**Midterms Web Tech Content:**

**Cascading Style Sheet (CSS)**

It is a language that is in charge of the layout, format and presentation of a structurally markup document like the HTML and XHTML.

**Brief History (https://www.w3.org/Style/CSS20/history.html)**

Hakon Wium Lei – developed CSS formerly HCSS (HTML Cascading Style Sheet)

Bert Boss – developed the Stream-based Style Sheet Proposal /SSP

**Versions of CSS**

CSS 1

CSS 2

CSS 3. This is the latest version of CSS language that has the goal of extending the features of CSS 2.1

**HTML Stylesheets**

**Sources of Styles:**

1. **Author stylesheets. The person creating the website provides his/her own design**
   1. **External stylesheets (Linked). It can be used across other styles**
   2. **Embedded styles(Internal)**
   3. **Inline styles**
      1. **Persistent**
      2. **Preferred**
      3. **alternate**
2. **User styles.**
3. **user**

|  |  |  |
| --- | --- | --- |
|  | **Normal** | **Important** |
| **Author** | **1** | **2** |
| **User** | **2** | **1** |
| **User Agent** | **3** | **3** |

**CSS Statements**

1. **At Rules**
   1. **@charset**
   2. **@import**
   3. **@media**
   4. **@page**
   5. **@viewport**
   6. **@font-face**
   7. **@key frames**

**CSS RULES SETS (aka CSS Rule)**

**This consists of selectors followed by sequence of elements/attributes you want to target.**

* **CSS Comments /\* \*/**

**CSS Selector**

Selectors. It is used as condition in CSS rule in order to determine without element in the document tree are matched by this selector.

* Universal selector
* Type
* Grouping
* ID
* Class
* Descendant
* Child
* Adjacent Sibling
* Attribute Acronym
* Pseudo classes
  + First child pseudo class
  + Link pseudo class
  + Dynamic pseudo class

Declaration Block

* Properties
  + Visual, aural, paged, interactive
  + Margins, borders and padding
  + Colors and background
  + Fonts
  + Text formatting
  + Lists and tables
  + Positioning and dimensions
  + Miscellaneous
* Values
  + Keywords
  + Numbers
  + Length
    - Relative and absolute
* Percentages
* Colors
* Strings
* URLs
* Counters
* Initial values, inherited values and actual values
  + By origin and importance
    - User important declarations
    - Author important declarations
    - Author normal declarations
  + By Specificity
    - Inline styles (1) or not (0)
    - Number of ID attributes in selector
    - Number of other attributes and pseudo-classes in selector
    - Number of element names and pseudo-element in selector
  + By Order

**Client-side web scripting**

Allows programs to be downloaded directly from the web server and executed in the client envirssonment (e.g. browser)

Common Client Side Technologies

1. JavaScript
2. ECMAScript
3. Jscript
4. ActionScript
5. VBScript

Vendor Specific Extension.

Prefix for the names of experimental or non-standard CSS properties (e.g. Webkit)

The experimental can be enabled or disabled.

Simple Selector

1. Type Selector
2. Universal Selector.
   1. P{}
   2. H1{}
3. Attribute Selector. Targeting a specific attribute which matches the given value inside a []. Example: **img [alt]** – This means that anything that is specifically an image that contains ‘alt’. **a [href $ = ‘.png’]** – This means that anything that inside a that ends with .png.
   1. [attr]
   2. [attr=val]
   3. [attr~=val]
   4. [attr|=val]
   5. [attr^=val]
   6. [attr$=val]
   7. [attr\*=val]
4. class Selector
5. ID Selector
6. Pseudo-class
   1. Dynamic pseudo-class
      1. Link pseudo-class
         1. :link
         2. :visited
      2. User action pseudo-class
         1. :target (css3), :hover, :active, :focus
   2. **Target pseudo-class**
      1. lang()
   3. **UI element states pseudo-class**
      1. **:enabled**
      2. **:disabled**
      3. **:checked**
      4. **:intermediate**
   4. **Structural pseudo classes**
      1. **:root**
      2. **:first-child**
      3. **:last-child**
      4. **:only-child**
      5. **:nth-child**
      6. **:nth-child()**
      7. **:nth-last-child()**
      8. **:first-of-type**
      9. **:last-of-type**
      10. **:only-of-type**
      11. **:nth-of-type**
      12. **:nth-of-type()**
      13. **:nth-last-of-type()**
      14. **:empty**
   5. **Negation pseudo-class**
      1. **:not (css 3)**

**Selector Syntax. It is a chain of one or more sequences of simple selectors separated by combinators.**

1. **Combinators. It is used to additional matching constraints, applied by prepending another sequence of selectors and a combinator to a sequence.** 
   1. **Descendant combinators (white space; space, tab, line feed, carriage, return form feed)**
   2. **Child combinators (>)**
   3. **Sibling combinators**
      1. **Adjacent sibling combinators (+)**
      2. **General Sibling Combinator (~)**
2. **Pseudo elements**
   1. **::first-letter (css 3), :first-letter**
   2. **::first-line (css 3), :first-line**
   3. **::before (css 3), :before**
   4. **::after (css 3), :after**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **id** | **Class attributes** | **Type pseudo** |
| **P** | **0** | **0** | **1** |
| **p#abc** | **1** | **0** | **1** |
| **p.xyz** | **0** | **1** | **1** |
| **div p** | **0** | **0** | **2** |
| **div>p** | **0** | **0** | **2** |

**CSS rule Precedence**

* + By origin and importance
    - User important declarations
    - Author important declarations
    - Author normal declarations
    - User normal declarations
    - User agent declarations
  + By Specificity
    - Inline styles (1) or not (0)
    - Number of ID attributes in selector
    - Number of other attributes and pseudo-classes in selector
    - Number of element names and pseudo-element in selector
  + By Order
    - The latest declarations override the previous declaration.

**CSS Declarations**

1. **Properties**
   1. **Shorthand properties**
   2. **Vendor specific extensions/ vendor prefixes**
2. **Values**
   1. **Keywords**
   2. **Numbers** 
      1. **int**
      2. **real numbers in decimal notation**
3. **lengths**
   1. **length units**
      1. **em**
      2. **px**
      3. **in**
      4. **xm**
      5. **mm**
      6. **pt (point 1/72”)**
      7. **pc (pica 1/6”)**
   2. **percentages**
   3. **URLs and URIs**
   4. **Counter**
   5. **Colors**
   6. **Strings**
4. **Others w/c are also used as the same, but not length**
   1. **angles: transformation**
      1. **deg**
      2. **rad**
      3. **grad**
      4. **turn**
   2. **time: animation**
      1. **ms**
      2. **s**
   3. **frequencies: speech**
      1. **hz**
      2. **khz**
   4. **resolution**

**CSS Common Issues**

1. **Browser Support**
   1. **No scripting support**
   2. **Scripting disabled**
   3. **Plugin Availability**
2. **Version incompatibilities, non-standard implementations**
3. **Capabilities Restriction**
   1. **Scripting languages are not for general purpose programming**
4. **Security Risk**
   1. **Browser implementation defects**

**JAVASCRIPT**

* Developed circa 1995 by Brendan Eich at Netscape Communication as the scripting language for the Netscape Navigator Browser
* Formerly known as Mocha, LiveScript then Java Script
* Standardized by ECMA International as ECMAScript
* Latest Version:

JavaScript code is executed on the fly as the document is rendered within the user-agent (browser). It executes in response with the document events

* Paradigm
  + Object-oriented, functional, imperative scripting language
* Java/C- like syntax
  + Implicit semicolon insertion for statement termination
  + Identifiers are alphanumeric, \_, and $ characters
  + Single-line (//) or block (/\*\*/) comments
* Type System and variable scoping rules
  + Dynamic / weak typing
  + Global or local scopes
  + Data Types
    - Primitive
      * Numbers (decimal, hexadecimal notation)
      * Boolean (true and false)
      * Strings (Single or Double Quote Delimited)
      * Undefined and Null
* Composite (object) Types
  + - Object, Number, Boolean, String, Date, Math, Global, RegExp, Error
    - Arrays
    - Functions and Arguments
* DOM Objects
  + - Anchor, Applet, Attr, Comment, DOMException, DOMImplementation, DocumentFragment, Element, Event, Form, Image, Input, Layer, Link, Node, Option, Select, Style, Text, TextArea
* Keywords
  + - Break, case, catch, continue, default, delete, do, else, finally, for, function, if, in, instanceof, new, return, switch, this, throw, try, typeof, var, void, with
* Statements and Constructs
  + var
    - Used to declare global/local scope variables
* If-else
  + - Condition expressions having values of 0, “”, null and undefined evaluate to false
* Switch-case-default-break
* while, do-while, for, for-in, break, continue
* try-catch-finally, throw
* function, return
* with

Document Object Model (DOM)

* Document Object Collections
  + Anchors []
  + Forms []
  + Images []
  + Links []
* Document Object Properties
  + Cookie
  + documentMode
  + domain
  + lastModified
  + ReadyState
  + Referrer
  + Title
  + URL
* Document Object Methods
  + close ()
  + getElementById()
  + getElementsByName()
  + getElementsByTagName()
  + open()
  + write()
  + writeIn()