.

<input id=”name” onkeyup=”functionName()”>

functionName(){

will have the code for getting the value and setting the valaue

so on the input itself we are getting it.}

Create element and append

Elemtnt can be created inside the js code.

Using

var element= document.createelement(“h1”)

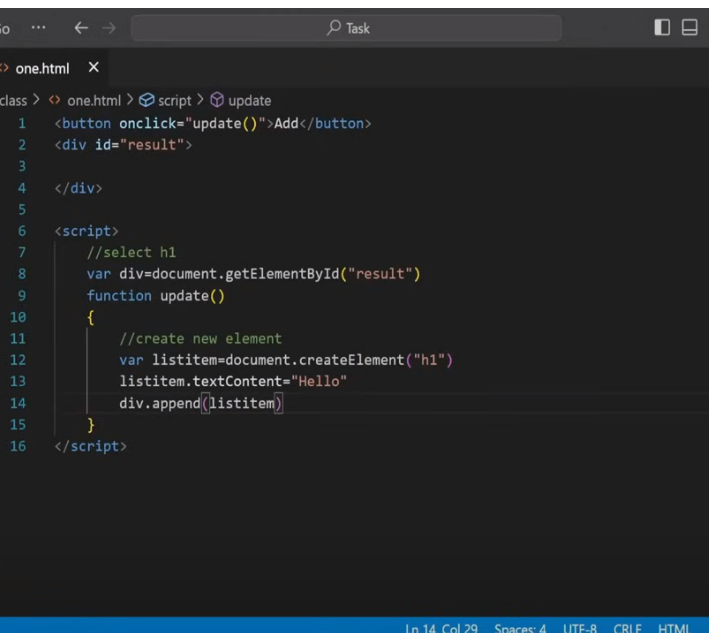
element.textcontent=”hello”

div.append(“value to be appended”)

the div should be declared in the html tag..

and on every click, div will be created and ppended.

Append and create element.



Js for changing button color.

<button id="btn" onclick="change()">color</button>

<script>

    var bb = document.getElementById("btn")

    function change() {

        bb.style.backgroundColor = "red"

    }

</script>

**For getting the text content and button content**

<h1 id="rr">result</h1>

<button id="bb" onclick="changes()">Samidurai</button>

<script>

    var res = document.getElementById("rr")

    var bs = document.getElementById("bb")

    function changes() {

        res.textContent = bs.textContent

    }

</script>

**Events**

For getting the event information of the particular element we can use event.

 function changes(event) {

        res.textContent=event.target.textContent

      console.log(event)

    }

For that first call the event in the element

And then pass the value event in the function

And then inside the function, get the value from

Event.target.textcontent.

Can use the event in all the elements.

<h1 id="rr">result</h1>

<button id="bb" onclick="changes(event)">Samidurai</button>

<button id="bb" onclick="changes(event)">bhuvana</button>

<button id="bb" onclick="changes(event)">viayn</button>

<script>

    var res = document.getElementById("rr")

    var bs = document.getElementById("bb")

    function changes(event) {

        res.textContent=event.target.textContent

      console.log(event)

    }

</script>

**For deleting an element**

Can use the method remove()

<h1 id="hh">heading</h1>

<button onclick="ch()">delete</button>

<script>

var hhh=document.getElementById("hh")

function ch(){

hhh.remove()

}

</script>

<h1 id="hh" onclick="ch()">heading</h1>

We can pass the mouse hover to the normal html elements too.

Click

<h1 id="hh" onclick="ch()">heading</h1>

<h2 id="h2" onclick="change()">heading</h2>

<script>

    var hhh = document.getElementById("hh")

    function ch() {

        hhh.remove()

    }

    var h22 = document.getElementById("h2")

    function change() {

        h22.style.color = "blue"

    }

</script>

To do list:

Innerhtml:

Div id=1

H1 hello

Script select

Inner html is used for getting the iner html content along with the text.

Wheras the text content will get only the text whereas the innerhtml can get inner html and its inner text..

Need to add along with the html content

<div id="Hello">

    Helloa

</div>

<script>

    var hell = document.getElementById("Hello")

    hell.innerHTML = "<h1> Samidurai </h1>"

//this will change the content of the div tag created above

</script>

Different ways of selecting an element:

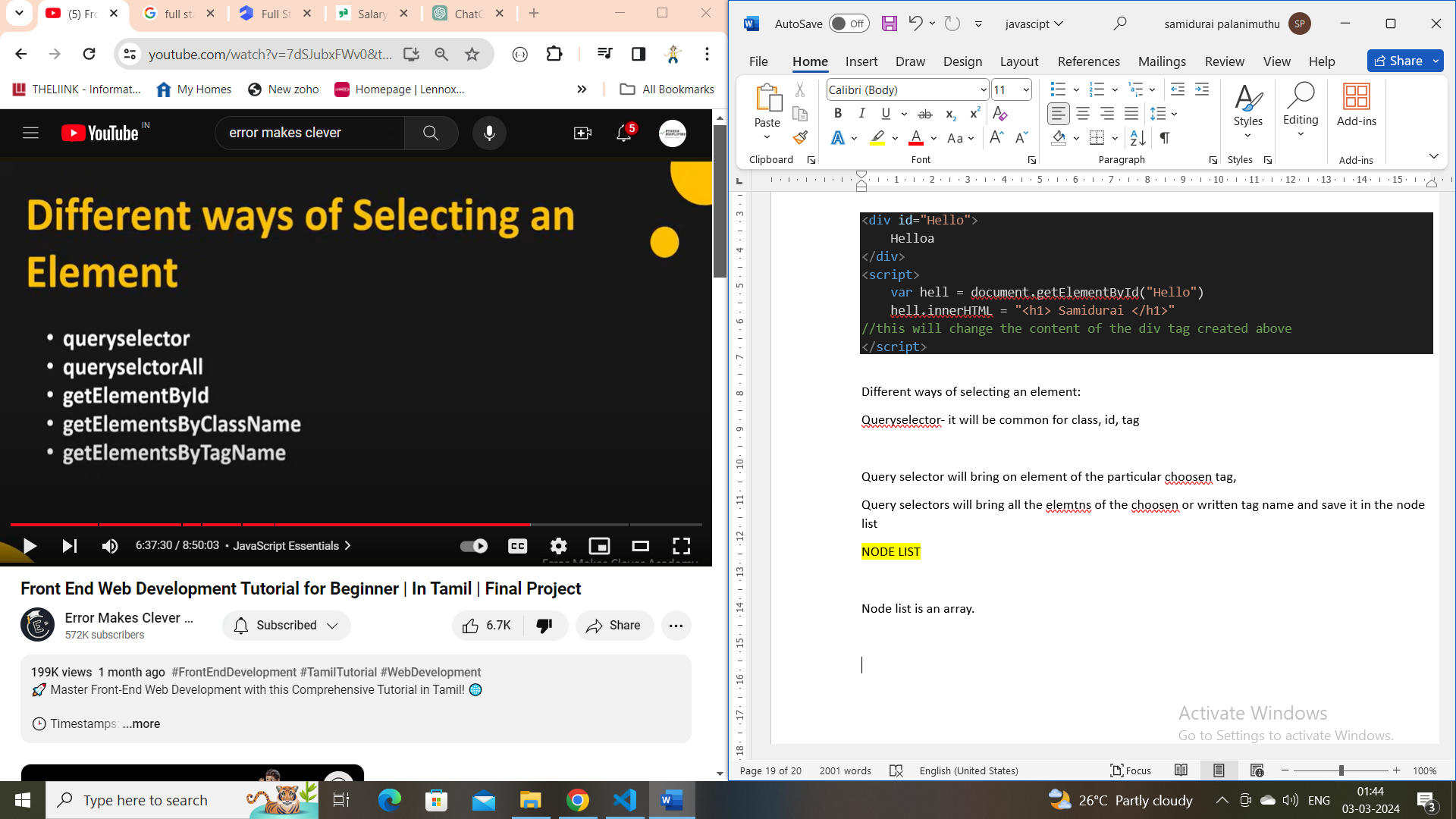
Queryselector- it will be common for class, id, tag

Query selector will bring on element of the particular choosen tag,

Query selectors will bring all the elemtns of the choosen or written tag name and save it in the node list

NODE LIST

Node list is an array.



Query selector

Queryselector all…dot class name

Below all we can eget the element by – directly call the name without dots

Getelement by id

Get element by class name

Getelemetns by tag name

Append will add the element one by one

Prepend will add a element before

**Insert adjacent element**

4 types of adding adjacent element

After begin after end, before begin, before end

Before begin- before the div elemtn

After end- after the div- outside the div

Before end At the end of the div inside div.

<div>

    <h1>

        john

    </h1>

    <h1>

        praveen

    </h1>

</div>

<script>

    var d = document.querySelector("div")

    d.append("hello")

    d.prepend("prepend")

    var a = document.createElement("h1")

    a.textContent = "a"

    d.insertAdjacentElement("afterend", a)

</script>

For deleting an element

Use remove() function

  var pp=document.getElementById("ppp")

pp.remove()

TODO list from scratch

<input id="in">

<button onclick="add()">add</button>

<ul id ="listContainer">

    <li>hello

    <button>Delete</button>

</li>

</ul>

<script>

var ul = document.getElementById("listContainer")

var ii=document.getElementById("in")

function add(){

var litem= document.createElement("li")

litem.textContent="lists"

ul.append(litem)

}

</script>

For adding the value to the list

First create a list

And then

In the script…

On clicking the button

First get the list

Function add(){

Create a list item.. element

Add value to the list items

And then append the values to the previous elemtn

When addin the value from the input box

Then add an id to the input box,

Then select the input from the script

Then doc.value

Then append the values

event.target.parentElement.remove()

// on using the event.target.parentElement.

// we can use the parent elements too.

For adding the element inside the element.

Instead of using text content.

Change it to innerhtml

It will create an element

When having double quotation.

If single then fine.

If double quotation inside double, it will consider it as closing the code.

So we have to use backslash

Make sure the call the classes and other values properly. Using . “” and other types

For

function show(){

    pops.style.display= "block"

    popup.style.display="block"

}

function clo(){

    pops.style.display= "none"

    popup.style.display="none"

}

For displaying the content and not displaying. We can use display none display block

Event listener

Event listener will give me options over the element.

As event selects the element

Event listener can get the whatever options available on the element.

Also event listener, we can use it in multiple options for the same element and also

We can us anonymous functions for it..

Proper program structure:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Booksky</title>

    <link rel="stylesheet" href="style.css">

</head>

<body>

    <script src="script.js"></script>

</body>

</html>

Css linking in the head with link tag

Js link in the body with the src location

<link rel=”stylesheet” href=style.css>

<script src=location.js> </script>

.book-container h2 {

    color: cyan;

}

.book-container h5 {

    color: darkolivegreen;

}

.book-container button {

    color: dodgerblue;

    border-radius: 10px;

    background-color: crimson;

    padding: 5px;

    padding-left: 10px;

    padding-right: 10px;

    border: none;

}

For styling the element inside the div or some tags

Use class name and then call the element name

.classname element type

.book-container h5 {

    color: darkolivegreen;

}

We can add two types to the same element

<button class="addButton" id="addpopupbutton">+</button>