

# Style and Layout in the Web

## CSS and Bootstrap

Add some style to our webpages. Intro to the Bootstrap framework.

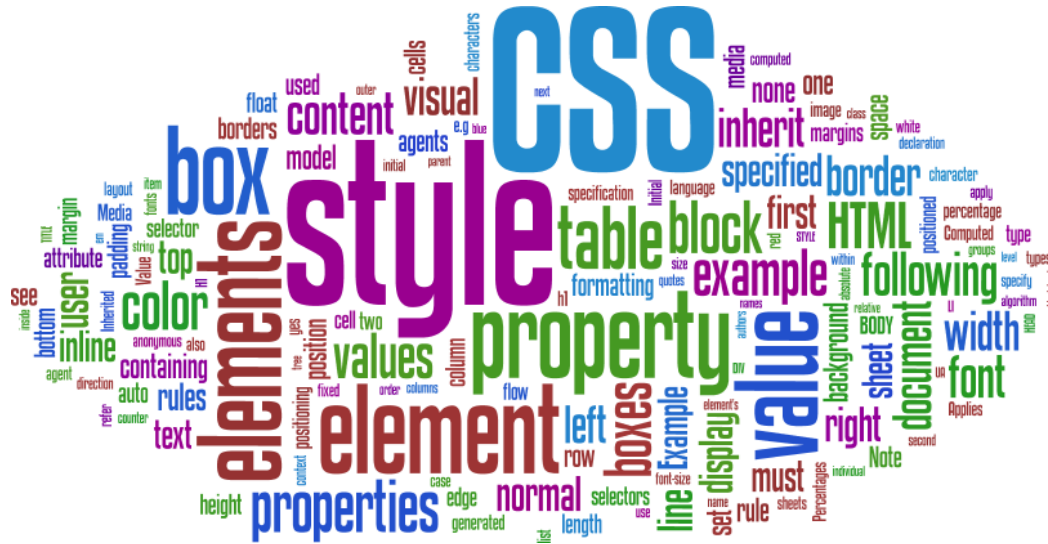


POLITECNICO  
DI TORINO



# Goal

- Styling web content
- Advanced layout in web pages
- Responsive layouts
- Libraries



Style & Layout in the web

# CSS – CASCADING STYLE SHEETS

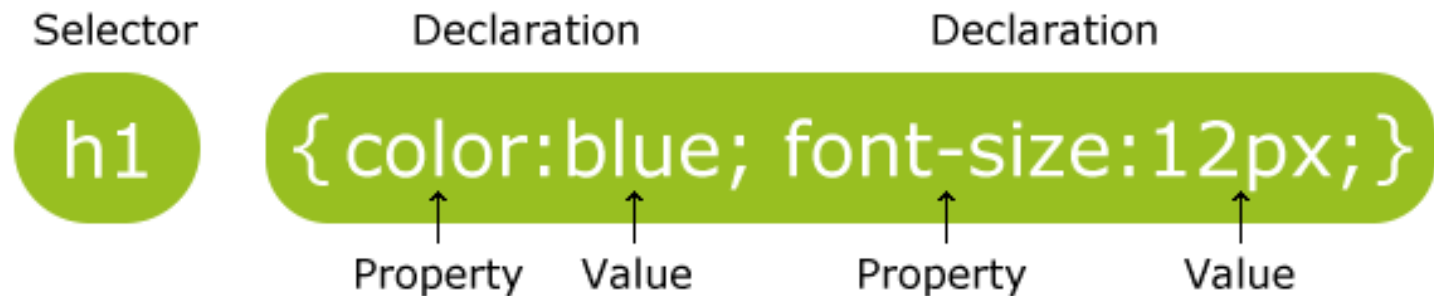
# Cascading Style Sheets



- CSS: Cascading Style Sheet
- CSS 1: W3C recommendation (17 Dec 1996)
- CSS 2.1: W3C Recommendation (7 June 2011)
- CSS 3: different stages (REC, PR, CR, WD)
  - see <http://www.w3.org/Style/CSS/current-work>
- Resources:
  - CSS 2.1 standard, <http://www.w3.org/TR/CSS21/>
  - W3C CSS Home: <http://www.w3.org/Style/CSS/>
  - W3C CSS Tutorial, <http://www.w3.org/Style/Examples/011/firstcss>

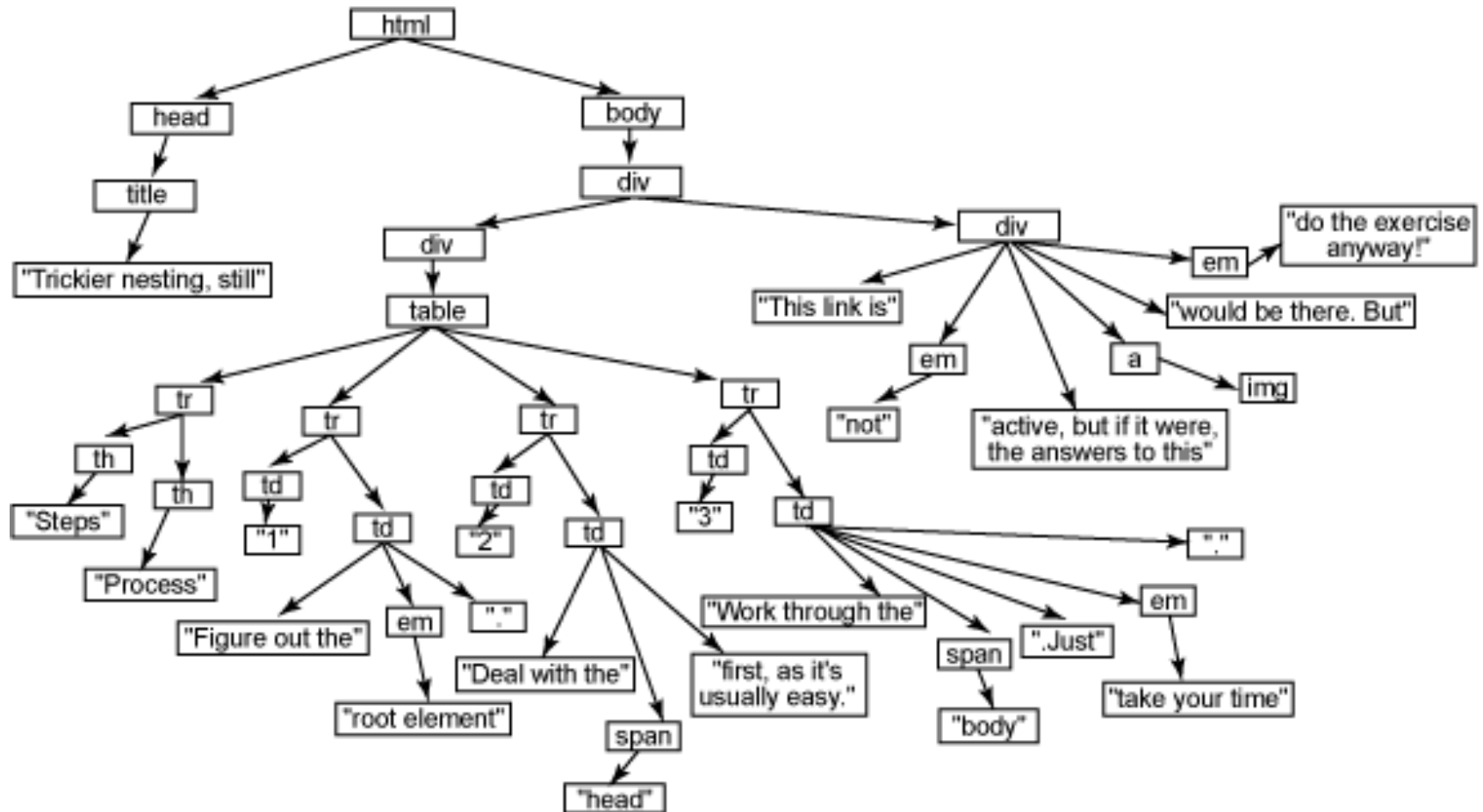
# CSS Syntax

- CSS is based on rules
- A rule is a statement about one stylistic aspect of one or more HTML elements
- A style sheet is a set of one or more rules that apply to an HTML document



# Tree structure

- HTML documents are trees



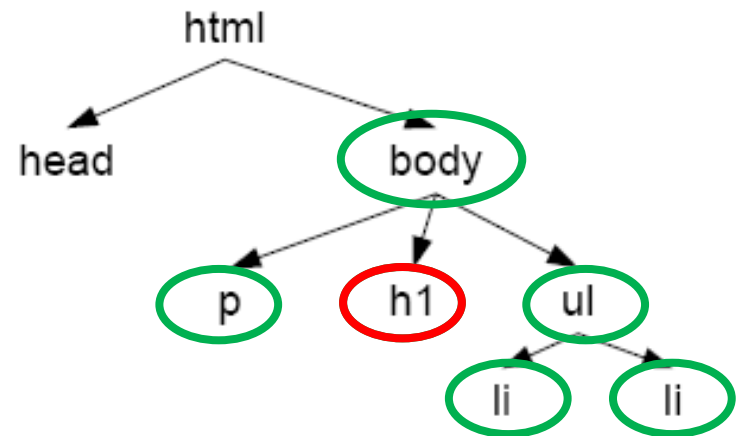
# Tree structure and inheritance

- XHTML documents are trees
- Styles are inherited along trees

- When two rules are in conflict the most specific wins
- Example

– body {color: green}

– h1 {color: red}



# Cascading Style Sheets

- The term “cascading” means that a document can include more than one style sheet
- In this case, visualization follows **priority** rules





# External style

- Link to an external style sheet using the `<link>` element

```
h1 { font-size:17px;
      font-family:verdana; color:green; }
h2 { font-size:18px;
      font-family:arial; color:red; }
```

style.css

```
<head>
<link rel=stylesheet type="text/css"
      href="style.css">
</head>
<body>
<h1>Green text on verdana 17 pixel font</h1>
<h2>Red text on arial a 18 pixel font</h2>
</body>
```

# External style

- Alternative method
- @import directive in the <style> element

```
<head>
  <style>
    @import url(style.css);
  </style>
</head>
<body>
  ...
</body>
```

# Internal style

- `<style>` element inside the document header

```
<head>
<style type="text/css">
h1 { font-size:17px; font-family:verdana;
color:green; }
h2 { font-size:18px; font-family:arial;
color:red; }
</style>
</head>
```

# Inline style

- `<style>` attribute within an HTML element

```
<h1 style="font-size:17px;  
    font-family:verdana; color:green; "> Green text  
on verdana 17 pixel font </h1>
```

# CSS properties

- <http://www.w3schools.com/cssref/>

## CSS Property Groups

- [Color](#)
- [Background and Borders](#)
- [Basic Box](#)
- [Flexible Box](#)
- [Text](#)
- [Text Decoration](#)
- [Fonts](#)
- [Writing Modes](#)
- [Table](#)
- [Lists and Cour](#)
- [Animation](#)
- [Transform](#)
- [Transition](#)
- [Basic User Int](#)
- [Multi-column](#)

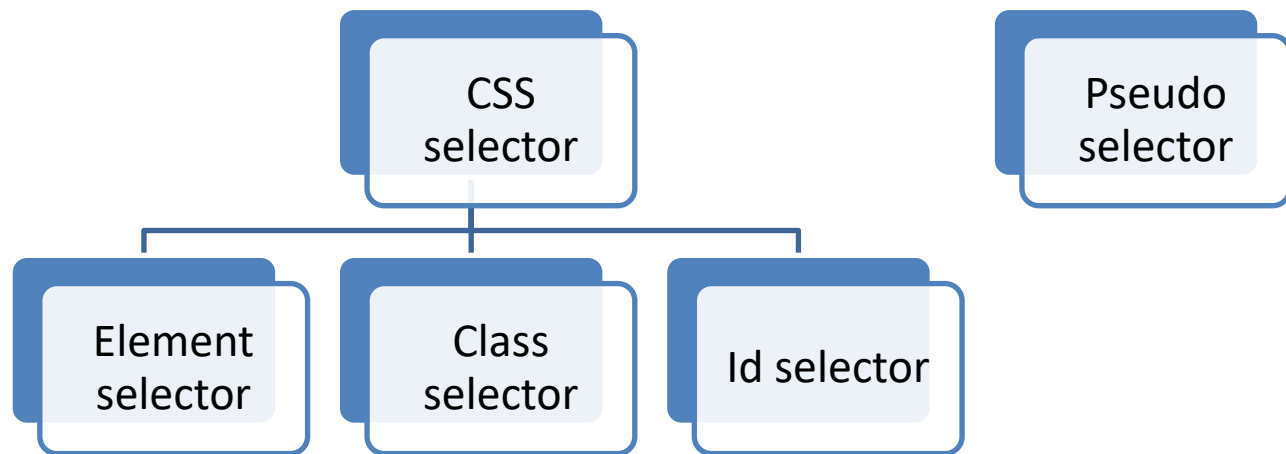
## Color Properties

Property	Description	CSS
<a href="#">color</a>	Sets the color of text	1
<a href="#">opacity</a>	Sets the opacity level for an element	3

<b>Default value:</b>	<i>not specified</i>
<b>Inherited:</b>	yes
<b>Animatable:</b>	yes. <a href="#">Read about animatable</a>
<b>Version:</b>	CSS1
<b>JavaScript syntax:</b>	<code>object.style.color="#0000FF"</code>

# CSS selectors

- Three types of selectors plus “pseudo-selectors”



# Main Selectors

	CSS Selector	HTML code
Element	<code>E</code>	<code>&lt;E&gt;</code>
Class	<code>.cc</code>	<code>&lt;E class="cc"&gt;</code>
ID	<code>#uu</code>	<code>&lt;E id="uu"&gt;</code>

## CSS2 Selectors

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

## CSS2 Pseudo Selectors and Pseudo Classes

:first-child	First child element
:first-line	First line of element
:first-letter	First letter of element
:hover	Element with mouse over
:active	Active element
:focus	Element with focus
:link	Unvisited links
:visited	Visited links
:lang(var)	Element with language "var"
:before	Before element
:after	After element



# Pseudo class selector

- Used to style an element based on something other than the structure of the document
  - e.g., the state of a form element or link

```
/* makes all unvisited links blue */  
a:link {color:blue;}  
/* makes all visited links green */  
a:visited {color:green;}  
/* makes links red when hovered or activated */  
a:hover, a:active {color:red;}  
/* makes table rows red when hovered over */  
tr:hover {background-color: red;}  
/* makes input elements yellow when focus is applied */  
input:focus {background-color:yellow;}
```

# Meaningful XHTML

- Meaningful elements
  - h1, h2, ...
  - ul, ol, and dl
  - strong and em
  - blockquote and cite
  - abbr, acronym, and code
  - fieldset, legend, and label
  - caption, thead, tbody, and tfoot
  - HTML5 adds many new “semantic” elements
- **id** and **class** names
  - Allow to give extra meaning
- **div** and **span**
  - Add structure to document

# DIV element

- Stands for “division”
- Used to group block-level elements
  - Provides a way of dividing a document into meaningful areas
- Use only if necessary and not redundant

```
<div id="mainNav">
  <ul>
    <li>Home</li>
    <li>About Us</li>
    <li>Contact</li>
  </ul>
</div>
```



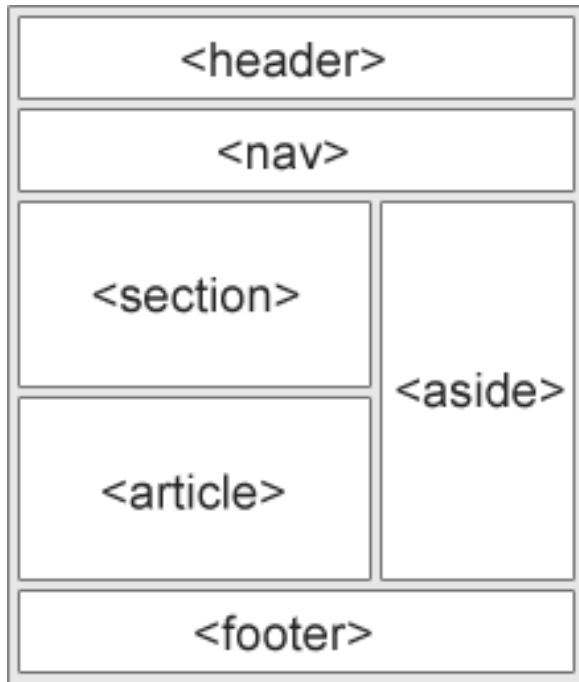
```
<ul id="mainNav">
  <li>Home</li>
  <li>About Us</li>
  <li>Contact</li>
</ul>
```

# SPAN element

- Used to group or identify inline elements

```
<h2>Where's Durstan?</h2>  
<p>Published on  
    <span class="date">March 22nd, 2005</span>  
by <span class="author">Andy Budd</span></p>
```

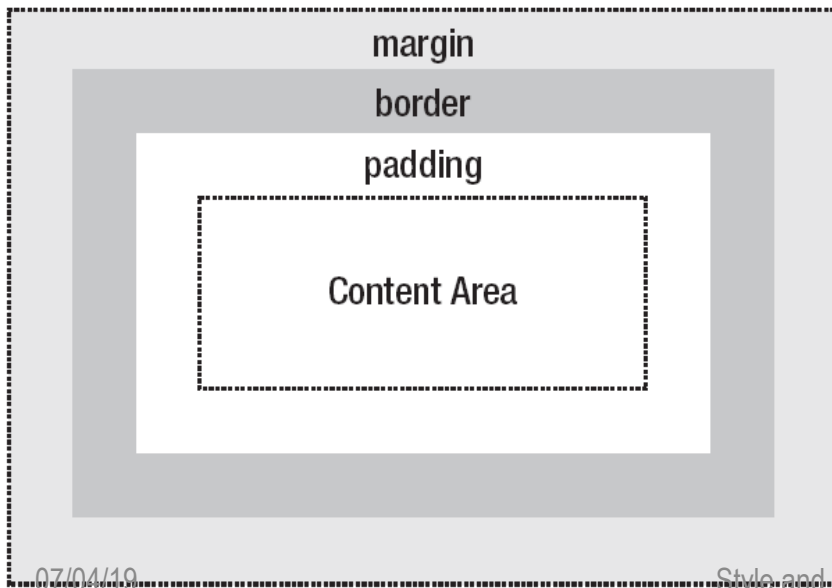
# HTML5 semantic tags



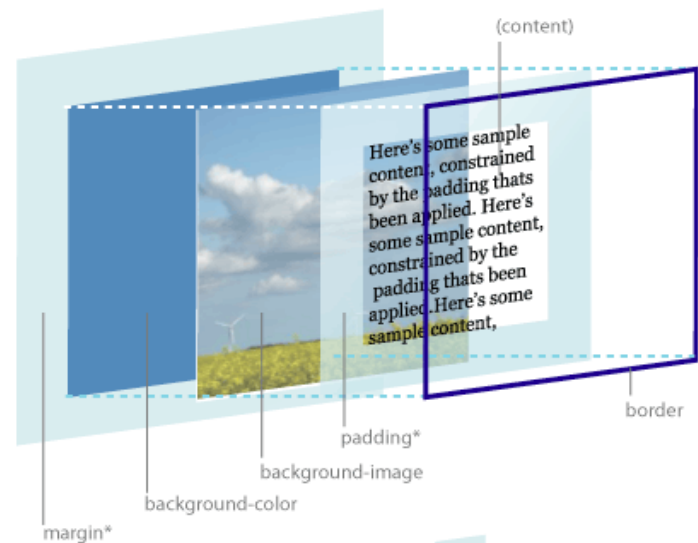
- <header>: defines a header for a document or a section
- <nav>: defines a container for navigation links
- <section>: defines a section in a document
- <article>: defines an independent self-contained article
- <aside>: defines content aside from the content (like a sidebar)
- <footer>: defines a footer for a document or a section
- <details>: defines additional details
- <summary>: defines a heading for the <details> element

# The box model

- One of the cornerstones of CSS
- Dictates how elements are displayed and, to a certain extent, how they interact with each other
- Every element on the page is considered to be a rectangular box



THE CSS BOX MODEL HIERARCHY



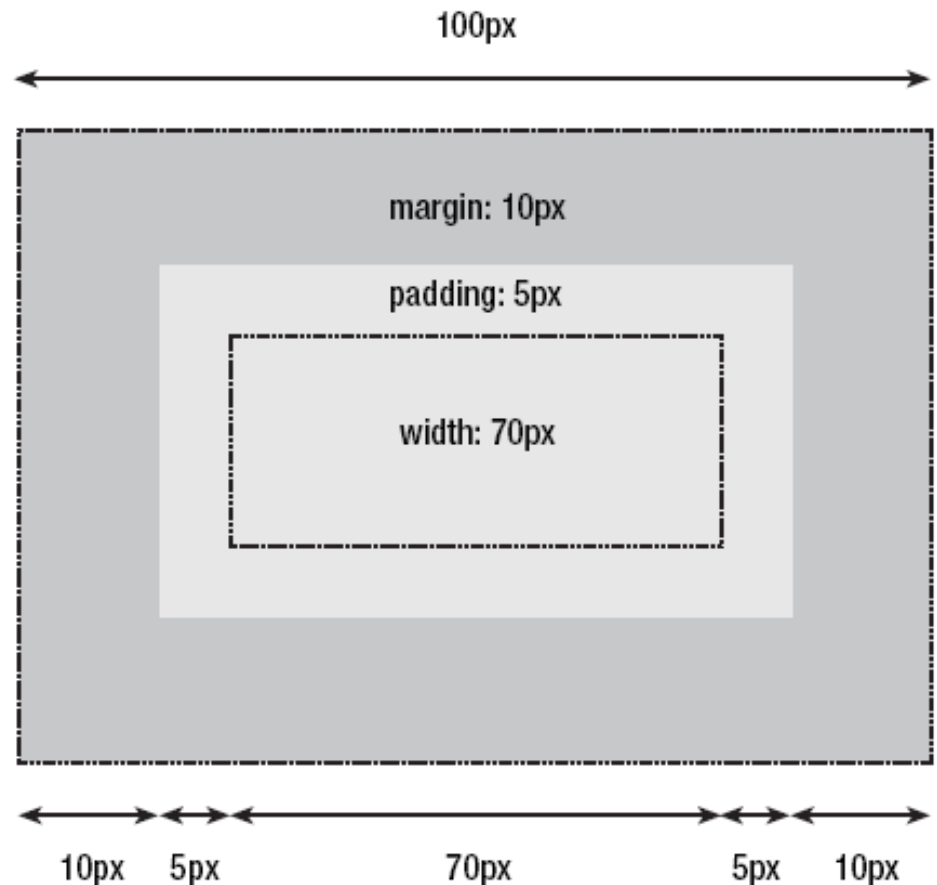
# The box model

- Content
  - The content of the box, where text and images appear
- Padding
  - Clears an area around the content
  - The padding is affected by the background color of the box
- Border
  - A border that goes around the padding and content
  - The border is affected by the background color of the box
- Margin
  - Clears an area around the border
  - The margin does not have a background color, it is completely transparent

# Example

- Padding, borders, and margins are optional and default to zero

```
#myBox {  
  margin: 10px;  
  padding: 5px;  
  width: 70px;  
}
```





# Positioning schemes

- Three basic positioning schemes in CSS
  - Normal flow
  - Floats
  - Absolute positioning
- Unless specified, all boxes start life being positioned in the normal flow
  - The position of an element's box in the normal flow will be dictated by that element's position in the (X)HTML

# Normal flow

- **Block-level** boxes will appear vertically one after the other

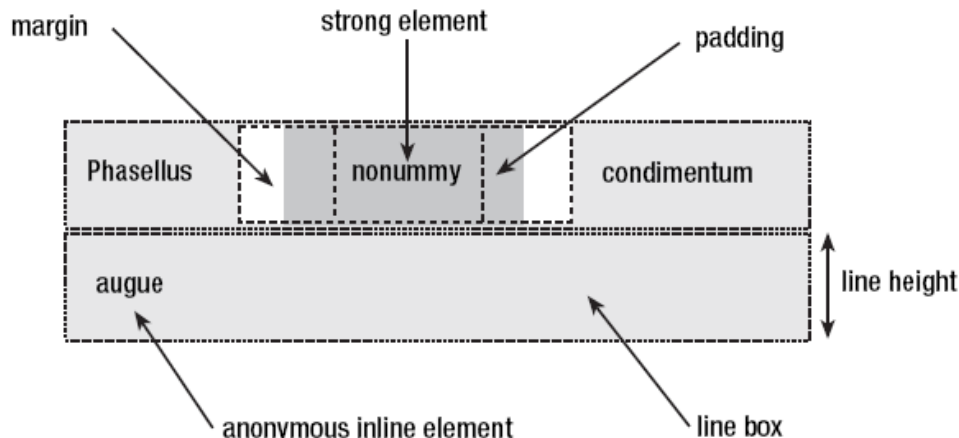
```
<div> ... </div>
```

- The vertical distance between boxes is calculated by the boxes' vertical margins

- **Inline** boxes are laid out in a line horizontally

```
<span> ... </span>
```

- Their horizontal spacing can be adjusted using horizontal padding, borders, and margins
- Vertical padding, borders, and margins will have no effect on the height of an inline box



# Display property

- Allows to control element visualization (block or inline)
- Changing an inline element to a block element, or vice versa, can be useful for making the page look a specific way
- Example
  - W3Schools.com
  - [http://www.w3schools.com/Css/css\\_display\\_visibility.asp](http://www.w3schools.com/Css/css_display_visibility.asp)

```
li {display:inline;}
```

```
div {display:none;}
```

```
span {display:block;}
```

# Box Positioning

- A block can be positioned in different ways to which correspond different positioning schemes
  - **Static**: normal block
  - **Relative**: the offset values are relative to the block position in the normal flow. If a relative block B follows a relative block A, the offset is respect to the position of A without the offset
  - **Absolute**: the box position is determined by the top, left, right, bottom properties and is relative to the containing block
  - **Fixed**: the box is fixed with respect to some reference (the viewport as an example)

# Examples

- W3Schools.com
  - [http://www.w3schools.com/Css/css\\_positioning.asp](http://www.w3schools.com/Css/css_positioning.asp)

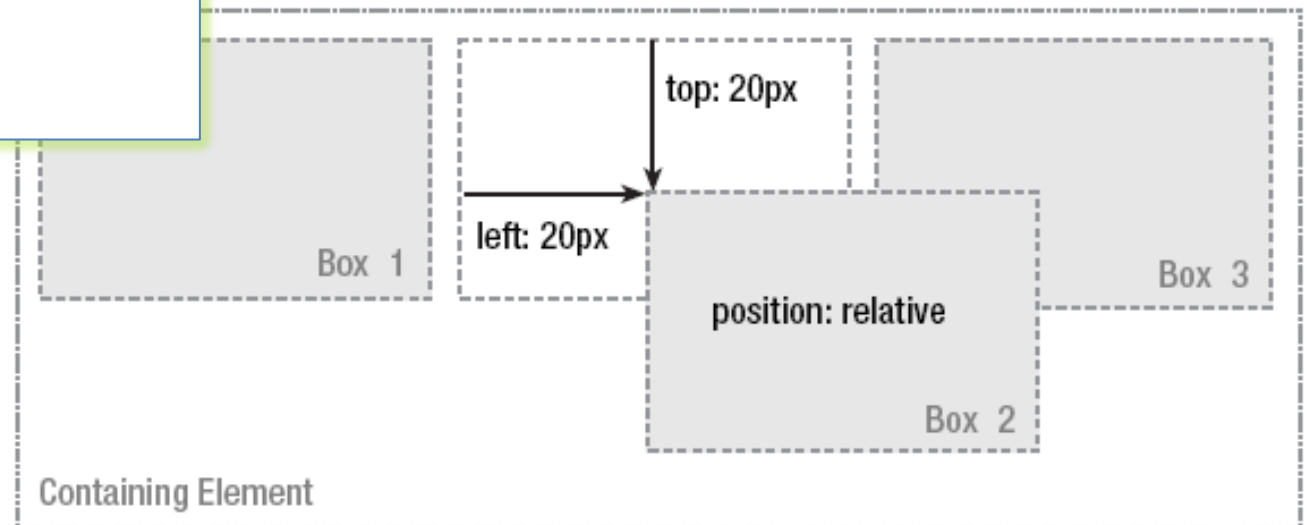
*The main problem people have with positioning is remembering which type of positioning is which. Relative positioning is “relative” to the element’s initial position in the flow of the document, whereas absolute positioning is “relative” to nearest positioned ancestor or, if one doesn’t exist, the initial container block.*

A. Budd, C. Moll, S. Collison,  
“CSS Mastery: Advanced Web Standards  
Solutions”, FriendsOfED, 2006

# Relative positioning

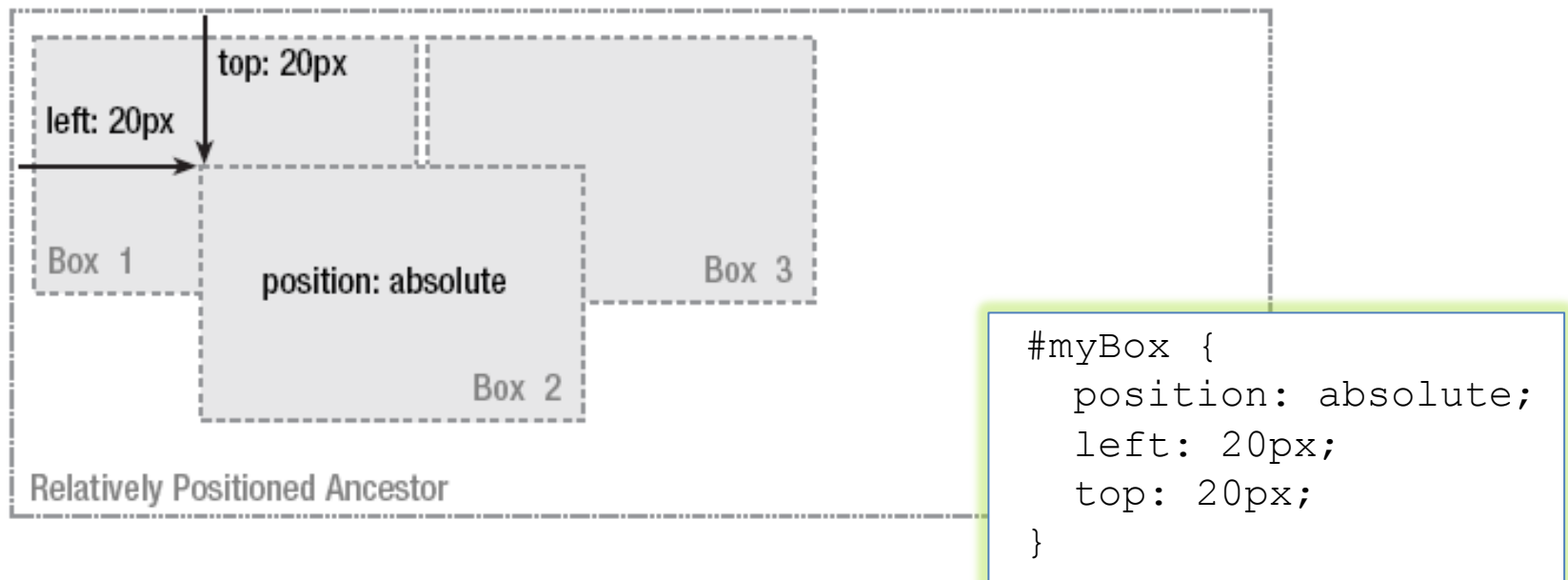
- It is possible to shift one element “relative” to its starting point by setting a vertical or horizontal offset

```
#myBox {  
  position: relative;  
  left: 20px;  
  top: 20px;  
}
```



# Absolute positioning

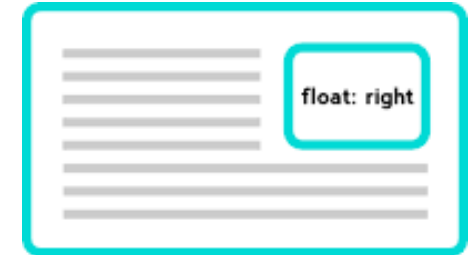
- Takes the element out of the flow of the document, thus taking up no space
- Other elements in the normal flow of the document will act as though the absolutely positioned element was never there



# Floating

- A floated box can either be shifted to the left or the right until its outer edge touches the edge of its containing box, or another floated box
- Often used for images and when working with layouts
  - [http://www.w3schools.com/Css/css\\_float.asp](http://www.w3schools.com/Css/css_float.asp)

```
img
{
  float:right;
}
```



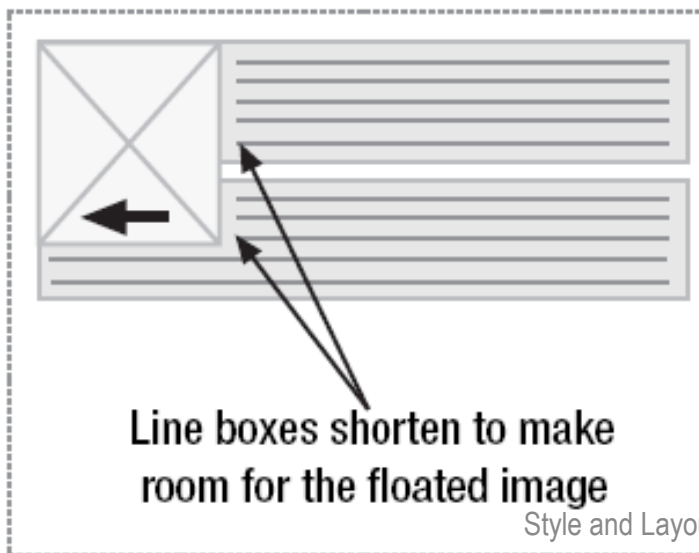


# Line boxes and clearing

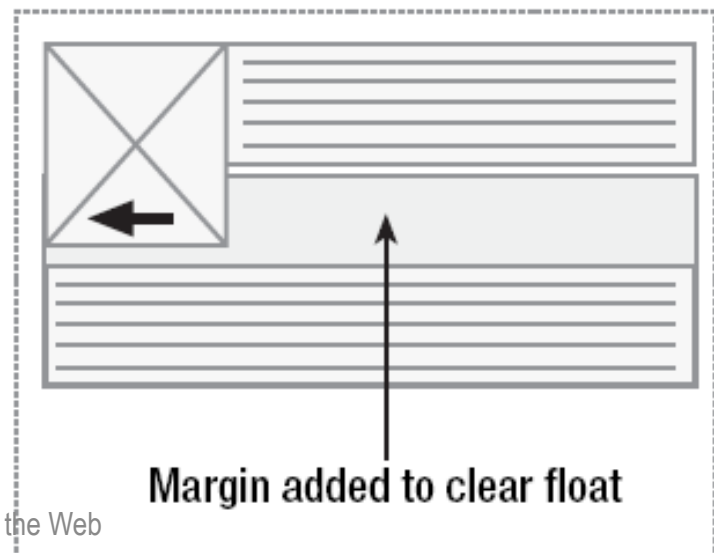
- To stop line boxes flowing around the outside of a floated box, you need to apply a clear to that box
  - The clear property can be left, right, both, or none, and indicates which side of the box should not be next to a floated box

```
p { clear: left }
```

**Image floated left**



**Second paragraph cleared**



# References

- Andy Budd, Cameron Moll, Simon Collison, “CSS Mastery, Advanced Web Standards Solutions”
  - [www.cssmastery.com/](http://www.cssmastery.com/)



Style & Layout in the web

# THE BOOTSTRAP FRAMEWORK

# Bootstrap

- Open Source CSS (and Javascript) framework
- Allows applying “modern” styles
  - Sensible and nice-looking defaults
  - Easy to apply custom themes
- Takes care of cross-browser issues
- Simplified layout model
  
- Developed by Twitter
  - <http://getbootstrap.com/>
  - <https://getbootstrap.com/docs/3.3/>

# Bootstrap versions

Bootstrap 3.3.7



Bootstrap 4.1.0



# Bootstrap philosophy

- Based on CCS classes
- Each class applies one “effect”
- Various classes may be combined in the same element
- Wide range of “standard” classes
- Wide range of additional “components”
  - Ready-to use interactive elements or groups of elements
- Mobile-first
  - Responsive

# Using Bootstrap

```
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg320mUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u" crossorigin="anonymous">

<!-- Optional theme -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap-theme.min.css"
integrity="sha384-rHyoN1iRsVXV4nD0JutlInGaslCJuC7uwjduW9SVrLvRYooPp2bWYgmgJQIXwl/Sp" crossorigin="anonymous">

<!-- Latest compiled and minified JavaScript -->
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js" integrity="sha384-
Tc5IQib027qvyjSMFhJ0MaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGNIcPD7Txa" crossorigin="anonymous"></script>
```

<https://getbootstrap.com/docs/3.3/getting-started/>

- 3 files
  - Bootstrap
  - Theme
  - Javascript support
- Or download locally

# Page structure

- Start with **Basic template**

- <https://getbootstrap.com/docs/3.3/getting-started/#template>

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- The above 3 meta tags *must* come first in the head; any other head content must come *after* these
tags -->
    <title>Bootstrap 101 Template</title>

    <!-- Bootstrap -->
    <link href="css/bootstrap.min.css" rel="stylesheet">

    <!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
    <!--[if lt IE 9]>
      <script src="https://oss.maxcdn.com/html5shiv/3.7.3/html5shiv.min.js"></script>
      <script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
    <![endif]-->
  </head>
  <body>
    <h1>Hello, world!</h1>

    <!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
    <!-- Include all compiled plugins (below), or include individual files as needed -->
    <script src="js/bootstrap.min.js"></script>
  </body>
</html>
```

- Or, choose from published **Examples**

- <http://getbootstrap.com/getting-started/#examples>

- Remember to include your page content in a

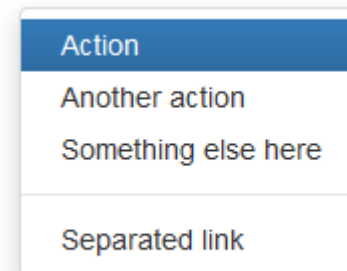
- `<div class="container">`



# Buttons & Menus



<https://getbootstrap.com/docs/3.3/css/#buttons>



<https://getbootstrap.com/docs/3.3/components/#dropdowns>

# Tables

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

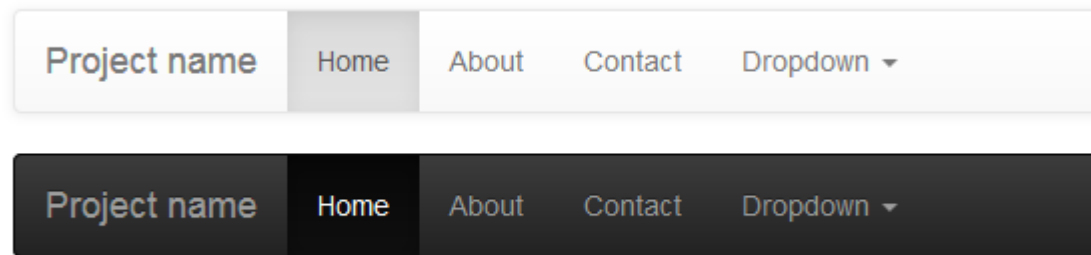
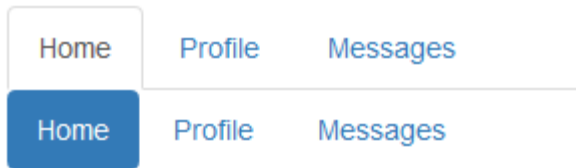
#	First Name	Last Name	Username
1	Mark	Otto	@mdo
	Mark	Otto	@TwBootstrap
2	Jacob	Thornton	@fat
3	Larry the Bird		@twitter

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry	the Bird	@twitter

#	First Name	Last Name	Username
1	Mark	Otto	@mdo
2	Jacob	Thornton	@fat
3	Larry the Bird		@twitter

<https://getbootstrap.com/docs/3.3/css/#tables>

# Navigation bars



# Grids & Columns

.col-md-4

.col-md-4

.col-md-4

.col-md-3

.col-md-6

.col-md-3

.col-md-8

.col-md-4

# More info, examples, help

- <http://www.w3schools.com/bootstrap/default.asp>
- <http://getbootstrap.com/css/>
- <http://getbootstrap.com/components/>

# Questions?

**01QZP AMBIENT INTELLIGENCE**

Fulvio Corno

Luigi De Russis

{name.surname}@polito.it



# License

- This work is licensed under the Creative Commons “Attribution-NonCommercial-ShareAlike Unported (CC BY-NC-SA 4.0)” License.
- You are free:
  - to **Share** - to copy, distribute and transmit the work
  - to **Remix** - to adapt the work
- Under the following conditions:
  - **Attribution** - You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work).
  - **Noncommercial** - You may not use this work for commercial purposes.
  - **Share Alike** - If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one.
- To view a copy of this license, visit <https://creativecommons.org/licenses/by-nc-sa/4.0/>