



HTML & CSS crash course

What is HTML

- HTML is a computer language devised to allow website creation.
- It is relatively **easy to learn**, with the basics being accessible to most people in one sitting;
- It is quite **powerful** in what it allows you to create
- It is constantly undergoing revision and evolution
- The most recent incarnation is HTML5 ([» W3C](#))
 - Has many improvements but
 - Basic grammar has not changed → one of the *most important standards for working with and on the Web*.



General characteristics

- HTML: HyperText Markup Language
 - Based on “tags” like latex or markdown
 - Relates to: tags-based text processing (TeX, IBM-script, Wordstar).
 - Combines power & simplicity
 - Hypertext and Hypermedia
 - HTML documents are text (ASCII) files.
 - HTML is (intended to be) “portable”.

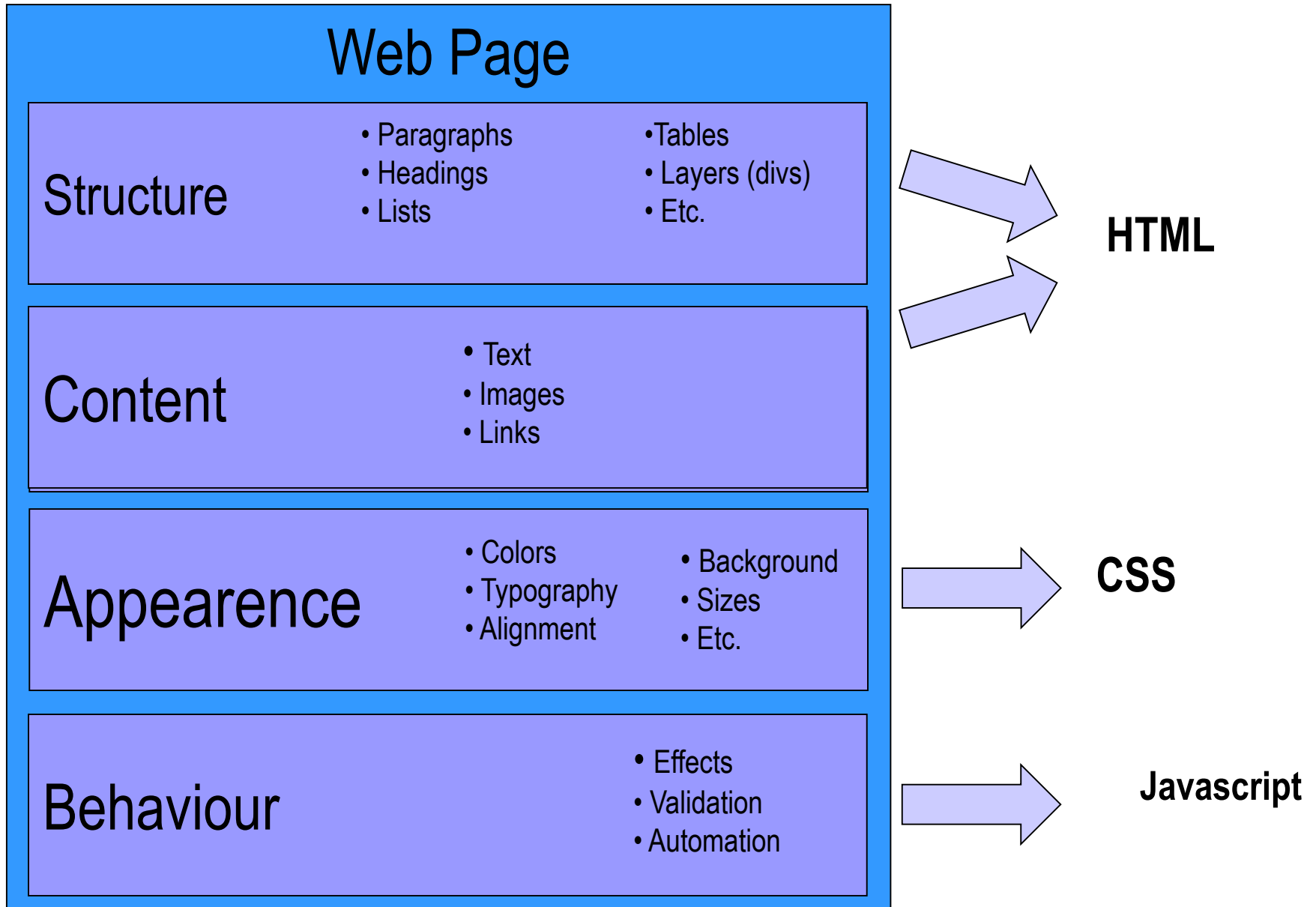
HTML editors

- General purpose editors
 - Emacs, Textpad, Notepad++
- Browser-related editors
 - Kompozer,
- Specific editors
 - HoTMetaL, Quanta, Front Page
- Word processors ☹

Reference guides

- W3school.com
- HTML5 Introduction (MOOC):
<https://www.edx.org/course/html5-introduction-w3cx-html5-0x-0>
- Google

HTML – CSS – Javascript





HTML files

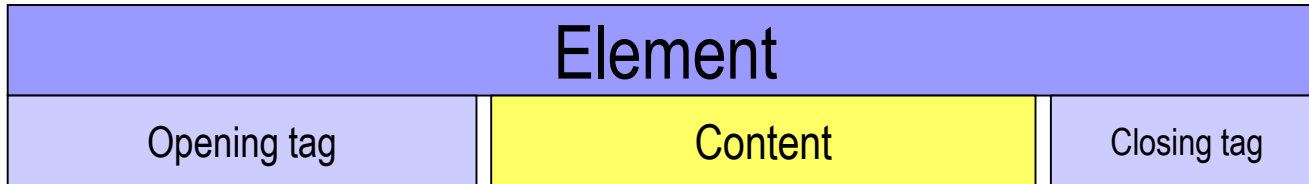
- An HTML file is basically plain text that can be opened and edited with any text editor.
- HTML's power lies in its *marked up structure*.
- HTML markup allows defining
 - the parts of a document that need to be displayed as headlines,
 - the parts that contain links,
 - the parts that should be organized as tables,
 - Etc.



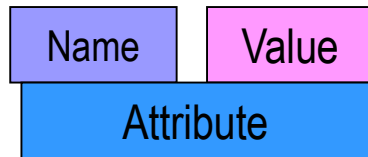
HTML structure

- HTML documents are made of HTML “elements”
- Elements: Text (content) enclosed between two «tags»
- Tags: Pairs of words enclosing content. Words are identical.
closing tag starts by «/»

Tags, Elements and Attributes



<p class="text">HTML course</p>





Minimal Document Structure

- Some labels are compulsory: Any HTML document must have them
 - `<HTML></HTML>`
 - `<HEAD></HEAD>`
 - `<BODY></BODY>`



Example 1. Simplest HTML

```
<!DOCTYPE html>
<html>
  <head>
    <title>First HTML</title>
  </head>
  <body>
    I am your first HTML-file!
  </body>
</html>
```

Example 1-HTML5 improvements

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>You Can Edit This</title>
</head>
<body>
<h1>I Mean, You Can Really Edit This</h1>
<p contenteditable=true>
Now is the time for all good cats to come
But in our brave new world of HTML5, all we need is:
to the aid of their catnip.
</p>
</body>
</html>
```

Basic text formatting

- Headings

- `<H1></H1>.....<H6></H6>`

- Paragraph breaks

- `<P></P>`

- Centering

- `<CENTER></CENTER>`

- Bold and Italic

- ` <I></I>`

Tag attributes

- Paragraph alignment
 - Left: **ALIGN=left** (default)
 - Right: **ALIGN=right**
 - Centered: **ALIGN=center**
- Use
 - **<p ALIGN=option>**

Lists

■ Unordered Lists

□ ``

□ ``

□ ``

■ Ordered Lists

□ ``

□ ``

□ ``

Example 2: lists

```
<!DOCTYPE html>
<html lang="en">
<head> <title>Ordered and unordered lists </title></head>
<body>
<h1> Ordered list</h1>
<OL> <LI> First
      <LI>Second
      <LI>Third
</OL>

<h1> Unordered list</h1>
<UL>
  <LI>Natural Sciences
    <UL> <LI>Biology
          <LI>Zoology
        </UL>
    <LI>Social Sciences
    <UL>      <LI>Geography
              <LI>History
            </UL>
  </UL>
</UL>
</body>
</html>
```




HTML grouping tags: DIV & SPAN

- Used to define sections in documents
 - `<div>` Defines a block-level section
 - Used as a container for other HTML elements
 - `` Defines an inline section
 - Used as container for some text
- No required attributes
 - often use style, class or id
- Often used in conjunction with CSS

DIV and SPAN example

```
<!DOCTYPE html >
<html lang="en">
<head> <title>Span and Div</title></head>
<body>
```

```
<h1> The &lt;DIV> tag </h1>
```

The <DIV> tag defines a division or a section in an HTML document.

In the example below <DIV> is used to create a section in a document that will have a light blue background color:

```
<div style="background-color:lightblue">
  <h3>This is a heading</h3>
  <p>This is a paragraph.</p>
</div>
```

```
<h1> The &lt;SPAN> tag </h1>
```

In the example below is used to change the color of a single word in a paragraph:

```
<p>My mother has <span style="color:blue">blue</span> eyes.</p>
</body> </html>
```

Tables

- Tables are used for
 - Storing tabular information
 - Create/organize the page layout.
- Tags used for managing tables are:
 - **<TABLE> </TABLE>** Begin and End of the table
 - <TR> </TR>** Define table rows
 - <TD> </TD>** Define table columns
- Table structure is very flexible.
- Table tags are **hierarchichal**: a table has **one or more** rows **each of which** is divided in one or more columns

Example HTML Table

```
<!DOCTYPE html>

<html>

<head>

    <title>HTML Tables</title>

</head>

<body>

    <h1>An HTML Table</h1>

    <table>

        <tr> <th>Rank</th> <th>Nominal GDP</th> <th>Name</th> </tr>
        <tr> <th></th> <th>(per capita, USD)</th> <th></th> </tr>
        <tr> <td>1</td> <td>170,373</td> <td>Lichtenstein</td> </tr>
        <tr> <td>2</td> <td>167,021</td> <td>Monaco</td> </tr>
        <tr> <td>3</td> <td>115,377</td> <td>Luxembourg</td> </tr>
        <tr> <td>4</td> <td>98,565</td> <td>Norway</td> </tr>
        <tr> <td>5</td> <td>92,682</td> <td>Qatar</td> </tr>

    </table>

</body>

</html>
```

Images

- Inserted using a graphical file (GIF, JPEG, PNG, BMP, ...)
- Sintax:
 - ``
- Attributes
 - `<IMG SRC= "image.gif" HEIGHT= value
WIDTH= value>`
- Alignment
 - ``

Hyperlinks

- Hyperlinks allow linking any two pages independently of their physical location

- Examples:

- ☐ ` Diario El País `
- ☐ ` Página1 `
- ☐ ``

Hyperlinks to different sections of HTML pages

- Link from a page (e.g: an author index authors.html) to a specific section in another documento (autores.html). (hiperenlace 2)
 - First create the reference in th first document:
 - ` Autor1 `
 - Next create the anchor in the second document:
 - ` Autor1 `

Hyperlinks between sections of same HTML page

- Links from one page (authors.html) to a section in the same document
- First create the reference:
 - ` Autor1 `
- Next create anchor in the same document:
 - ` Autor1 `

HTML CHEAT SHEET

*`
` empty tags

DOCUMENT OUTLINE

<code><!DOCTYPE></code>	Version of html
<code><html></code>	HTML document
<code><head></code>	Page information
<code><body></code>	Page documents

COMMENTS

`<!--comment text-->`

PAGE INFORMATION

<code><base/></code>	Base URL
<code><meta/></code>	Meta data
<code><title></code>	Title
<code><link/></code>	Relevant resource
<code><style></code>	Style resource
<code><script></code>	Script esource

LISTS

<code></code>	Ordered list
<code></code>	Unordered list
<code></code>	List item
<code><dl></code>	Definition list
<code><dt></code>	Definition term
<code><dd></code>	Term description

DOCUMENT STRUCTURE

<code><h[1-6]></code>	Heading
<code><div></code>	Page section
<code></code>	Inline section
<code><p></code>	Paragraph
<code>
</code>	Line break
<code><hr/></code>	Horizontal rule

LINKS

<code></code>	Page link
<code></code>	Email link
<code></code>	Anchor
<code></code>	Link to anchor

TABLES

<code><table></code>	Table
<code><caption></code>	Caption
<code><thead></code>	Table header
<code><tbody></code>	Table body
<code><tfoot></code>	Table footer
<code><colgroup></code>	Column group
<code><col/></code>	Column
<code><tr></code>	Table row
<code><th></code>	Header cell
<code><td></code>	Table cell

IMAGES AND IMAGE MAPS

<code></code>	Image
<code><map></code>	Image map
<code><area/></code>	Area of image map

TEXT MARKUP

<code></code>	Strong emphasis
<code></code>	Empahasis
<code><blockquote></code>	Long quotation
<code><q></code>	Short quotation
<code><abbr></code>	Abbreviation
<code><acronym></code>	Acronym
<code><address></code>	Address
<code><pre></code>	Pre-formatted text
<code><dfn></code>	Definition
<code><code></code>	Code
<code><cite></code>	Citation
<code></code>	Deleted text
<code><ins></code>	Inserted text
<code><sub></code>	Subscript
<code><sup></code>	Superscript
<code><bdo></code>	Text direction

FORMS

<code><form></code>	Form
<code><fieldset></code>	Collection of fields
<code><legend></code>	Form legend
<code><label></code>	Input label
<code><input/></code>	Form input
<code><select></code>	Drop-down box
<code><optgroup></code>	Group of options
<code><option></code>	Drop-down options
<code><textarea></code>	Large text input
<code><button></code>	Button

CORE ATTRIBUTES

<code>class</code>	style
<code>id</code>	title



Synthesis exercise

- Select a topic you feel comfortable with.
- Write or get a text that is organized in, at least two or three sections.
- Allow the code to have all the elements we have learnt about: text, tables, images, hyperlinks...
- Organize it in sections
 - Start with a table of contents
 - Formatted by an ordered list
 - Each element must link to each section's title.
 - Continue with sections
 - Title: Headings of same type
 - End each section with a link to the table of contents followed by a horizontal bar.



Cascading Style Sheets (CSS)



What are CSS

- CSS complements HTML by providing a look and feel to web pages.
- “raw” HTML pages can look fairly plain, with a default font and font size.
- Using CSS, you can spice up that look,
 - adding color and background images,
 - changing fonts and font sizes,
 - drawing borders around areas, and
 - even changing the layout of the page itself.



Why use CSS

- Before CSS, an HTML developer changed fonts and colors by changing attributes on each element.
 - If the developer wanted all the headings to look a certain way, she had to change each of those headings.
 - Imagine doing this on a page with ten headings, and then imagine doing it on 50 pages.
- CSS alleviates this burden of individually updating elements and makes it so that *you can apply one single style across one or more elements*.
 - You can apply multiple styles to the same element, and you can target a certain style down to the individual element.
 - For example, if you want all headings to be bold font but a certain heading should have italic, you can do that with CSS.

Separating content from presentation

Without CSS

Formats are set as tag attributes

```
<h2 align="center">
  <font color="blue" size="3"
    face="Times New Roman, serif">
    <i>Web programming</i>
  </font> |
</h2>
```

With CSS

Define a new h2 format in a separate style sheet:

```
<style type="text/css">
  h2 {
    text-align: center;
    color: blue;
    font: italic large "Times New Roman", serif;
  }
</style type="text/css">
```

Automatically applied to your header

```
<h2 > Web programming </h2>
```

CSS styles

A CSS is a *set of rules*.

Each rule is formed by:

- ✓ The Selector (style name)
- ✓ Declaration (defines the style)
 - ✓ Property
 - Value

With CSS we can ...

- ✓ Redefine HTML tag styles
- ✓ Create personalized styles for generic use (Classes)
- ✓ Create styles for a specific HTML tag

```
h2 {
    text-align: center;
    color: blue;
    font: italic large "Times New Roman", serif;
}

.highlightedText{
    font-family: Arial, Helvetica, sans-serif;
    font-size: 12px;
    font-style: normal;
    font-weight: bold;
    color: #000000;
}

#logo {
    background-image: url("/img/logo.gif");
    background-position:center;
    background-repeat:no-repeat;
    height: 50px; width: 150px;
    position: absolute; left: 0px; top: 0px;
```



CSS Syntax

CSS syntax is made up of 5 parts

1. Selector
2. Property/Value
3. Declaration
4. Declaration block
5. Curly braces

Selectors

- Selectors are used to ***declare which part of the markup a style applies to***, a kind of match expression.
- **3 types of selectors**
 - 1) Tag (or Type) selectors (body, p, div, a): *Redefine existing tags*
 - 2) Class selectors (.content, .menu): *Define a new type of tag*
 - 3) ID selectors (#wrapper, #sidebar): *Create and name a new tag*
- The selector is normally the HTML element you want to style
- Selectors should never start with a number, nor should they have spaces in them

Tag selectors

Target elements by their element type

CSS

```
div { ... }
```

HTML

```
<div>...</div>
```

```
<div>...</div>
```

Example

```
span{  
    background-color:  
    skyblue;  
}
```

```
<span>Here's a span with  
    some text </span>  
<p>Here's a p with some  
    text </p>
```

Class selectors

Allow selecting an element based on the element's class attribute value.

CSS

```
.awesome { ... }
```

HTML

```
<div class="awesome">...</div>
```

Example

```
.center {text-align:center;}  
p.center {text-align:left;}
```

```
<h1 class="center">Center-  
aligned heading</h1>
```

```
<p class="center">Center-  
aligned paragraph.</p>
```

```
<p class="left">Center-  
aligned paragraph.</p>
```

ID selectors

More precise than class selectors, as they target only one unique element at a time

CSS

```
#anacidre { ... }
```

HTML

```
<div id="anacidre">...</div>
```

Example

```
#wrapper  
{text-align:center;  
border:1px solid red;  
Width: 200px;  
Height: 100px  
}
```

```
<div id="wrapper">... </div>
```

Class vs ID selectors

- The difference between an ID and a class is that
 - an ID can be used to identify ***one element***,
 - whereas a class can be used to identify ***more than one***.

```
#top { background-color: #ccc;  
padding: 20px }
```

```
.intro { color: red;  
font-weight: bold; }
```

```
<div id="top">  
<h1>Chocolate curry</h1>  
<p class="intro">  
This is my recipe for  
making curry purely  
with chocolate</p>  
<p class="intro">Mmm mm mmmmm</p>  
</div>
```

How CSS styles are used

✓ Inline Styles

Use the “style” attribute to redefine an HTML element individually.

```
<h2 style="color: blue; background: green;">  
    GIE Mod. 2. Web Programming  
</h2>
```

✓ Embedded Style

Define CSS rule from within an HTML document.

It can be applied to any element **in this document**.

```
<head>  
    <style type="text/css">  
        h2 {  
            font-style: italic;  
            font-weight: bold;  
            color: blue;  
        }  
    </style>  
</head>  
<body>  
    <h2> GIE Mod. 2. Web Programming</h2>  
</body>
```

✓ External Style Sheets

Independent CSS files which one references from the HTML documents that are being used

```
<head>  
<link rel="stylesheet" type="text/css" href="estilos.css">  
</head>
```

An exemple CSS file

```
/* Applies to the entire body of the HTML document (except where overridden
   by more specific selectors). */
body {
    margin: 25px;
    background-color: rgb(240,240,240);
    font-family: arial, sans-serif;
    font-size: 14px;
}

/* Applies to all <h1>...</h1> elements. */
h1 {
    font-size: 35px;
    font-weight: normal;
    margin-top: 5px;
}

/* Applies to all elements with <... class="someclass"> specified. */
.someclass { color: red; }

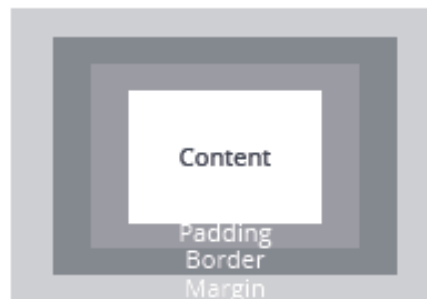
/* Applies to the element with <... id="someid"> specified. */
#someid { color: green; }
```

CSS CHEAT SHEET

SELECTORS

*	All elements
div	<div>
div*	All elements within <div>
div span	 within <div>
div, span	<div> and
div > span	 with parent <div>
div + span	 preceded by <div>
.class	Elements of class "class"
div.class	<div> of class "class"
#itemid	<div> with "itemid"
div#itemid	<div> with "itemid"
a[attr]	<a> with attribute "attr"
a[lang ='en']	<a> when lang begins "en"

BOX MODEL



TEXT MARKUP

margin	border-color
margin-top	border-top-color
margin-right	border-right-color
margin-bottom	border-bottom-color
margin-left	border-left-color
padding	border-style
padding-top	border-top-style
padding-right	border-right-style
padding-bottom	border-bottom-style
padding-left	border-left-style
border	border-width
border-top	border-top-width
border-bottom	border-right-width
border-right	border-bottom-width
border-left	border-left-width

PSEUDO-SELECTORS /CLASSES

:first-child	:link
:first-line	:visited
:first-letter	:lang(var)
:hover	:before
:active	:after
:focus	

FONTS

font	font-weight
font-family	font-stretch
font-style	font-size
font-variant	font-size-adjust

POSITIONING

display	clear
position	z-index
top	direction +
right	unicode-bidi
bottom	overflow
left	clip
float	visibility

COLOR/BACKGROUND

color
background
background-color
background-attachment
background-repeat
background-image
background-position

TEXT

text-indent	word-spacing
text-align	text-transform
text-decoration	white-space
text-shadow	line-height
letter-spacing	

TABLES

caption-side	border-spacing
table-layout	empty-cells
border-collapse	speak-header

DIMENSIONS

width	min-height
min-width	max-height
max-width	vertical-align
height	



Exercise

- Assuming you have created a basic web site with, at least two pages, create a CSS file that modifies, at least:
 - Color, font and text properties
 - Affecting to general elements such as the body and particular ones such as headings
 - Link that CSS to all pages in the site
- Create a new CSS file by making some changes to the first one.
 - Link each file to a different page
 - Link both CSS to one of the pages. See the effects of cascading.