 ==================================================================

   Video-2: Create Script tag and connect external Script file

 ===================================================================

আমরা external / internal CSS use করতে পারি

তেমন আমরা JavaScript এ ও external / internal CSS file use করতে পারি।

====  ====  ====  ====  ====

Internal JavaScript <script> tag এর ভিতর <body> tag এর last tag

হিসেবে ব্যবহৃত হয়।

<Head> এ ও লেখা হয়। কখন কোথায় সেট করা হচ্ছে তার ও কারণ আছে। আমরা

এখন আপাতত <body> tag এর last tag হিসেবে লিখবো।

====  ====  ====  ====  ====

External JS file link করতে:

আমরা একটা js folder open করে তাতে কয়েকটা file খুলে সেগুলো add করে

দিতে পারি html file এ। এখানে আমরা <script:src> tag use করবো

javascript file order follow করে console এ output দেখাবে।

// ================================================================

//          Video-3: What is DOM (Document Object Model)

// ================================================================

browser console এ document লিখে enter দিলে #document এ html file

এর পুরো body দেখা যাবে। অর্থাৎ document এর সাথে internal connection আছে

console / JS এর।

console.log(document.body) লিখে দিলে <body> tag এর elements দেখাবে console এ।

\* Model - Demo/Replica [কোনো জিনিসের। Original না]

\* object এ property থাকে আমরা জানি। এছাড়া object এ অনেকধরণের

function ও থাকে। এদেরকে method বলে।

# DOM - JS দিয়ে html কে Access করার system বাতলে দেয়।

'Document Object Model' - The 'DOM' represents a document with a logical tree.

Total html document কে 'DOM' দ্বারা represent করা হয়।

JS দিয়ে এই document এর element গুলোকে আমরা access করতে পারি 'DOM'

এর সূত্রানুসারে।

প্রতিটা node কে object বলা হচ্ছে যা document এর প্রতিটা part কে

represent করে।

// ================================================================

//  Video-4: Get different HTML elements in JS by using tag name

// ================================================================

<script> tag এর console এ কোন object লিখে তা browser console এ

displayed হবে অর্থাৎ object ও তার property access করা যাবে।

# // 1. write object on JS and we can access object from browser console.

// javascript

    <script>

        const student = {

            ID: 15,

            Name: 'Kismot Ali',

            age: 16,

            height: 60,

        }

    </script>

// browser console --> objectname.property name

    [ex - student.name] --> output : Kismot Ali

# // 2. To access the tag with elements from browser console

// browser console --> document.tagname [ex- document.body]

    Ex - document.getElementsByTagName('h3');

# // 3. To access the 'h3' tag from document

\*\* browser console shows the result of function.

// javascript

<script>

    const blogTitles = document.getElementsByTagName('h3');

        console.log(blogTitles);

</script>

# // 4. we can make loop to see the elements of "blogTitles" in browser console.

// javascript

    <script>

        for (const 'h3' of blogTitles) {

            console.log(h3); // h3 তে quotation হবে না। highlight করার জন্য দেয়া হলো এখানে

        }

    </script>

# // 5. we can also access inner text of h3 and it will be displayed on browser console.

// javascript

    <script>

        for (const 'h3' of blogTitles) {

            console.log(h3.innerText); // h3 তে quotation হবে না। highlight করার জন্য দেয়া হলো এখানে

        }

    </script>

# // 6. If we want to access paragraph tag

// javascript

    <script>

        const blogDetails = document.getElementsByTagName('p');

        console.log(blogDetails);

    </script>

    # // or, run for loop

// javascript

    <script>

        for (const 'p' of blogDetails) { // we can also use 'blog' of details

        console.log(p); // p তে quotation হবে না। highlight করার জন্য দেয়া হলো এখানে

    }

    </script>

    # // to access inner text of <p> tag

// javascript

    <script>

        for (const 'p' of blogDetails) { // we can also use 'blog' of details

        console.log(p.innerText); // p তে quotation হবে না। highlight করার জন্য দেয়া হলো এখানে

    }

    </script>

\*\* we can change text, save text by using JS.

================================================================

  Video-5: Capture and change Element by using 'getElementById'

================================================================

\*\* we can change text, save text by using JS.

/\*

    sometimes we need to make change for a particular part.

    ex- we need to change one 'paragraph' of <p> tags. We can do

    it by 'CSS' [set 'id' for that part] but here we do that by JS.

// CSS style

<!-- Just use internal style to practice we will do by JS -->

    <style>

        second-title {

            background-color: rgba(0, 255, 255, 0.281);

        }

    </style>

// \*\*\*\* browser console ==>

    1. document.getElementByID('second-title') // --> [to access this ID]

    2. document.getElementByID('second-title').style //

    browser recommends all CSS properties. [Names are in camelCase]

    # \*\*\* here we can set the value to check

    document.getElementByID('second-title').style.color = 'red' //

// javascript : \*\*\* to make change from document

<script>

    # second-title

    const secondTitle = document.getElementById('second-title');

    secondTitle.style.color = 'red'; // color wil be set

    secondTitle.style.textAlign = 'center'; // text alignment done

    # third-title

    const thirdTitle = document.getElementById('third-title');

    thirdTitle.innerText = 'Title Updated By JS'; // title update

    thirdTitle.style.color = 'purple'; // color wil be set

    thirdTitle.style.backgroundColor = 'aqua'; // Background color

    thirdTitle.style.padding = '10px'; // padding wil be set

    thirdTitle.style.marginLeft = '20px'; // margin-left wil be set

</script>

================================================================

Video-6: Explore 'getElementsByClassName' and 'querySelectorAll'

================================================================

# 1. Explore 'getElementsByClassName'

// \*\*\*\* browser console ==>

    1. // document.getElementsByClassName('blog') // --> [to access this 'blog' class.]

    2. // document.getElementsByClassName('blog').style //

    browser recommends all CSS properties. [Names are in camelCase]

# Explore 'getElementsByClassName'

// javascript: \*\* to make change from document [getElementByClassName]

    <script>

        const blogs = document.getElementsByClassName('blog');

        # // to see the blog class serially in console

        for (const blog of blogs) {

            console.log(blog);

        }

        # // to add the style on blog

        for (const blog of blogs) {

            blog.style.border = '2px solid red';

            blog.style.borderRadius = '10px';

            blog.style.margin = '10px';

            blog.style.padding = '10px';

        }

    </script>

# 2. \*\* CSS selector [querySelector]

// \*\*\*\* browser console ==>

    1. // document.querySelector('.blog') // --> here we got one 'blog' class

    2. To get all blog class we need to use 'querySelectorAll'

        // document.querySelectorAll('.blog') //

    3. To get the 'paragraph' of id="second-blog"

        // document.querySelectorAll('#second-blog p') //

# [Summary]

\*\*\*\*\* Five Things to know: [To access html document by JS]

    '1'. To access by 'tag' Name: 'getElementsByTagName'

    Ex-

    \* document.getElementsByTagName('p') --> To access all <p> tags

    \* document.getElementsByTagName('article') --> To access all <article> tags

    \* document.getElementsByTagName('h3') --> To access all <h3> tags

    '2'. To access specific element by ID Name: 'getElementByID'

    Ex-

    \* document.getElementByID('second-title')

    \* document.getElementByID('second-blog')

    '3'. To access class by Class Name: 'getElementsByClassName'

    Ex-

    \* document.getElementsByClassName('blog')

    \* document.getElementsByClassName('blogs')

    '4'. To select class or ID by CSS selector: 'querySelector'

        // document.querySelector(); //

    Ex-

    \* document.querySelector('.blog') --> To access one 'blog' class.

    '5'. To select class or ID by CSS selector: 'querySelectorAll'

        // document.querySelectorAll(); // to get all matchings

    \* document.querySelectorAll('.blog') --> To access all 'blog' class.

    \* document.querySelectorAll('#second-blog p') --> To access 'second-blog' ID and its <p> tag.

// ================================================================

//       Video-7: NodeList, htmlcollection, setAttribute,

//                   parentNode, childnodes

// ================================================================

[To get Class We use two things and they show the same result but there object name is different. They are- HTMLCollection & Nodelist]

# 1. HTMLCollection:

// Browser Console ==>

    // document.getElementsByClassName('blog') > Enter //

        we get all blog class on an object which name is 'HTMLCollection'

    \* The Result is:

        HTMLCollection [0: article.blog, 1: article#second-blog.blog, 2: article.blog, 3: article.blog, length: 4, … ]

# 2. Nodelist:

Browser Console ==>

    // document.querySelectorAll('.blog') > Enter //

        we get all blog class on an object which name is 'NodeList'

    \* The Result is:

    NodeList(4) [ article.blog, article#second-blog.blog, article.blog, article.blog]

# 3. // To access attribute of any element

// \*\*\*\* browser console ==>

    1. // document.getElementByID('fourth-title').getAttribute('id') // --> [to access this IDs Attribute ('id')]

// javascript

\*\*\* In document file we will declare variable to call function

<script>

    # // getAttribute

    const fourthTitle = document.getElementById('fourth-title');

    fourthTitle.getAttribute('id'); // id name পাওয়া যাবে

    fourthTitle.getAttribute('style'); // style গুলো পাওয়া যাবে

    # // set attribute

    fourthTitle.setAttribute('title', 'I added a tooltip using JS');

    // \*\*\*\* 'In browser console we also declare variable and call function to check but to apply on document we need to do program and save on document file.' \*\*\*\*\*

    # // parentNode : by this I can check the parentNode of any element

    fourthTitle.parentNode // to see the parentNode of fourthTitle

    fourthTitle.parentNode.parentNode // to see the parentNode of parentNode.

    fourthTitle.parentNode.parentNode.parentNode

    # // childNodes

    const secondBlog = document.getElementById('second-blog');

    secondBlog.childNodes # // nodeList is shown

</script>

// ================================================================

//  Video-8: Create HTML elements using Javascript and appendChild

// ================================================================

# create and add element by using JS [createElement()]

// javascript

<script>

    # // create element [createElement()]

        const li = document.createElement('li');

        li.innerText = 'Blog-5'

    # // add element which are created [there are two steps]

        # 1. where to add

        '[ We need to parent ('ul') of the list

        to add our new 'li'.

        to get element we use getElementByID]'.

        const ul = document.getElementById('blog-list');

        # 2. append new element [add element]

        ul.appendChild(li); // set as new 'li' of 'ul'

    # \*\*\* Add article

        # // Bangla - system

        const article = document.createElement('article');

        article.classList.add('blog'); // add blog class

        /\* const h3 = document.createElement('h3');

        h3.innerText = 'My Awesome Blog - 5';

        const p = document.createElement('p');

        p.innerText = 'Lorem ipsum dolor sit amet consectetur, adipisicing elit. Alias vitae distinctio asperiores accusamus'; \*/

        # // add child for article

        /\* article.appendChild(h3);

        article.appendChild(p); \*/

        /\* 'we can make article by innerHTML `` [add carret. Under the ESC key] - May be it can done by string also but we need to use string in every line. carret can be used at a time once' \*/

        article.innerHTML = `

        <h3>My Awesome Blog - 5</h3>

        <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Alias vitae distinctio asperiores accusamus</p>

        `

    # now article add in blog section [parent of article]

        const blogSection = document.getElementById('blogs');

        blogSection.appendChild(article);

    'we can add here the blog style code to apply same code on this article. [just copy the previous code and set here]'

    # to add the style on blog [explore getElementsByClassName part] [এটা আপাতত solution]

        for (const blog of blogs) {

            blog.style.border = '2px solid red';

            blog.style.borderRadius = '10px';

            blog.style.margin = '10px';

            blog.style.padding = '10px';

        }

</script>

================================================================

//      Video - 9 : Module summary and DOM Review

================================================================

# Summary

\*\* HTML document এর এই element / tag গুলোকে আমরা JS দিয়ে access করতে

পারি, read করতে পারি, modify করতে পারি, add করতে পারি।

Total document টা কে তাই Document Object Model ('DOM') বলে। কারণ সব

element কে আমরা access করবো document দিয়ে।

# \*\* javascript আমরা দুইভাবে add করতে পারি।

1. <body> tag এর last tag হিসেবে <script> tag add করে সেখানে javascript

লিখতে পারি। একে internal javascript বলে।

2. <script:src> দিয়ে external javascript add করে সেই javascript file

এ আমরা javascript লিখবো।

বাকিটা 'js file' এ

# Practical

// HTML

<ul id="friends">

    <!-- emmet shortcut: li{friend-$}\*10 -->

    <li>friend-1</li>

    <li>friend-2</li>

    <li>friend-3</li>

    <li>friend-4</li>

    <li>friend-5</li>

    <li>friend-6</li>

    <li>friend-7</li>

    <li>friend-8</li>

    <li>friend-9</li>

    <li>friend-10</li>

</ul>

<section>

    <p>'paragraph - 1'</p>

    <p>'paragraph - 2'</p>

    <p>'paragraph - 3'</p>

    <p id="special-blog">'paragraph - 4'</p>

    <p>'paragraph - 5'</p>

</section>

// javascript

<script>

# 1. access all paragraphs by tag name

    const blogs = document.getElementsByTagName('p');

    // console.log(blogs); // to see the paragraphs in console

    [optional console]

    # // here we can also apply for loop to get blogs

    for (const blog of blogs) {

    // console.log(blog);

    # // apply style on blogs

    blog.style.backgroundColor = 'navy';

    blog.style.color = 'white';

    # // margin, padding, border, radius add করা যাবে।

    }

# 2. id দিয়ে ধরে নিয়ে আসা যাবে [natus এ id দেয়া হলো special-blog]

    const natus = document.getElementById('special-blog');

    # // natus এর style, inner text change করা যাবে।

    # // change inner text

    natus.innerText = 'New Natus';

    #// change inner html

    natus.innerHTML = `

    <h4>Special inner HTML</h4>

    <div>

        <ul>

            <li>First list</li>

            <li>Second list</li>

        </ul>

    </div>

`

# // 3. class দিয়ে ধরে নিয়ে আসা যাবে 'getElementsbyClassName'

# 4. console এ document.getElementsbyTagName('p')

দিলে paragraph গুলো দেখাবে HTMLCollections দিয়ে

কারণ সে html element গুলোকে access করছে।

# 5. console এ document.querySelectorAll('p')

দিলে paragraph গুলো দেখাবে NodeList দিয়ে

html tag ছাড়াও অন্যান্য element কে document এ Node ধরা হয়।

তাই CSS selector এদেরকে nodelist দিয়ে নিয়ে আসবে।

# 6. element remove করা। parent টা কে দিয়ে child কে access করে তাকে remove করতে হবে।

    const friends = document.getElementById('friends');

    // console.log(friends.childNodes); // যত childNodes আছে সব দেখাবে

    console.log(friends.children); // শুধু child html দেখাবে

    const fifth = friends.children[4];

    console.log(fifth); // check the fifth 'li' is selected

    friends.removeChild(fifth); // to remove fifth child from friends

# 7. element add করা। [index অনুযায়ী add না করে last element হিসেবে add]

    // step 1: create element

    const friend = document.createElement('li');

    // step 2: set inner text

    friend.innerText = 'friend-11';

    // step 3: add  / append on ul [parent Node]

    friends.appendChild(friend);

</script>

# // important things to know from this module - 12:47 [video]