HTML

* What is HTML?
* What is HTML’s role in modern web development?
* Who invented HTML? -- Tim Berners-Lee
* Who oversees the development of HTML? – World Wide Web Consortium
* What are the versions of HTML? -- HTML 5 (Latest)
* Which documents specify the HTML standards?
* What is the syntax of HTML?

HTML- derived from SGML  
 - relationship with XML

Mark Up Language 🡪 HTML  
aspects: -structure and content = HTML  
 -presentation/aesthetics = CSS  
 - behavioural 🡪 JavaScript

Versions of HTML W3C  
🡪 HTML 1.0 (Corsina) - Recommendation (Technical Doc)  
🡪 HTML 2.0 – RFC 1866 - To capture what should be followed  
 - First W3C standard 🡪 HTML 3.0  
 🡪 HTML 3.2  
 🡪 HTML 4.0, 1997  
 🡪 HTML 4.01, 1999  
 🡪 XHTML (January 26, 2008)  
 (Extensible HyperText Markup Language)  
 1.0, 1.1, 2.0   
 HTML 4.01 🡪 Strict, Transitional, Frameset

Ian Hickson et. Al

4.01 🡪 Evolve it!  
 🡪 WHATWG  
 🡪 HTML5 – 2014  
WHATWG- Web Hypertext Application Technology Working Group

What is the syntax of HTML?

1. <!DOCTYPE html> 🡪 triggers standard compliance rendering   
   Two modes: “standards”  
    “quirks”
2. <html> 🡪 Root element
3. <head> 🡪 meta-info
4. <body> 🡪 doc content </body>  
   (start tag) (content) (end tag)

void element 🡪 no closing tag

Index of Elements

DOM 🡪 Dominant Object Model (extends in Java)

XHTML 🡪 </img>

HTML Elements:

Html

* Head  
  Title, base, link, meta, style hr – thematic break
* Body

Article, aside, now, section

Header, footer figure-figcaption

Main

Address

div

h1, h2, h3, h4, h5, h6

p

hr

pr

blockquote

ol, ul

li

dl, dt, dd

a

em, strong, small, s

cite, q

dfn, abbr

ruby, rb, rt, rte, rp

data, time

code, var, samp, kbd

sup, sub

I, b, u, mark

bdi, bdo

span

br, wbr

ins, del

table

caption

colgroup, col

thead, tbody, tfoot

tr

th, td

form

label

input

HTML Attributes:

Global Attributes

Element – Specific Attributes

Data- \*attributes

Role, aria- \*Attributes

WAI – ARIA WCAG

Cascading Style Sheets (CSS)

* Language used to specify the presentation aspects (e.g., layout and fomatting) of structurally marked up documents
* Developed by Hakon Wium Lie (CHSS) and Bert Bos (SSP)

-resources

Centralized – CSS 1 (1996)

Rep—CSS 2.1

Specs- CSS preprocessors, CSS Frameworks  
 - Sass, Less, 960 Grid System, Bootstrap, Foundation, Materialize, etc.

HTML/XHTML Stylesheets

* Author styles- external stylesheets (recommended), embedded styles, inline styles
* User styles
* User agent-styles (example default CSS 2.1 swtylesheet for HTML 4)

\*Reset CSS- CSS that defines initial CSS

CSS Statements

* At- Rules

-@charset

-@import

-@media

-@font-face

-@keyframes

-@page

* CSS Rule Sets (a.k.a CSS Rules, Style Rules)
* Consist of a selector

CSS Selectors

Selector

-structure used as a condition in a CSS rule to determine which elements in the document tree are matched by the selector and are thus requested

* Selector syntax:  
   - chain of one or more sequences of simple selectors separated by combinators, with one pseudo- element possibly appended to the lost sequence.
* Sequence of simple selectors

-chain of simple selectors not separated by combinators

-always starts with a type selector or universal selectors

* Group of selectors

-comma –separated list of selectors

* Simple selectors
* Type selector
* universal selector
* attribute selector—[attr, [attr=value], [attr~=value], [attr1=value],

[attr^=value](CSS3), [attr$=value](CSS3), [attr=value](CSS3)

* pseudo-class

-dynamic pseudo class

* link pseudo class
* :link
* :visited
* User action pseudo classes
* :hover
* :active
* :focus

-target pseudo-class

-language pseudo-class

-language pseudo-class

-UI element states pseudo classes

* :enabled
* :disabled
* :checked
* :indeterminate

-structural pseudo classes

* root
* :first-child()
* :last-child()
* :only:-child()
* :nth-last-child()
* :first-of-type
* :last-of-type
* :only-of-type
* :nth-of-type()
* :nth-last-of-type()
* :empty
* Combinators
* descendant combinators (whitespace such as space, tab, linefeed, carriage, return, form feed)
* child combinator (>)
* Sibling combinators
* Adjacent sibling combinatory (+)
* General sibling combinatory (~)
* pseudo-elements
* :: first-letter :first-letter
* ::first-line : first-line
* ::before :before
* ::after :after
* CSS Rule Precedence

-by origin and importance

* User agent important declarations
* User important declarations
* Author important declarations
* Author normal declarations
* User agent normal declarations

-by specifity

* Inline style
* Number of ID selectors
* Number of class selectors, attribute selectors and pseudo classes
* Number of type selectors and pseudo-elements
* CSS Declarations

-properties

* shortened properties
* vendor specific extensions

-values

* keywords
* numbers
* dimension

-length, angle, duration, frequency, resolution

-length units:

* font-relative: em, ex, eh, rem
* viewport-percentage: vn, vh, vmn, vmax, absolute length
* frequency units: hz, khz
* resolution units
* ……
* ….. (incomplete)
* CSS Preprocessors CSS Frameworks
* Sass (Syntactically Awesome Style Sheets), Less, SCSS, 960, Grid System, bootstrap, foundation, materialize
* …..

JavaScript

Client-side Scripting

* JavaScript (most used)
* Facilitate actions with users
* Interact with the document

Client-side JavaScript

1. External
2. Embedded
3. Inline script

\*engine parses from top

\*If script is encountered during rendering, it gets executed

\*defer while fetching script, continue, rendering of document

\*async-fetch, as soon it is available, execute

\*cannot be used at the same time

<noscript> -can be turned on/off in the browser

-interact with the document

* window.document
* document.getElementById(‘h’)
* h.textConten=’hi’
* h.innerText=’hello’
* h.innerHTML=’ ’
* h.outerHTML
* h.lang or h[“lang”]
* h.data-extra
* h.setAttribure
* h.getAttribute
* h.style=’color:blue’
* h.style.fontSize=’4em’
* document.getElementsbyClassName (class is set)
* document.getElementsbyName
* document.getElementsbyTagNumbers
* document.querySelectorAll(‘p:first-child’)
* p.matches(‘p’)
* document.children
* document.head.childNodes
* document.head.children
* document.body.firstChild
* document.body.firstElementChild
* document.body.firstChild.previousElementSibling
* document.body.parentElement

JS 🡪 object-oriented

var- you are declaring a data type (implicit)

Variables

1. var
2. let
3. const

Function Scope 🡪 Local Variable

\*You cannot create local script variable before Ecma Scripts

\*Constants are constants-no re-assignment allowed

Data types

* Simple
* Primitive

‘falsy’ values- not really false but equals to false

‘truthly’ values – not really true but equal to true

Undefined=false

Null=false

0=false

Empty string=false

typeof- no difference between strings and integers

JavaScript- dynamically typed

Standard Objects

* Array, Boolean, Date, Error, Function, JSON, Math, Number

Expression and Operators- …

Variable declarations

1. “String syntax”
2. Fat arrow syntax- only one argument, you can use drop the parenthesis

\*arguments object, only inside the function

\*function arguments can have default values

\*memory allocation will only occur if you give values to arrays

2 Ways of Making Arrays in JS

1. var emptyArray=newArray(); //using Array object
2. var alsoEmptyArray=[]; //using Array Literal

\*arrays can be any type

Methods on Arrays

1. Mutators

* array.prototype.pop()
* array.prototype.push()
* array.prototype.shift()
* array.prototype.sort()
* array.prototype.splice()
* array.prototype.unshift()

1. Accessors

* array.prototype.concat()
* array.prototype.includes()

1. Iteration Methods

* array.prototype.filter()
* array.prototype.find()
* array.prototype.findIndex()
* array.prototype.forEach()
* array.prototype.keys()
* array.prototype.map()
* array.prototype.reduce()
* array.prototype.reduceRight()
* array.prototype.values()

\*constructor- a function  
 -capital letter convention

Server-Side Scripts Web Scripts

* Java (servlets/JSP)
* PHP
* Node.js
* ASP.Net

Client Side Technology (CST)

* HTML
* CSS
* JS—sandbox execution

NetBeans IDE

* Java SE
* Java ME
* Java EE – focus on Java Web Applications

-Server: Apache Tomcat 8.0.27.0

: GlassFish Server 4.1.1