HTML

(Hyper Text Markup Language**)**

- HTML is used to design web page.
- Web Page is a Hyper Text Document that provide UI for interacting with resources.
- Hyper Text: The term Hyper is derived from Greek term which means “Beyond”
- Markup: It is general computer terminology derived from “Marking up”.
- Marking up is the process of preparing information to present according to requirement.
- Markup Language is a language used for Presentation.
- HTML is a presentation language.

- HTML is used to present information on browser.

FAQ: What is difference between programming language and markup language?

- Programming language handles user interactions by dynamically accepting input from user, process the request and generate an output.
- Presentation language can used only for presenting content.

Evolution of Markup Language:

- Internet start with a browser called “Mosaic”.
- The early markup language used for internet were “GML & SGML”

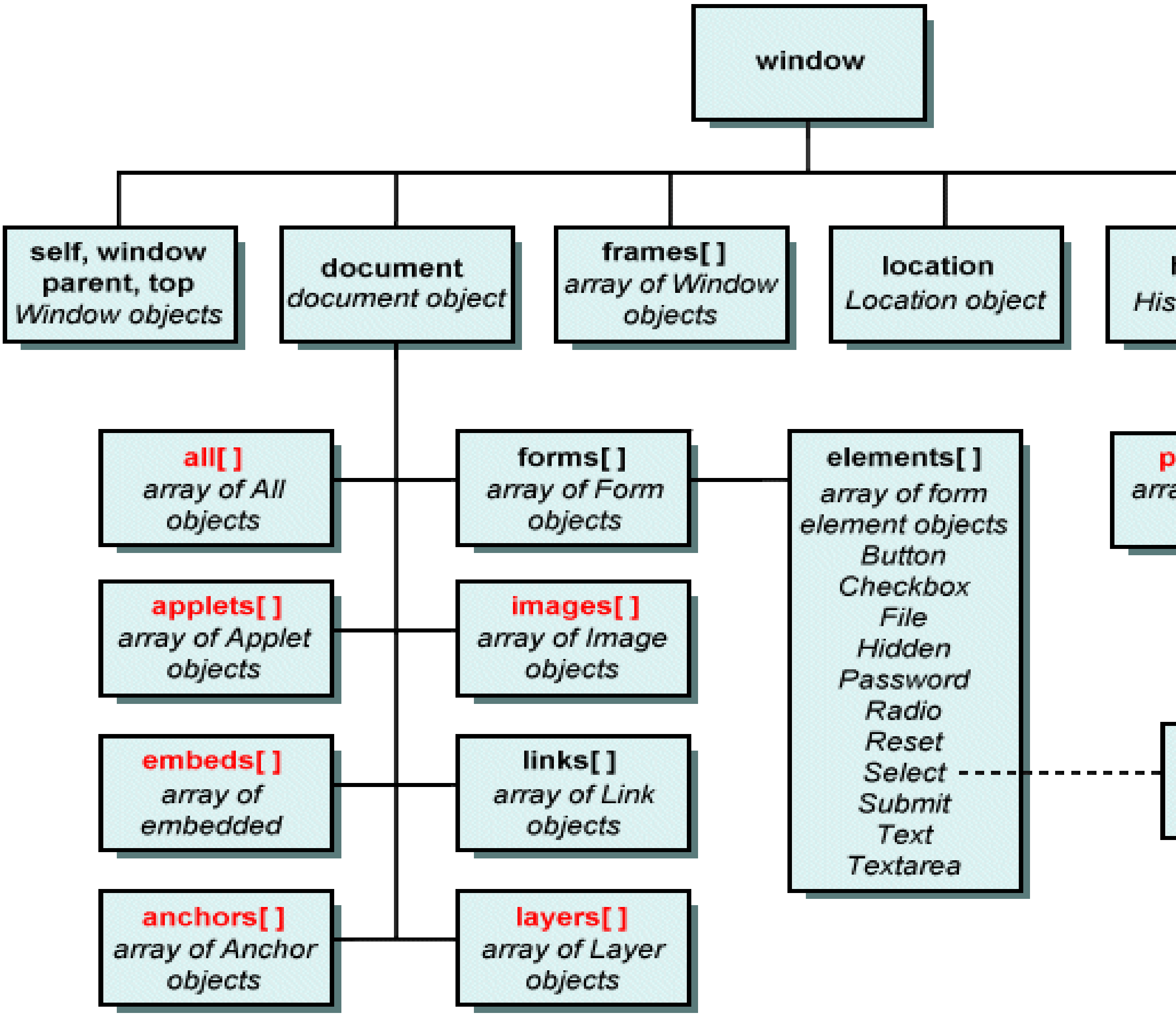
- Generic Markup Language and Standard Generic Markup Language.
- These languages were used for presentation on a browser called “Mosaic”.
- In the early 1990’s at CERN [Council for European Research and Nuclear] labs **“Tim Berners Lee”** introduced a language called “HTML”.
- Tim Berners Lee introduced the concept of Web. [Father of web]
- HTML is super set to GML and SGML.
- **1995** IETF [Internet Engineering Task Force] developed HTML version **HTML 2.0**

- **1997** W3C [World Wide Web Consortium] developed **HTML 3.2** [Jan-1997]
- **1997** December W3C developed **HTML 4.0**
- **2004** WHATWG [Web Hypertext Application Technology Work Group] started contributing to HTML along with W3C.
- **2014** W3C developed **HTML 5.** Supported by 2 groups: W3C & WHATWG

What HTML Comprises of?

- HTML is a markup language that comprises of elements.
- HTML presents everything by using elements.

- Elements are presented in a hierarchy called “DOM” [Document Object Model]



- HTML elements are classified into 5 groups
  - ○Normal Elements

- ○ Void Elements
- ○ RC Data Elements
- ○ Raw Text Elements
- ○ Foreign Element

How website starts?

- Every website by default starts with a page called “index.html”
- If there is no “index.html” then you have to manually request any page.
- Index.html is the start-up page for website.

Ex:

- Go to your project in VS code
- Add a new file by name “index.html”
- Write some text:

"Welcome to HTML"

- Request your website from browser
  http://localhost/fullstackweb
- This will automatically load index.html
- If you want to manually access any another page, then you can request directly in URL.
  http://localhost/fullstackweb/home.html

What should be the extension for Static page?

- You can define static page with extension
  - .html
  - .htm

What is difference between HTML & Htm?

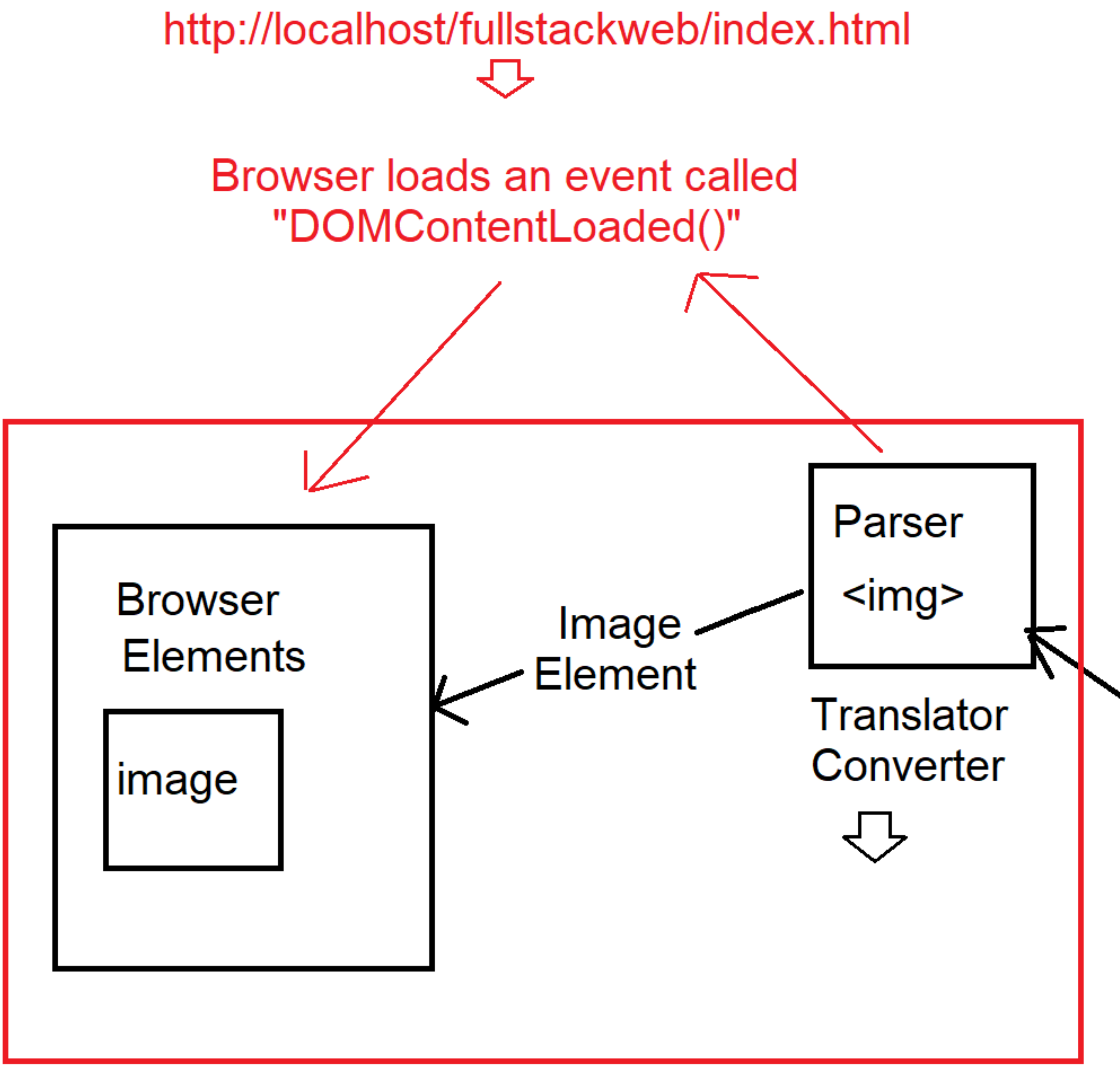
- Technically both are same.
- Always recommended to define extension “.html”
- “.htm” is the extension often given by tools that can publish webpages.

Ex:

- Open Ms-Word
- Write some text in a file.
- Save as type “Web Page” with name “welcome”
- It will save the file by name “welcome.htm”

What are Elements and Tags?

- HTML is a collection of elements arranged in a hierarchy called DOM.
- Browser can understand only Elements.
- The presentation in HTML is defined by using Elements, like Images, Anchor, Form etc.
- Elements are designed by using 2 techniques
    - ○Dynamically
    - ○Statically
- Statically elements are presented by using “tag”
- “Tag” is used to build and present elements statically.



http://localhost/fullstackweb/index.html

Browser loads an event called "DOMContentLoaded()"

Browser Elements

image

Image Element

Parser

<img>

Translator Converter

- You can also present element dynamically.
- You can build element dynamically

Types of Elements in HTML

- HTML elements are classified into following types
  - ○Normal Element
  - ○Void Element
  - ○RC Data Element
  - ○Raw Text Element
  - ○Foreign Element

Element Type	Description
Normal Element	- A normal element directly returns some presentation directly on call back. - It starts returning the presentation and will never stop. - Usually it stops only at the end of

	document. - These types of elements are designed by using Start Tag <tagname> End Tag </tagname> Ex: <b> </b>
Void Element	- Terms void defines no-return value. - Void element will not return anything directly on call back. - You have to define what to return by using attributes. - It will return only the content that you asked to. And stops automatically.

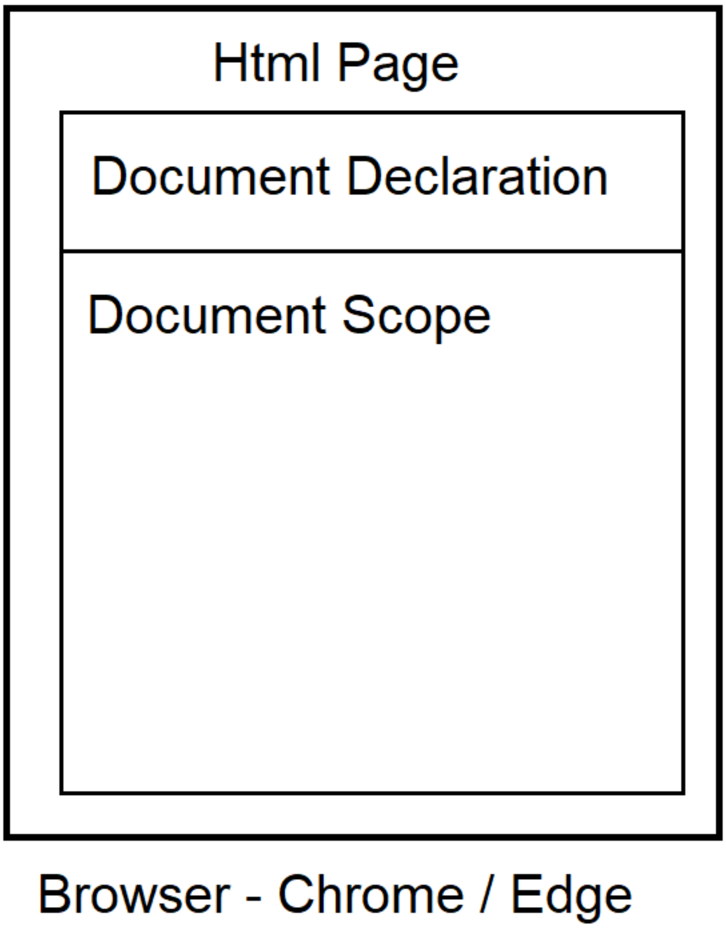
	- These elements don’t require and end tag. - They are self-ending elements Ex: <img>
RC Data Element	- Rich Content elements. - The elements are used for presenting text. - You can’t embed any another content with in the context. - You can use only for text without any formats. - Any content you defined will be treated as plain text. Ex: <textarea>

	</textarea>
Raw Text Element	- It is an HTML element or literal [plain text, code, or symbol]. - It is presented without using a tag. - It uses raw text for presentation. - The raw text elements are defined with < > or &. Ex: &#8377; &copy; ©
Foreign element	- It is HTML element but not native to HTML. - These elements require additional library, browser can't

	understand these elements directly. - You have to import library or define a plugin in order to use that element. Ex: SVG, Canvas, MathML

Structure of HTML Page

- HTML page at high level comprises of two sections.
  - ○Document Declaration
  - ○Document Scope



Document Declaration

- Every markup document starts with document declaration.
- The document declaration comprises of information about the document given to browsers parser [Translator].

- The information comprises of details like
    - Culture of document
    - Language of document
    - Encoding of document
    - Version of document etc.
- The document declaration is for parser not to display in browser.
- HTML document declaration specifies the language and version.
- Document declaration in HTML is defined by using the following

  <!DOCTYPE html>

  W3C.org, WHATWG

Document Scope

- Every browser can display multiple documents merged into one body section.
- You can display multiple documents into one body.
- Every document scope is defined by using “<html>” element.

  Syntax:

  ```
  <html>
    Document root scope
  </html>
  ```
- It is also known as document root scope.
- It is technically referred as “:root”.

Ex:

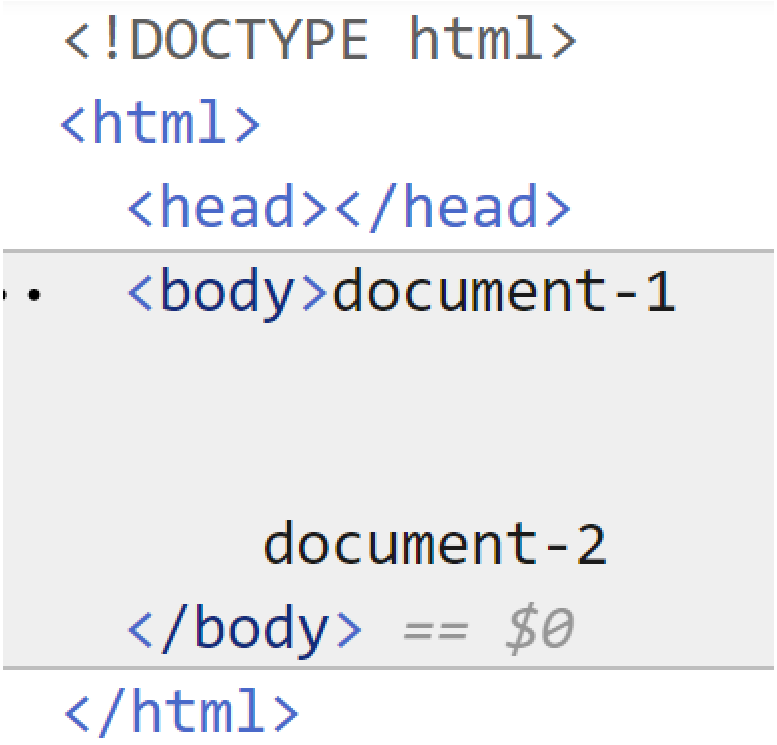
```
<!doctype html>

<html>

  document-1
```

```
</html>
<html>
    document-2
</html>
```

O/P:



```
<!DOCTYPE html>
<html>
  <head></head>
  <body>document-1


        document-2
  </body> == $0
</html>
```

- Every document scope comprises for information about which culture related content it is displayed.

- The cultures can be like “en-US, en-IN, en-UK etc.”
- The culture for HTML page is defined by using “lang” attribute.
- Culture specifies the display format of numbers, currency, date and time etc.

Syntax:

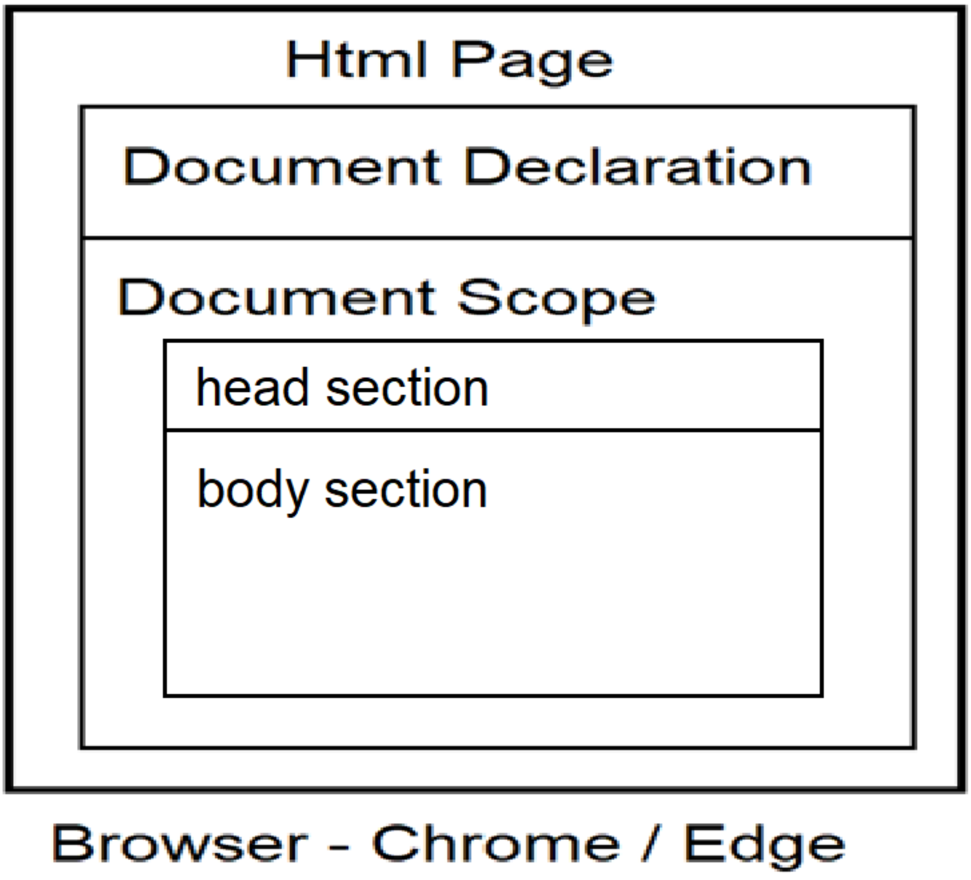
```
<html lang=”en-in”> </html>
```

Ex:

```
<!doctype html>
<html lang="en-in">
</html>
```

Structuring Document Scope

- At high level the document scope comprises of 2 sections.
  - ○ Head Section
  - ○ Body Section



Head Section:

- It comprises of content that is intended to load into browser memory and give access later to the browser or page.

- It is defined by using <head> element.
- Typically, HTML head section comprises of contents like
  - **Title**
  - **Link**
  - **Meta**
  - **Style**
  - **Script**

Syntax:

```
<!DOCTYPE html>

<html>

<head>


</head>

</html>
```