Client Side Scripting

* HTML
* CSS
* Javascript

HTML – (Hypertext Markup Language)

Webpage – a document that is formatted in a special way

* structure and content
* document in electronic form
* aesthetics or presentation
* behavioral

HTML Versions

* HTML 1.0
* HTML 2.0
* HTML 3.0
* HTML 3.2
* HTML 4.0
* HTML 4.01
* HTML 5.0
* HTML 5.1

XHTML – Extensible Hypertext Markup Language

* It is a reformulation of HTML to XML
* rules in XML are very strict (case sensitive)

XML Parser –

WHATWG – Ian Hickson

* Started this group and develop the next version of HTML since XHTML is not that great because of its strict syntax.

HTML

* Content Model

Categories:

* + Heading
  + Sectioning
  + Embedded
  + Phrasing
  + Flow
  + Interactive
  + Metadata
* Content Attributes
  + Global Attributes
* DOM Interface
* HTML
  + Head
    - Title
    - Base
    - Link
    - Meta
    - style
  + Body
    - Article
    - Aside
    - Nav
    - Section
    - Header
    - Footer
    - Div
    - (h1, h2, h3, h4, h5, h6)
    - P
    - Hr
    - Pre
    - Blockquote
    - ol, ul, li
    - dl, dt, dd
    - figure, figcaption
    - a, em, strong, small
    - dfn, abbr
    - ruby, rb, rt, rtc, 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, check, radio, button
    - embed
    - math, svg, canvas
    - script, noscript
    - templates

HTML Attributes

* Global Attributes
* Element Specific Attributes
* Data - \* Attributes
* Role, aria - \* Attributes

CSS Statements

* CSS Selectors
  + subject
* Selector syntax
  + Simple selectors
    - Type selector
    - Universal selector
    - Attribute selector
    - Class selector
    - ID selector
    - Pseudo class
      * Dynamic pseudo class
      * Link pseudo class
        + :link
        + :visited
      * User action pseudo class
        + :hover
        + :active
        + :focus
      * Target pseudo class
        + :target
      * Language
        + :lang()
      * UI element state
        + :enabled
        + :disabled
        + :checked
        + :indeterminate
      * Structural pseudo class
        + :root
        + :first-child
        + :last-child
        + :only-child
        + :nth-child()
        + :nth-last-child()
        + :first-of-type
        + :last-of-type
        + :only-of-type
        + :nth-of-type()
        + :nth-last-of-type()
        + :empty
      * Negation pseudo class
  + Type selectors
  + Group selector
  + Combinators
    - Decendant combinatory
    - Child combinatory (>)
    - Sibling combinatory
      * Adjacent sibling combinatory (+)
      * General sibling combinatory (-)
    - Pseudo elements
      * ::first-letter, :first-letter
      * ::first-line, :first-line
      * ::before, :before
      * ::after, :after

CSS Rule Precedence

Cascading

* By origin and importance
* By specificity
* By order
* By origin and importance
  + User agent important declarations
  + User important declarations
  + Author important declarations
  + Author normal declarations
  + User normal declarations
  + User agent normal declarations
* By specificity
  + Inline-style
  + Number of ID selectors
  + Number of class selectors, attribute selectors and pseudo classes
  + Number of type selectors and pseudo elements
* By order
  + The last is given the priority

Id value – is unique and important

Pseudo class – ex. first-child

Pseudo elements – ex. before, after

Type selectors – ex. p

CSS Declarations

* Properties
  + All properties has an initial value
* Values
  + Keywords
  + Numbers
  + Dimensions
    - Length, angle, duration, frequency, resolution
    - Length units
      * Font-relative: em, ex, ch, rem
      * Viewport-percentage: vm, vh, vmin, vmax
      * Absolute lengths: cm, mm, q, in, pt, pc, px
    - Angle units: deg, grad, rad, turn
    - Duration units: s, ms
    - Frequency units: hz, khz
    - Resolution units: dpi, dpcm, dppx
  + Percentage
  + URL’s and URI’s
  + Colors
  + Strings
  + Functions: calc(), attr(), counter(), counters(), linear-gradient(), radial-gradient(), translate(), scale(), rotate() etc.

CSS Preprocessor, CSS Framework

* Saas
* Less
* 960 Grid System
* Bootstrap
* Foundation
* Materialize

HTML Preprocessor

* Markdown
* Slim
* Pug

CSS Framework

* Bootstrap
* Materialize

Javascript

* + Externally linked script
  + Embedded script
  + Inline script
  + Externally linked script
    - Defer attribute
    - Async
    - Noscript element
* Global Attribute
* Custom Attribute
  + .setAttribute(`data-extra’, `value’)
  + .getAttribute(`data-extra’)
  + .style `color:blue’
  + .style.visibility = `hidden’
  + .style.fontSize = `3em’
* Document
  + document.getElementById
  + document.getElementByClass
  + document.getElementByName
  + document.getElementByTagName
  + document.getElementByTagNameNS
  + document.querySelector
  + document.querySelectorAll
  + document.childNodes
  + document.children
  + document.childrenElementCount
  + document.head.childNodes

element node – ex.p

text node – ex. link

attribute node – ex. href

DOM – Document Object Model

4 Levels pf DOM

XML

HTML

XHTML

Javascript – imperative

* not a strongly type system
* dynamically type
* weakly type

Java – strongly type and very rigid

Console – is a function but it can be overwritten as a variable

Alert – can also be overwritten

Document – assignment of variable will be ignored silently

Global variable

Const – same behavior with let, difference is it is for constant declaration

Simple / Primitive – Boolean, numbers, string, undefined, null

Structured/ Refer

Truthy – not true but equivalent to true

Falsy – not false but equivalent to false ex. null, undefined, empty string, zero

Standard Objects

* Array
* Boolean
* Date
* Error
* Function
* JSON
* Math
* Number
* Object
* RegExp
* String
* Map
* Set
* WeakMap
* WeakSet

Expression and Operators

* instanceOf
* typeOf
* new
* this

Methods

3 Categories:

* Mutator Method
  + Array.prototype.copyWithin()
  + Array.prototype.fill()
  + Array.prototype.pop()
  + Array.prototype.push()
  + Array.prototype.reverse()
  + Array.prototype.shift()
  + Array.prototype.sort()
  + Array.prototype.splice()
* Accessor Method
  + Array.prototype.concat()
  + Array.prototype.indexOf()
  + Array.prototype.join()
  + Array.prototype.lastIndexOf()
  + Array.prototype.slice()
  + Array.prototype.toString()
* Iteration or Iterator Method
  + Array.prototype.filter()
  + Array.prototype.every()
  + Array.prototype.find()
  + Array.prototype.forEach()
  + Array.prototype.entries()
  + Array.prototype.values()
  + Array.prototype.reduce()
  + Array.prototype.[@@iterator]()