**What is window object ?**

The window object represents an open window in a browser . it is browser’s object( not JavaScript object) and it is automatically created by browser. It is a global object with lots of properties and method. like alert(),prompt()….

console.dir(Window); //Access the window object.

console.dir(window.document); // Access the document

**HTML DOM (Document Object Model)**

When a web page is loaded , the browser creates a Document Object Model of the page.

The HTML DOM model is constructed as a tree of Objects:

DOM

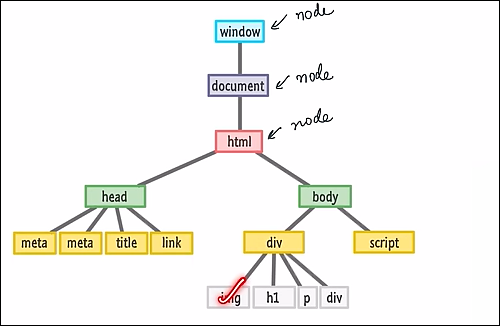
When we write a HTML code attached with JavaScript we can access all the code of HTML inside the JavaScript.

How we will access the HTML code

Automatically all the HTML code convert in object inside the JavaScript and objects are known as document(document is model or Representation of HTML code and this representation of code look like tree) of HTML code ) and this document available inside the window object as a sub document of window object because in window object already exist a document. And we can access all the HTML code through the sub document.

With the HTML DOM, JavaScript can access and change all the elements of an HTML document.

In other words  The HTML DOM is a standard for how to get, change, add, or delete HTML elements.



**How we will access HTML in JavaScript**

Console.dir() is used to print properties and method of special object , special object like document object

console.dir(Window); //Access the window object.

console.dir(window.document); // Access the document

console.dir(document.body)  // print all body object

console.log(document.body) //print the all body elements of html

console.dir(document.head)  // print all head object

console.log(document.head) //print the all head elements of htm

**Benefit of creating individual file of HTML, CSS and JavaScript**

* Improve the readability of code.
* Promote the molecularity
* Browser caching means decrease the re-load time. When we load the js file browser automatically save these file . next time when we re-load the web page browser use save js file .
* When we load script tag before the body tag in that case DOM is not accessible .so that is the meager reason why we write a script tag just before the ending of body tag

**We use DOM for dynamic manipulation in our web page.**

DOM Manipulations (Accessing are element)

Suppose you open a web page in any browser and you want to change something like Styling and content related. on that page in that case we can not do any changes in HTML and CSS file but we can dynamically change or manipulation on that page using  DOM. in other word we use DOM for dynamic manipulation or changes in our web page.

                          H**ow to do DOM Manipulations**

**Step :1 Access the HTML elements**

way of accessing  elements from the HTML page

1. Selecting with id (Id has always unique name for all elements)

    let Heading =  document.getElementById("Heading");

       // if id elements(id name) is not exist in that case return null

       console.log(Heading)

       console.dir(Heading)

2. Selecting with class (Class Name may or may not be same for multiple elements)

getElementsByClassName() returns a HTML collection (HTML collection is similar to an array object but not array) If class elements(class name) is not exist in that case return empty HTM collection

const c=document.getElementsByClassName("navItaem")

console.log(c)

console.log(c[2]) // get the element from array like object

console.log(typeof c)// check type of

console.log(Array.isArray(c))// check is it array or not

3. Selecting with tag name.

  let container = document.getElementsByTagName("div");

        console.log(container);

        console.dir(container);

4.Query Selector

* it is the best way and frequently usable method for accessing elements from the HTML page
* If we use Query Selector then we can select element with tag name ,id name and class name.
* Query selector has two method to access HTML element with tag name ,id name and class name.

1. querySelector(): It is use to select the first element from the HTML page which comes first. whatever we passed inside the querySelector().

let FirstMethdo=document.querySelector("p");// with tag name

            console.log(FirstMethdo);

            console.dir(FirstMethdo);

2. querySelectorAll() :It is use to select all element from the HTML page. whatever we passed inside the querySelectorAll().

//document.querySelectorAll return  the Node(tag) list

const  navItem=document.querySelectorAll(".navItaems")

console.log(navItem)

console.log(navItem[2]) // get the element from array like object

console.log(typeof navItem)// check type of

console.log(Array.isArray(navItem))// check is it array or not

// Add , Remove , toggle class from ClassList

// Toggle is basally check if the class is already exist then it remove that class otherwise add this class.

const SelectClass=document.querySelector(".nav")

console.log(SelectClass)

SelectClass.classList.add("bg-color");

SelectClass.classList.remove("bg-color");

SelectClass.classList.toggle("bg-color");

SelectClass.classList.toggle("bg-color");

//We can also target HTML element with these  method.

// it return HTML page document

let content=documen

console.log(content)

// It return all HTML tag in form array list.

let content1=document.all;

 console.log(content1)

// we can access particular element through index number

let content3=document.all[5];

console.log(content3)

// we can access particular element through index number

 let content4=document.body;

 console.log(content4)

// we can access particular element through index number

let content5=document.head;

 console.log(content5)

// we can access particular element through index number

 let content6=document.title;

console.log(content4)

// we can access particular element through index number

let content7=document.links;

console.log(content7)

// INSERT ELEMENTS THROUGH JAVA SCRIPT ON HTML PAGE

// IF WE WANT TO INSERt(ADD) ELEMENT ON HTML PAGE FIRST WE HAVE TO CREATE THAT ELEMENT.

// Create node

    let btn=document.createElement("button");// Create node

    btn.innerText= "Click Me";

    let div=document.querySelector("div"); // Call the node where to add created node.

    div.append(btn);//node.append(element): Add at the end of the node (inside)

    let btn2=document.createElement("button");

    btn2.innerText="Submit";

    let container=document.querySelector("div");

    container.prepend(btn2); // node.prepend(element) :Add at the begin of the node (inside)

     let box=document.createElement("div");

     box.innerText="Circul";

     box.style.width="50px" ;

     box.style.height="50px" ;

     box.style.backgroundColor="green";

     box.style.border="2px solid red";

     box.style.borderRadius="100%";

     let cant=document.querySelector("div");

     cant.before(box); // node.before(element) :Add at the befour of the node (outside)

    let patila=document.createElement("table");

    patila.innerText="Studen Info Table";

    patila.style.height="200px";

    patila.style.width="200px" ;

    patila.style.backgroundColor="green";

    let info=document.querySelector("div");

    info.after(patila); // node.after(element) :Add at the after of the node (outside)

    //node.remove()// remove the node

    let x=document.querySelector("p");

    x.remove();

  //read about that

    //apendChild()

    //removeChild()

**Step-2 :After Accessing the html element we use some properties for checking(get) the value and changing(set) the value.**

1.tagName : Returns tag name for an element nodes.

  let head=document.querySelector("#Heading");

    console.log(head);

    console.dir(head);

    console.log(head.tagName);

2.innerText:The innerText property returns only the visible text content of an element and all its child .

     let info=document.querySelector(".info");

     console.log(info);

     //get the text

     console.log(info.innerText);

     //set the text

     info.innerText="This is my paragraph";

4.textContent: The textContent properties return visible text and  hidden text of element and its chield.

    let hide=document.querySelector("#Heading");

    console.log(hide);

    console.log(hide.innerText);

    console.log(hide.textContent);

3.innerHtml:The innerHTML property returns The text content and all spacing and inner HTML tags of an element and all its child.

     let info2=document.querySelector("#SecondBox");

     console.log(info2);

      //get the text with tag.

     console.log(info2.innerHTML);

      //set the text with tag.

     info2.innerHTML="<div> hello</div>";

**Read about first child, last child and Children’s( these are used for navigate the element) and also read about Text node, Comment node and element node**

**DOM Manipulations**

// getAttribute(): it is use to get the attribute value of any Node.

       let c=x.getAttribute("id");

       console.log(c);

       let h=head.getAttribute("class");

       console.log(h);

// setAttribute(atr,val): it is use to set the attribute value of any Node.

       let p=head.setAttribute("class","NewHeading");

       console.log(p);

// get and set attribute

const  img=document.querySelector("img")

console.log(img);

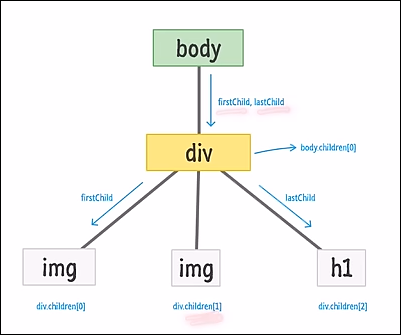
console.dir(img)

const get=img.getAttribute("alt");

console.log(get)

const set=img.setAttribute("alt", "A beautifull Girl standing with Umbrala")

console.log(set)

****

With the object model, JavaScript gets all the power it needs to create dynamic HTML:

* JavaScript can change all the HTML elements in the page
* JavaScript can change all the HTML attributes in the page
* JavaScript can change all the CSS styles in the page
* JavaScript can remove existing HTML elements and attributes
* JavaScript can add new HTML elements and attributes
* JavaScript can react to all existing HTML events in the page
* JavaScript can create new HTML events in the page

Notes