



jQuery



Au Mau Duong
Technical Trainers

September 2013						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

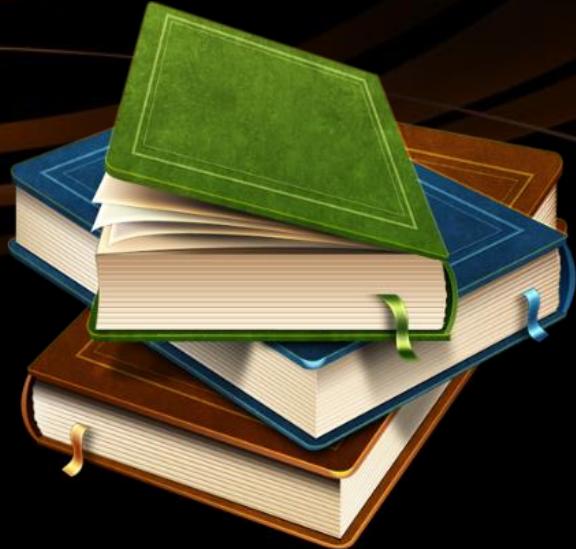
Selectors and DOM Manipulation, jQuery HTML

jQuery



Table of Contents

- What is jQuery?
- Selectors and DOM Manipulation
 - All selector, Multiple selector, Class selector, Id selector
 - Element selector, Filter Selectors, Attribute Selectors
- jQuery HTML
 - Get/Set Content and Attributes
 - Add/Remove Elements
 - Get/Set CSS Classes
 - Dimensions





What is jQuery?

The World's Most Popular JavaScript Library

What is jQuery?

- jQuery is a cross-browser JavaScript library
 - Dramatically simplifies DOM manipulation
 - Simplifies AJAX calls and working with RESTful services
 - Free, open-source software: <https://jquery.com>



```
<script src="https://code.jquery.com/jquery-  
3.1.1.min.js"></script>
```

Load jQuery from
its official CDN

```
$('.li').css('background', '#DDD');
```

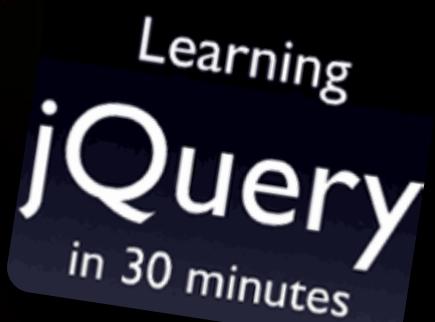
Change the CSS for all tags

Why jQuery is So Popular?

- Easy to learn
- Easy to extend
 - Create new jQuery plugins by writing JavaScript functions
- Powerful DOM Selection
 - Powered by CSS 3.0
- Lightweight
- Community Support
 - Large community of developers and geeks



Selectors and DOM Manipulation



Selection with jQuery

- jQuery **selectors** returns a collection of the items
 - Even if there is only one item
- Can be stored in a variable or used right away

```
$('div') // Like: document.querySelectorAll('div');  
$('.menu-item') // Like: document.querySelectorAll('.menu-item');  
$('#navigation') // Like: document.getElementById('navigation');  
$('ul.menu li') // Like: document.querySelectorAll('ul.menu li');
```

```
$('div').css('background', 'blue'); // Make all DIVs blue
```

- <http://learn.jquery.com/using-jquery-core/selecting-elements/>

jQuery Selectors

- DOM elements selection in jQuery is much like as in JavaScript

- All selector

```
$('*') // Selects all elements
```

- Class selector

```
$('.class') // Selects all elements by class name
```

- Element selector

```
$('section') // Selects all elements by tag name
```

- Id selector

```
$('#id') // Selects a element by given id attribute
```

- Multiple selector

```
($('selector1, selector2') // Combines two selectors
```

Filter Selectors

```
$('li:even') // Even <li>
```

```
$('li:odd') // Odd <li>
```

```
$('li:first') // First <li>
```

```
$('li:last') // Last <li>
```

```
$('li:first-child') // Selects the first child of <li>
```

```
$('li:has(p)') // Selects all <li> holding <p> inside
```

```
$('li:contains("Sofia")') // Selects <li> holding given text
```

```
$('li:eq(2)') // Selects the element at index 2
```

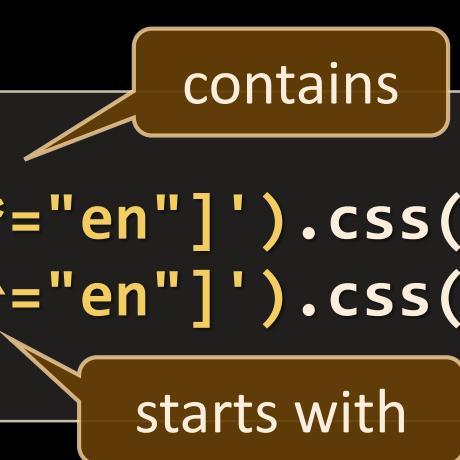
```
$('li:not(:checked)') // Elements not matching the selector
```

Attribute Selectors

- Attribute selectors support "contains" / "starts with" semantic

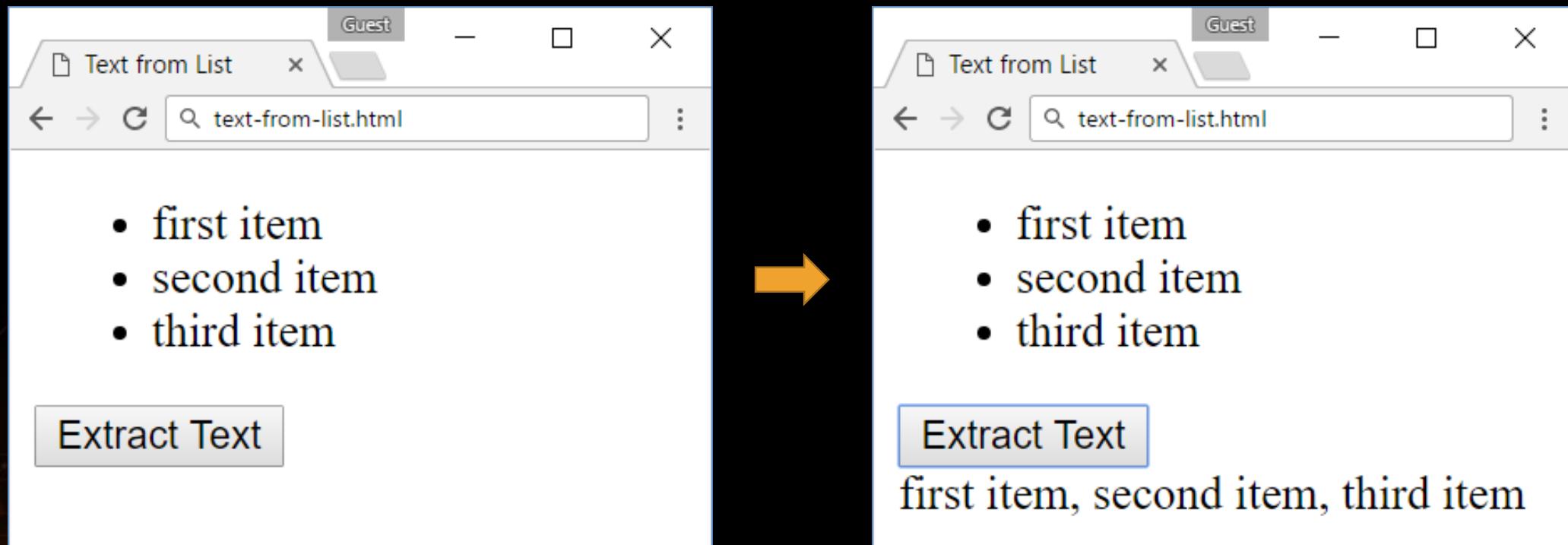
```
<body>
  <a href="index.html" hreflang="en">Home</a>
  <a href="about.html" hreflang="en-UK">About</a>
  <a href="contacts.html" hreflang="bg">Contacts</a>
</body>
```

```
<script>
  $('a[hreflang*="en"]').css('color', 'red');
  $('a[hreflang^="en"]').css('color', 'green');
</script>
```



Problem: Text from List

- An HTML page holds a **list** of text items + [Extract Text] button
- Write a JS function to display all list items, separated by ", "

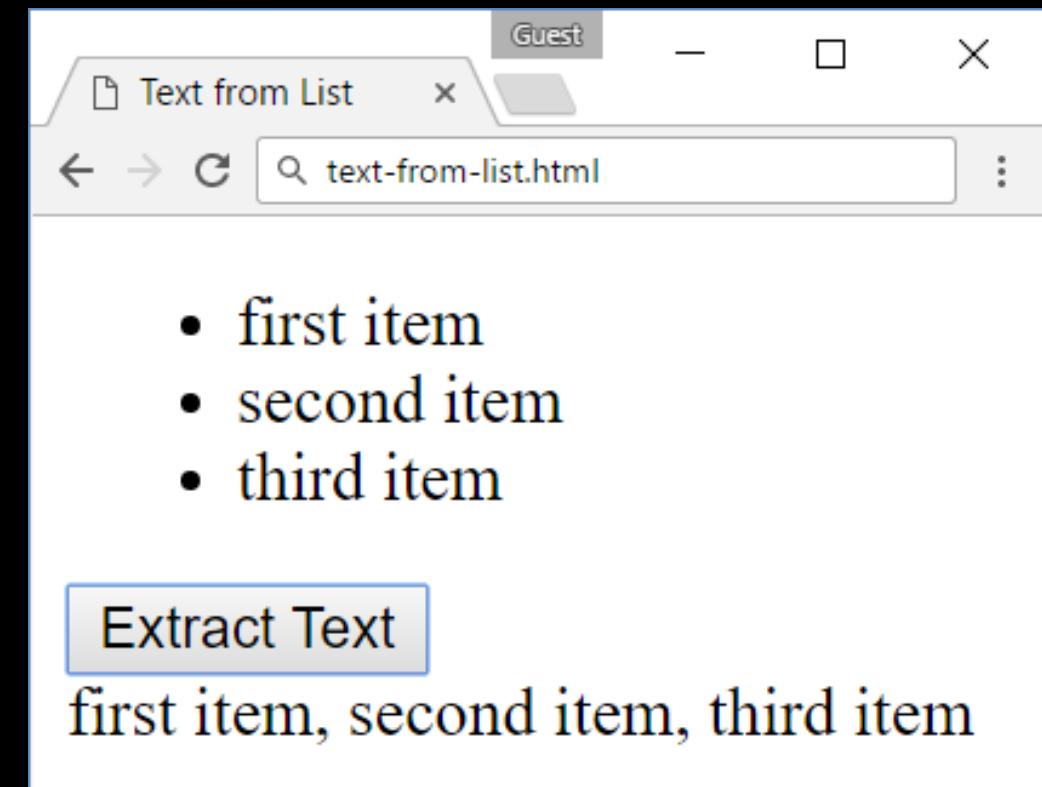


Problem: Text from List – HTML

```
<ul id="items">
  <li>first item</li>
  <li>second item</li>
  <li>third item</li>
</ul>

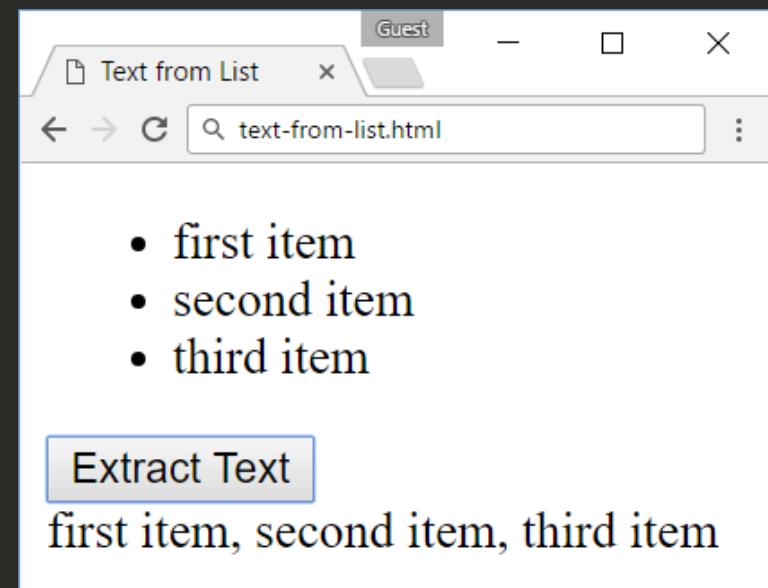
<button
  onclick="extractText()"
  Extract Text</button>

<div id="result"></div>
```



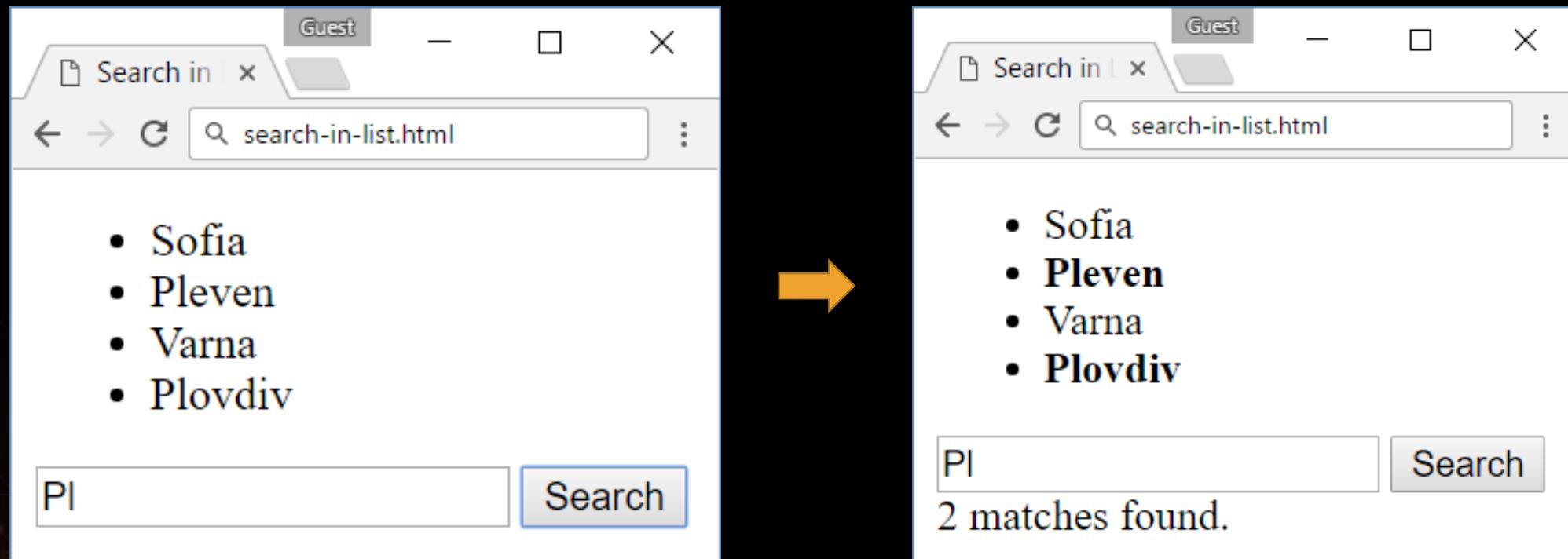
Solution: Text from List

```
function extractText() {  
  let items = $("ul#items li")  
    .toArray()  
    .map(li => li.textContent)  
    .join(", ");  
  $("#result").text(items);  
}
```



Problem: Search in List

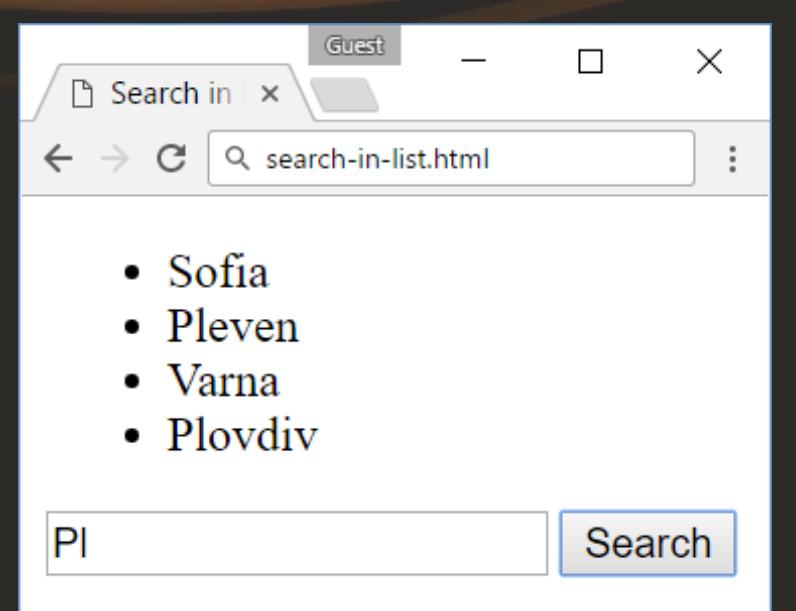
- A HTML page holds a list of towns + search box + [Search] button
- Write a JS function to highlight all towns holding the search text



Problem: Search in List – HTML

```
<ul id="towns">
  <li>Sofia</li>
  <li>Pleven</li>
  <li>Varna</li>
  <li>Plovdiv</li>
</ul>

<input type="text" id="searchText" />
<button onclick="search()">Search</button>
<div id="result"></div>
```

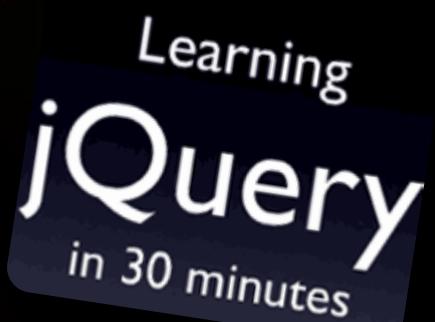


Solution: Search in List

```
function search() {  
    let searchText = $('#searchText').val();  
    let matches = 0;  
    $("#towns li").each((index, item) => {  
        if (item.textContent.includes(searchText)) {  
            $(item).css("font-weight", "bold");  
            matches++;  
        } else  
            $(item).css("font-weight", "");  
    });  
    $('#result').text(matches + " matches found.");  
}
```

Check your solution here: <https://judge.softuni.bg/Contests/329>

jQuery HTML



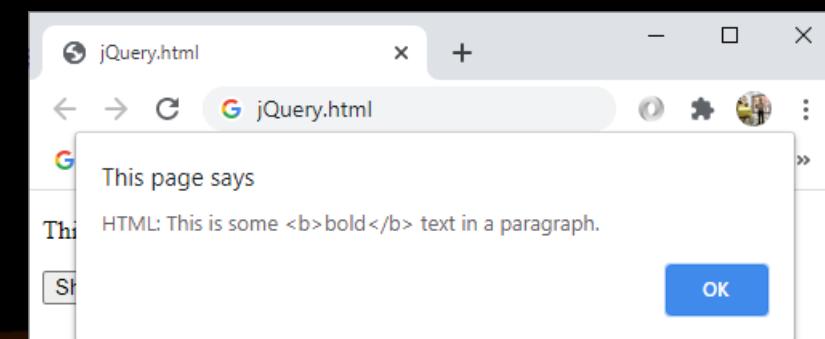
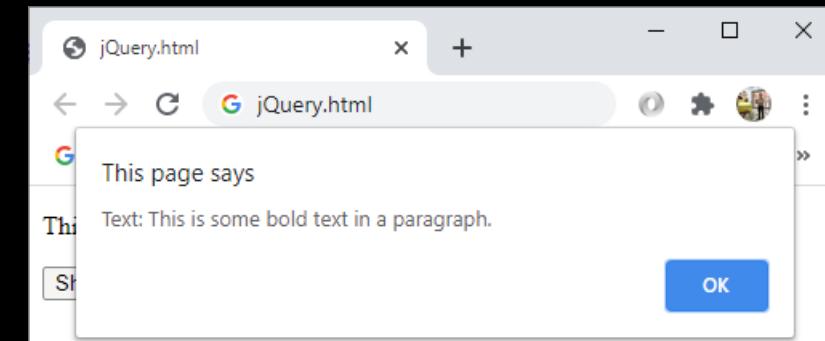
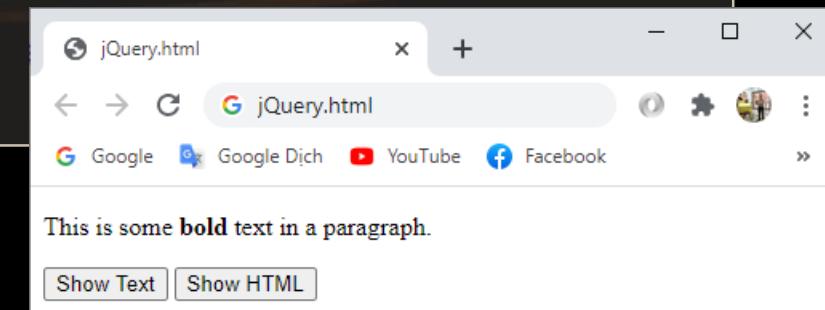
jQuery HTML - Get Content and Attributes

- jQuery contains powerful methods for changing and manipulating HTML elements and attributes
- Get Content:
 - `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
- Get Attributes:
 - `attr()`: get attribute values

jQuery HTML - Get Content - `text()` and `html()`

```
<p id="test">This is some <b>bold</b> text in a paragraph.</p>
<button id="btn1">Show Text</button>
<button id="btn2">Show HTML</button>
```

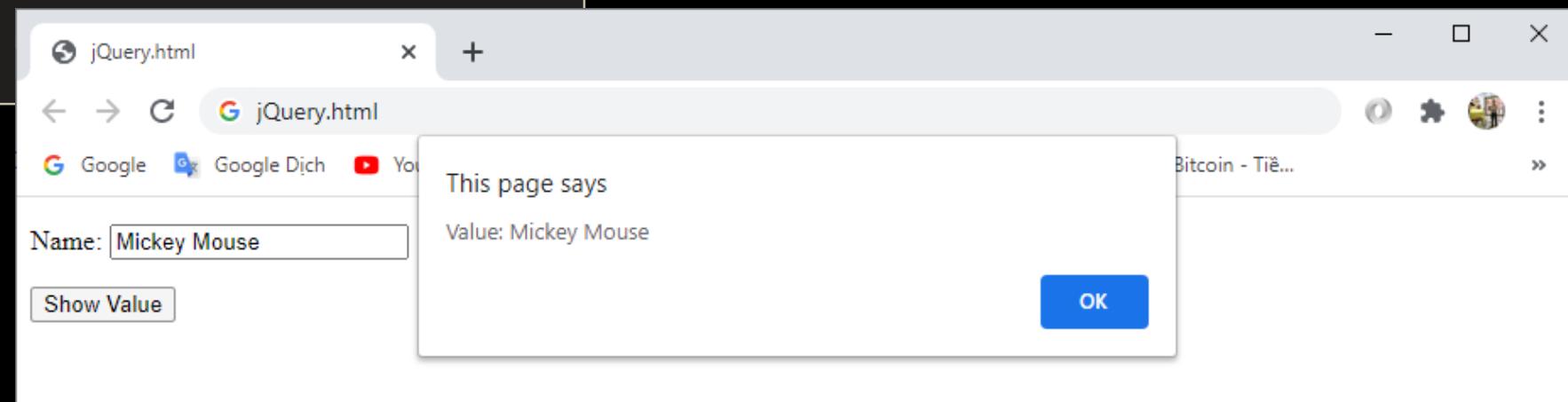
```
<script>
$(document).ready(function(){
    $("#btn1").click(function(){
        alert("Text: " + $("#test").text());
    });
    $("#btn2").click(function(){
        alert("HTML: " + $("#test").html());
    });
});
</script>
```



jQuery HTML - Get Content - val()

```
<p>Name: <input type="text" id="test" value="Mickey Mouse"></p>
<button>Show Value</button>
```

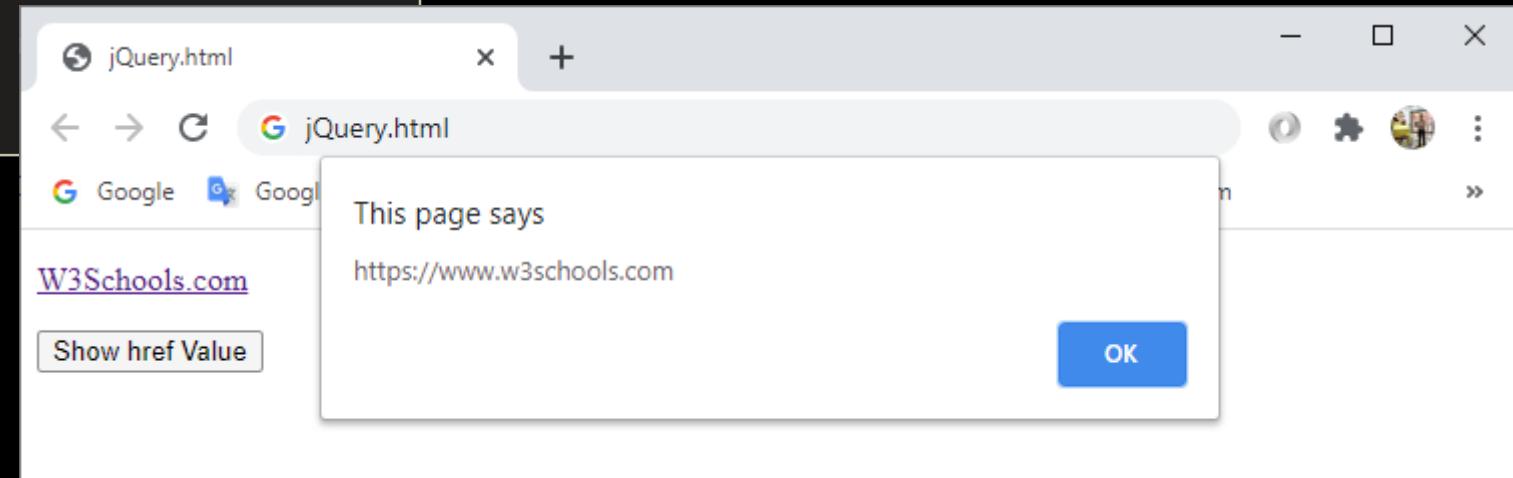
```
<script>
$(document).ready(function(){
    $("button").click(function(){
        alert("Value: " + $("#test").val());
    });
});
</script>
```



jQuery HTML - Get Attributes - attr()

```
<p><a href="https://www.w3schools.com" id="w3s">W3Schools.com</a></p>
<button>Show href Value</button>
```

```
<script>
$(document).ready(function(){
    $("button").click(function(){
        alert($("#w3s").attr("href"));
    });
});
</script>
```



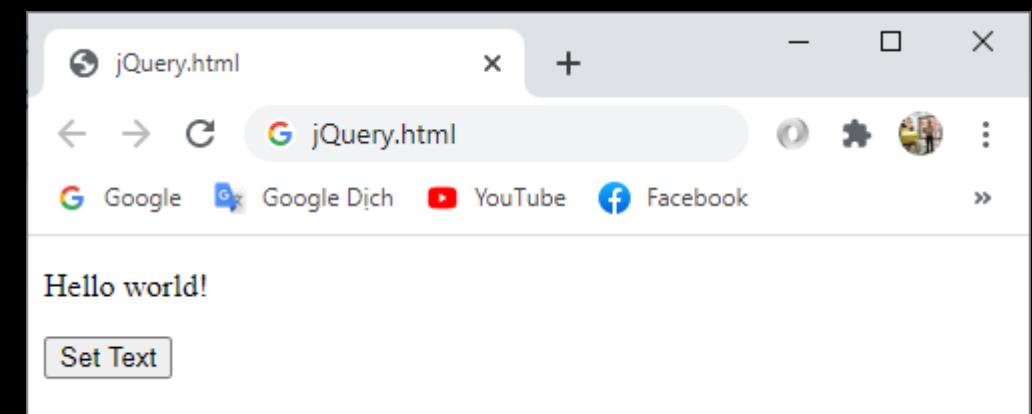
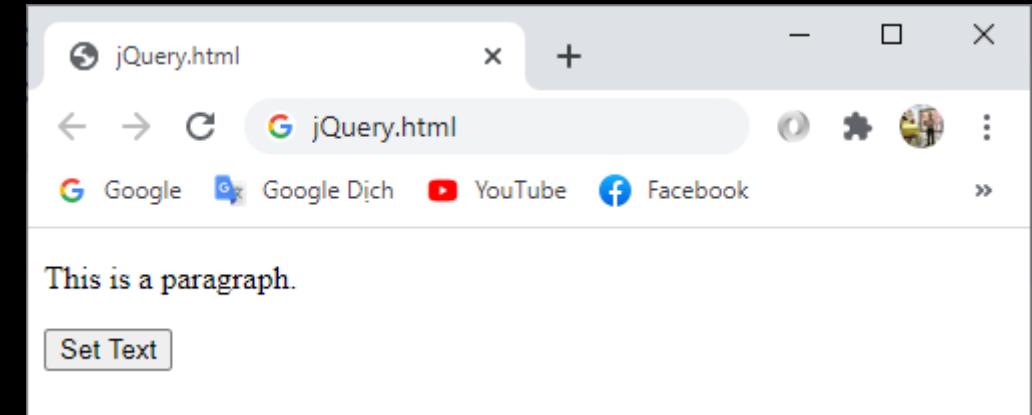
jQuery HTML - Set Content and Attributes

- jQuery contains powerful methods for changing and manipulating HTML elements and attributes
- Set Content:
 - `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
- Set Attributes:
 - `attr()`: set/change attribute values

jQuery HTML - Set Content - text()

```
<p id="test">This is a paragraph.</p>
<button id="btn">Set Text</button>
```

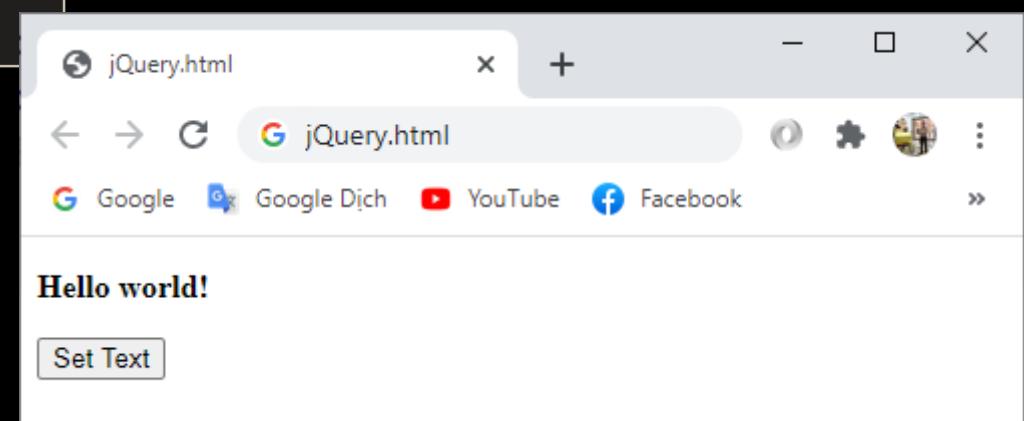
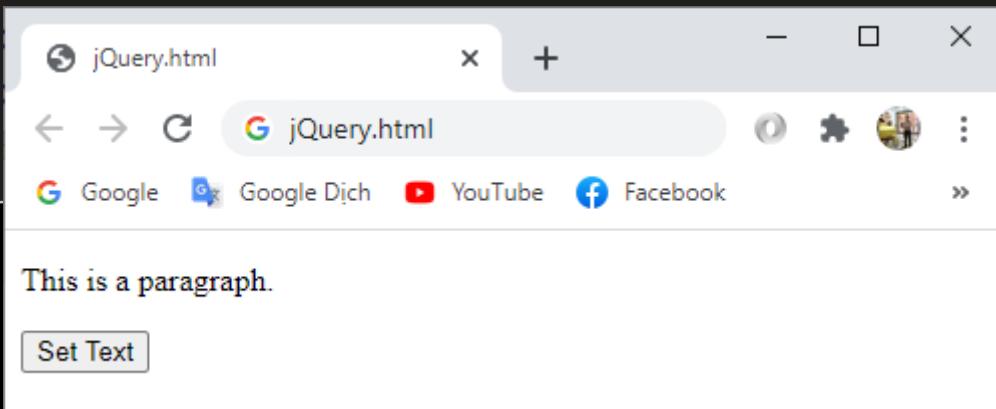
```
<script>
$(document).ready(function(){
    $("#btn").click(function(){
        $("#test").text("Hello world!");
    });
});
</script>
```



jQuery HTML - Set Content - html()

```
<p id="test">This is a paragraph.</p>
<button id="btn">Set Text</button>
```

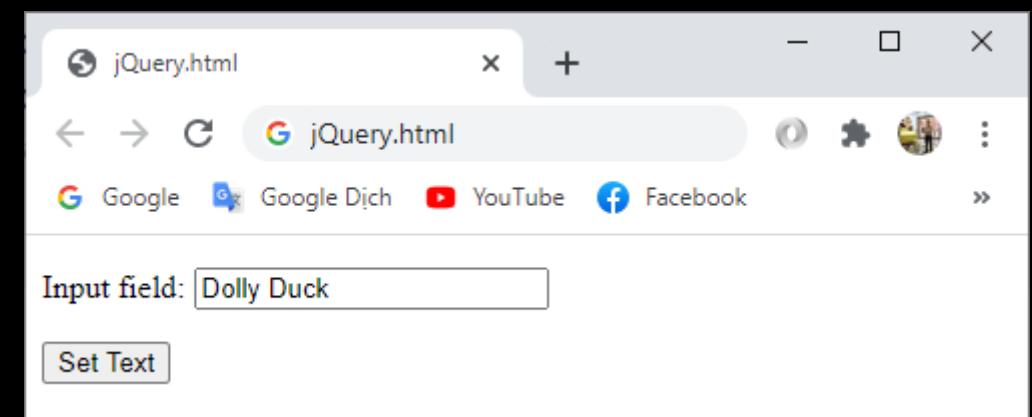
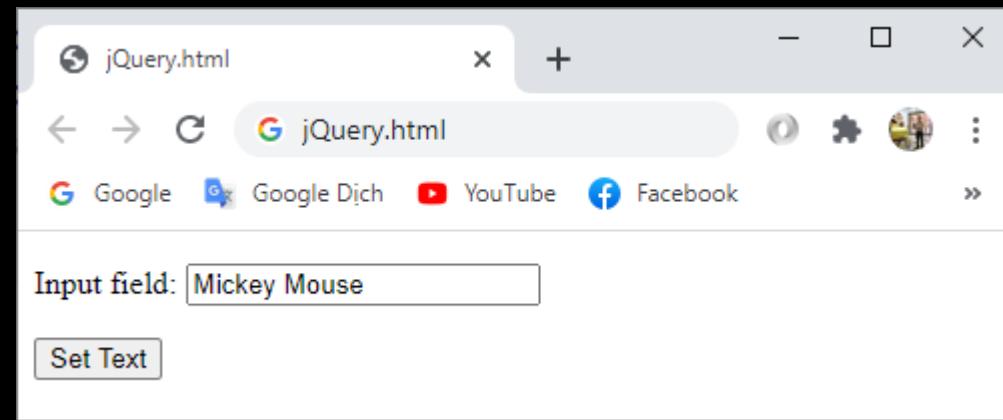
```
<script>
$(document).ready(function(){
  $("#btn").click(function(){
    $("#test").html("<b>Hello world!</b>");
  });
});
</script>
```



jQuery HTML - Set Content - val()

```
<p>Input field: <input type="text" id="test" value="Mickey Mouse"></p>
<button id="btn">Set Text</button>
```

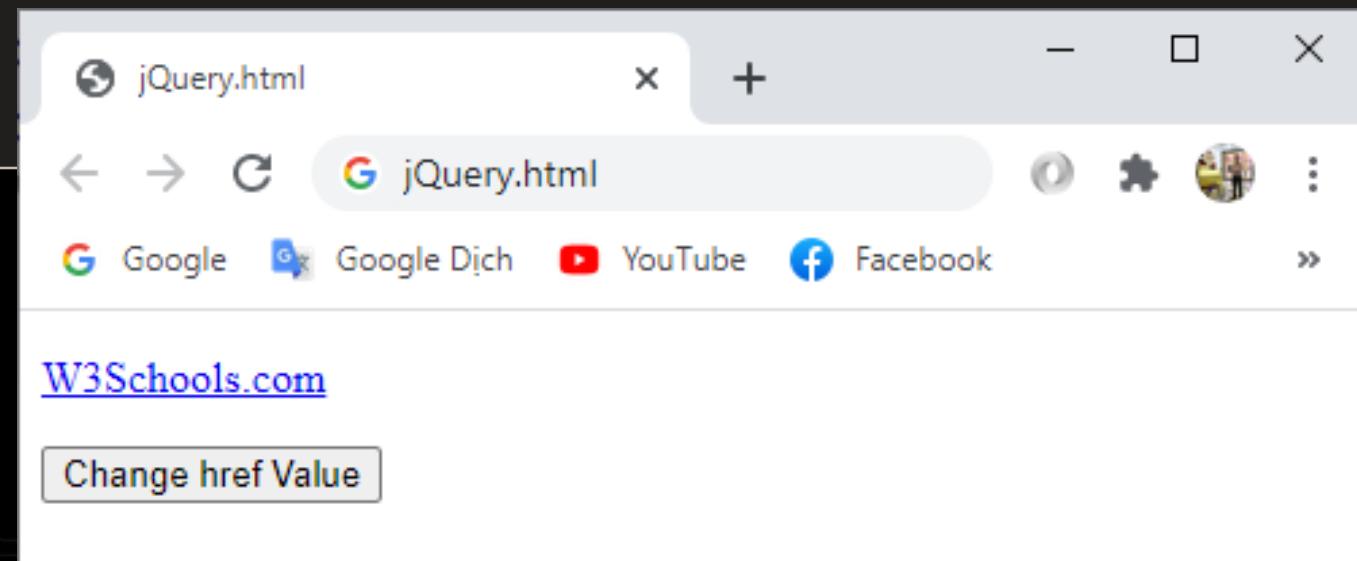
```
<script>
$(document).ready(function(){
    $("#btn").click(function(){
        $("#test").val("Dolly Duck");
    });
});
</script>
```



jQuery HTML - Set Attributes - attr()

```
<p><a href="https://www.w3schools.com" id="w3s">W3Schools.com</a></p>
<button>Change href Value</button>
```

```
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("#w3s").attr("href", "https://www.w3schools.com/jquery/");
    });
});
</script>
```



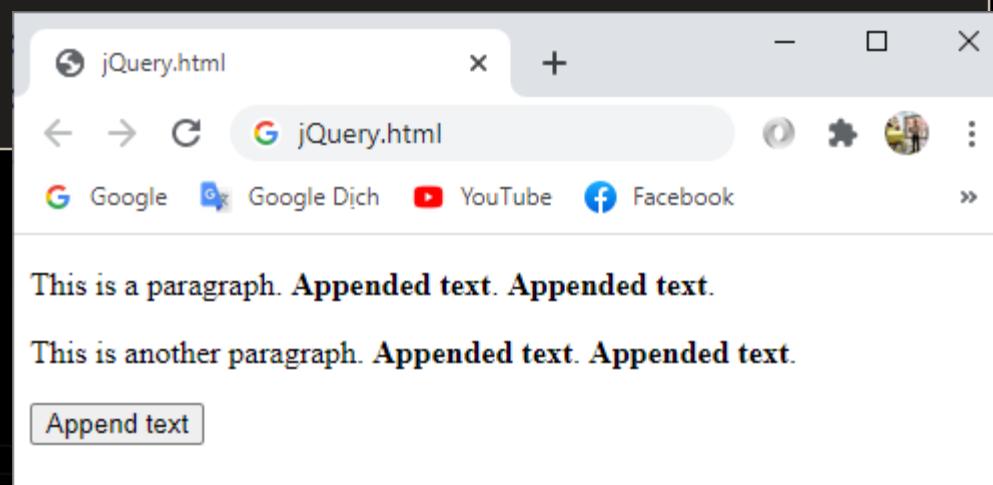
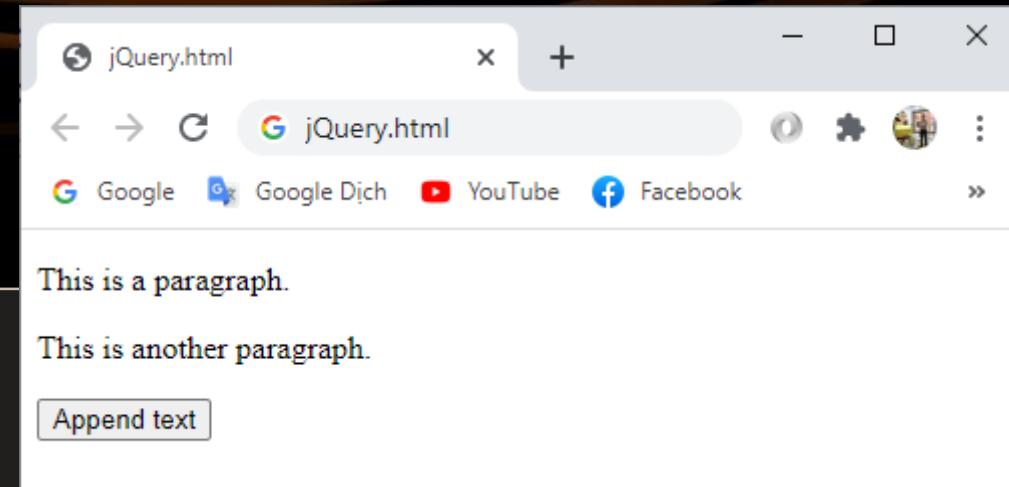
jQuery HTML - Add Elements

- With jQuery, it is easy to add new elements/content
- We will look at four jQuery methods that are used to add new 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

jQuery HTML - Add Elements - append()

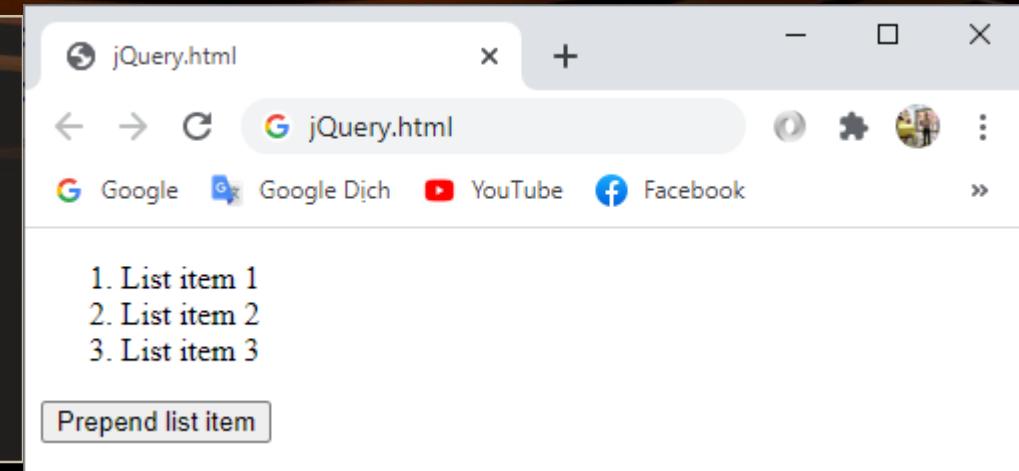
```
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
<button id="btn">Append text</button>
```

```
<script>
$(document).ready(function(){
    $("#btn").click(function(){
        $("p").append(" <b>Appended text</b>.");
    });
});
</script>
```

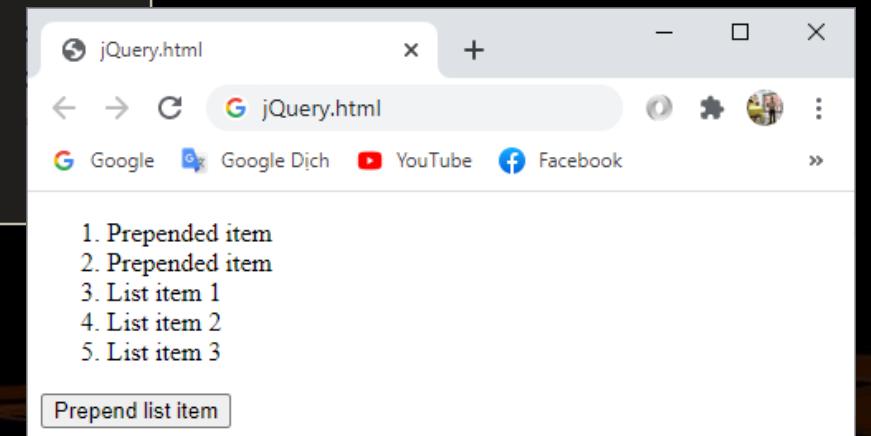


jQuery HTML - Add Elements - prepend()

```
<ol>
  <li>List item 1</li>
  <li>List item 2</li>
  <li>List item 3</li>
</ol>
<button id="btn">Prepend list item</button>
```



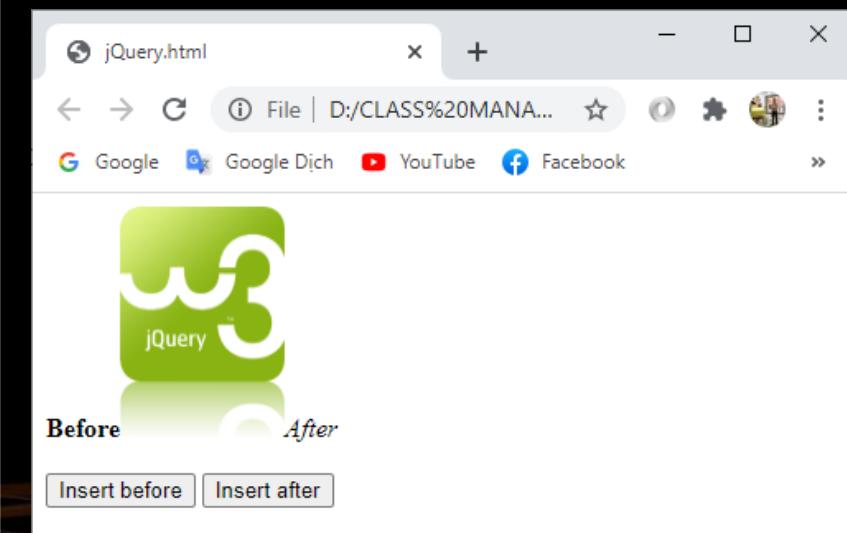
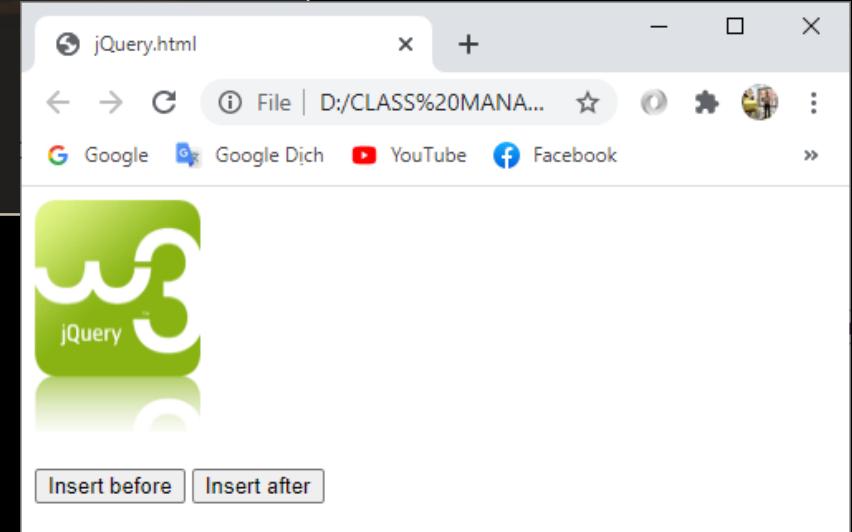
```
<script>
$(document).ready(function(){
  $("#btn").click(function(){
    $("ol").prepend("<li>Prepended item</li>");
  });
});
</script>
```



jQuery HTML - Add Elements - before() and after()

```
<br><br>  
<button id="btn1">Insert before</button>  
<button id="btn2">Insert after</button>
```

```
<script>  
$(document).ready(function(){  
    $("#btn1").click(function(){  
        $("img").before("<b>Before</b>");  
    });  
  
    $("#btn2").click(function(){  
        $("img").after("<i>After</i>");  
    });  
});  
</script>
```



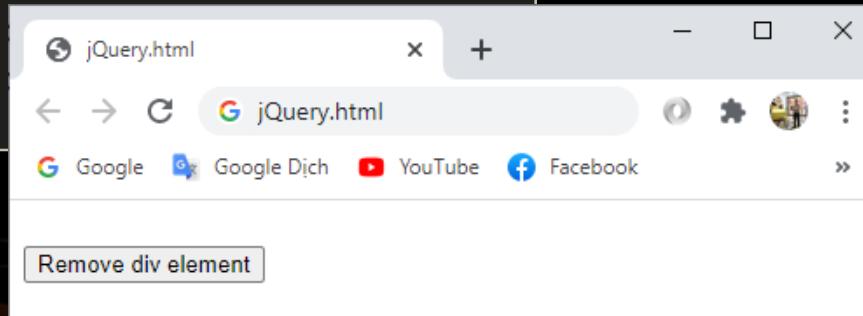
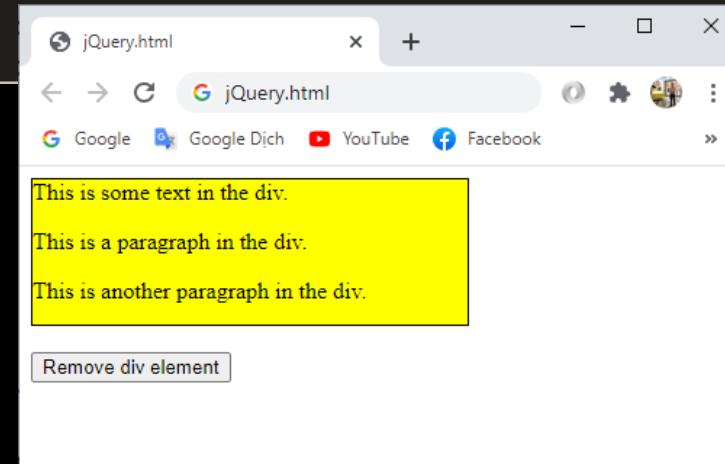
jQuery HTML - Remove Elements/Content

- With jQuery, it is easy to remove existing HTML elements
- To remove elements and content, there are mainly two jQuery methods:
 - `remove()`: Removes the selected element (and its child elements)
 - `empty()`: Removes the child elements from the selected element

jQuery HTML - Remove Elements/Content - remove()

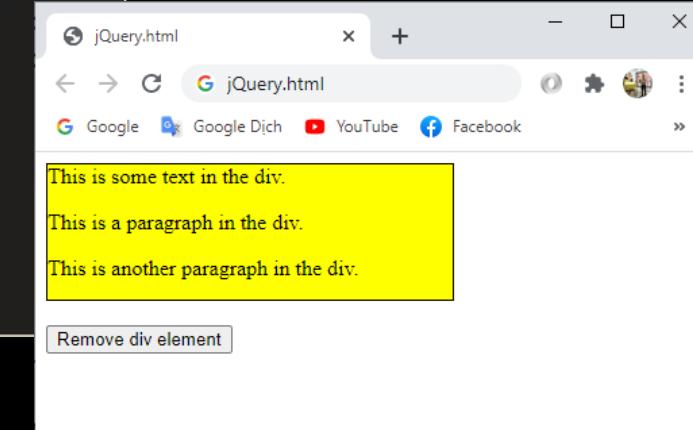
```
<div id="div1" style="height:100px;width:300px;border:1px solid black;background-color:yellow;">  
This is some text in the div.  
<p>This is a paragraph in the div.</p>  
<p>This is another paragraph in the div.</p></div><br>  
<button>Remove div element</button>
```

```
<script>  
$(document).ready(function(){  
    $("button").click(function(){  
        $("#div1").remove();  
    });  
});  
</script>
```

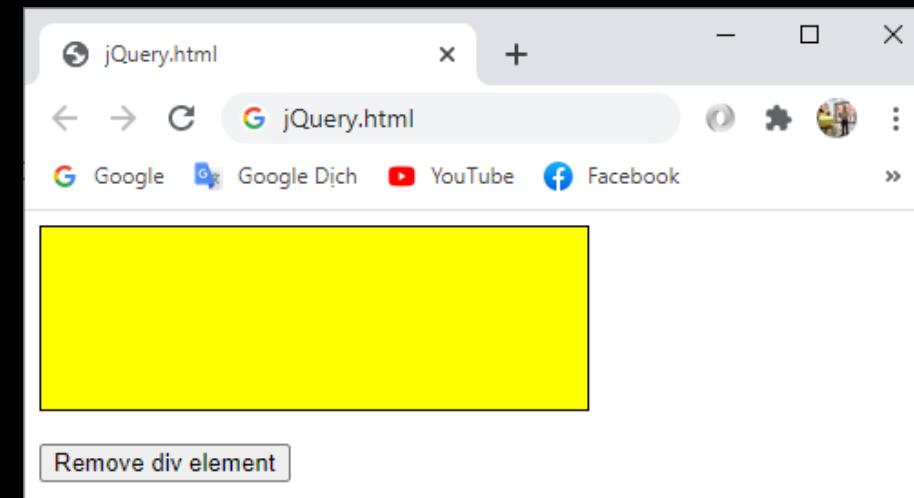


jQuery HTML - Remove Elements/Content - empty()

```
<div id="div1" style="height:100px;width:300px;border:1px solid black;background-color:yellow;">  
This is some text in the div.  
<p>This is a paragraph in the div.</p>  
<p>This is another paragraph in the div.</p></div><br>  
<button>Remove div element</button>
```



```
<script>  
$(document).ready(function(){  
    $("button").click(function(){  
        $("#div1").empty();  
    });  
});  
</script>
```

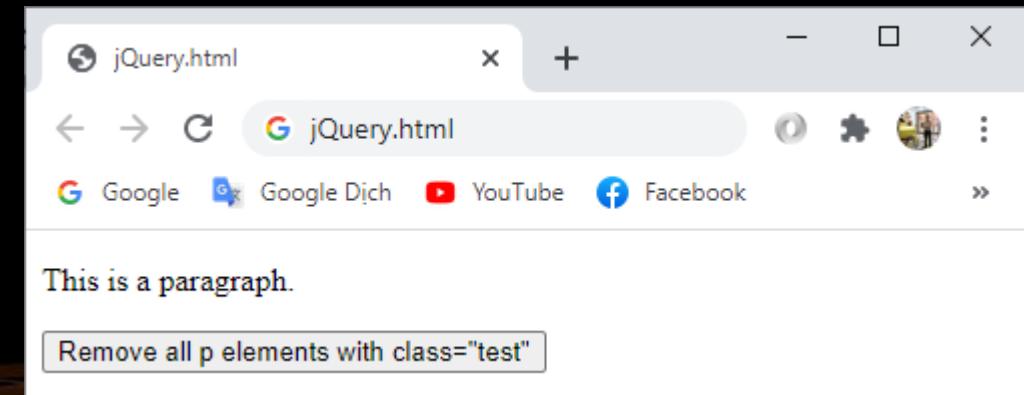
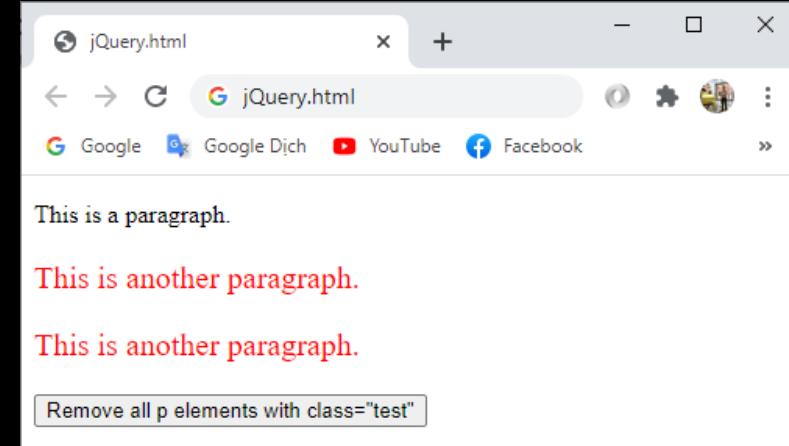


jQuery HTML - Filter the Elements to be Removed

```
<p>This is a paragraph.</p>
<p class="test">This is another paragraph.</p>
<p class="test">This is another paragraph.</p>
<button>Remove all p elements with class="test"</button>
```

```
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("p").remove(".test");
    });
});</script>
```

```
<style>.test {
    color: red;
    font-size: 20px;
}</style >
```



jQuery HTML - Get and Set CSS Classes

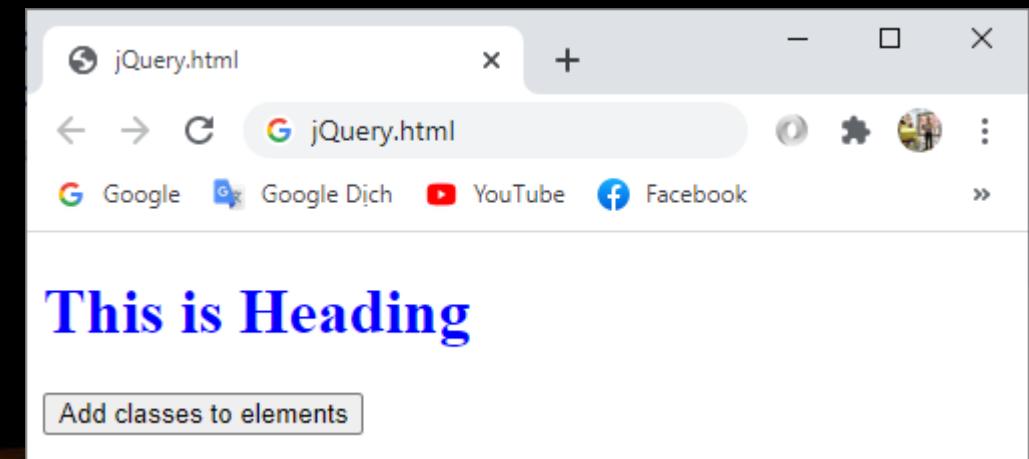
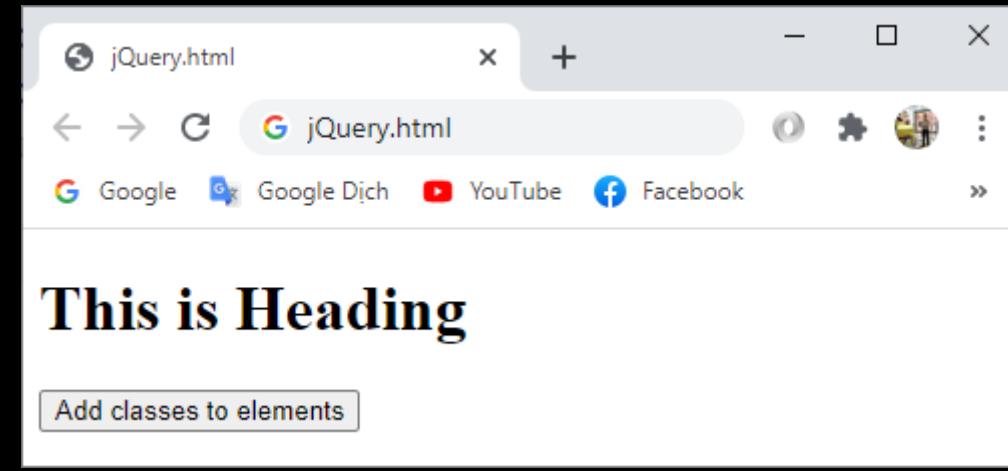
- With jQuery, it is easy to manipulate the style of elements
- jQuery has several methods for CSS manipulation. We will look at the following methods:
 - `addClass()`: Adds one or more classes to the selected elements
 - `removeClass()`: Removes one or more classes from the selected elements
 - `toggleClass()`: Toggles between adding/removing classes from the selected elements
 - `css()`: Sets or returns the style attribute

jQuery HTML - Get and Set CSS Classes - addClass()

```
<h1>This is Heading</h1>
<button>Add classes to elements</button>
```

```
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("h1").addClass("blue");
    });
});
</script>
```

```
<style>
.blue {
    color: blue;
}
</style>
```

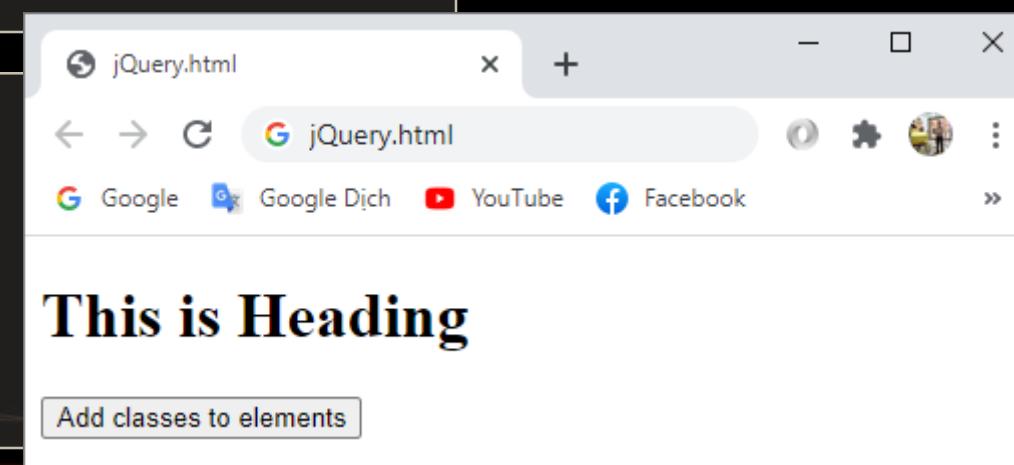
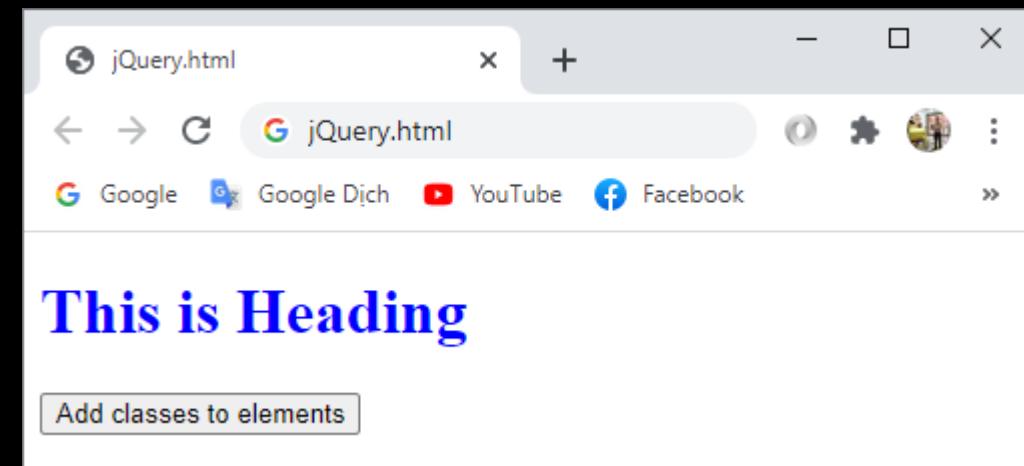


jQuery HTML - Get and Set CSS Classes - removeClass()

```
<h1>This is Heading</h1>
<button>Add classes to elements</button>
```

```
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("h1").removeClass("blue");
    });
});
</script>
```

```
<style>
.blue {
    color: blue;
}
</style>
```



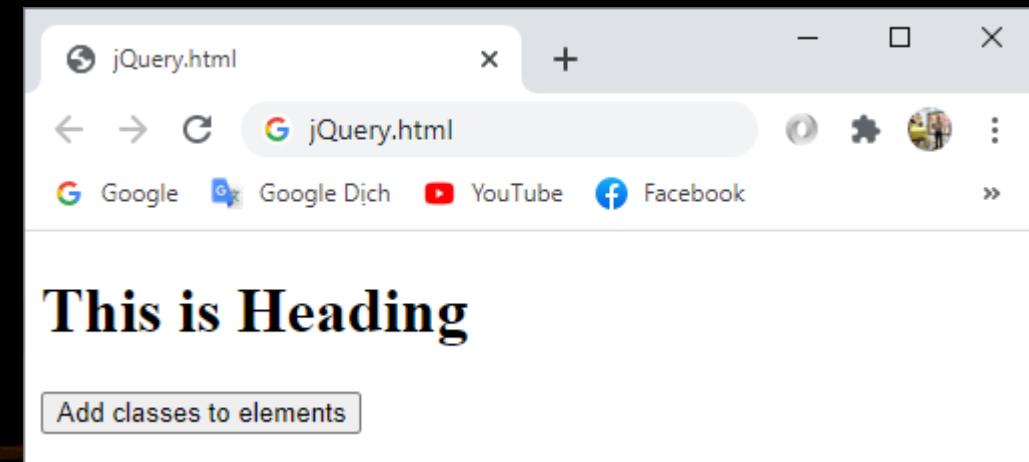
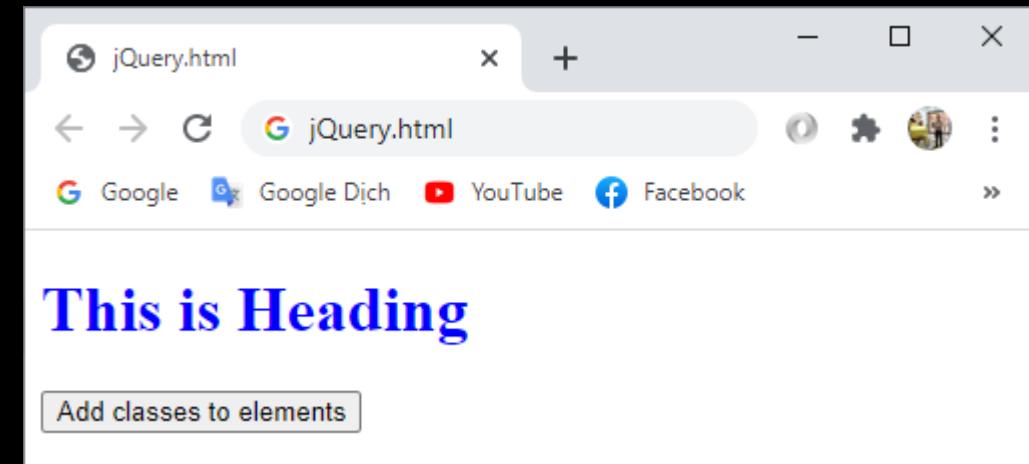
jQuery HTML - Get and Set CSS Classes - toggleClass()

```
<h1>This is Heading</h1>
<button>Add classes to elements</button>
```

```
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("h1").toggleClass("blue");
    });
});
</script>
```

adding/removing classes

```
<style>
.blue {
    color: blue;
}
</style>
```



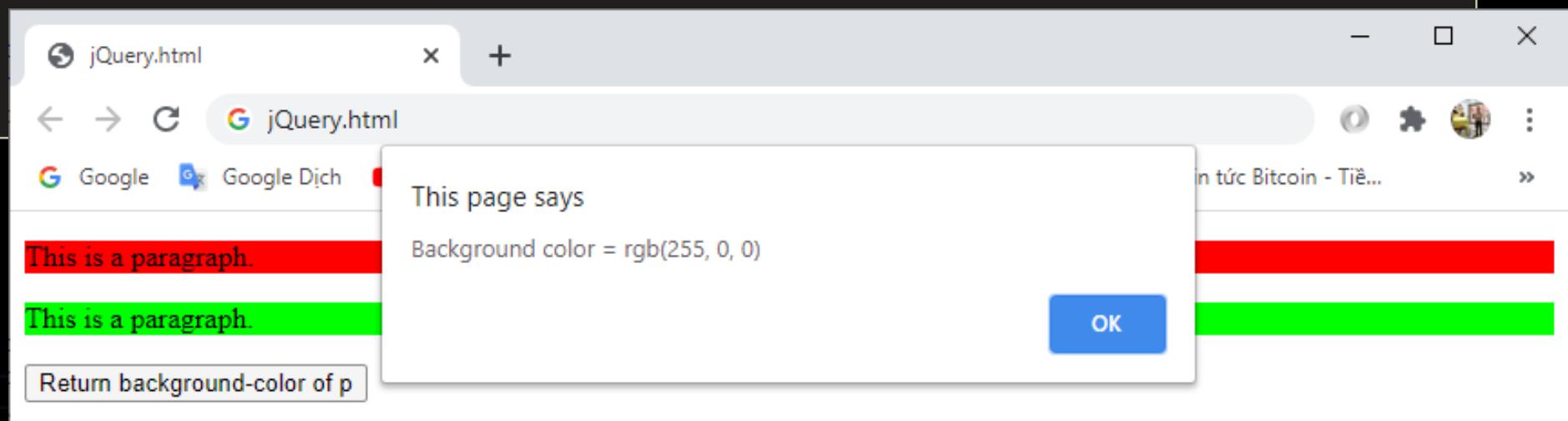
jQuery HTML - css() Method

- The `css()` method sets or returns one or more style properties for the selected elements
- Return a CSS Property: `css("propertyname");`
- Set a CSS Property: `css("propertyname","value");`
- Set Multiple CSS Properties:
 - `css({"propertyname":"value","propertyname":"value",...});`

jQuery HTML - css() Method - Return a CSS Property

```
<p style="background-color:#ff0000">This is a paragraph.</p>
<p style="background-color:#00ff00">This is a paragraph.</p>
<button>Return background-color of p</button>
```

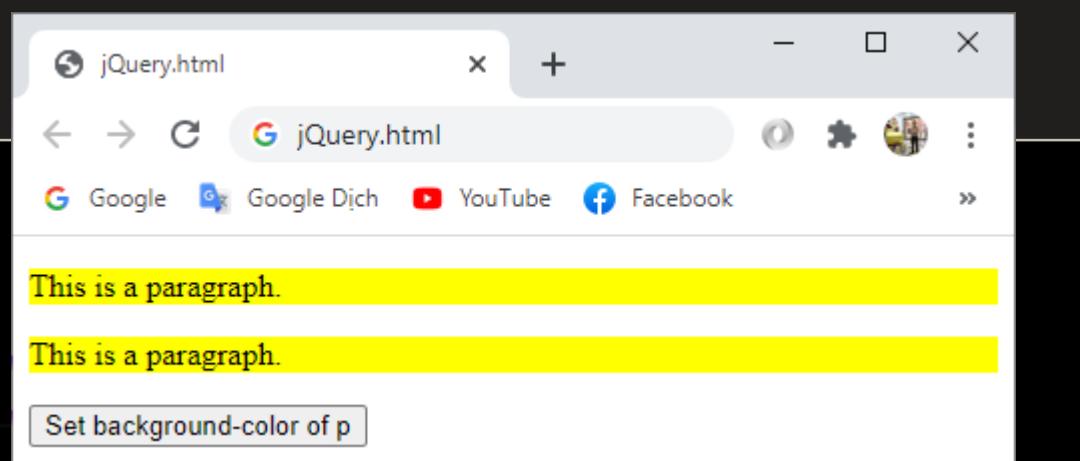
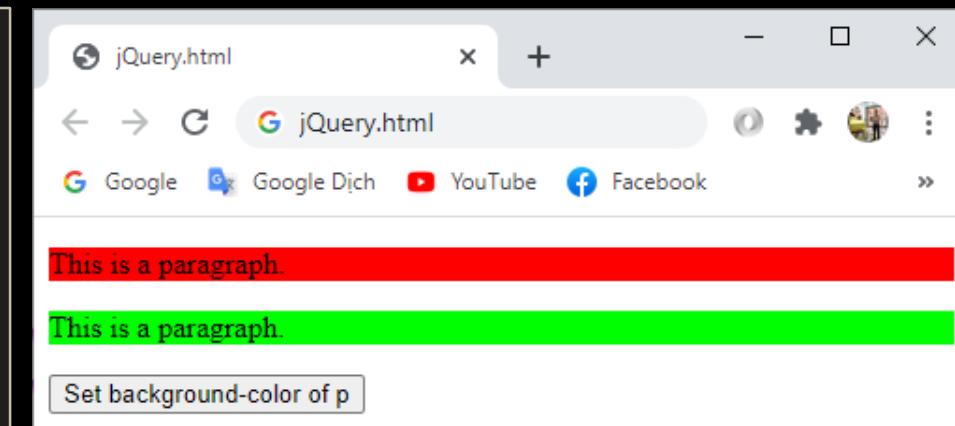
```
<script>
$(document).ready(function(){
    $("button").click(function(){
        alert("Background color = " + $("p").css("background-color"));
    });
});
</script>
```



jQuery HTML - css() Method - Set a CSS Property

```
<p style="background-color:#ff0000">This is a paragraph.</p>
<p style="background-color:#00ff00">This is a paragraph.</p>
<button>Return background-color of p</button>
```

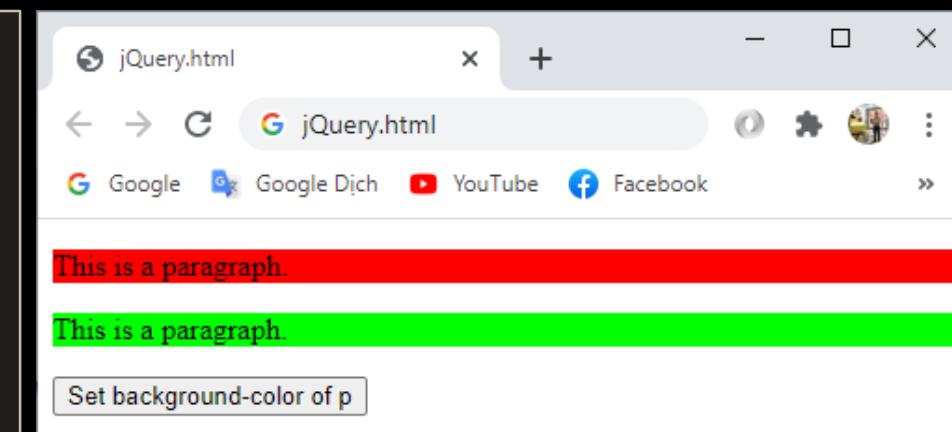
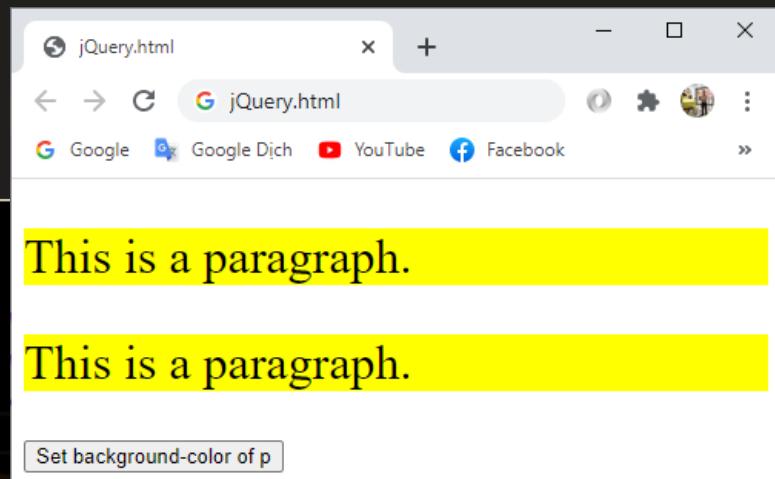
```
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("p").css("background-color", "yellow");
    });
});
</script>
```



jQuery HTML - `css()` Method - Set Multiple a CSS Property

```
<p style="background-color:#ff0000">This is a paragraph.</p>
<p style="background-color:#00ff00">This is a paragraph.</p>
<button>Return background-color of p</button>
```

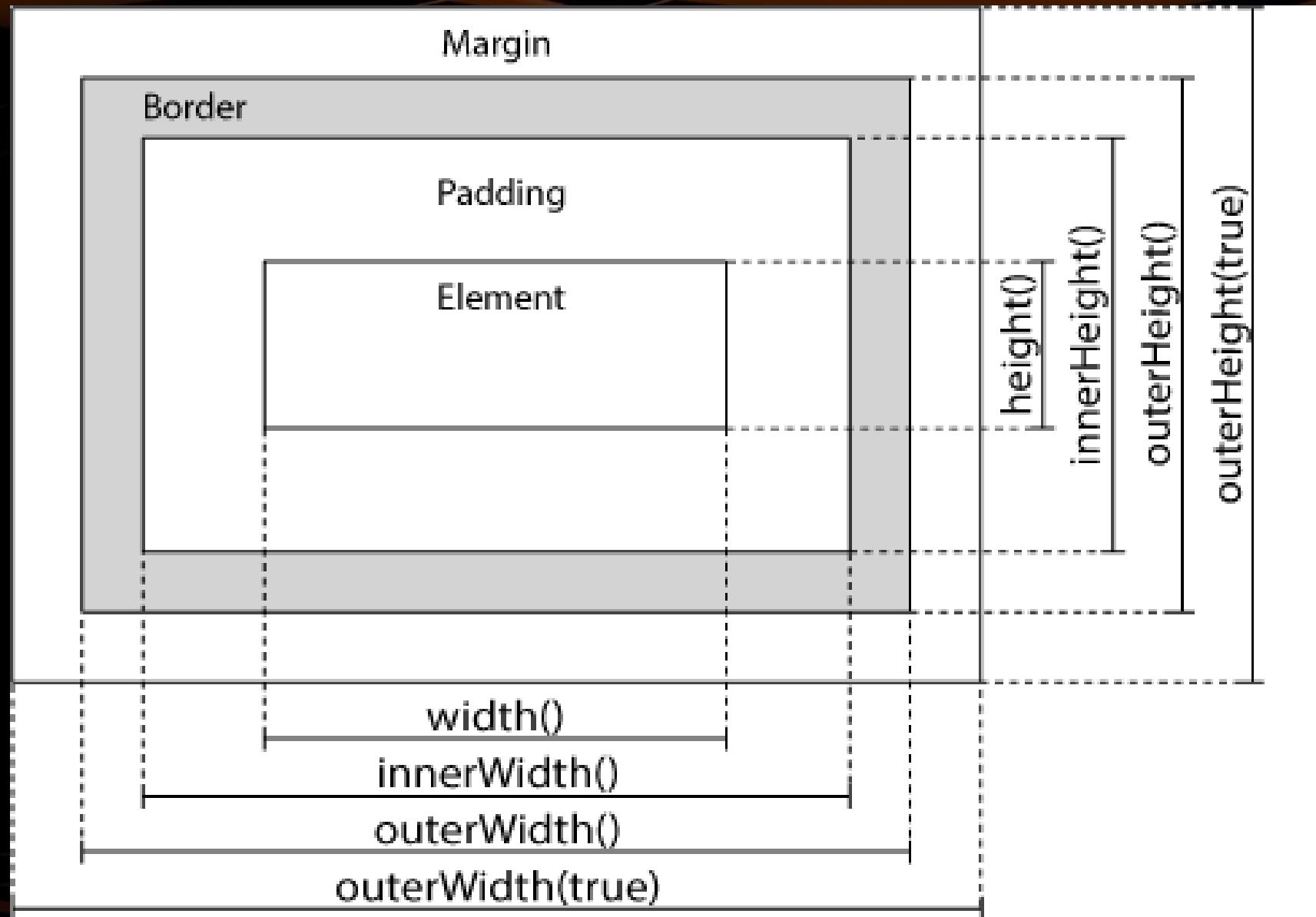
```
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("p").css({"background-color": "yellow",
        "font-size": "200%"});
    });
});
</script>
```



jQuery HTML - Dimensions

- With jQuery, it is easy to work with the dimensions of elements and browser window
- jQuery has several important methods for working with dimensions:
 - `width()`
 - `height()`
 - `innerWidth()`
 - `innerHeight()`
 - `outerWidth()`
 - `outerHeight()`

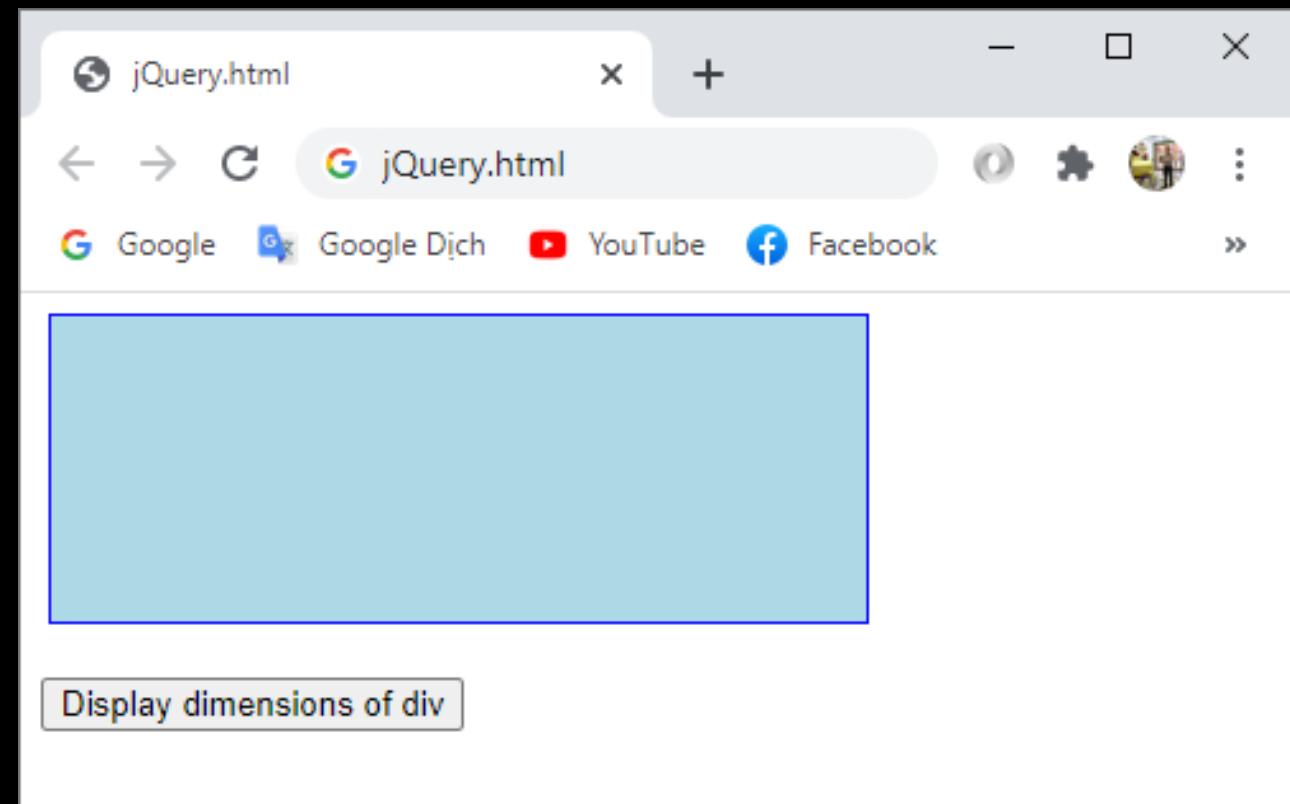
jQuery HTML - Dimensions



jQuery HTML - Dimensions - width() and Height() (1)

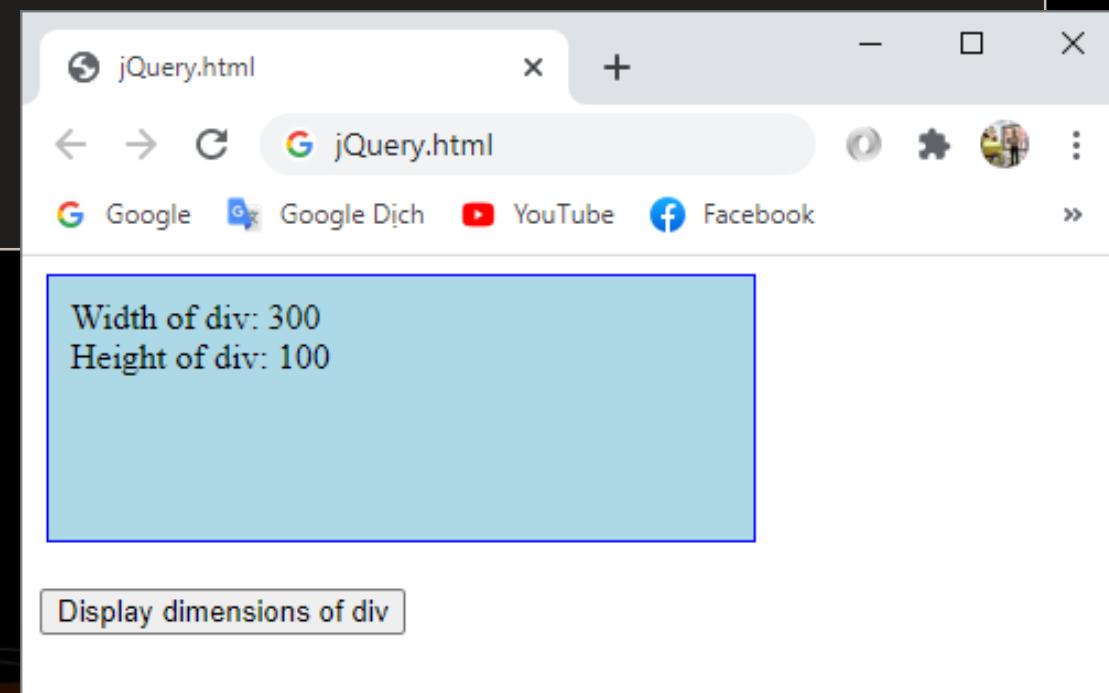
```
<div id="div1"></div><br>
<button>Display dimensions of div</button>
```

```
<style>
#div1 {
    height: 100px;
    width: 300px;
    padding: 10px;
    margin: 3px;
    border: 1px solid blue;
    background-color:
lightblue;
}
</style>
```



jQuery HTML - Dimensions - width() and Height() (2)

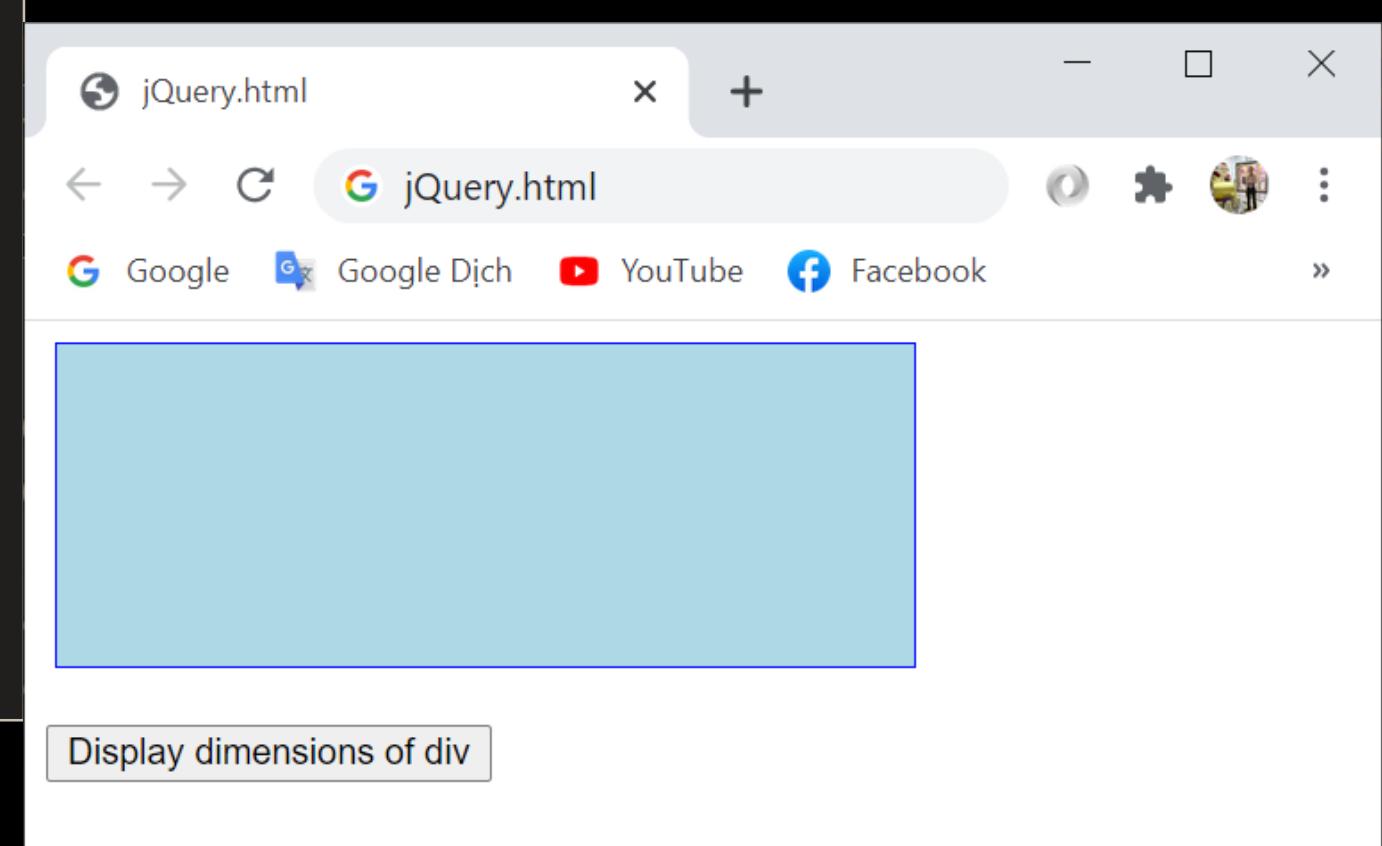
```
<script>
$(document).ready(function(){
    $("button").click(function(){
        var txt = "";
        txt += "Width of div: " + $("#div1").width() + "<br>";
        txt += "Height of div: " + $("#div1").height();
        $("#div1").html(txt);
    });
});
</script>
```



jQuery HTML - Dimensions - innerWidth() and innerHeight() (1)

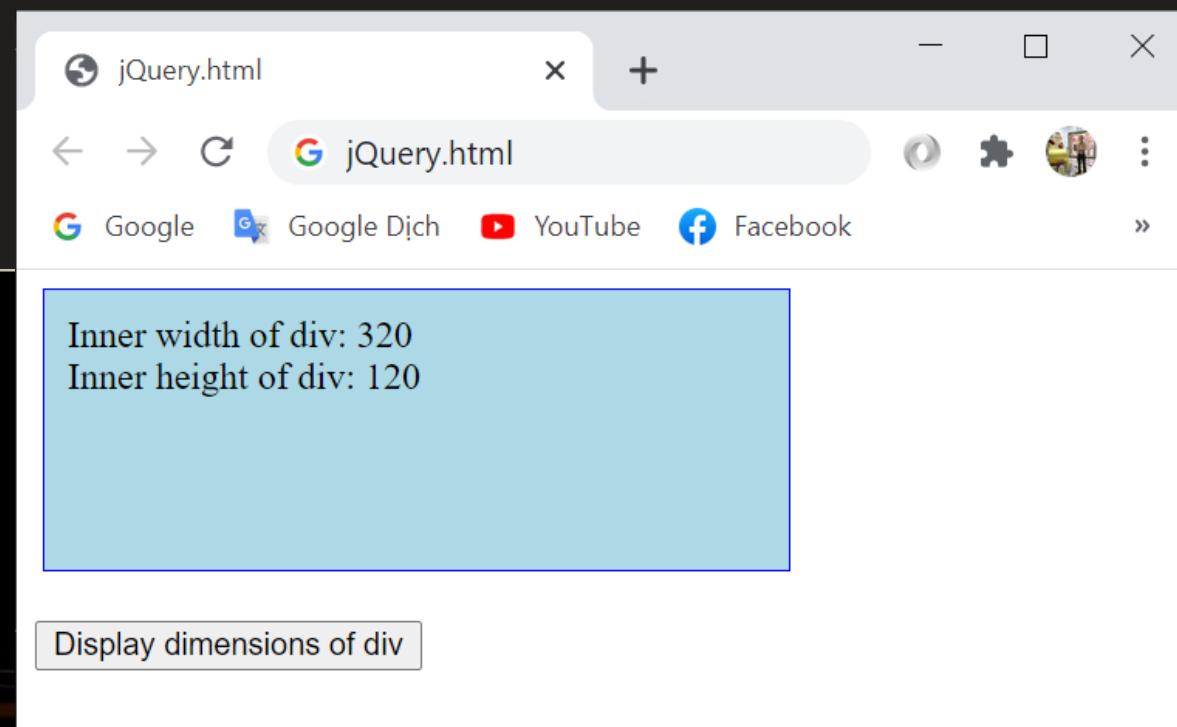
```
<div id="div1"></div><br>
<button>Display dimensions of div</button>
```

```
<style>
#div1 {
    height: 100px;
    width: 300px;
    padding: 10px;
    margin: 3px;
    border: 1px solid blue;
    background-color: lightblue;
}
</style>
```



jQuery HTML - Dimensions - innerWidth() and innerHeight() (2)

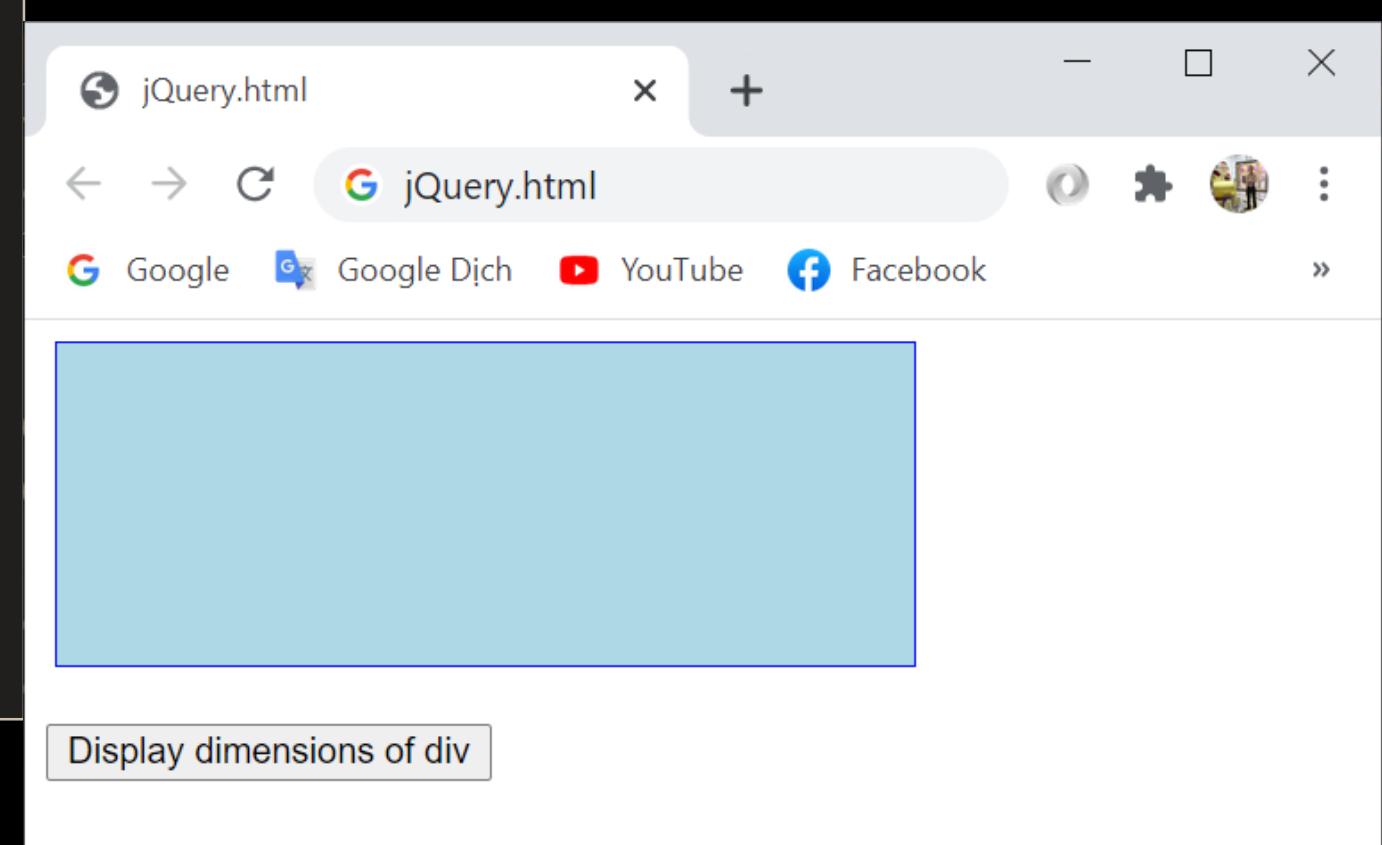
```
<script>
$(document).ready(function(){
    $("button").click(function(){
        var txt = "";
        txt += "Inner width of div: " + $("#div1").innerWidth() + "<br>";
        txt += "Inner height of div: " + $("#div1").innerHeight();
        $("#div1").html(txt);
    });
});
</script>
```



jQuery HTML - Dimensions - outerWidth() and outerHeight() (1)

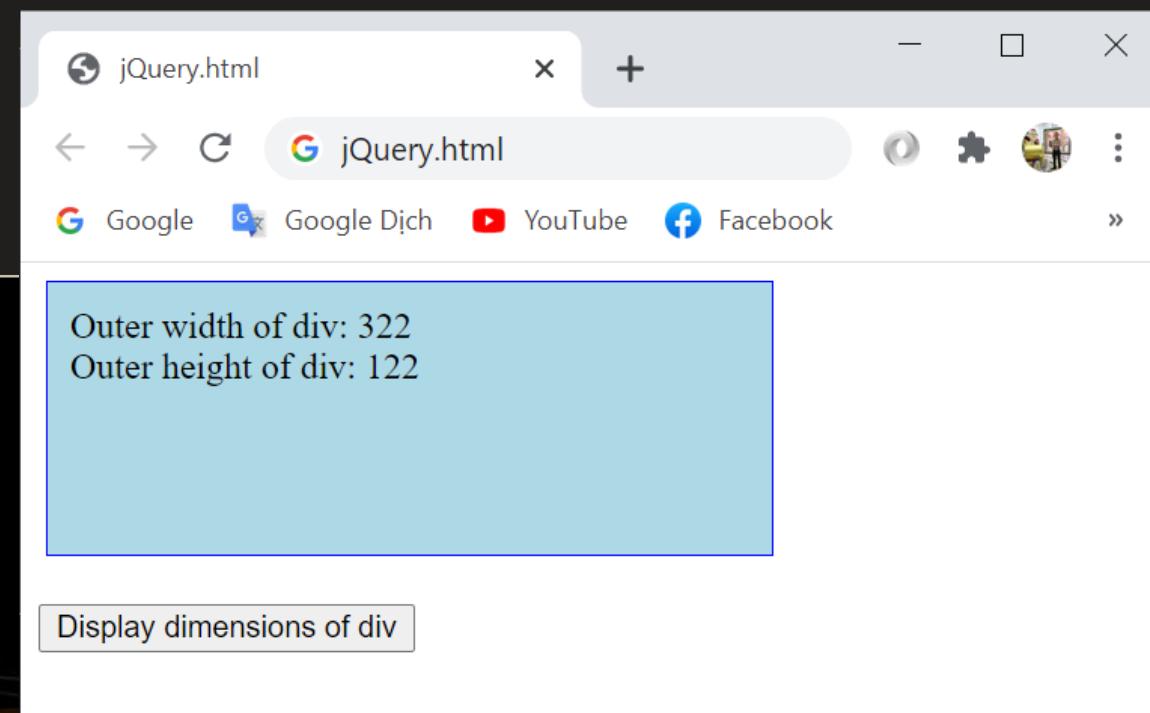
```
<div id="div1"></div><br>
<button>Display dimensions of div</button>
```

```
<style>
#div1 {
    height: 100px;
    width: 300px;
    padding: 10px;
    margin: 3px;
    border: 1px solid blue;
    background-color: lightblue;
}
</style>
```



jQuery HTML - Dimensions - outerWidth() and outerHeight() (2)

```
<script>
$(document).ready(function(){
    $("button").click(function(){
        var txt = "";
        txt += "Outer width of div: " + $("#div1").outerWidth() + "<br>";
        txt += "Outer height of div: " + $("#div1").outerHeight();
        $("#div1").html(txt);
    });
});
</script>
```



Problem: Countries Table

- You are given a HTML table holding countries with their capital
- Implement add / delete / move up / move down actions for the table rows
- Initially add these countries:
 - Bulgaria / Sofia
 - Germany / Berlin
 - Russia / Moscow

Country	Capital	Action
Poland	Warsaw	[Create]
Bulgaria	Sofia	[Up] [Down] [Delete]
Germany	Berlin	[Up] [Down] [Delete]
Russia	Moscow	[Up] [Delete]

Problem: Countries Table – HTML

```
<style>
  td, th { background: #DDD; padding: 5px 10px }
  input[type='text'] { width: 60px }
</style>

<table id="countriesTable">
  <tr><th>Country</th><th>Capital</th><th>Action</th></tr>
  <tr><td><input type="text" id="newCountryText" /></td>
    <td><input type="text" id="newCapitalText" /></td>
    <td><a href="#" id="createLink">[Create]</a></td>
  </tr>
</table>

<script>$(() => initializeTable())</script>
```

Solution: Countries Table – Initialize Table

```
function initializeTable() {  
    $("#createLink").click(createCountry);  
  
    addCountryToTable("Bulgaria", "Sofia");  
    addCountryToTable("Germany", "Berlin");  
    addCountryToTable("Russia", "Moscow");  
    fixRowLinks();  
  
    // Functions addCountryToTable(country, capital),  
    // createCountry(), moveRowUp(), moveRowDown(),  
    // deleteRow(), fixRowLinks() come here ...  
}
```

Solution: Countries Table – Create Country

```
function createCountry() {  
    let country = $('#newCountryText').val();  
    let capital = $('#newCapitalText').val();  
    addCountryToTable(country, capital, true);  
    $('#newCountryText').val('');  
    $('#newCapitalText').val('');  
    fixRowLinks();  
}
```

Solution: Countries Table – Add Country Row

```
function addCountryToTable(country, capital) {  
    let row = $('|')  
        .append($(' ').text(country))         .append($(' ').text(capital))         .append($(' ')             .append($('\[Up\]').click(moveRowUp))             .append(" ")             .append($('\[Down\]).click(moveRowDown))             .append(" ")             .append($('\[Delete\]').click(deleteRow)));     row.css('display', 'none');     $('#countriesTable').append(row);     row.fadeIn(); } | | |

```

Solution: Countries Table – Row Up

```
function moveRowUp() {  
    let row = $(this).parent().parent();  
    row.fadeOut(function() {  
        row.insertBefore(row.prev());  
        row.fadeIn();  
        fixRowLinks();  
    });  
}
```

Solution: Countries Table – Row Down

```
function moveRowDown() {  
    let row = $(this).parent().parent();  
    row.fadeOut(function() {  
        row.insertAfter(row.next());  
        row.fadeIn();  
        fixRowLinks();  
    });  
}
```

Solution: Countries Table – Delete Row

```
function deleteRow() {  
    let row = $(this).parent().parent();  
    row.fadeOut(function() {  
        row.remove();  
        fixRowLinks();  
    });  
}
```

Solution: Countries Table – Fix Row Links

```
function fixRowLinks() {  
    // Show all links in the table  
    $('#countriesTable a').css('display', 'inline');  
  
    // Hide [Up] link in first table data row  
    let tableRows = $('#countriesTable tr');  
    $(tableRows[2]).find("a:contains('Up')")  
        .css('display', 'none');  
  
    // Hide the [Down] link in the last table row  
    $(tableRows[tableRows.length - 1]).find("a:contains('Down')")  
        .css('display', 'none');  
}
```

Summary

- jQuery – the most popular client-side JS library
- Select DOM elements with jQuery
 - `$([selector])`
- Altering the DOM:
 - `$([selector]).html(...)` / `append(...)`



jQuery - Selectors and DOM Manipulation, jQuery HTML



Questions?

