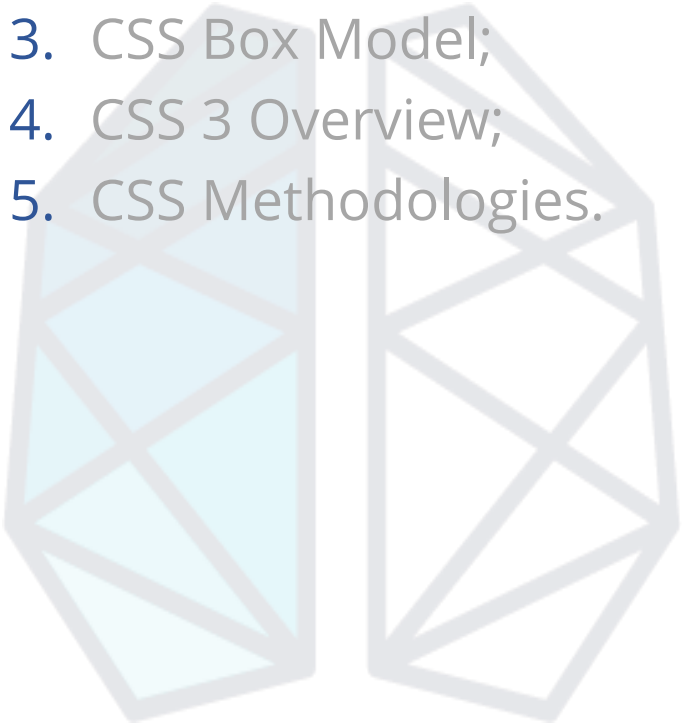
HTML5 & CSS3 - markup fundamentals

Module 1.3: HTML & CSS, CSS3

BRAIN ACADEMY

Training program

1. Fundamentals of HTML;
2. Fundamentals of CSS;
3. CSS Box Model;
4. CSS 3 Overview;
5. CSS Methodologies.



Module contents

- Fundamentals of HTML:
  - What Are HTML & CSS?
  - What are HTML tags;
  - HTML Document Structure;
  - Tag Attributes;
  - The HEAD Section Tags;
  - The base BODY Section Tags;
  - SEO optimization.

What Are HTML & CSS 1/3

- HTML - HyperText Markup Language:
  - Was created by Tim Berners-Lee in late 1991.
  - HTML is the special format that can display the text document as a web page.
  - HTML is a markup language that web browsers use to interpret and compose text, images and other material into visual or audible web pages.

What Are HTML & CSS 2/3

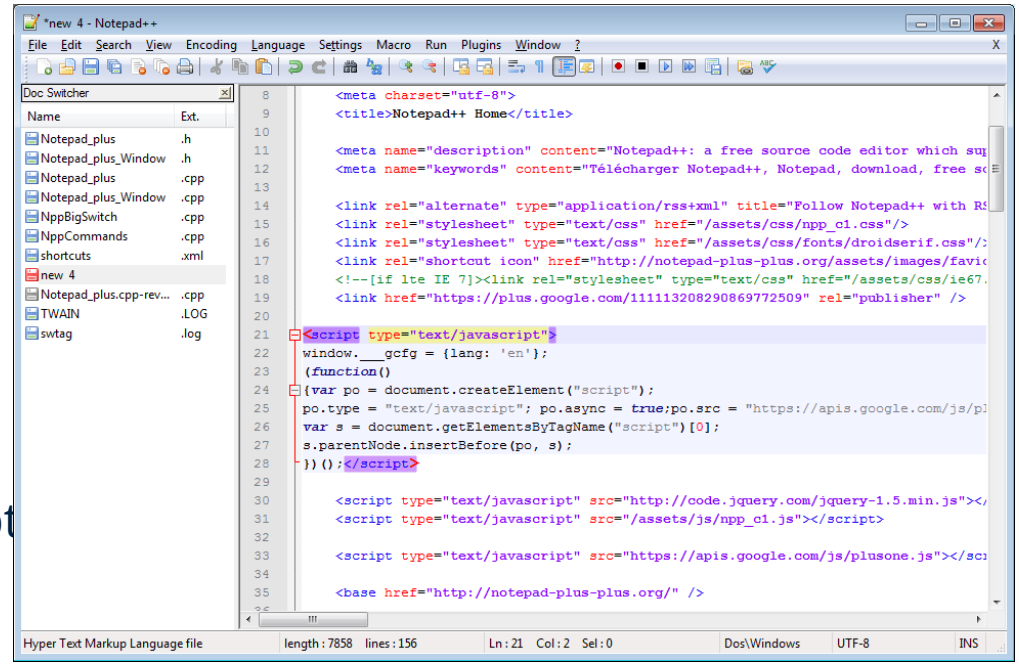
- CSS - Cascading Style Sheets:
  - CSS was creates by Håkon Wium Lie in late 1994.
  - CSS is a style sheet language used for describing the presentation of a document written in a markup language.

What Are HTML & CSS 3/3

- W3C - World Wide Web Consortium:
  - Founded and currently led by Tim Berners-Lee.
  - W3C develops uniform principles and standards for the Internet that make the World Wide Web (WWW) more versatile and comfortable.

Notepad++

- Distributed as free software
- Supports:
  - syntax highlighting
  - code folding
  - over 50 programming, script and markup languages.



Module contents

- Fundamentals of HTML:
  - What Are HTML & CSS?
  - What are HTML tags;
  - HTML Document Structure;
  - Tag Attributes;
  - The HEAD Section Tags;
  - The base BODY Section Tags;
  - SEO optimization.

What are HTML tags

- Tags are instructions that are embedded directly into the text of the HTML document.
- Tags begin with (<) and end with (>).
- Tags are not case sensitive : <html> is equivalent to <HTML>.
- HTML tags can be of two types. They are:
  - Paired Tags: <html></html>, <head></head> etc.
  - Unpaired Tags: <meta>, <link>, <img> etc.

Module contents

- Fundamentals of HTML:
  - What Are HTML & CSS?
  - What are HTML tags;
  - HTML Document Structure;
  - Tag Attributes;
  - The HEAD Section Tags;
  - The base BODY Section Tags;
  - SEO optimization.

HTML Document Structure

HTML tags:

- <!DOCTYPE HTML5>
- <html> … </html>
- <head> … </head>
- <body> … </body>
- <!– comments -->

```
<!DOCTYPE HTML >

<html>
 <head>
 </head>
 <body>

   This is a sample web page.

 </body>
</html>
```

Module contents

- Fundamentals of HTML:
  - What Are HTML & CSS?
  - What are HTML tags;
  - HTML Document Structure;
  - Tag Attributes;
  - The HEAD Section Tags;
  - The base BODY Section Tags;
  - SEO optimization.

Tag Attributes

- The <tag> tells the browser do something
- An attribute tells the browser how to do it
- Attributes can come in pair name="value"
- In some cases, attributes have no a value

```
<body bgcolor="green" hidden></body>
```

Module contents

- Fundamentals of HTML:
  - What Are HTML & CSS?
  - What are HTML tags;
  - HTML Document Structure;
  - Tag Attributes;
  - The HEAD Section Tags;
  - The base BODY Section Tags;
  - SEO optimization.

The HEAD Section Tags

- <title>My First Web Page</title>
- <meta/>
  - http-equiv = "Content-Type"
  - content ="text/html"
  - charset ="UTF-8"
- <style> ...</style>
- <link/>
  - rel ="stylesheet"
  - type ="text/css“
  - href

Lab work #1

Get practice to create a simple web-page

Module contents

- Fundamentals of HTML:
  - What Are HTML & CSS?
  - What are HTML tags;
  - HTML Document Structure;
  - Tag Attributes;
  - The HEAD Section Tags;
  - The base BODY Section Tags;
  - SEO optimization.

The BODY Section Tags 1/10

- Headings:

```
<h1>This is a heading</h1>
<h2>This is a heading</h2>
<h3>This is a heading</h3>
<h4>This is a heading</h4>
<h5>This is a heading</h5>
<h6> This is a heading</h6>
```

- ...
-
 - line breaks
- <hr/> - horizontal rule
- <font> - HTML Fonts tag in HTML5 is deprecated
- ...

Nested Tags 2/10

- It doesn't matter which tag is first, but they must be closed in the proper order:

```
<p><b><em>This is NOT the proper way
          to close nested tags.</p></b></em>
<p><b><em>This is the proper way
          to close nested tags.</em></b></p>
```

HTML Lists 3/10

There are three types of lists in the HTML:

- An unordered list
  ...
- An ordered list
  ...
- Definition list
  <dl>...</dl>

```
<ul>
        <li>List Item</li>
</ul>

<ol>
        <li>List Item</li>
</ol>

<dl>
        <dt> Term </dt>
        <dd>Definition</dd>
</dl>
```

HTML Images 4/10

- The <img> tag is singular (stand-alone) tag
- The value of the src attribute is the URL of the image
- The alt attribute is used to define an alternate text for an image

```
<p><img src="img/logo.jpg" alt="logo"/></p>
```

- Image Dimensions set by attributes width and height
- Attributes that not supported in HTML5: align, border, vspace and etc.

HTML Links 5/10

- The syntax of creating an anchor:

  Text to be displayed
- Attributes: href, target,type, rel
- ‘url’ may be:
  - Relative links - refer to a page in relation to the current document;
  - Absolute links — reference files based on the absolute location on the local file system or WWW;
  - mailto:name@domen.com.

HTML Tables 6/10

- Tables are defined with the <table> tag;
- A table is divided into rows that defined with the <tr> tag;
- Each row is divided into data cells that defined with the <td> tag;
- A data cell can contain text, images, lists, paragraphs, forms, horizontal rules, tables, etc.

HTML <iframe> tag 7/10

- An inline frame is marked up as follows:

```
<iframe src="https://www.youtube.com/watch?v=z-pJV20yD-w"></iframe>
```

- Dimensions set by attributes width and height
- Attribute name specifies the name of an <iframe>

HTML <form> tag 8/10

- <form> … </form>
- Attributes:
  - action
  - method (get, post)
  - target (default: _self)
  - novalidate
  - name
- <fieldset><legend>…</legend> … </fieldset>

HTML form’s elements 9/10

- <input > attributes:
  - type (text, radio, submit , button, checkbox, date, time, datetime etc);
  - Title;
  - Value;
  - Checked;
  - Name.
- <select> <option>...</option> ... <select> defines a drop-down list;
- <textarea>;
- <button>.

Example using <label>

```
<input id="a" type="checkbox"><label for="a">First checkbox</label>
```

HTML <script> tag 10/10

- <script>...</script>:
  - Type;
  - Src;
  - Async;
  - Defer;
  - Language.

Lab work #2

Get practice to create a page with:

- a refer to a page in relation to the current document;
- a link to another web page;
- a picture/graphic;
- a list of information;
- an iframe.

For example, it will be image gallery with ability to see a picture in all size in new window

Module contents

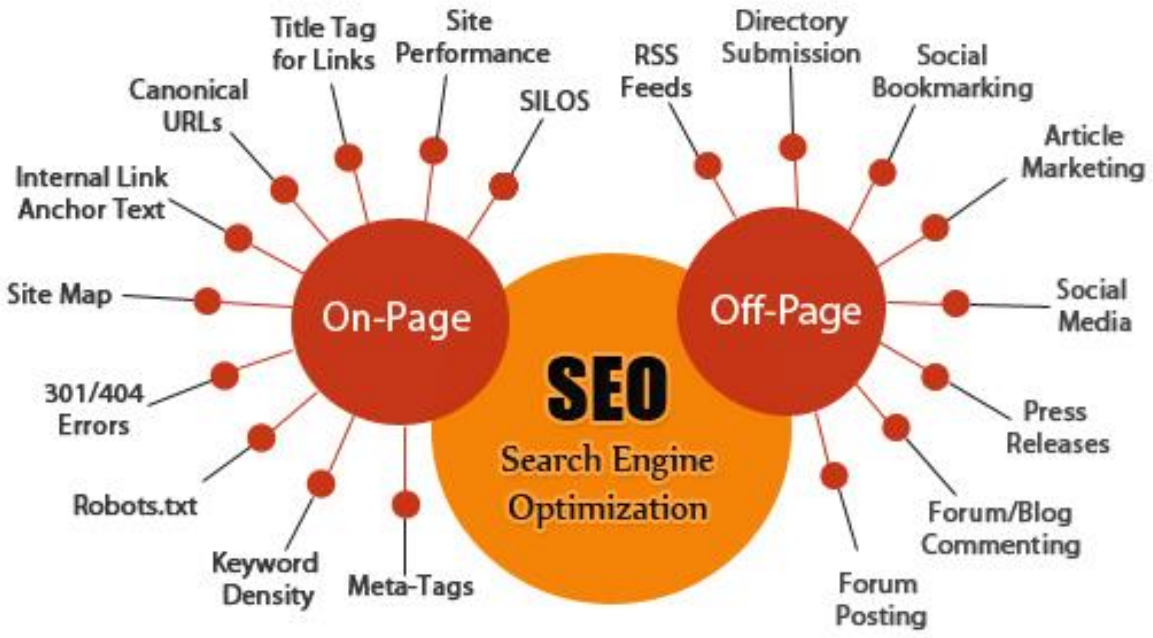
- Fundamentals of HTML & CSS:
  - What Are HTML & CSS?
  - What are HTML tags and Tag Attributes;
  - HTML Document Structure;
  - The HEAD Section Tags;
  - The base BODY Section Tags;
  - SEO optimization.

Search Engine Optimization (SEO)

http://www.worldwidemyanmar.com/search-engine-optimization

http://canadianmarketer.ca/events/event/five-seo-tips-2015/

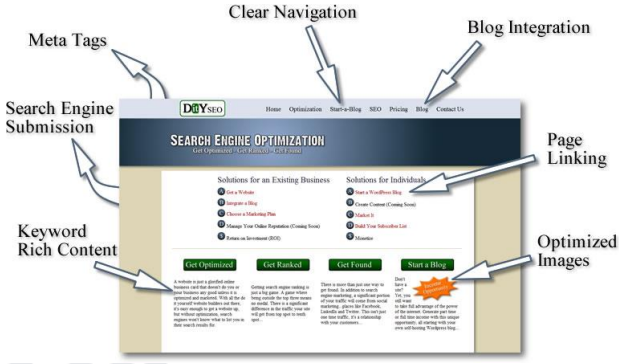
On-page and of-page tasks



http://surewin.ca/services/search-engine-optimization.html

Meta tag

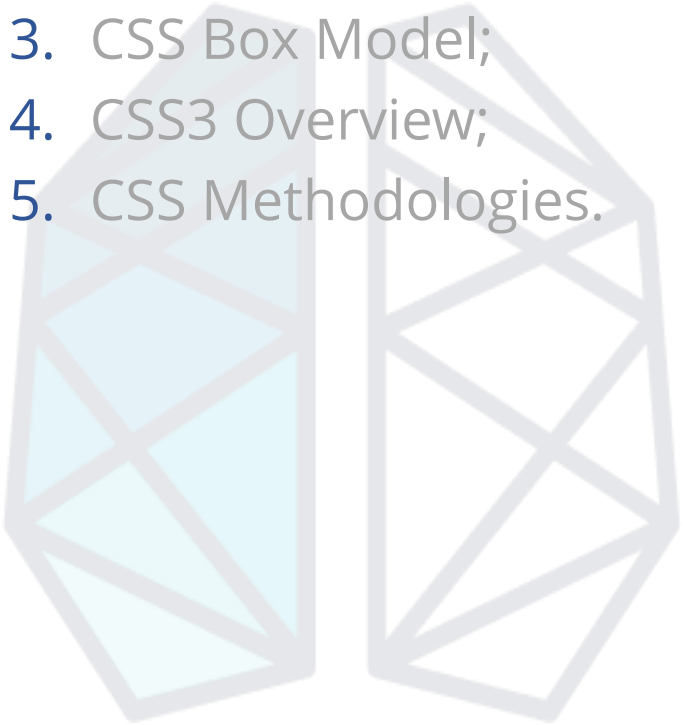
- description
  - <meta name="description" content="My description" />
- keywords tag
- H: Header tags:
  - <h1>, <h2>, <h3> etc (up to 6)
- I: Image file names and alt tags
  -
- Readable, interesting, knowledgeable content
- Relevancy
- Reputable links



http://www.wordtracker.com/academy/learn-seo/on-page-optimization/how-to-optimize-web-page

Training program

1. Fundamentals of HTML;
2. Fundamentals of CSS;
3. CSS Box Model;
4. CSS3 Overview;
5. CSS Methodologies.



Module contents

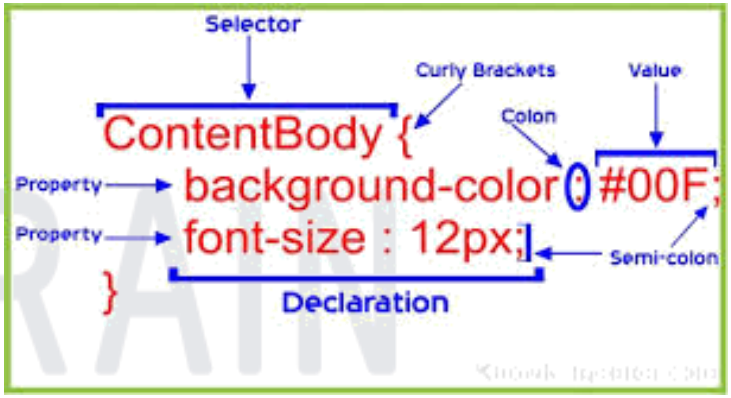
- Fundamentals of CSS:
  - CSS Selectors;
  - CSS Backgrounds;
  - CSS Text;
  - CSS Font;
  - CSS Link;
  - CSS List;
  - CSS Image.

Module contents

- Fundamentals of CSS:
  - CSS Selectors;
  - CSS Backgrounds;
  - CSS Text;
  - CSS Font;
  - CSS Link;
  - CSS List;
  - CSS Image.

Syntax 1/6

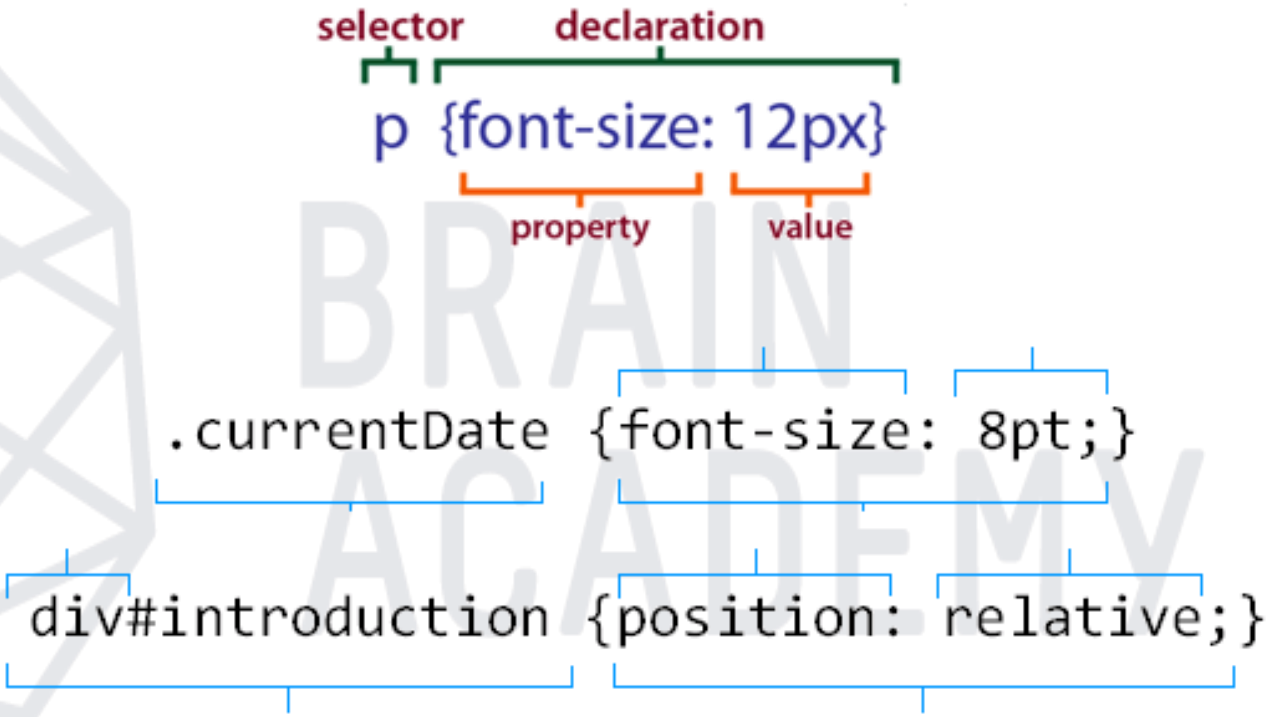
- Syntax CSS selector:



- A CSS comment starts with /* and ends with */
- Three ways of inserting a style sheet
- Multiple Styles
- Validate CSS online: http://jigsaw.w3.org/css-validator/

The element, id, class CSS selectors 2/6

- CSS selector:
  - html-tag
  - class attribute
  - id attribute



Grouping selectors 3/6

```
h1 {
      text-align: center;
      color: red;
   }

h2 {
      text-align: center;
      color: red;
   }

p {
      text-align: center;
      color: red;
   }
```

```
h1, h2, p  {
      text-align: center;
      color: red;
    }
...
<body>
<h1>Hello World!</h1>
<h2>Smaller heading!</h2>
<p>This is a paragraph.</p>
</body>
```

Cascade selectors 4/6

```
<style>
  p { color: yellow; }
  p { color: red; }
  p { color: green; }
</style>
```

```
h1 {
  color: gray;
  font-family: sans-serif;
}

h1 { border-bottom: 1px solid black; }
```



```
h1 {
  color: gray;
  font-family: sans-serif;
  border-bottom: 1px solid black;
}
```

CSS Combinators 5/6

- Descendant selector

  div p {background-color : yellow;}
- Child Selector

  div > p {background-color : yellow;}
- Adjacent sibling selector

  div + p {background-color : yellow;}
- General sibling selector

  div ~ p {background-color : yellow;}

Priority selectors 6/6

- Calculating a selector's specificity
- Concatenating the four numbers a-b-c-d (in a number system with a large base) gives the specificity.

```
*               /* a=0 b=0 c=0 ->    0 */
li              /* a=0 b=0 c=1 ->    1 */
ul li           /* a=0 b=0 c=2 ->    2 */
ul ol+li        /* a=0 b=0 c=3 ->    3 */
h1 + *[rel=up]  /* a=0 b=1 c=1 ->   11 */
ul ol li.red    /* a=0 b=1 c=3 ->   13 */
li.red.level    /* a=0 b=2 c=1 ->   21 */
#x34y           /* a=1 b=0 c=0 ->  100 */
#s12:not(p)     /* a=1 b=0 c=1 ->  101 */
```

CSS selector	Priority/specificity
style=""	1000
#id #id2 {}	0200
#id .class {}	0110
div#id {}	0101
#id {}	0100
div.class .class2 {}	0021
div.class {}	0011
div p {}	0002
div {}	0001

Module contents

- Fundamentals of CSS:
  - CSS Selectors;
  - CSS Backgrounds;
  - CSS Text;
  - CSS Font;
  - CSS Link;
  - CSS List;
  - CSS Image.

HTML Colors 1/3

- Hexadecimal Colour Representation in HTML – RGB (red, green and blue).
- A color is most often specified by:
  - a HEX value - like "#ff0000“;
  - an RGB value - like "rgb(255,0,0)“;
  - a color name - like "red“.

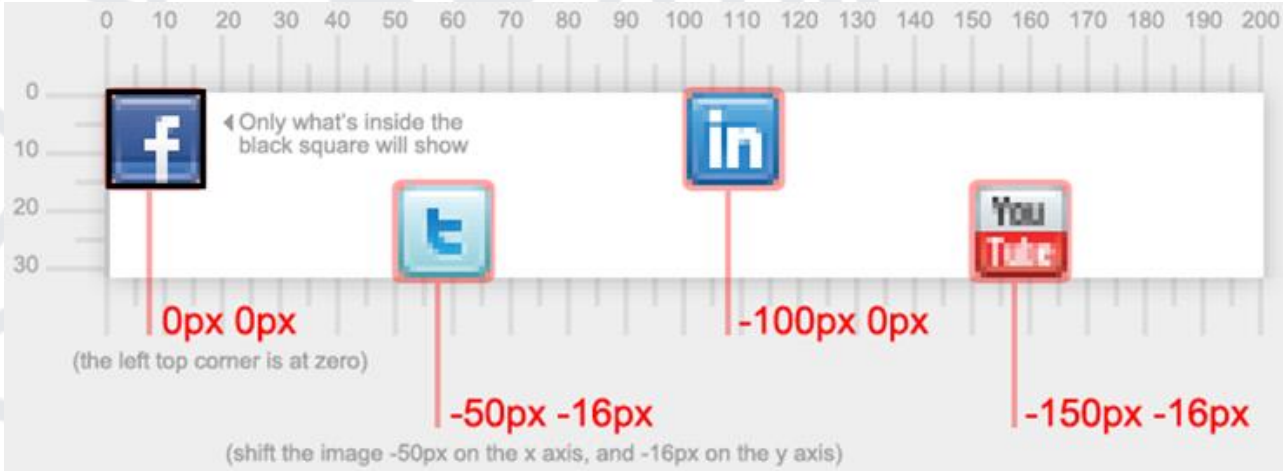
CSS background properties 2/3

- CSS properties used for background effects:
  - background-color;
  - background-image;
  - background-repeat (repeat-x, repeat-y, no-repeat);
  - background-attachment (scroll, fixed, local, initial, inherit);
  - background-position.

How to set background position 3/3

- CSS Syntax: background-position: *value*;
- Value:
  - left top, left center, left bottom
  - right top, right center, right bottom
  - center top, center center, center bottom
  - initial
  - 50% 50%
  - 25% 75%
  - 75% 25%
  - 10px 200px
  - 50px 50px

  etc



Module contents

- Fundamentals of CSS:
  - CSS Selectors;
  - CSS Backgrounds;
  - CSS Text;
  - CSS Font;
  - CSS Link;
  - CSS List;
  - CSS Image.

Main CSS text properties

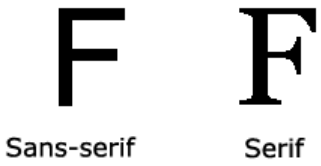
- text-color;
- text-align (center, right, left, justify);
- text-decoration (overline, line-through, underline, none);
- text-transform (uppercase, lowercase, capitalize);
- text-indent (px).

Module contents

- Fundamentals of CSS:
  - CSS Selectors;
  - CSS Backgrounds;
  - CSS Text;
  - CSS Font;
  - CSS Link;
  - CSS List;
  - CSS Image.

CSS Font Properties

- font-family (Serif, Sans-serif, Monospace)
- font-style (normal, italic, oblique)
- font-size (from pixels to em: *pixels*/16=*em*)
- font-weight (normal, bold, *number,* initial
- font-variant (normal, small-caps, initial)



Using external API

- https://www.google.com/fonts
  - choose the styles you like

Module contents

- Fundamentals of CSS:
  - CSS Selectors;
  - CSS Backgrounds;
  - CSS Text;
  - CSS Font;
  - CSS Link;
  - CSS List;
  - CSS Image.

CSS Link properties

- Links states are:
  - a:link;
  - a:visited;
  - a:hover;
  - a:active.
- The mostly used property is text-decoration.

Module contents

- Fundamentals of CSS:
  - CSS Selectors;
  - CSS Backgrounds;
  - CSS Text;
  - CSS Font;
  - CSS Link;
  - CSS List;
  - CSS Image.

CSS Lists properties

- list-style:
  - list-style-type (circle, square, upper-roman, lower-alpha);
  - list-style-image;
  - list-style-position.

Module contents

- Fundamentals of CSS:
  - CSS Selectors;
  - CSS Backgrounds;
  - CSS Text;
  - CSS Font;
  - CSS Link;
  - CSS List;
  - CSS Image.

CSS Image

- Links states:
  - img:link,
  - img:visited,
  - img:hover,
  - img:active
- CSS Dimension Properties:
  - height,max-height, min-height
  - width,max-width, min-width
- opacity (0.0 - 1.0)
- Image Sprites

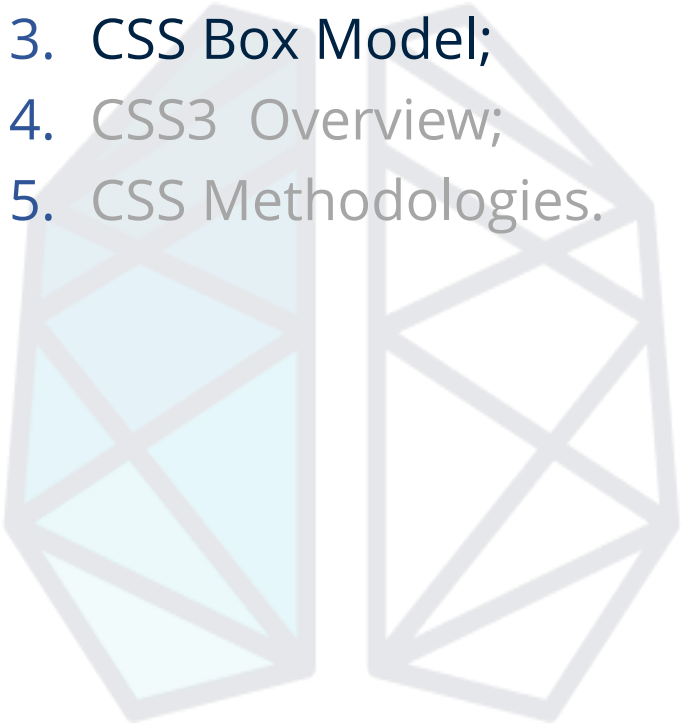
Lab work #3

Experiment with CSS properties and values to change styling of a page.

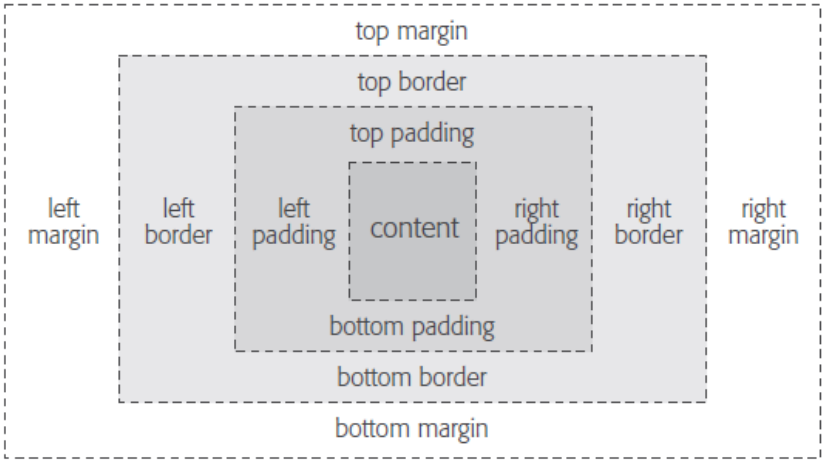
- Practice in using image sprites. With CSS, we can show just the part of the image we need.

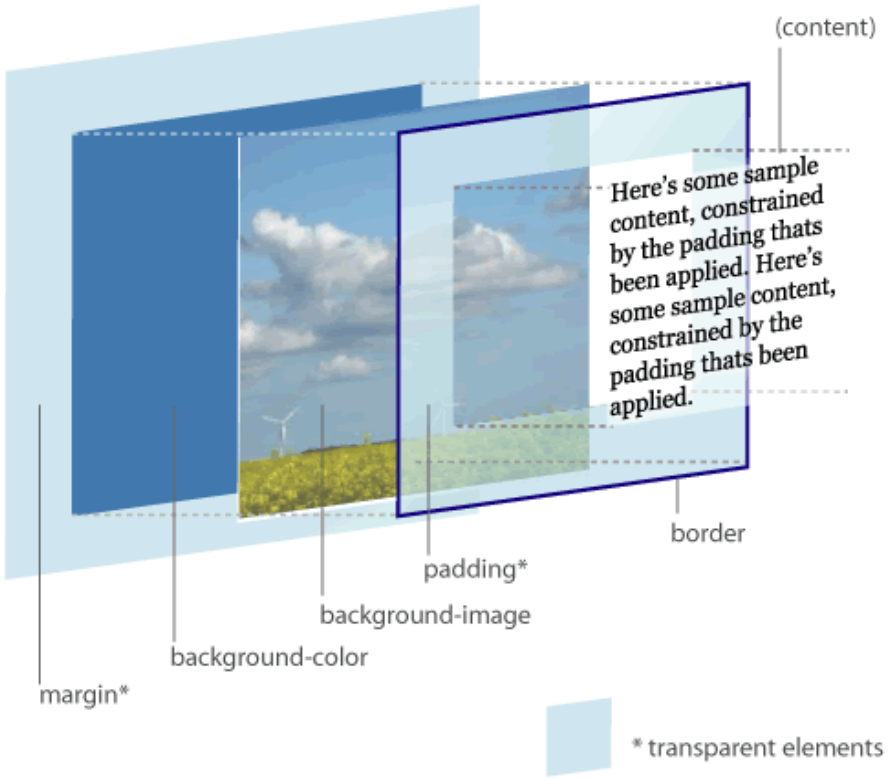
Training program

1. Fundamentals of HTML;
2. Fundamentals of CSS;
3. CSS Box Model;
4. CSS3 Overview;
5. CSS Methodologies.



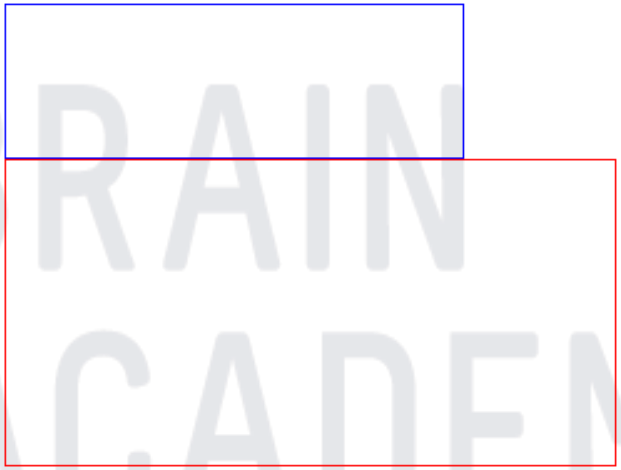
Define CSS Box Model 1/3





Code Example 2/3

```
<!--  box-sizing Property -->
<div class="div1"></div>
<div class="div2"></div>
<style>
.div1 {
    width: 300px;
    height: 100px;
    border: 1px solid blue;
}
.div2 {
    width: 300px;
    height: 100px;
    padding: 50px;
    border: 1px solid red;
}
</style>
```

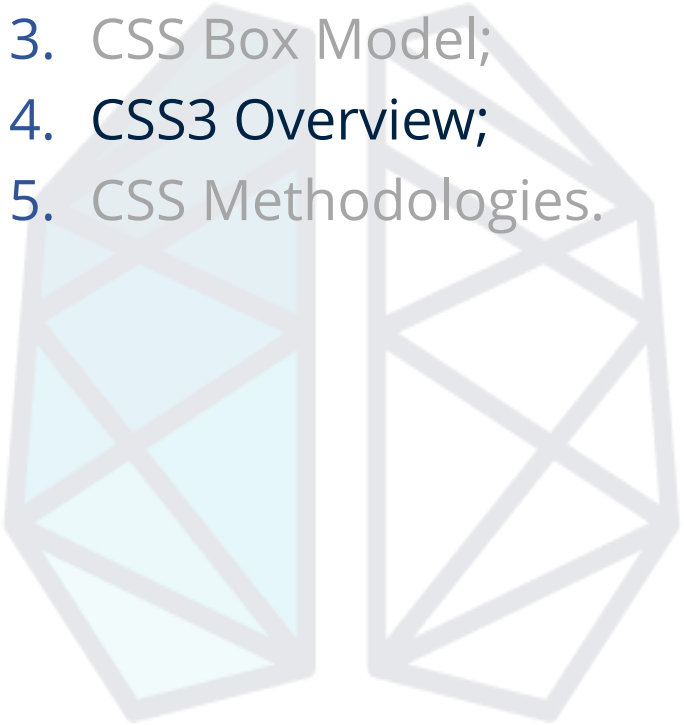


CSS Layout 3/3

- CSS Layout - The position property:
  - position (static, relative, fixed, absolute);
  - bottom, left, right, top;
  - z-index.
- CSS Layout - float and clear;
- CSS Layout - The display property (inline, block, none);
- Compare:
  - display: none;
  - visibility : hidden;

Training program

1. Fundamentals of HTML;
2. Fundamentals of CSS;
3. CSS Box Model;
4. CSS3 Overview;
5. CSS Methodologies.



Module contents

- CSS3 Overview:
  - New property;
  - Gradient;
  - 2D and 3D Transforms;
  - Transition;
  - Animation;
  - Flexible Box.

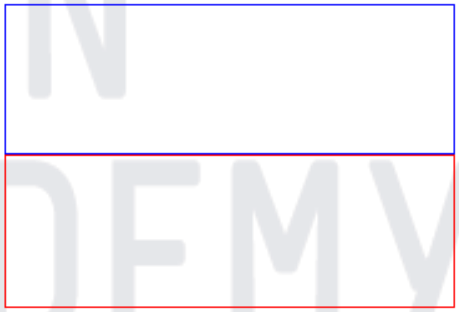
CSS3 New property

- border-radius:
- Background: size, origin, clip:
- Text:
  - text-overflow clip|ellipsis|*string*|initial|inherit;
  - word-wrap normal|break-word|initial|inherit;
  - word-break: normal|break-all|keep-all|initial|inherit;
- Shadow Effects:
  - text-shadow: none|*h-shadow v-shadow blur-radius color*|initial|inherit;
  - box-shadow: none|*h-shadow v-shadow blur spread color*|inset|initial|inherit;

CSS3 box-sizing property

- Allows to include the padding and border in an element's total width and height:
  - box-sizing

```
.div1 {
    width: 300px;
    height: 100px;
    border: 1px solid blue;
    box-sizing: border-box;
}
.div2 {
    width: 300px;
    height: 100px;
    padding: 50px;
    border: 1px solid red;
    box-sizing: border-box;
}
```



CSS3 Gradients

- linear-gradient for text
  - (*direction*, *color-stop1*, *color-stop2*, ...);
- linear-gradient for color
  - (*angle*, *color-stop1*, *color-stop2*);
- radial-gradient(*shape size* at *position, start-color, ..., last-color*);
- repeating-radial-gradient();

A linear gradient with a specified angle:

```
#grad {
  background: -webkit-linear-gradient(180deg, red, blue); /* For Safari 5.1 to 6.0 */
  background: -o-linear-gradient(180deg, red, blue); /* For Opera 11.1 to 12.0 */
  background: -moz-linear-gradient(180deg, red, blue); /* For Firefox 3.6 to 15 */
  background: linear-gradient(180deg, red, blue); /* Standard syntax */
}
```

CSS3 2D and 3D Transforms

- 2D transformation methods:
  - translate(x,y)
  - rotate(*angle*)
  - scale(x,y)
  - skewX(*angle*)
  - skewY(*angle*)
  - matrix()

- 3D transformation methods:
  - rotateX(*angle*)
  - rotateY(*angle*)
  - rotateZ(*angle*)

Code Example 2D Transforms

```
<div class="divTransform">Transformed</div>
<style>
.divTransform {
    width: 100px;
    height: 50px;
    border: 1px solid green;

  //transform: translate(200px,0px);
  //transform: rotate(20deg);
  //transform: scale(5,10);
  //transform: skewX(20deg);
  //transform: skewY(20deg);
  //transform: skew(20deg, 10deg);
  //transform: matrix(1, -0.5, 0, 1, 0, 0);
}
</style>
```

CSS3 Transition

- Allows to change property values smoothly (from one value to another), over a given duration:
  - transition-property;
  - transition-duration;
  - transition-timing-function;
    - ease/linear/ease-in/ease-out/ease-in-out/cubic-bezier(n,n,n,n)
  - transition-delay;
  - Transition.
- Draws the animation faster (also mobile)

Code Example

```
<img class="example" src="cat.jpg">
<style>
.example { opacity: 0.3;}
.example:hover {
  opacity:1;
  transition:opacity 0.4s linear; }
```

CSS3 Animation

BRAIN ACADEMY

- Allows animation of most HTML elements without using JavaScript or Flash;
- Properties:
  - Animation;
    - animation –name;
    - animation-duration;
    - animation-delay;
    - animation-iteration-count;
    - animation-direction;
    - animation-timing-function.
  - @keyframes.

The @keyframes Rule

- Formal syntax:

```
@keyframes <identifier> {
  [
    [ from | to | <percentage> ]
    [, from | to | <percentage> ]*
    block ]*
  }
```

Code example

```
<h1>My animation</h1>
<style>
h1 {
  animation-duration: 3s;
  animation-name: slidein;
}
@keyframes slidein {
  from {
    margin-left: 100%;
    width: 300%;
  }

  to {
    margin-left: 0%;
    width: 100%;
  }
}
</style>
```

```
<div class="animated"></div>
<style>
.animated {
  width: 50px;
  height: 50px;
  animation-name: bgclranimate;
  animation-duration: 3s;
  animation-timing-function: linear;
  animation-iteration-count: infinite;
}
@keyframes bgclranimate{
  0% {background:yellow}
  100% {background:green}
}
</style>
```

The Difference Between CSS3 Transition and Animation

BRAIN ACADEMY

- Transitions:
  - state changes, positions, fills;
  - generally response to a user action.

- Animations:
  - sprites animation;
  - uninterrupted and looped animation.

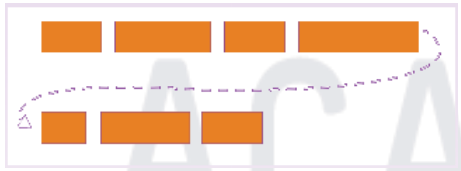
CSS3 Flexbox

- Flexbox defines how flex items are laid out inside a flex container.
- A **flex container** is declared by
  - setting the *display* property of an element to either flex (rendered as a block)
  - inline-flex (rendered as inline)
- A **flex items** are inside a flex container

Flex Container Properties 1/4

- display : flex; /* or inline-flex */
- flex-direction : column-reverse | column | row | row-reverse ;

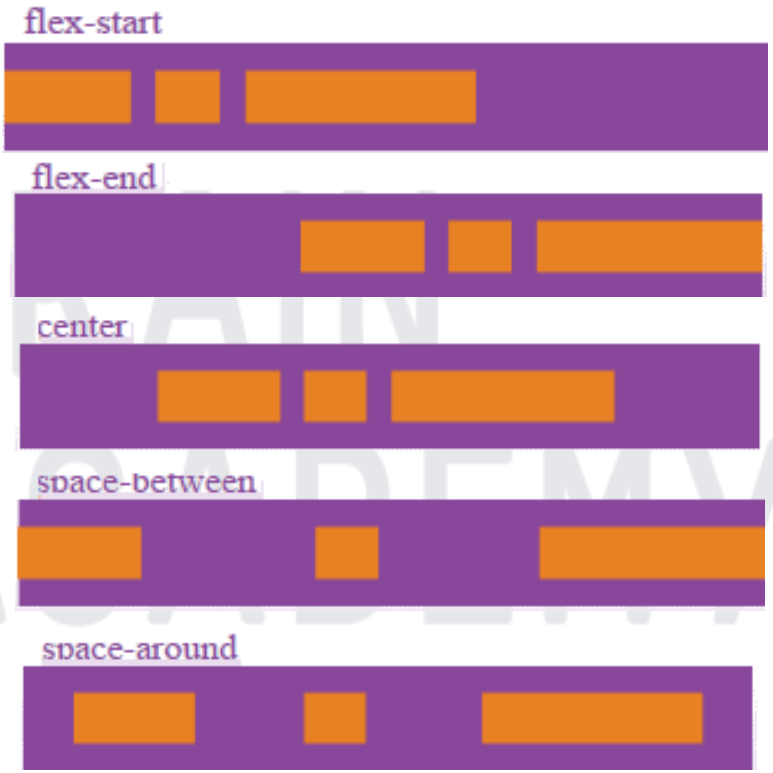
- flex-wrap : nowrap | wrap | wrap-reverse;



- flex-flow: <'flex-direction'> || <'flex-wrap'>
  - Applies to: parent flex container element

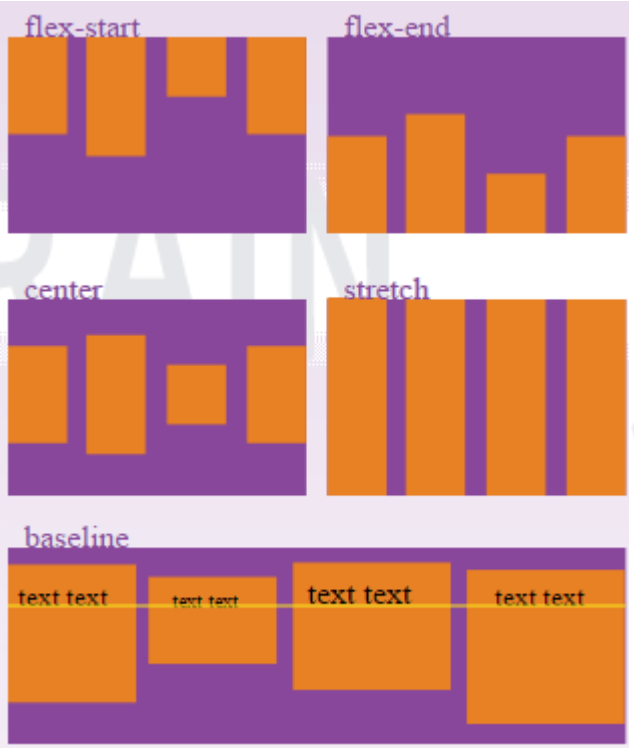
Flex Container Properties 2/4

- justify-content: flex-start |
  flex-end |
  center |
  space-between |
  space-around;



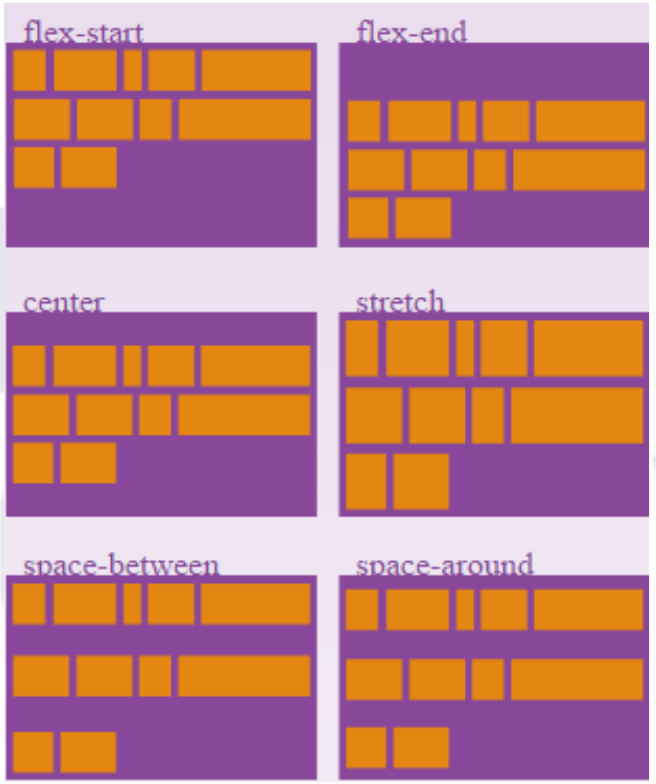
Flex Container Properties 3/4

- align-items: flex-start | flex-end | center | baseline | stretch;



Flex Container Properties 4/4

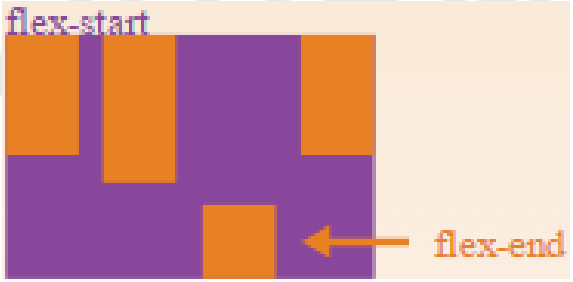
- align-content : flex-start |
  flex-end |
  center |
  space-between |
  space-around |
  stretch;



Flex Items Properties

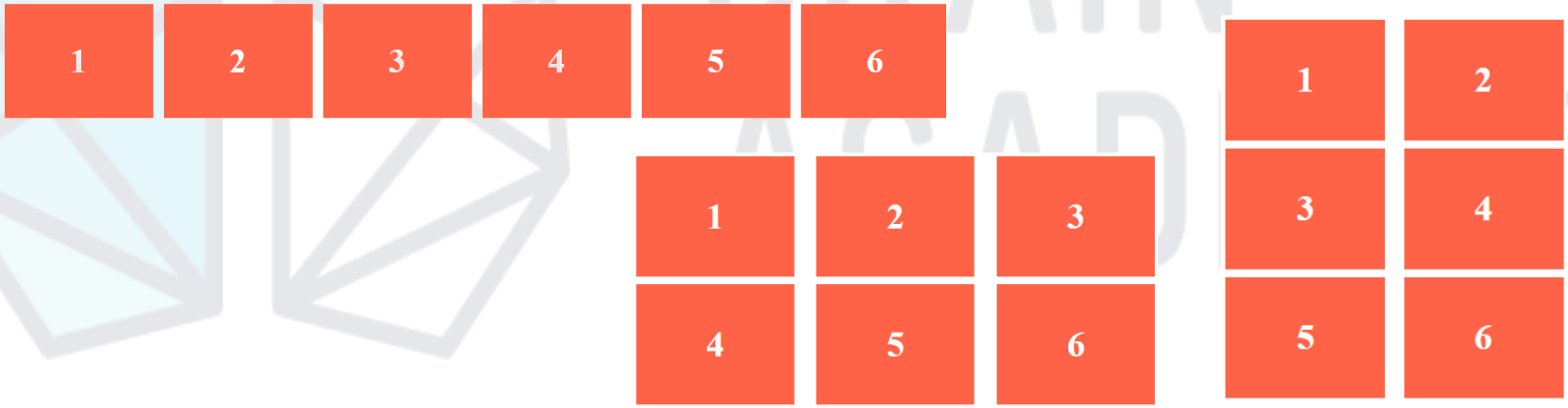
- order: <integer>;

- flex-grow : <number>; /* default 0 */
- flex-shrink : <number>; /* default 1 */
- flex-basis: <length> | auto; /* default auto */
- flex: none | [<'flex-grow'> <'flex-shrink'>? || <'flex-basis'>]
- align-self : auto | flex-start | flex-end | center | baseline | stretch;



Code example

- Consider a list of 6 items, all with a fixed dimensions in a matter of aesthetics but they could be auto-sized.
- Do them evenly and nicely distributed on the horizontal axis
- Resize the browser to see how it happens



Flexbox bugs

BRAIN ACADEMY

- The best Flexbox bugs collection s an open source place to track all of them
  - Philip Walton and Greg Whitworth's Flexbugs

Chrome	Safari	Firefox	Opera	IE	Android	iOS
20- (old) 21+ (new)	3.1+ (old) 6.1+ (new)	2-21 (old) 22+ (new)	12.1+ (new)	10 (tweener) 11+ (new)	2.1+ (old) 4.4+ (new)	3.2+ (old) 7.1+ (new)

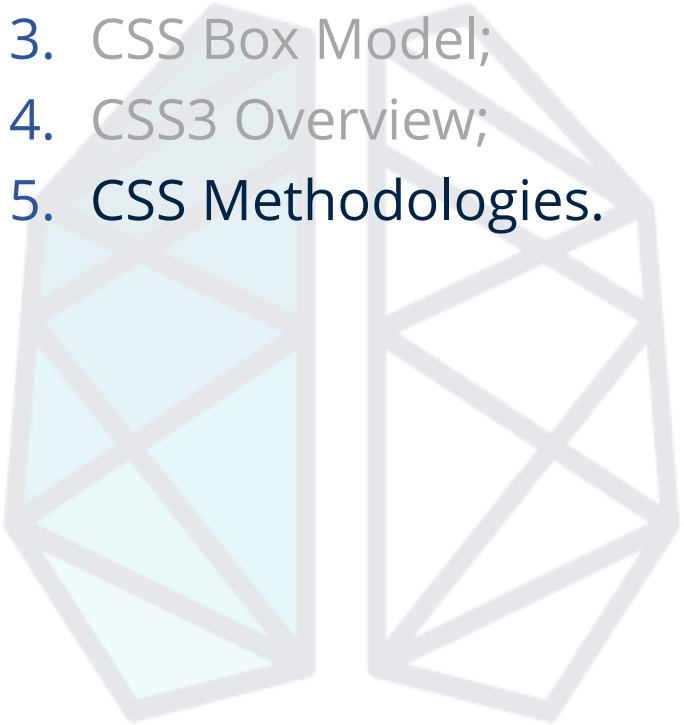
Lab work #4

Apply and explain CSS “cascade” including: importance, specificity and inheritance.

- Apply and explain CSS “cascade” including: importance, specificity and inheritance.
- Describe the use of Normalize and reset.css files.

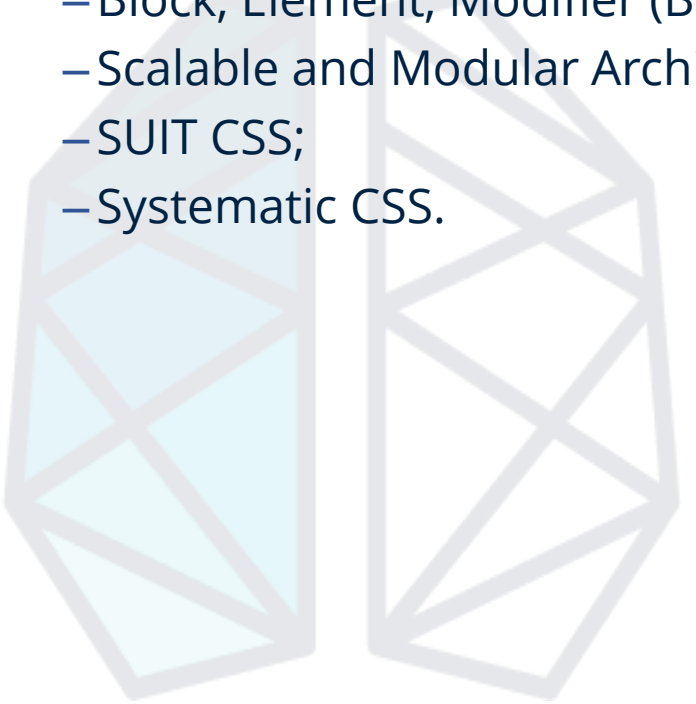
Training program

1. Fundamentals of HTML;
2. Fundamentals of CSS;
3. CSS Box Model;
4. CSS3 Overview;
5. CSS Methodologies.



Module contents

1. CSS Methodologies:
   - Object-Oriented CSS (OOCSS);
   - Block, Element, Modifier (BEM);
   - Scalable and Modular Architecture for CSS (SMACSS);
   - SUIT CSS;
   - Systematic CSS.



Object-Oriented CSS (OOCSS)

Object-oriented CSS is a coding paradigm that styles "objects" or "modules"—nestable chunks of HTML that define a section of a webpage—with robust, reusable styles.

- There are two main principles to separate:
  - *the structure from the skin;*
  - *the container from the content.*
- The main steps are:
  - Determine your object.
  - Set your reset, base elements, and base object styles.
  - Apply classes to objects and components for the desired look and feel.

Block, Element, Modifier (BEM)

BEM — Block Element Modifier is a methodology, that helps you to achieve reusable components and code sharing in the front-end.

- The block is an independent entity. Block names must be unique in all project.
- The element is a part of a block with a specific function, they are block dependent.
- HTML elements must not be used in CSS selectors.
- Syntax: block_element—modifier.

Scalable and Modular Architecture for CSS (SMACSS)

There are five main principles, divided by its rule types:

- Base Rules: The default styles;
- Modules: Dictate minor layout components;
- Layout Rules: Dictate major layout components;
- State: Describe the appearance of a module in various states;
- Theme: Describe the appearance of a module in various contexts.

SUIT CSS

SUIT CSS is a reliable and testable styling methodology for component-based UI development.

- A collection of CSS packages and build tools are available as modules.
- SUIT CSS plays well with React, Angular, Ember and other component-based approaches to UI development.

Systematic CSS

Systematic CSS is meant to be a simpler alterative to existing CSS methodologies: There are fewer naming-conventions to remember, and the class-naming convention is intended to be more intuitive.

- It is a new CSS methodology, 2015.
- The process of developing a new web design is broken up into four phases:
  - Layout;
  - Element;
  - Widgets;
  - Modifiers.