**Casading Style Sheet**

**1. The Basic**

- not HTML

- many ways to add css to website

- selector {property: value; declaration; declaration}

h1 {color: red}

p {color: blue}

- default value for tags

- **Class** selector (.myClass):

+ Class selectors are designed to be applied many times in a page.

- **ID** selector (#unique):

+ An ID selector should only be applied once in a page.

- CSS cascade: rules that decide which CSS selectir wins when there is a conflict

- Don't repeat code

- CSS layout:

**Static** : page width does not change even

**Liquid** : flows and base on the size of the screen but does not resize nor respond, has limits.

**Adapted** : static + some responsive

**Responsive**: most flexible, respond base on the size of the window, text and images relow, resize and reposition themselves

**2. CSS styling**

- <link >: links to external CSS

- <nav></nav>: navigation tag (HTML5)

CSS: styling

+color : hex, name, rgb, hsl

+text-transform : uppercase, lowercase,..

+text-align : justify (every line have sae width)

+text-decoration :

+text-indent : thục đầu dòng

+text-style : italic

+font-weight : bold

+font-family : first, second, third, fourth option.

("Arial Black", Tahoma, "Time New Roma", serif) +font-size :

- pixel (**px**) : fixed-size

- viewport(**vw**) : relative, 1 vw = 1% viewport width

- (**em** ) : relative to base font-size, container font-size

-------------------------------------------------------------------------------------------------------------------

**Complex selector**

p.italic (no space between) : tag p has class="italic";

p. italic(has space between): class="italic" inside p tag

Parent/Child tag: inherit, last code win

-------------------------------------------------------------------------------------------------------------------**Google font**

Source: font.google.com

Embed: <link href="">

Use: font-family:

-------------------------------------------------------------------------------------------------------------------

**Who is the boss in CSS**

- a style applied to a tag will always win: BOSS

- class selectors override tag selectors

- last code wins

\* last code wins OR the rule that is more specific wins!

-------------------------------------------------------------------------------------------------------------------

**Background**

Background: { color - url - repeat - position - attachment}

Background-size: cover

**Box Model**

* Box model order: Content - padding - border - margin - position
* Margin, padding:

- top - right - bottom - left;

- top/bottom - right/left;

- top - right/left - bottom

* Total width = margin + border + padding + width
* Centering:   
  - margin: 10px auto;
* Height: for block level; **%** (only if parent tag has height set as pixel) or **px**
* Opacity: 0 -> 1; vd: 0.5

**Links and List**

* Pseudo-class: \*follow oder: l – v – h – a (love and hate)
  + a: link (unvisited link)
  + a: visited
  + a: hover
  + a: active (selected link)
* Creating buttons:
* Styling list with img: li { background-img:
  + - * + Background-reapeat:none;
        + Padding-left: 25px
        + Padding-right: 10px
        + Background-position: 0px 9px;
        + Margin-bottom:10px;
        + Overflow: hidden;

}

* CSS navbars:
  + - A tag: inline elements
    - a {

display: block;

width: 60px; }

**Display and Position**

px = fixed layout (width: 600px)

% = fluid (width: 90%)

* + max-width: 1200px;
  + min-width: 600px;

**height:**

* + overflow: hidden; auto; scroll

**Block and inline refresher**

**\***: universal selector

\* {margin: 0; padding: 0}

**Display, visibility, float**

Display: none -> never display; taken out of the flow

Visibility: hidden -> just hidden; the tag cannot be seen, but it is STILL in flow.

* { Clear: both } It pushes the tag that has 'clear: both' applied, past any floated tags.

**Position:**

* Static: defaut – not allow top, right, left, bottom
* Fixed: Pulls the tag outside of the normal document flow; relative with the viewport
  + - Bottom: 0;
    - Left: 200px;
    - Right: 0;
    - Top: 0;
* Relative: relative with where it was begin with
* Absolute: The element is positioned relative to its first positioned (not static) ancestor element
  + Usually inside relative tag;

**Select child tag: nth-child(***positon***)**

**8. Media Queries**

**@media** screen and (max-width: 600px)

If width <= 600px do…

**9. CSS Layout**

* Simple: liquid + media queries
* Static: container (width in pixel)
* Static 🡪 liquid: width 🡪 % + max/min width
* SHARK: last lesson
  + Display: table
  + Display: table-cell;…