

HYP TECHNOLOGY / FRONT-END

JAVASCRIPT - JQUERY

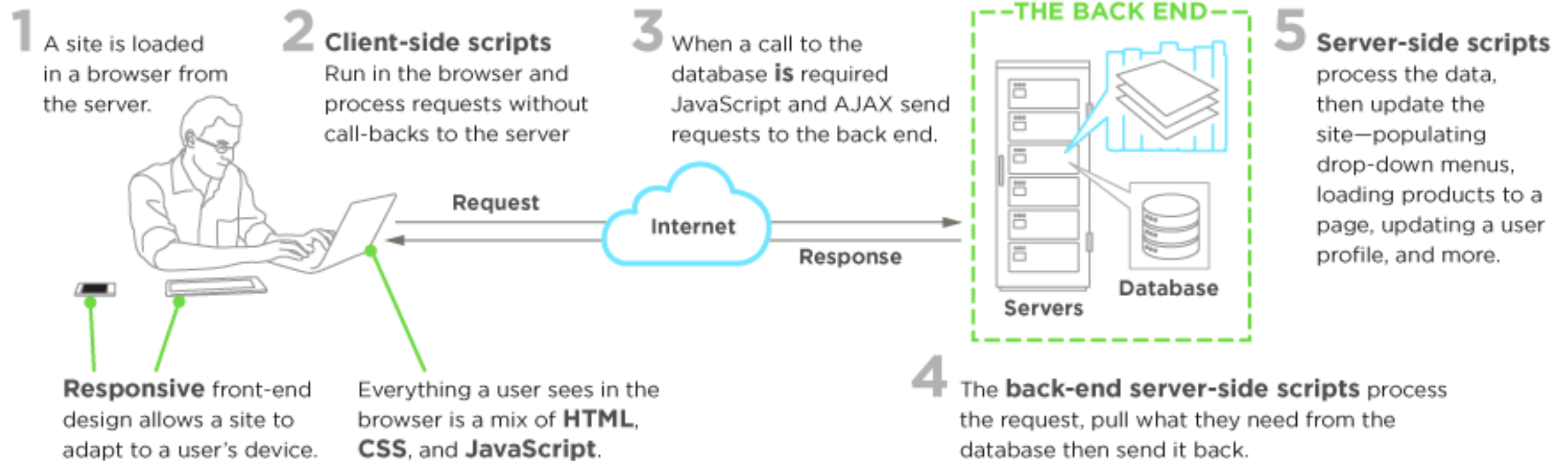
Mirko Gelsomini

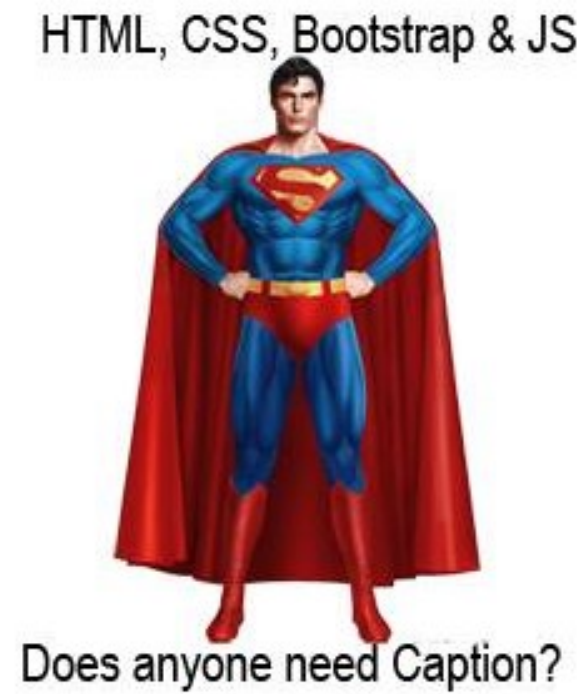
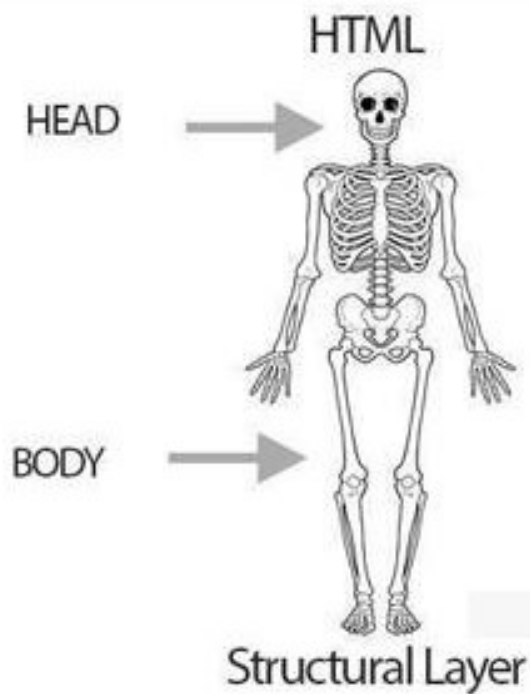
Ways to get help:

- o. Google or StackOverflow
- 1. After class – most immediate way
- 2. Via e-mail: mirko.gelsomini@polimi.it
- 3. private meeting at I3Lab, Via Golgi 39, building 21



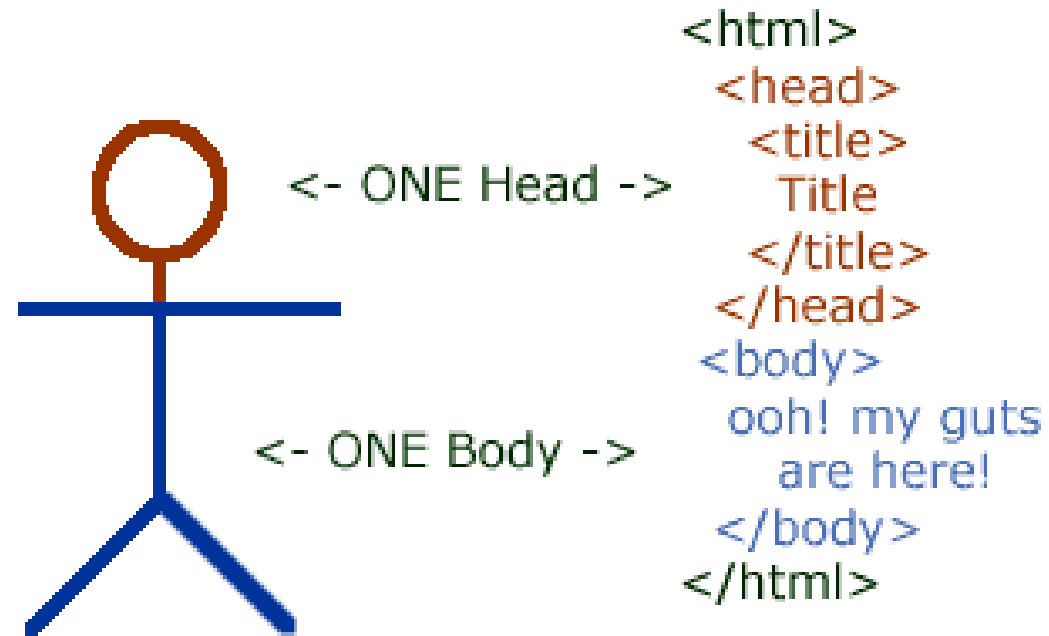
FRONT-END DEVELOPMENT





HTML - HyperText Markup Language

- <!DOCTYPE html>
- <html>
 - <head>
 - </head>
 - <body>
 - </body>
- </html>



And the head always comes first !

Connecting HTML, CSS and JS



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

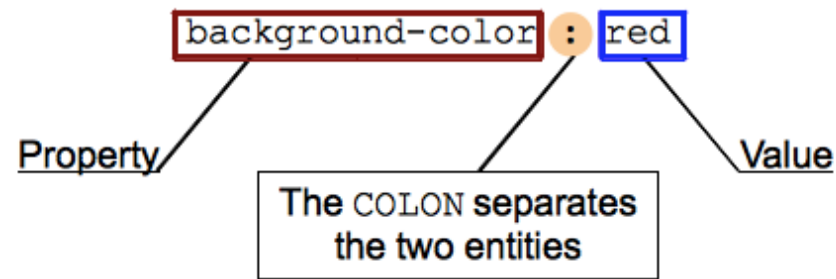


```
<script src="assets/js/script.js"></script>
```

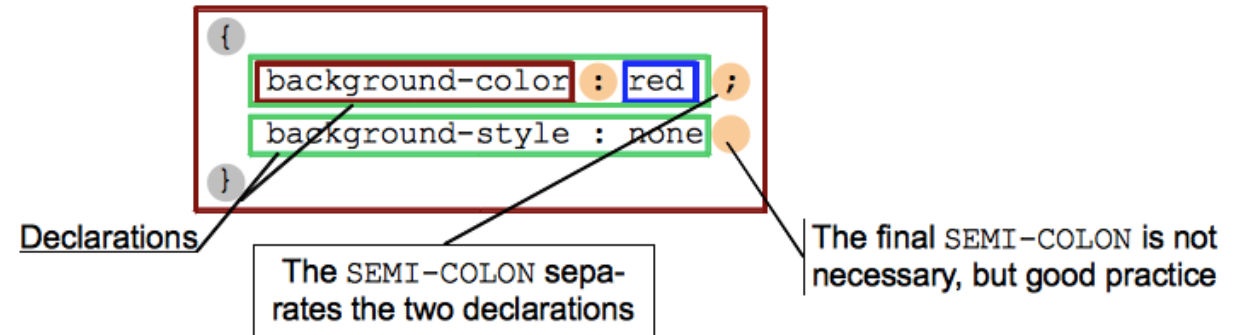


CSS Syntax

A CSS declaration :



A CSS declarations block:



Comments in CSS begin with `/*` and end with `*/`.

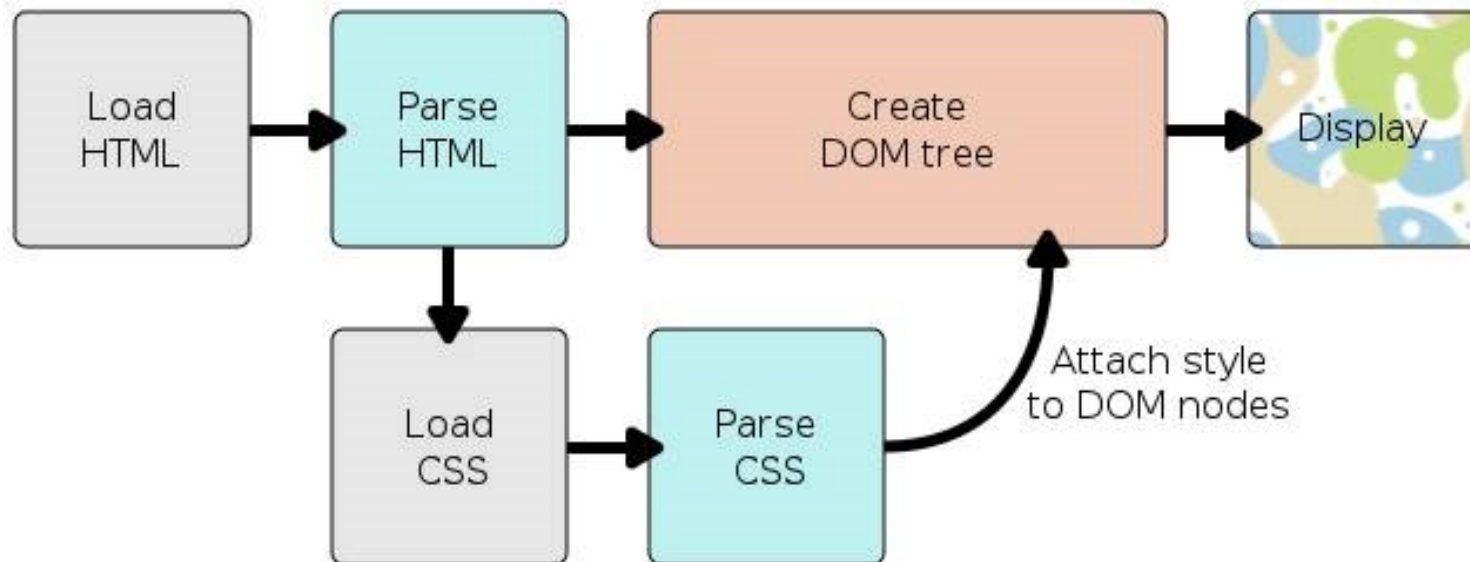
Important:

If a property is unknown or if a value is not valid for a given property, the declaration is deemed *invalid* and is wholly ignored by the browser's CSS engine.

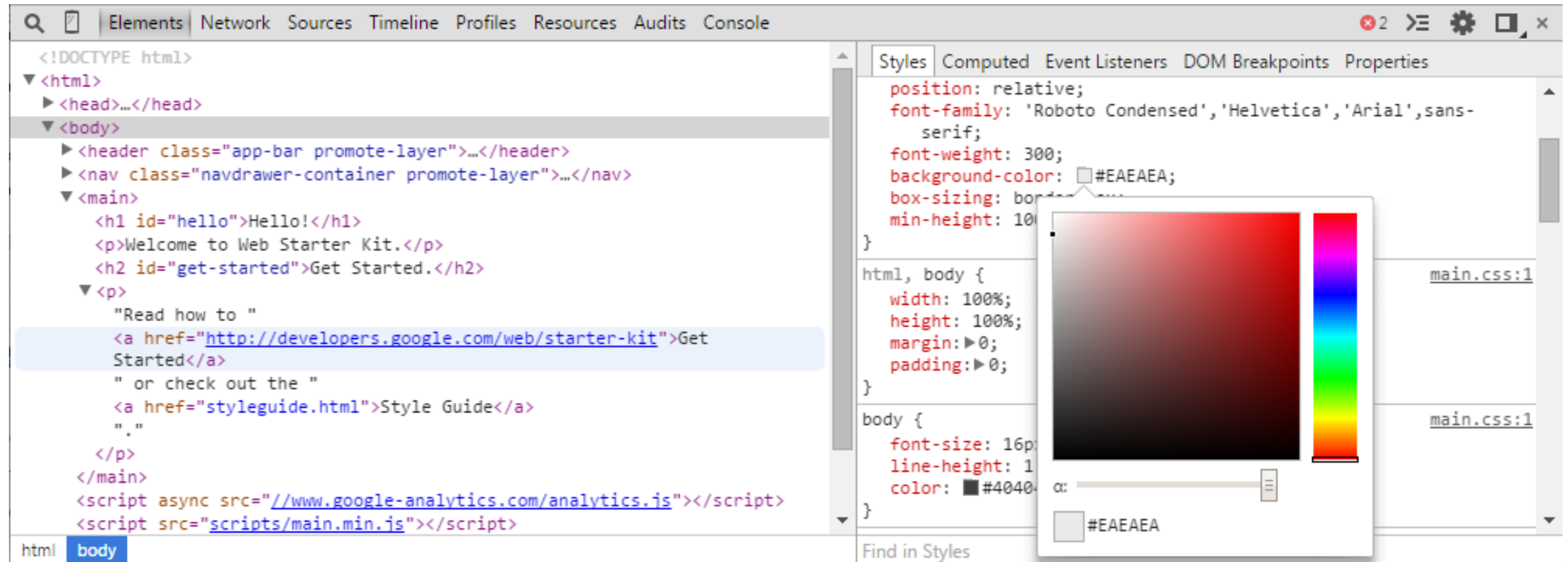
In CSS (and other web standards), US spelling has been agreed on as the standard to stick to where uncertainty arises. For example, color (as seen in the above code) should always be spelt color. colour won't work.

Browser's job

- The browser converts HTML into the DOM (Document Object Model). The DOM represents the document in the computer's memory. It combines the document's content with its style.
- The browser displays the contents of the DOM.



Dev Tools (F12)



Javascript

- JavaScript is the programming language of HTML and the Web.
- JavaScript is easy to learn

Javascript syntax

- JavaScript statements are composed of: Values, Operators, Expressions, Keywords, and Comments.
- JavaScript statements are separated by **semicolons (;)**

```
var n1, n2, sum, prod, c, str, sentence;  
n1 = 5;  
n2 = 6;  
sum = n1 + n2;  
c = 'h';  
str = "ello";  
sentence = c + str;  
prod = myFunction(n1,n2);
```

```
function myFunction(p1, p2) {  
    return p1 * p2;  
}
```

Javascript Objects

```
var person = {  
  firstName: "John",  
  lastName: "Doe",  
  age: 50,  
  eyeColor: "blue",  
  fullName: function() {return this.firstName + " " + this.lastName;}  
};
```

You can access object properties in two ways:

objectName.propertyName

or

objectName["propertyName"]

You access an object method with the following syntax:

objectName.methodName()

JavaScript Statements

```
document.getElementById("demo").innerHTML = "Hello Dolly.";
```

Javascript – Display variables

- JavaScript can "display" data in different ways:
 - Writing into an HTML element, using **innerHTML**.
 - Writing into the HTML output using **document.write()**.
 - Writing into an alert box, using **window.alert()**.
 - Writing into the browser console, using **console.log()**.

jQuery – definition from W3C

- jQuery is a **lightweight**, "write less, do more", **JavaScript library**.
- The purpose of jQuery is to make it much **easier to use** JavaScript on your website.
- jQuery takes a lot of **common tasks** that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.
- jQuery also **simplifies a lot of the complicated things from JavaScript**, like AJAX calls and DOM manipulation.
- The jQuery library contains the following features:
 - HTML/DOM manipulation
 - CSS manipulation
 - HTML event methods
 - Effects and animations
 - AJAX
 - Utilities
- **Tip:** In addition, jQuery has plugins for almost any task out there.

Why jQuery?

- There are lots of other JavaScript frameworks out there, but jQuery seems to be the most popular, and also the most extendable.
- Many of the biggest companies on the Web use jQuery, such as:
 - Google
 - Microsoft
 - IBM
 - Netflix

The jQuery team knows all about **cross-browser issues**, and they have written this knowledge into the jQuery library. jQuery will run exactly the same in all major browsers, including Internet Explorer 6!

jQuery

There are several ways to start using jQuery on your web site. You can:

- Download the jQuery library from jquery.com and save it in your js folder
- Include jQuery from a CDN, like Google

There are two versions of jQuery available for downloading:

- Production version - this is for your live website because it has been minified and compressed
- Development version - this is for testing and development (uncompressed and readable code)

Include jQuery in the head:

```
<head>  
  <script src="jquery-xxx.min.js"></script>  
</head>
```

Or before the end of the body

```
<script src="jquery-xxx.min.js"></script> ← suggested  
</body>
```


HTML tags and CSS Selectors

<p>Hello Mickey!!</p>

```
p {  
  color:red;  
}
```

<p class="blue">Hello Pluto!!</p>

```
.blue{  
  color:blue;  
}
```

<p id="pluto">Hello Pippo!!</p>

```
#pluto{  
  color:green;  
}
```

HTML tags and CSS and JQuery Selectors

HTML

<p>Hello Mickey!!</p>

<p class="blue">Hello Pluto!!</p>

<p id="pluto">Hello Pippo!!</p>

CSS

```
p {  
  color:red;  
}
```

```
.blue {  
  color:blue;  
}
```

```
#pluto {  
  color:green;  
}
```

JQUERY

```
$("p")
```

```
$(".blue")
```

```
$("#pluto")
```

jQuery Syntax

- The jQuery syntax is tailor-made for **selecting** HTML elements and performing some **action** on the element(s).
- Basic syntax is: ***\$(selector).action()***
 - A \$ sign to define/access jQuery
 - A (*selector*) to "query (or find)" HTML elements
 - A jQuery *action()* to be performed on the element(s)

Document Ready Event

```
$(document).ready(function(){  
    // jQuery methods go here...  
});
```

This is to prevent any jQuery code from running before the document is finished loading (is ready). It is good practice to wait for the document to be fully loaded and ready before working with it. This also allows you to have your JavaScript code before the body of your document, in the head section.

Here are some examples of actions that can fail if methods are run before the document is fully loaded:

- Trying to hide an element that is not created yet
- Trying to get the size of an image that is not loaded yet

jQuery Events

- All the different visitor's actions that a web page can respond to are called events.
- An event represents the precise moment when something happens.
- Examples:
 - moving a mouse over an element
 - selecting a radio button
 - clicking on an element
- The term "**fires/fired**" is often used with events. Example: "The keypress event is fired, the moment you press a key".

List of Events

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

```
$("#p").event(function(){  
    //some instructions  
});
```

=

```
$("#p").on("event",function(){  
    //some instructions  
});
```

```
$("#p").click(function(){  
    //some instructions  
});
```

=

```
$("#p").on("click",function(){  
    //some instructions  
});
```

Hide, show, fade, slide

`$(selector).show(speed,callback); / $(selector).hide(speed,callback); / $(selector).toggle(speed,callback);`

`$(selector).fadeIn(speed,callback); / $(selector).fadeOut(speed,callback); / $(selector).fadeToggle(speed,callback);`

`$(selector).slideDown(speed,callback); / $(selector).slideUp(speed,callback); / $(selector).slideToggle(speed,callback);`

Where *speed* = “slow”, “fast”, <milliseconds> - optional field

And *callback* is the function will be called after the effect terminates – optional field

Callback vs Sequential

```
$("#button").click(function(){  
    $("#p").hide("slow", function(){  
        alert("The paragraph is now hidden");  
    });  
});
```

```
$("#button").click(function(){  
    $("#p").hide(1000);  
    alert("The paragraph is now hidden");  
});
```


Chaining

```
$(selector).action1().action2();
```

```
$("#p1")  
  .slideUp(2000)  
  .slideDown(2000);
```

slideUp -> 2000ms -> slideDown -> 2000ms

Get

jQuery methods for DOM manipulation are:

- text() - Sets or returns the **text content** of selected elements
- html() - Sets or returns the **content of selected elements (including HTML markup)**
- val() - Sets or returns the **value** of form fields

```
$("#btn1").click(function(){  
    alert("Text: " + $("#test").text());  
});
```

```
$("#btn2").click(function(){  
    alert("HTML: " + $("#test").html());  
});
```

```
$("#btn3").click(function(){  
    alert("Value: " + $("#test").val());  
});
```

Set

jQuery methods for DOM manipulation are:

- `text()` - Sets or returns the **text content** of selected elements
- `html()` - Sets or returns the **content of selected elements (including HTML markup)**
- `val()` - Sets or returns the **value** of form fields

```
$("#btn1").click(function(){  
    $("#test1").text("Hello world!");  
});
```

```
$("#btn2").click(function(){  
    $("#test2").html("<b>Hello world!</b>");  
});
```

```
$("#btn3").click(function(){  
    $("#test3").val("Dolly Duck");  
});
```

Add, Remove

Add New HTML Content:

- `append()` - Inserts content at the end of the selected elements
- `prepend()` - Inserts content at the beginning of the selected elements
- `after()` - Inserts content after the selected elements
- `before()` - Inserts content before the selected elements

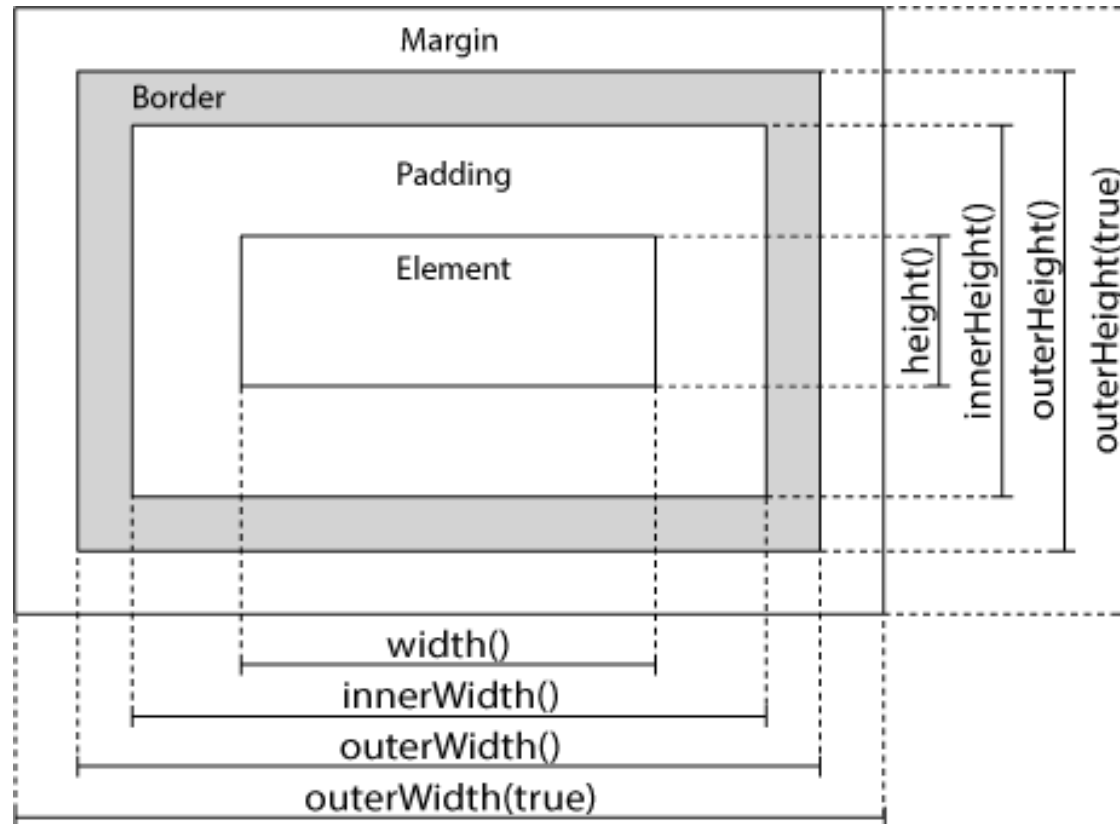
Remove Elements/Content:

- `remove()` - Removes the selected element (and its child elements)
- `empty()` - Removes the child elements from the selected element

Dimension

jQuery has several important methods for working with dimensions:

- `width()`
- `height()`
- `innerWidth()`
- `innerHeight()`
- `outerWidth()`
- `outerHeight()`



Css classes

jQuery has several methods for CSS manipulation

- `addClass()` - Adds one or more classes to the selected elements
- `removeClass()` - Removes one or more classes from the selected elements
- `toggleClass()` - Toggles between adding/removing classes from the selected elements
- `css()` - Sets or returns the style attribute

Set a CSS Property

```
$(element).css("propertyname", "value");
```

```
$("#p").css("background-color", "yellow");
```

JSON

- JSON is a format for storing and transporting data.
- JSON is often used when data is sent from a server to a web page.
- JSON stands for **JavaScript Object Notation**
- JSON is lightweight data interchange format
- JSON is language independent *
- JSON is "self-describing" and easy to understand
- * The JSON syntax is derived from JavaScript object notation syntax, but the JSON format is text only. Code for reading and generating JSON data can be written in any programming language.

Quiz #1

What is jQuery?

1. A wonderful css framework
2. It is Javascript, no?
3. A powerful Javascript Framework

Quiz #2

How can one select an element in jQuery?

- `/(selector)`
- `$(selector)`
- `@(selector)`

Quiz #3

Why one has to use `$(document).ready()`?

- To wait for the DOM to load
- To wait, just a bit
- To hide an element that is not created yet

Quiz #4

Why one has to use `$(document).ready()`?

- To wait for the DOM to load
- To wait, just a bit
- To hide an element that is not created yet

Quiz #5

What is the right term when an event is “called”?

- The event **blows**
- The event **sinks**
- The event **fires**

Quiz #6

What is a “callback”?

- A sequential procedure
- A variable
- A function that is called when a procedure finishes

Quiz #7

Can I set CSS using jQuery?

- No
- Yes
- Maybe, my group mate will surely know how

On the web – in case you missed this class

- <https://developers.google.com/web/fundamentals/> (en - free)
- <http://www.codecademy.com/tracks/web> (en - free)
- <http://www.html.it/guide/guida-html5/> (ita - free)
- <https://www.codeschool.com/paths/html-css> (en)
- <http://teamtreehouse.com/tracks/front-end-web-development> (en)
- Bootstrap: <https://www.tutorialspoint.com/bootstrap/index.htm>
- Official Ref (en):
 - HTML: <http://www.w3schools.com/html/>
 - CSS: <http://www.w3schools.com/css/>

Being stuck?

0. Google (Stack Overflow suggested)
1. Meet Mirko just After class – most immediate way
2. Via e-mail: mirko.gelsomini@polimi.it
3. Meet Mirko at I3Lab, Via Golgi 39, building 21

