



Oum El Bouaghi University

**Faculty of Exact Sciences, Natural and Life Science
Department of Mathematics and Computer Science**

Advanced Web Programming

jQuery Library

Dr. Saighi Asma

Concerned students

Faculty

Department

Level

Speciality

ESNL

MI

B3

ISSE

Birth Context of jQuery

- There are many JavaScript libraries.
 - jQuery is the **most popular and extensible**.
 - Largest web companies use it: Google, Microsoft, IBM, Netflix, etc.
 - jQuery is lightweight, open-source, cross-browser JavaScript library: *write less, do more*.
 - **jQuery goal**: ease use of JavaScript on a web site.



jQuery Framework

- **jQuery Provides functionalities:**
 - HTML/DOM manipulation.
 - HTML event methods.
 - Effects and animations.
 - AJAX requests.
 - CSS manipulation.
- **jQuery strengths:**
 - Power of expression and functional richness.
 - Lightweight (about 94 KB).
 - Free and open-source.
 - Ease of selecting elements on a web page.
 - Ease of AJAX requests.
 - Ability to write code outside of HTML.

First steps with jQuery

- **Different ways to start using jQuery on website:**
 - Downloading jQuery library from the official website: jQuery.com.
 - Including jQuery from Content Delivery Network (CDN).
- **Available versions of jQuery:**
 - Production version
 - Development version
- **jQuery library is a javascript file that can be referenced using the HTML `<script>` tag:**

```
<head>  
<script src="path/jquery-3.5.1.min.js"></script>  
</head>
```

- **jQuery can be included from CDN: example google**

```
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>  
</head>
```

jQuery syntax

- **The use of jQuery involves two steps:**
 - **Selecting** a part of the document whose result is contained in an object.
 - **Acting** upon it.
- Each jQuery object consists of a **set of DOM nodes**  **set of tags.**
- **"jQuery ()" function**, abbreviated as **\$()** to **create** jQuery object :
 - **Input:** string containing a « selector ».
 - **Output:** jQuery object.
- Basic syntax **\$ (selector) .action ()** :
 - **\$** sign: for defining/accessing jQuery
 - **(selector)**: to query HTML elements.
 - jQuery **action()**: for performing the selected element(s).

- **Examlpe**

- `$("#div").hide()` returns an **object** which contains **all "div" elements in the document.**
- `$("#div").hide()` **hides every "div" elements** in the document.
- It is preferable to execute **jQuery methods** from a **"document ready"** event type.

```
$(document).ready(function(){  
    // jQuery methods are invoqued her..  
});
```

Element Selection

- **\$() function:** Magic selector `$('anything')`, offers various possibilities for selecting elements: By block type, By ID, etc.
- **Selection of elements by block type, ID, class:**
 - Returning elements from a document by block type, the name of the targeted tag is passed as an argument to the `$()` function.
- **Example:**
 - To return all tags on the page, the `*` selector is used: `$('*')`
 - To return all `<div>` tags on the page, `$('div')` is used.
- **Selection by combining criteria:** jQuery library returns elements from a document by combining the block type, ID, and class of the targeted tags as arguments to the `$()` function
- **Example:**

```
// all divs with class "main"
$("div.main");
```

- **jQuery:** contains powerful **methods** to modify and manipulate HTML elements and attributes.
- **Retrieval/setting of content:**
 - **Text():** returns the text content of the selected elements.
 - **Html():** returns the content of the selected elements (including HTML markup).
 - **Val():** returns the value of form fields.
 - **Attr():** returns the value of an attribute whose name is passed as an argument.

- **jQuery methods for Adding/Removing HTML Elements:**
 - **Append():** *Inserts* one or more content to the end of the selected elements.
 - **Prepend():** *Inserts* one or more content at the beginning of the selected elements.
 - **After():** *Adds* one or more content after the selected elements.
 - **Before():** *Inserts* content before the selected elements.
 - **Remove():** *Deletes* the selected element.
 - **Empty():** *Removes* the child elements of the selected element.

- **Manipulation of CSS Classes:**

- jQuery offers methods for CSS manipulation:

- **addClass():** *Adds* one or more classes to the selected elements.
- **removeClass():** *Toggles* between add/remove classes.
- **Css():** *Sets or returns* the style attribute.

- It is possible to define multiple style properties using **css() function**:

```
css ({ "propertyname": "value", "propertyname": "value", ... });
```

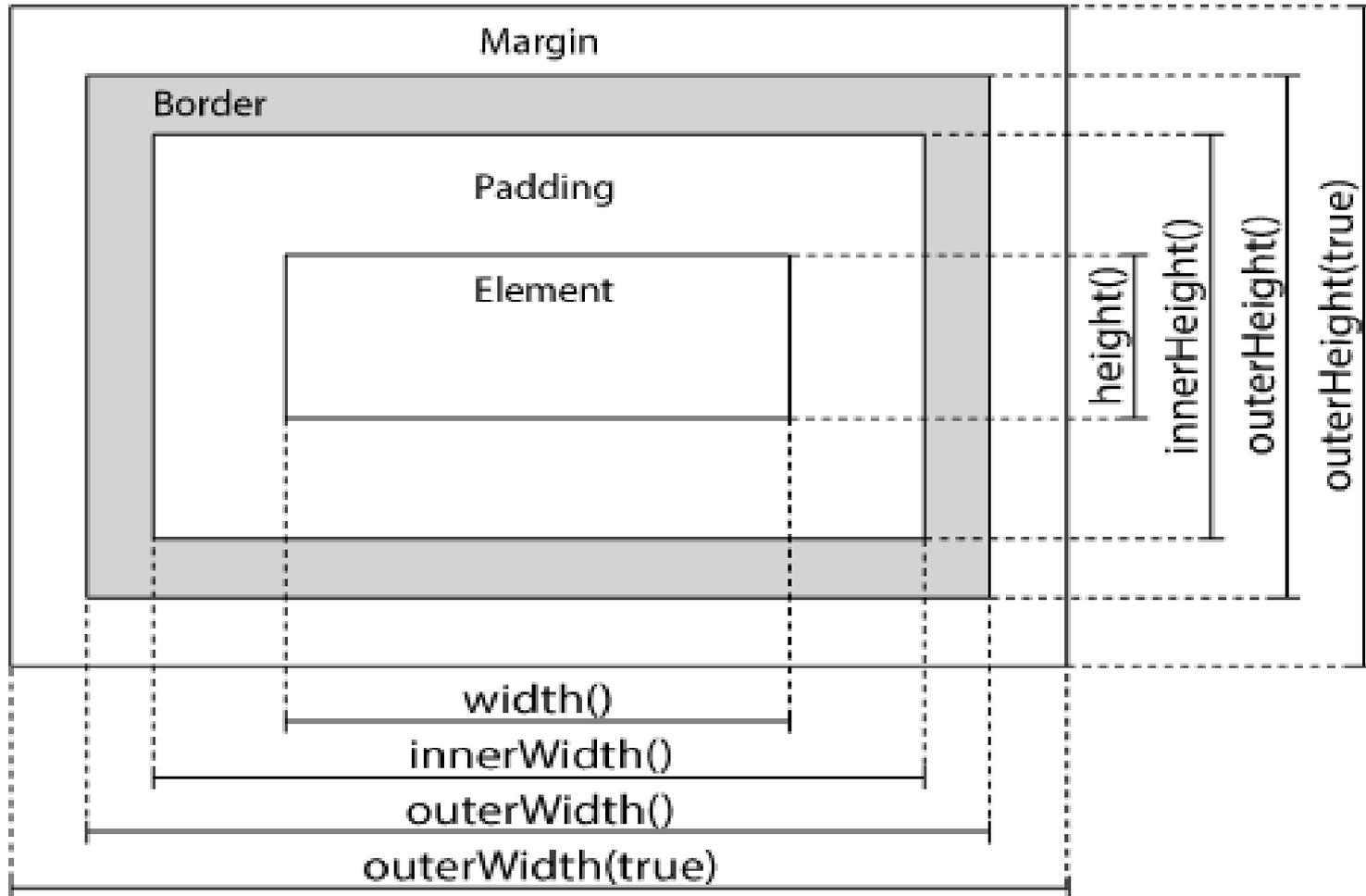
- **Example:**

```
$("#p").css ({ "background-color": "yellow", "font-size": "200%" });
```

- **Dimension Manipulation with jQuery:**
 - jQuery offers **methods** to utilize **dimensions**:
 - **Height():** *Sets or returns* the height of an element without padding, border, and margin.
 - **Width():** *Sets or returns* the width of an element without padding, border, and margin.
 - **innerWidth():** *Returns* the width of an element.
 - **innerHeight():** *Returns* the height of an element.
 - **outerWidth():** *Returns* the width of an element including padding and border.
 - **outerHeight():** *Returns* the height of an element including padding and border.

HTML Manipulation

- Dimension Manipulation with jQuery:



HTML Manipulation

- **Event Handling:**
 - jQuery library is tailor-made to handle events on an HTML page.
 - Events correspond to all the actions of different visitors to which a web page can respond.
- **Some common DOM 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

- **Click Event method: `click ()`**

```
$("#p").click(function() {  
    $(this).hide();  
});
```

- **Double click event method: `dblclick ()`**

```
$("#p").dblclick(function() {  
    $(this).hide();  
});
```

- **Mouse enter event method: `mouseenter ()`**

```
$("#p1").mouseenter(function() {  
    alert("You entered p1!");  
});
```

- **Mouse exit event method : `mouseleave ()`**

```
$("#p1").mouseleave(function() {  
    alert("Bye! You now leave p1! ");  
});
```

HTML Manipulation

- **Mouse down Event method: `mousedown ()`**

```
$("#p1").mousedown(function() {  
    alert("Mouse down over p1!");  
});
```

- **Mouse released event method: `mouseup ()`**

```
$("#p1").mouseup(function() {  
    alert("Mouse up over p1!");  
});
```

- **Event when an HTML element becomes active in the UI: `focus ()`**

```
$("#input").focus(function() {  
    $(this).css("background-color", "#cccccc");  
});
```

- **Event when an HTML element becomes inactive in the UI: `blur()`**

```
$("#input").focus(function() {  
    $(this).css("background-color", "#ffffff");  
});
```

- **Showing/ Hiding HTML Elements:**
 - **show ()** : *Displays* an HTML element.
 - **hide ()** : *Hides* an HTML element.
 - **toggle ()** : *Toggles* between hiding and showing HTML element.
- **Syntax:**
 - `$(selector).show (speed, callback);`
 - `$(selector).hide (speed, callback);`
 - `$(selector).toggle (speed, callback);`
- **'speed'** determines the speed of the action and can take the values: "slow", "fast" or milliseconds.
- **'callback'** executed once the action is completed.