

jQuery

Part 1 I : Introduction to jQuery

Overview

- An open source JavaScript library that simplifies the interaction between HTML and JavaScript.
- Provides easy ways of selecting sets of DOM elements.
- Provides functions that can be applied to the selected elements.
- Creates, deletes, shows, hides DOM elements
- Defines event behavior on a page (click, mouse movement, dynamic styles, animations, dynamic content)
- AJAX calls

jQuery Philosophy

**Find content
in the HTML
document
(selector)**



**Do something
to it
(method)**

- a. Manipulate elements
- b. Animate elements
- c. Communicate with the server

The Focus of jQuery

Find some elements.

Do something with them.

```
$("div").addClass("special");
```



The jQuery Object

```
$("div").addClass("special");
```

- `$` is the jQuery Object (also named jQuery)
- `"div"` finds some elements using CSS selectors
- `$("div")` returns a jQuery set (containing 0 to many DOM nodes).
- `addClass(...)` modifies all the divs

Including jQuery

- jQuery can be downloaded and used as a local library.
- Include the library in the `<script>` tag in the `<head>`.

```
<script src="./js/jquery-2.2.3.min.js"></script>
```

Standard jQuery Syntax

- A jQuery statement typically starts with the dollar sign (\$) and ends with a semicolon (;).
- In jQuery, the dollar sign (\$) is just an alias for jQuery.
- The dollar sign (\$) specifies that any code that follows should be interpreted using jQuery library

```
<html>
  <head>
    <title>jQuery Hello World Example</title>
    <script src="./js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        // Some code to be executed...
        alert("Hello World!");
      });
    </script>
  </head>
  <body>

  </body>
</html>
```

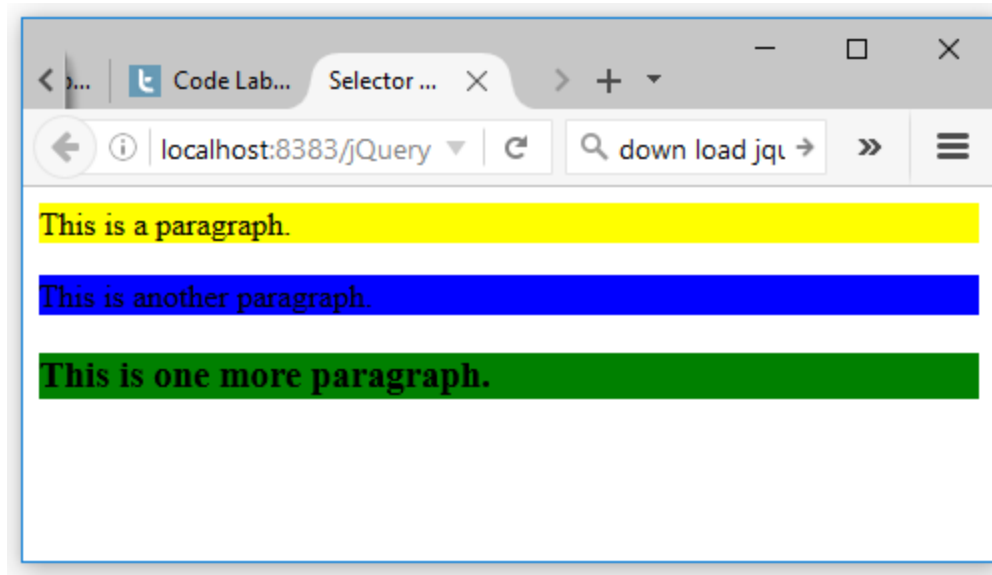

jQuery Selectors

- jQuery allows you to select an element or a set of DOM elements.
- Use the dollar sign/parentheses and provide a selector.
 - *example:* `$("p")`
- We can specify multiple selectors, combinators.
- In jQuery a DOM element can be selected by:
 - ❑ Selecting Elements by ID.
 - ❑ Selecting Elements by Class Name.
 - ❑ Selecting Elements by Name.
 - ❑ Selecting Elements by Attribute.
 - ❑ Selecting Elements by Compound CSS Selector.
- In addition to the CSS defined selectors, jQuery provides its own custom selector to further enhancing the capabilities of selecting elements on a page.

Example:

```
<html>
  <head>
    <title>Selectors Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        // selecting by ID
        ① $("#para").css("background", "yellow");
        // selecting by Class Name
        ② $(".para1").css("background", "blue");
        // selecting by Name
        ③ $("h3").css("background", "green");

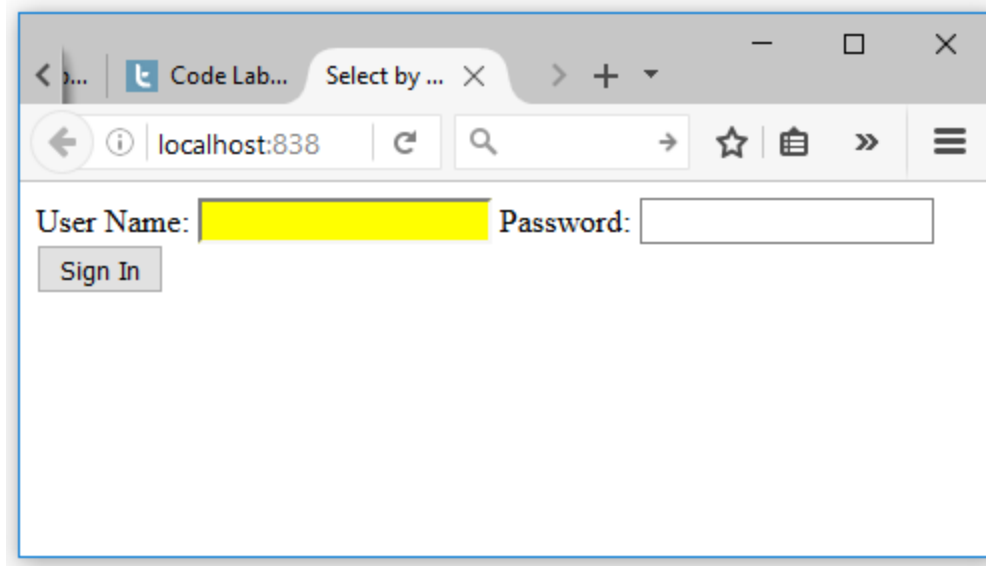
      });
    </script>
  </head>
  <body>
    ① <p id="para">This is a paragraph.</p>
    ② <p class="para1">This is another paragraph.</p>
    ② <h3>This is one more paragraph.</h3>
  </body>
</html>
```



Example: Selecting Elements by Attribute

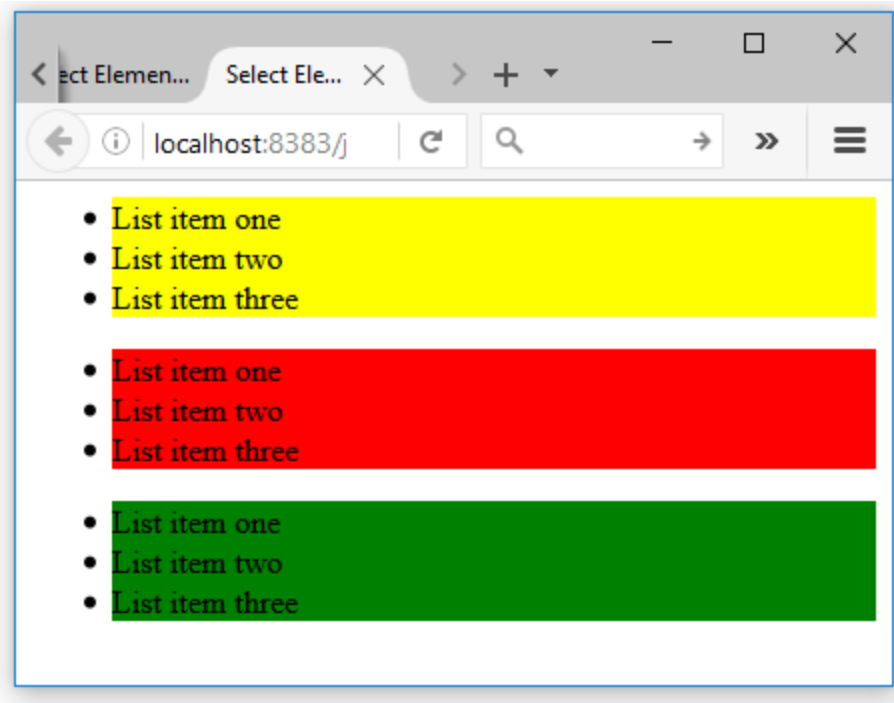
```
<html>
  <head>
    <title>Select by Attribute Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script type="text/javascript">
      $(document).ready(function () {
        // select by Attribute
        $('input[type="text"]').css("background", "yellow");
      });
    </script>
  </head>
  <body>
    <form>
      <label>User Name: <input type="text"></label>
      <label>Password: <input type="password"></label>
      <input type="submit" value="Sign In">
    </form>
  </body>
</html>
```

Example: Selecting Elements by Attribute



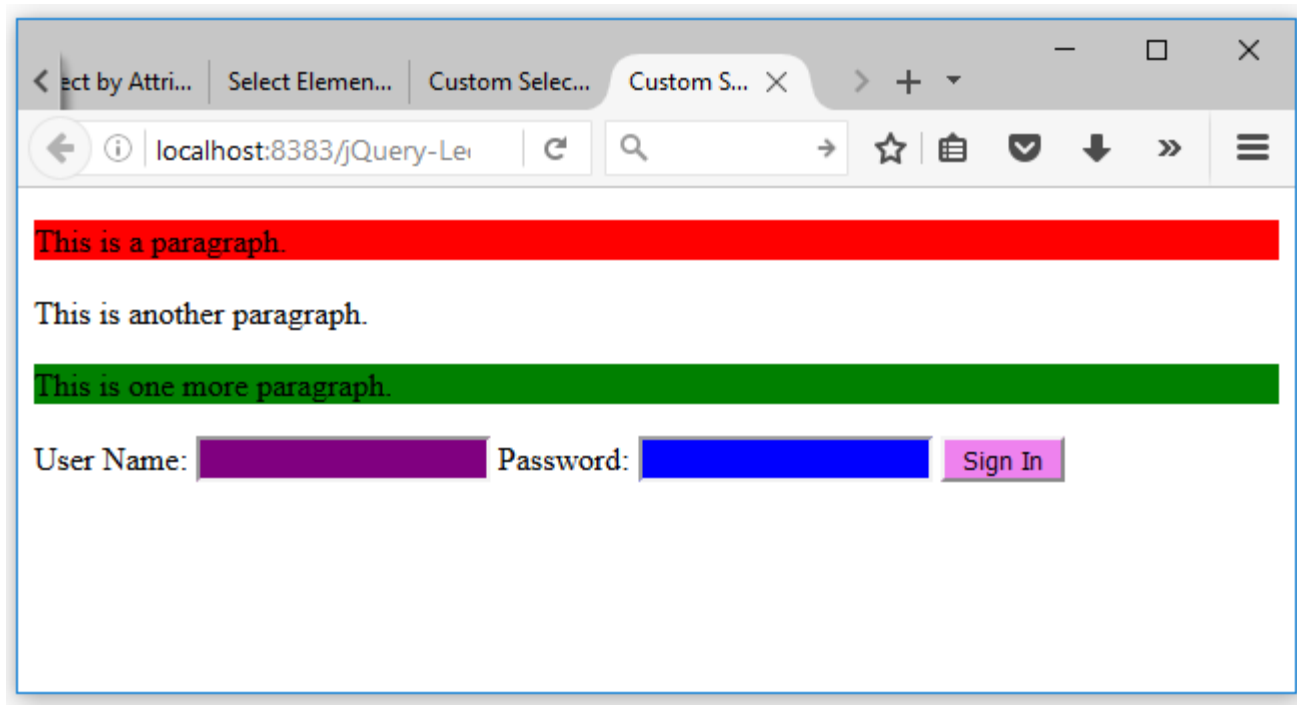
Example: Selecting Elements by Compound CSS Selector

```
<html>
  <head>
    <title>Select Element by Compound Selector Example </title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        ① // select li elements inside the ul elements
          $("ul li").css("background", "yellow");
        ② // select li elements only inside the ul element with id list
          $("ul#list li").css("background", "red");
        ③ // select li elements inside all the ul element with class list
          $("ul.list li").css("background", "green");
      });
    </script>
  </head>
  <body>
    ① <ul>
      <li>List item one</li>
      <li>List item two</li>
      <li>List item three</li>
    </ul>
    ② <ul id="list">
      <li>List item one</li>
      <li>List item two</li>
      <li>List item three</li>
    </ul>
    ③ <ul class="list">
      <li>List item one</li>
      <li>List item two</li>
      <li>List item three</li>
    </ul>
  </body>
</html>
```



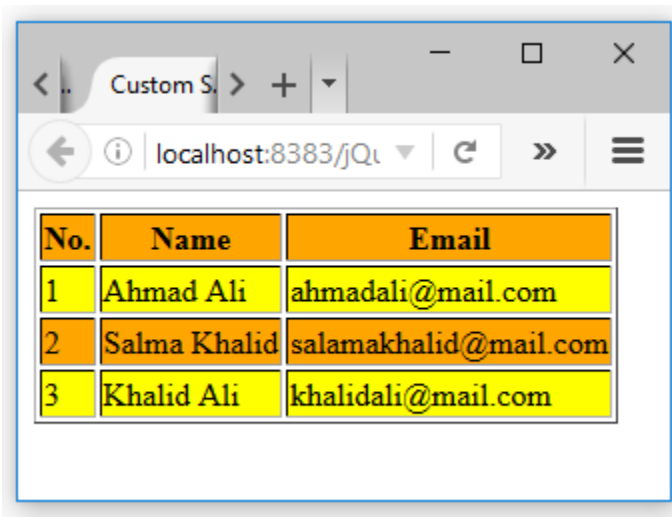
Example1: Custom Selector

```
<html>
  <head>
    <title>Custom Selector Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        // select first paragraph element
        ① $("p:first").css("background", "red");
        // select last paragraph element
        ② $("p:last").css("background", "green");
        // select all input elements with type text inside a form
        ③ $("form :text").css("background", "purple");
        // select all input elements with type password inside a form
        ④ $("form :password").css("background", "blue");
        // select all input elements with type submit inside a form
        ⑤ $("form :submit").css("background", "violet");
      });
    </script>
  </head>
  <body>
    ① <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
    ② <p>This is one more paragraph.</p>
    <form>
      ③ <label>User Name: <input type="text"></label>
      ④ <label>Password: <input type="password"></label>
      ⑤ <input type="submit" value="Sign In">
    </form>
  </body>
</html>
```

Example2: Custom Selector

```
<html>
  <head>
    <title>Custom Selector Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        // select table rows appearing at odd places
        ① $("tr:odd").css("background", "yellow");
        // select table rows appearing at even places
        ② $("tr:even").css("background", "orange");
      });
    </script>
  </head>
  <body>
    <table border="1">
      <thead>
        ② <tr>
          <th>No.</th>
          <th>Name</th>
          <th>Email</th>
        </tr>
      </thead>
      <tbody>
        ① <tr>
          <td>1</td>
          <td>Ahmad Ali</td>
          <td>ahmadali@mail.com</td>
        </tr>
        ② <tr>
          <td>2</td>
          <td>Salma Khalid</td>
          <td>salamakhalid@mail.com</td>
        </tr>
        ① <tr>
          <td>3</td>
          <td>Khalid Ali</td>
          <td>khalidali@mail.com</td>
        </tr>
      </tbody>
    </table>
  </body>
</html>
```



A screenshot of a web browser window displaying a table. The browser's address bar shows 'localhost:8383/jQuery'. The table has three columns: 'No.', 'Name', and 'Email'. The data is as follows:

No.	Name	Email
1	Ahmad Ali	ahmadali@mail.com
2	Salma Khalid	salamakhalid@mail.com
3	Khalid Ali	khalidali@mail.com

jQuery Events

What are Events ?

- Events are often triggered by the user's interaction with the web page, such as
 - when a link or button is clicked.
 - text is entered into an input box or textarea.
 - selection is made in a select box.
 - key is pressed on the keyboard,.
 - the mouse pointer is moved.
- jQuery enhances the basic event-handling mechanisms by offering the events methods for most native browser events, some of these methods are `ready()`, `click()`, `keypress()`, `focus()`, `blur()`, `change()`.

jQuery Events

- `.blur()`
- `.change()`
- `.click()`
- `.focus()`
- `.hover()`
- `.select()`
- `.submit()`
- `.mousedown()`
- `.mouseenter()`
- `.mouseleave()`
- `.keypress()`
- `.keyup()`

Ready Event (**`$(document).ready()`**)

- In order to traverse and manipulate the page we must wait until it's ready to be used.
- jQuery has a ready event that fires the instant the DOM is ready to be worked with.

```
$(document).ready(function(){  
    // Your jQuery code goes in here  
})
```

Example: The ready() Method

```
<html>
  <head>
    <title>Selectors Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        // selecting by ID
        $("#para").css("background", "yellow");
        // selecting by Class Name
        $(".para1").css("background", "blue");
        // selecting by Name
        $("h3").css("background", "green");

      });
    </script>
  </head>
  <body>
    <p id="para">This is a paragraph.</p>
    <p class="para1">This is another paragraph.</p>
    <h3>This is one more paragraph.</h3>
  </body>
</html>
```


Example: The click() Method

```
<html>
  <head>
    <title>click Event Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        $("p").click(function () {
          $(this).slideUp();
        });

        $("h1").click(function () {
          $(this).slideUp();
        });
      });
    </script>
  </head>
  <body>
    <h1> Header One</h1>
    <p>Click on me and I'll disappear.</p>
    <p>Click on me and I'll disappear.</p>
    <p>Click on me and I'll disappear.</p>
  </body>
</html>
```

Common jQuery Functions

jQuery Functions

- jQuery provides many functions that can be applied to the selected elements.

Example:

- The `css()` function.
 - it is used to alter styling properties .
 - it takes two argument : property and value.
- To use jQuery function the following syntax is used:

selector + dot + function + argument
`$("p").css("font-size", "14pt");`

Chaining

- You can have multiple actions against a single set of elements.
 - jQuery allows to chain function calls into one statement
-
- `$("#p").css("font-size", "14pt").css("color", "red");`

Example:

```
$("#div").show();  
$("#div").show().css("color","blue");  
$("#div").show().css("color","blue").remove();
```

Common jQuery Functions

CSS()	To alter styling properties.
addClass()	To add a class from an element.
removeClass()	To remove a class from an element.
text()	To get/set the inner text of an element.
html()	To get/set the full inner HTML of an element.
attr()	To get/set the attribute value of an element.
val()	To get/set values of form element.
ready()	The code in this function will only be executed after the page is downloaded, loaded, and ready.

Getting and Setting Attributes

- **Getting attributes:**

```
$("#em").attr("title")  
$("#label").html()  
$("#p:first").text()  
$("#input").val()
```

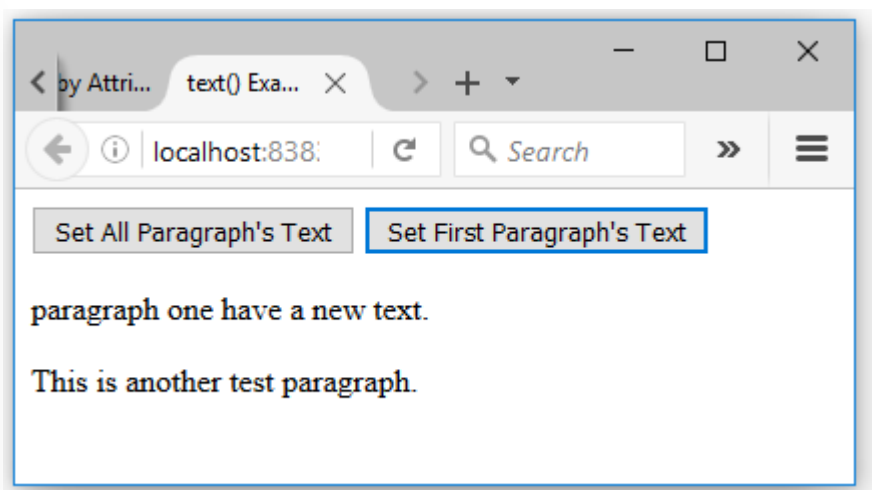
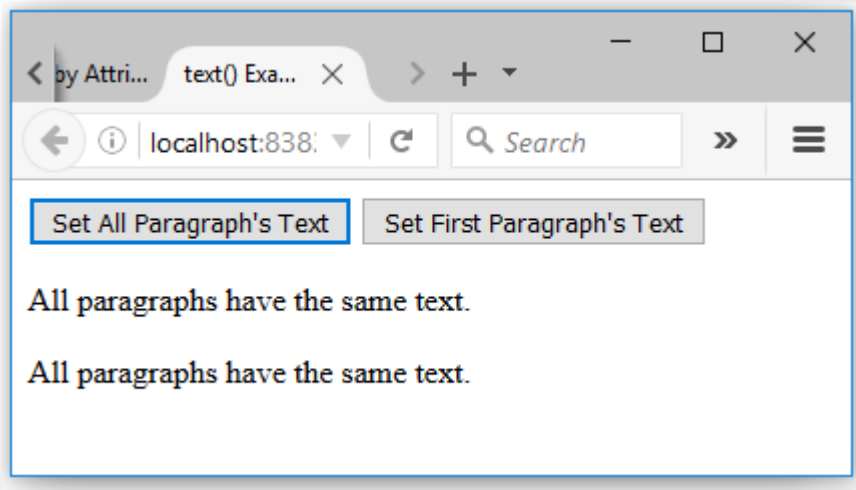
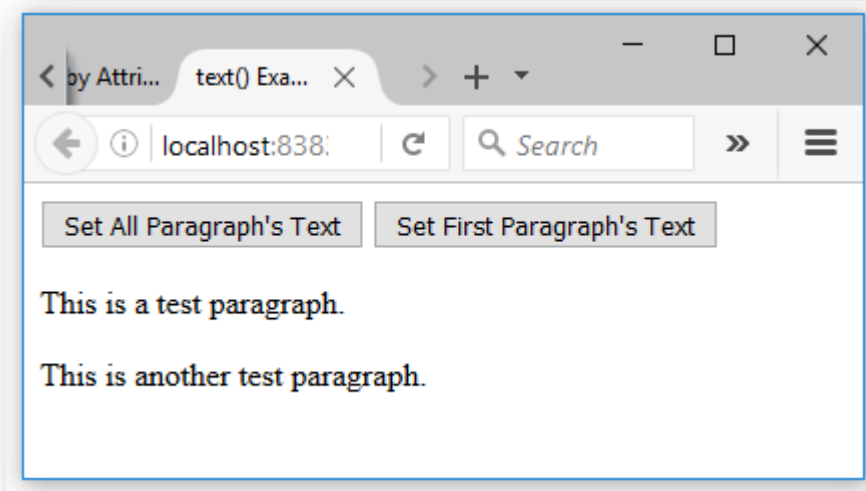
- **Setting attributes:**

```
$("#em").attr("title", "hello")  
$("#label").html("hello")  
$("#p:first").text("hello")  
$("#input").val("hello")
```

Example: **text()** function

```
<html>
  <head>
    <title>text() Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        $(".btn-one").click(function () {
          $("p").text("All paragraphs have the same text.");
        });

        $(".btn-two").click(function () {
          $("p:first").text(" paragraph one  have a new text.");
        });
      });
    </script>
  </head>
  <body>
    <button type="button" class="btn-one">Set All Paragraph's Text</button>
    <button type="button" class="btn-two">Set First Paragraph's Text</button>
    <p>This is a test paragraph.</p>
    <p>This is another test paragraph.</p>
  </body>
</html>
```

Manipulating DOM Elements

DOM Manipulation

- jQuery uses a set of functions to add/remove DOM element.

<code>append()</code>	Adds to the inner part of the element at the end.
<code>prepend()</code>	Adds to the inner part of the element at the beginning.
<code>after()</code>	Adds content to the DOM after the matched element.
<code>before()</code>	Adds content to the DOM before the matched element.
<code>remove()</code>	Removes the element (and its inner text and all descendants) entirely.
<code>attr()</code>	To get/set the attribute value of an element.
<code>val()</code>	To get/set values of form element.
<code>addClass()</code>	Adds one or more classes to the selected elements
<code>toggleClass()</code>	Toggles between adding/removing classes from the selected elements

DOM Manipulation

```
$("#target").addClass("css_class");
```

```
$("#target").toggleClass("css_class");
```

```
$("p").append("<strong>Hello</strong>");
```

```
$("p").after("<b>Hello</b>");
```

```
$("p").before("<b>Hello</b>");
```

Example: before - after

```
<html>
  <head>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        $("#btn1").click(function () {
          $("p").before(" <b>Appended before text</b> <br>");
        });

        $("#btn2").click(function () {
          $("p").after(" <b>Appended after text</b> <br>");
        });
      });
    </script>
  </head>
  <body>

    <p>This is a paragraph.</p>

    <button id="btn1">Append before</button>
    <button id="btn2">Append after</button>

  </body>
</html>
```

Example: append-remove

```
<html>
  <head>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        $("#btn1").click(function () {
          $("ol").append("<li>List item "+($("li").size()+1)+"</li>");
        });
      });
      $(document).ready(function () {
        $("#btn2").click(function () {
          $("li").last().remove();
        });
      });
    </script>
  </head>
  <body>

    <ol>
      <li>List item 1</li>
      <li>List item 2</li>
      <li>List item 3</li>
    </ol>

    <button id="btn1">Append list items</button>
    <button id="btn2">Remove list items</button>

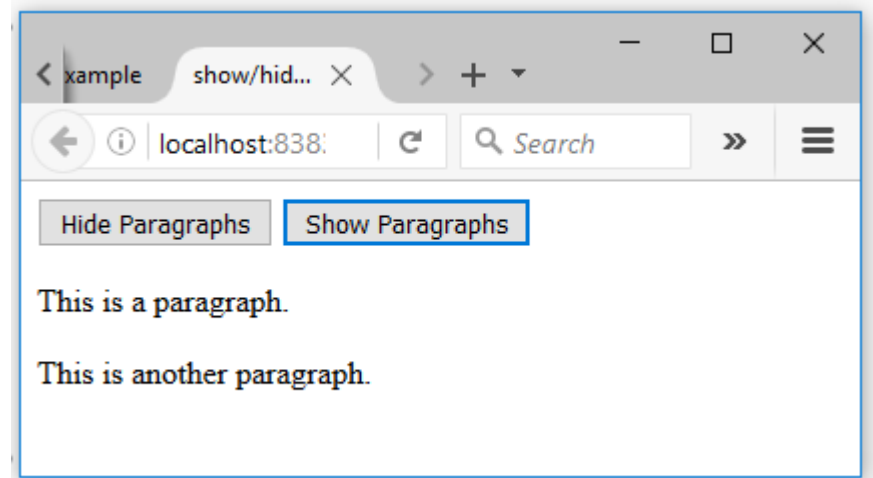
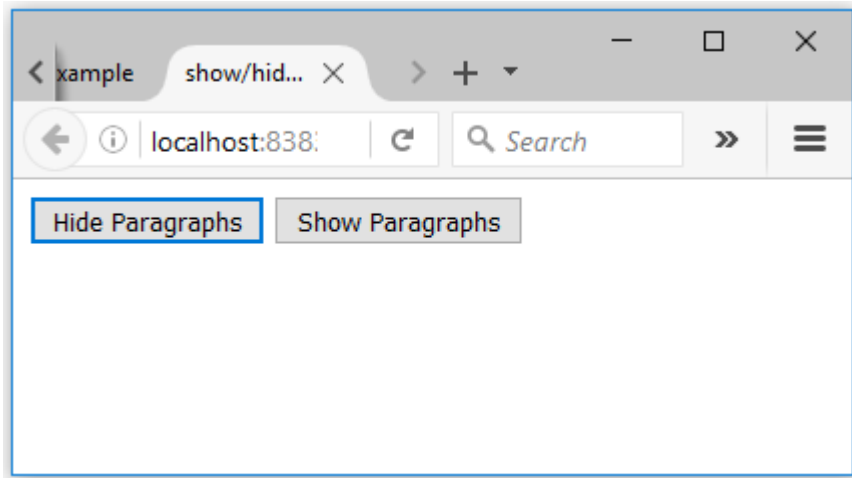
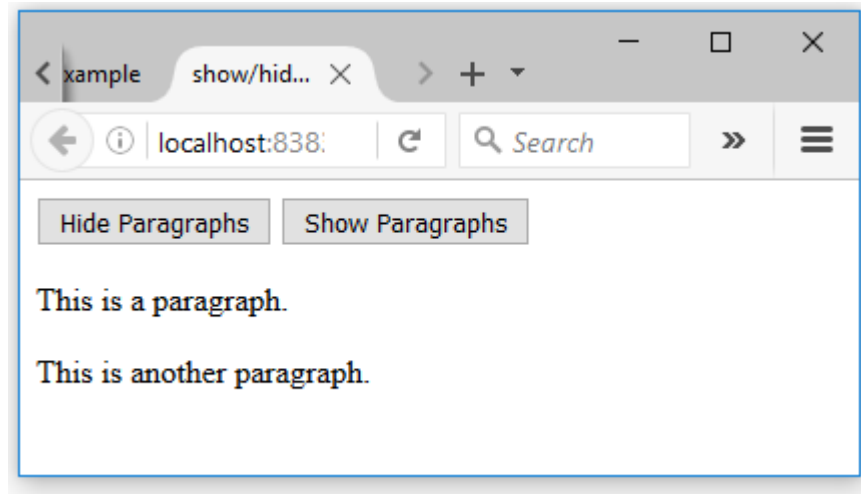
  </body>
</html>
```

jQuery Effects

- Using jQuery elements can be made visible/invisible without removing them from the DOM.
- Using jQuery, we can show/hide elements with some effects such as:
 - `fadeIn()`.
 - `fadeOut()`.
 - `slideUp()`.
 - `slideDown()`.

Example: show - hide

```
<html>
  <head>
    <title>show/hide Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        $(".hide-btn").click(function () {
          $("p").hide();
        });
      });
      $(document).ready(function () {
        $(".show-btn").click(function () {
          $("p").show();
        });
      });
    </script>
  </head>
  <body>
    <button type="button" class="hide-btn">Hide Paragraphs</button>
    <button type="button" class="show-btn">Show Paragraphs</button>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

Example: slideUp-slideDown

```
<html>
  <head>
    <title>slideUp/slideDown Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        $(".slideUp-btn").click(function () {
          $("p").slideUp();
        });
      });
      $(document).ready(function () {
        $(".slidDown-btn").click(function () {
          $("p").slideDown();
        });
      });
    </script>
  </head>
  <body>
    <button type="button" class="slideUp-btn">slideUp Paragraphs</button>
    <button type="button" class="slidDown-btn">slideDown Paragraphs</button>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

Example: fadeIn-fadeOut

```
<html>
  <head>
    <title>fadeIn/fadeOut Example</title>
    <script src="../js/jquery-2.2.3.min.js"></script>
    <script>
      $(document).ready(function () {
        $(".fadeUp-btn").click(function () {
          $("p").fadeIn();
        });
      });
      $(document).ready(function () {
        $(".fadDown-btn").click(function () {
          $("p").fadeOut();
        });
      });
    </script>
  </head>
  <body>
    <button type="button" class="fadeUp-btn">fadeIn Paragraphs</button>
    <button type="button" class="fadDown-btn">fadeOut Paragraphs</button>
    <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
  </body>
</html>
```

Thanks!