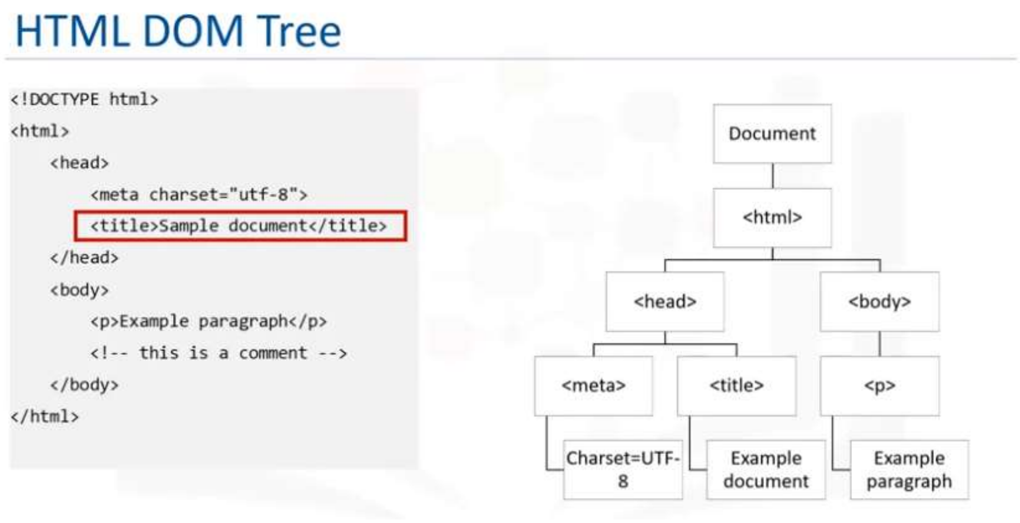
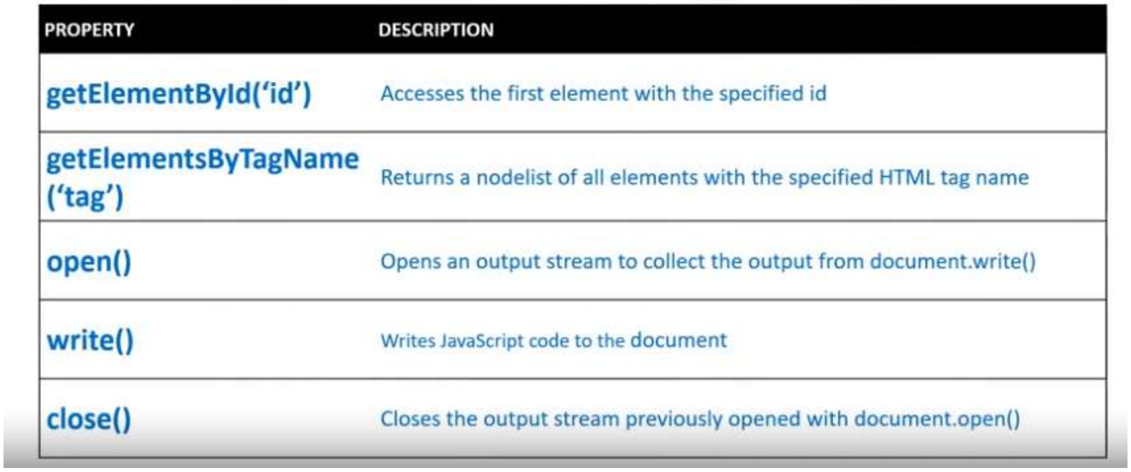
- Why HTML (Hyper Text Markup Language)
- Why HTML DOM Tree.

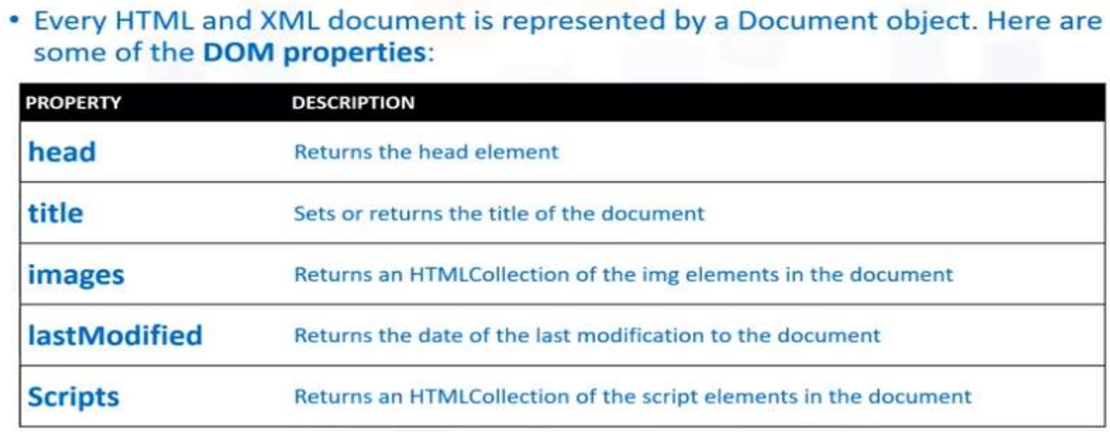


HTML DOM Tree

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="utf-8">
        <title>Sample document</title>
    </head>
    <body>
        <p>Example paragraph</p>
        <!-- this is a comment -->
    </body>
</html>
```



PROPERTY	DESCRIPTION
getElementById('id')	Accesses the first element with the specified id
getElementsByTagName ('tag')	Returns a nodelist of all elements with the specified HTML tag name
open()	Opens an output stream to collect the output from document.write()
write()	Writes JavaScript code to the document
close()	Closes the output stream previously opened with document.open()



- Every HTML and XML document is represented by a Document object. Here are some of the **DOM properties**:

PROPERTY	DESCRIPTION
head	Returns the head element
title	Sets or returns the title of the document
images	Returns an HTMLCollection of the img elements in the document
lastModified	Returns the date of the last modification to the document
Scripts	Returns an HTMLCollection of the script elements in the document

- Using HTML or XHTML (They are similar but XHTML has XML tag on top)

HTML VS XHTML

Use same tags

SGML application / XML application

Not case sensitive / Case sensitive

HTML requires a lenient HTML-specific parser / XHTML parse with a standard XML parser

- HTML scripting is enabled in a browser context when these conditions apply:
    - A support scripting Browser
    - User has not disabled scripting for the current browser context
    - No sandboxed browsing content flag set
- HTML document —--> turn on scripting —> document object
- All **HTML document**s
    - start with the declaration. This line tells the web browser that the file contains HTML code. This is not an HTML tag. `<!DOCTYPE html>`
    - An HTML file begins with the <html> tag and ends with </html> tag.
    - The HTML document has a head and a body
    - Title define in head section
    - H1 tag type in body section

- How to use CSS (Cascade style sheet)
    - First, we choose layout
        - Fluid layout
        - Fixed layout
    - Apply CSS to HTML
        - Using the <style> tag
        - Using the link tag <link href = “/styles/style.css” rel =”stylesheet”>

(Universal selector in CSS is *)

- **Properties of CSS**: Inline CSS has the highest priority, then comes Internal/Embedded followed by External CSS which has the least priority. Multiple style sheets can be defined on one page. If for an HTML tag, styles are defined in multiple style sheets then the below order will be followed.
    - Inline CSS
    - Internal or Embedded CSS
    - External CSS

- Identify the basic **structural elements** of HTML5.

    HTML5 use division-based layouts that separates areas in a document into divisions

        - <header>: Used to contain the header of a site.
        - <footer>: Contains the footer of a site.
        - <nav>: Contains the navigation functionality for the page.
        - <article>: Contains a standalone piece of content that would make sense if syndicated as an RSS item, for example a news item.
        - <section>: Used to either group different articles into different purposes or subjects, or to define the different sections of a single article.
        - <time>: Used for marking up times and dates.
        - <aside>: Defines a block of content that is related to the main content around it, but not central to the flow of it.
        - <hgroup>: Used to wrap more than one heading if you only want it to count as a single heading in the page's heading structure.
        - <figure> and <figcaption>: Used to encapsulate a figure as a single item, and contain a caption for the figure, respectively.

- Describe the attributes of HTML5 **input elements**.
    - <input type = “date”/>
    - <input type = “color”/>
    - <input type = “datetime-local”/>
    - <input type = “email”/>
    - <input type = “number” min = “5” max = “10”>
    - <input type = “range” min = “5” max = “10”>
    - <input type = “search/>
    - <input type = “tel” pattern=”[0-9]{3}-[0-9]{3}-[0-9]{4}”>
    - <input type = “URL”>

- Describe the basic syntax of JavaScript.
- Explain how to work with variables in JavaScript.
- Describe how to control the flow of a JavaScript application with control statements.
- Outline the use of functions and prototypes in JavaScript.
- Explain the use of client-side JavaScript with HTML.
- Describe how JavaScript integrates with the document object model (DOM).
- List the common APIs used in JavaScript Applications.
- Validate user input on an HTML5 form by using JavaScript.
- Develop and run JavaScript on the browser console.
- Create and use variables, conditional statements, loops, and methods in JavaScript.

JavaScript

JavaScript is a loosely typed and dynamic language. Variables in JavaScript are not directly associated with any particular value type, and any variable can be assigned (and re-assigned) values of all types

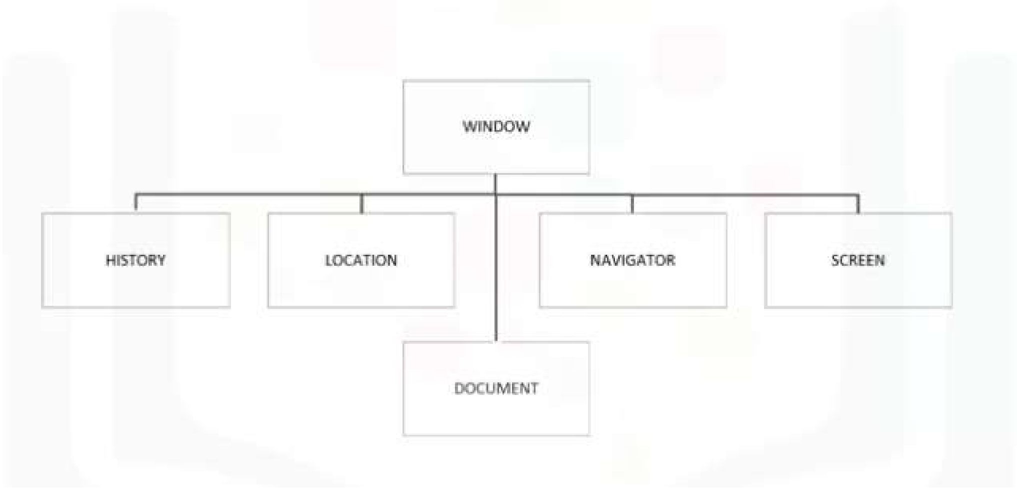
JavaScript data type (Dynamic, Object)

- Create custom **object** by create a **function**
- **Prototype** allows user to add a new function to the template for the object
- **Self-Executing functions** start running immediately after declared and often used to initialized data or to declare DOM elements

Client side script is a program that accompanies an HTML document or may be embedded directly in it, <script/> <noscript/> and it can bound to an event

Document object model (DOM)

Is the interface between HTML or XHTML and JAVASCRIPT



JavaScript DOM Object