

CI6206

Internet Programming



Lecture

Client-side Web Development:
jQuery

Dynamic HTML (DHTML)






- ❑ Manipulating the web page's structure is essential for creating a highly responsive UI
- ❑ Two main approaches
 - Manipulate page via plain Javascript
 - Manipulate page using Javascript + **library** (e.g., jQuery)



What is jQuery?

jQuery is a fast and concise JavaScript Library that simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development. (jquery.com)

What can you do with jQuery?

	find	elements in an HTML document
	change	HTML content
	listen	to what a user does and react accordingly
	animate	content on the page
	talk	over the network to fetch new content

What can you do with jQuery?

- ❑ Read the contents of DOM nodes (tag)
- ❑ Modify the contents of DOM nodes
- ❑ Modify the appearance of DOM nodes
- ❑ Create and attach new DOM nodes
- ❑ Remove DOM nodes
- ❑ Run a function right when the page is ready
- ❑ Add and remove event handlers
- ❑ Retrieve content from a web server
- ❑ Send content to a web server



To start ...

Get latest library from : <http://jquery.com>



The screenshot shows the jQuery website homepage. At the top, there is a navigation bar with social media icons and links for Plugins, Contribute, Events, Support, and jQuery Foundation. Below this is a blue header with the jQuery logo and tagline "write less, do more.". To the right of the logo is a banner for the "jQuery Accessibility Summit" presented by deque Systems, scheduled for October 10-11 in Herndon, VA, with a "CLICK FOR INFO" button. Below the header is a dark blue navigation bar with links for Download, API Documentation, Blog, Plugins, and Browser Support, along with a search bar. The main content area features three large icons: a box for "Lightweight Footprint", a stylized 'E' for "CSS3 Compliant", and a globe with arrows for "Cross-Browser". Each icon has a brief description of its feature. To the right of these icons is a large orange button labeled "Download jQuery" with the versions "v1.10.2 or v2.0.3". Below this button are two links: "View Source on GitHub" and "How jQuery Works".

jQuery
write less, do more.

deque Systems presents the
jQuery Accessibility Summit
OCTOBER 10-11, HERNDON, VA [CLICK FOR INFO](#)

Download API Documentation Blog Plugins Browser Support Search

Lightweight Footprint
Only 32kB minified and gzipped.
Can also be included as an AMD module

CSS3 Compliant
Supports CSS3 selectors to find elements as well as in style property manipulation

Cross-Browser
IE, Firefox, Safari, Opera, Chrome, and more

[View Source on GitHub](#) →
[How jQuery Works](#) →

Download jQuery
v1.10.2 or v2.0.3

Reference

- ▣ Add more reference if required.

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

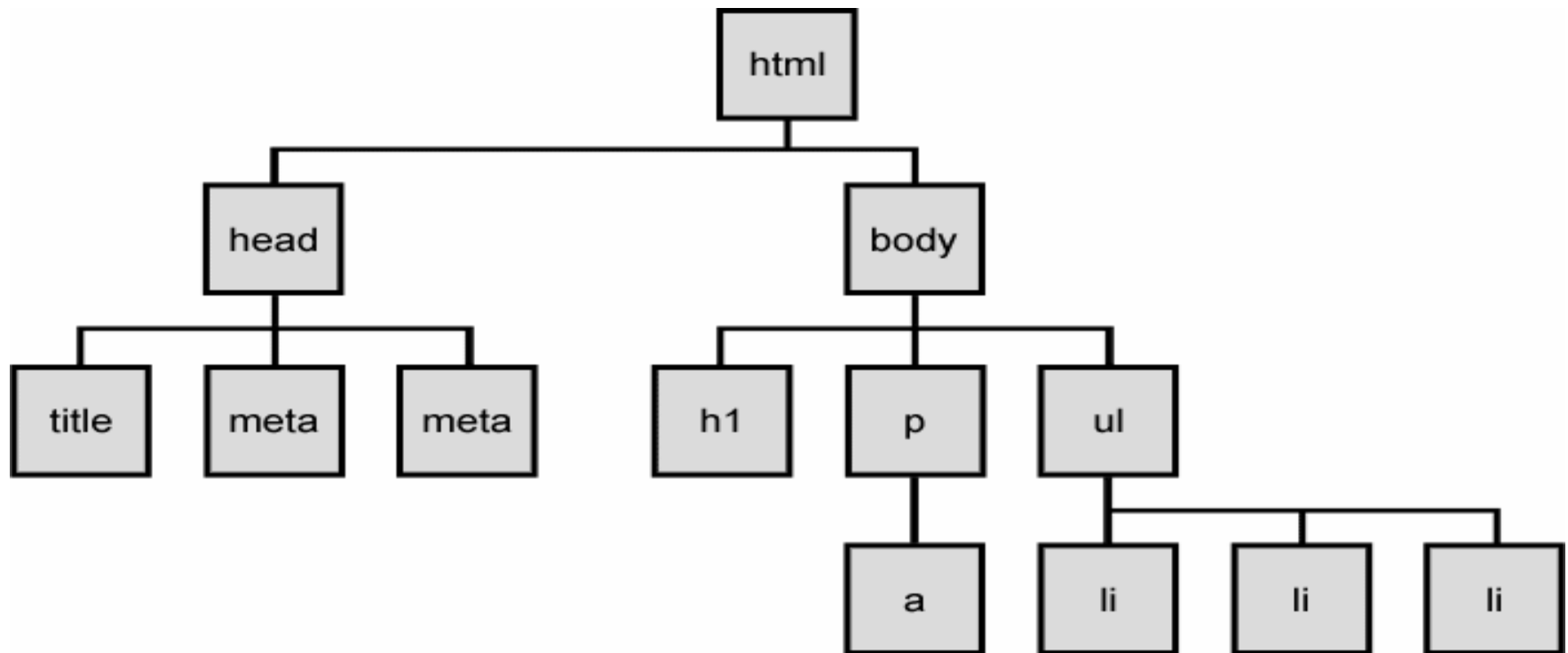
```
<script  
src="https://ajax.googleapis.com/ajax/libs/  
jquery/3.5.1/jquery.min.js">  
</script>
```

Document Object Model (DOM)

- Web page's structure
 - A tree structure
 - One node per HTML element
 - Each node can have attributes



Document Object Model (DOM)



Document Object Model (DOM)

What does that DOM structure look like?

HTML document

```
<!DOCTYPE html>
<html>
<head>
  <title>jQuery Adventures</title>
</head>
<body>
  <h1>Where do you want to go?</h1>
  <p>Plan your next adventure.</p>
</body>
</html>
```

Inside the DOM, HTML elements become "nodes" which have relationships with one another.

The DOM



node types: element text

window.onload

- ❑ We cannot use the DOM before the page is constructed. jQuery gives us a more compatible way to do this.

- The DOM way

```
window.onload = function() { // do stuff with the DOM }
```

- The direct jQuery translation

```
$(document).ready(function() { // do stuff with the DOM });
```

- The jQuery way

```
$(function() { // do stuff with the DOM });
```

Javascript example

```
window.onload = (function(){  
    return function(){  
        var aarr = [];  
        for (var z=1; z<=10; z++) {  
            aarr.push(document.getElementById("a"+z));  
            alert(aarr[z-1].id); }  
        };  
    })  
();
```

```
var body = document.getElementsByTagName('BODY')[0];  
if(body && body.readyState == 'loaded') {  
    DoStuffFunction(); }  
else {  
    if (window.addEventListener) { window.addEventListener('load',  
        DoStuffFunction, false);  
    }  
    else { window.attachEvent('onload',  
        DoStuffFunction);  
    } }  
}
```

Jquery Example

```
$(document).ready(function() {  
  var aarr = [];  
  for (var z=1; z<=10; z++) {  
    c = z-1;  
    aarr[c] = document.getElementById("a"+z); }  
});
```

When the DOM is ready...

```
$(document).ready(function(){  
    //...  
});
```

- ❑ Think of the document ready function as a self-executing function which fires after the page elements have loaded.
- ❑ Uses advanced listeners for detecting.
- ❑ `window.onload()` is a fallback.

.ready(function() ..

```
$(document).ready(function() {  
    //do jQuery stuff when DOM is ready  
});
```

```
$(function(){ //jQuery code here });
```

Explanation

Both literally mean the same thing.

.ready(function() ..

```
jQuery(document).ready(function($) {  
    //do jQuery stuff when DOM is ready  
});
```

Explanation

Adding the jQuery can help prevent conflicts with other JS frameworks. (Ample, AngularJS, Dojo etc)

Why do conflicts happen?

Many JavaScript Libraries/Frameworks use the same shortcut name which is the dollar symbol \$ which might cause confusion. Browser become confused.

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

```
jQuery(document).ready(function(){  
    $("p").click(function(){  
        $(this).hide();  
    });  
});
```


Aspects of the DOM and jQuery

- ❑ **Identification:** how do I obtain a reference to the node that I want.
- ❑ **Traversal:** how do I move around the DOM tree.
- ❑ **Node Manipulation:** how do I get or set aspects of a DOM node.
- ❑ **Tree Manipulation:** how do I change the structure of the page.

Selecting groups of DOM objects

name	description
<u>getElementById</u>	returns array of descendents with the given tag, such as "div"
<u>getElementsByTagName</u>	returns array of descendents with the given tag, such as "div"
<u>getElementsByName</u>	returns array of descendents with the given name attribute (mostly useful for accessing form controls)
<u>querySelector</u>	returns the first element that would be matched by the given CSS selector string
<u>querySelectorAll</u> *	returns an array of all elements that would be matched by the given CSS selector string

jQuery / DOM comparison

DOM method	jQuery equivalent
<code>getElementById("id")</code>	<code>\$("#id")</code>
<code>getElementsByTagName("tag")</code>	<code>\$("tag")</code>
<code>getElementsByName("somename")</code>	<code>\$("[name='somename']")</code>
<code>querySelector("selector")</code>	<code>\$("selector")</code>

The Magic `$()` function

```
var divTails = $("")
```

Create HTML elements on the fly

```
<!DOCTYPE html>
<html>
<head>
<script src="..."></script>
<script>
$(document).ready(function(){
    $("#btn1").click(function(){
        $("#p").append(" <b>Appended text</b>.");
    });

    $("#btn2").click(function(){
        $("#p2").append("<li>Appended item</li>");
    });
});
</script>
</head>
<body>

<p>This is a paragraph.</p>
<p id="p2">This is another paragraph.</p>

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

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

</body>
</html>
```

http://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_html_append

The Magic `$()` function

```
$(“div”).hide();  
$(“div”, $(“p”)).hide();
```

Selects document elements
(more in a moment...)

Example (find `<p>` and hide it)

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
</head>
```

```
<body>
```

```
<p>If you click on the "Hide" button, I will  
disappear.</p>
```

```
<button id="hide">Hide</button>
```

```
<button id="show">Show</button>
```

```
</body>
```

```
</html>
```

Example (find <p> and hide it)

```
<!DOCTYPE html>
<html>
<head>
<script
  src="http://ajax.googleapis.com/ajax/libs/jquery/1.12.2/jquery.min.js">
</script>
<script>
$(document).ready(function(){
  $("#hide").click(function(){
    $("p").hide();
  });
  $("#show").click(function(){
    $("p").show();
  });
});
</script>
</head>
<body>
<p>If you click on the "Hide" button, I will disappear.</p>
<button id="hide">Hide</button>
<button id="show">Show</button>
</body>
</html>
```


Example (find <p> and hide it)

```
<script
  src="http://ajax.googleapis.com/ajax/libs/jquery/1.12.2/jquery.min.js">
</script>
<script>
$(document).ready(function(){
  $("#hide").click(function(){
    $("p").hide();
  });
  $("#show").click(function(){
    $("p").show();
  });
});
</script>
```

jQuery Selectors

<http://api.jquery.com/category/selectors/>

All Selector

```
$("*") // find everything
```

Selectors return a pseudo-array of jQuery elements

Basic Selectors

By Tag:

```
$("div")  
// <div>Hello jQuery</div>
```

By ID:

```
$("#usr")  
// <span id="usr">John</span>
```

By Class:

```
$(".menu")  
// <ul class="menu">Home</ul>
```

Yes, jQuery implements CSS Selectors!

More Precise Selectors

`$(“div.main”)`

- *tag and class*
- *class selector within div tag*

`$(“table#data”)`

- *tag and id*
- *id selector within table tag*

Combination of Selectors

// find by id + by class

`$("#content, .menu")`

// multiple combination

`$(“h1, h2, h3, div.content”)`

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>multiple demo</title>
  <style>
    div, span, p {
      width: 126px;
      height: 60px;
      float: left;
      padding: 3px;
      margin: 2px;
      background-color: #eee;
      font-size: 14px;
    }
  </style>
  <script src="https://code.jquery.com/jquery-3.5.0.js"></script>
</head>
<body>

<div>div</div>
<p class="myClass">p class="myClass"</p>
<p class="notMyClass">p class="notMyClass"</p>
<span>span</span>

<script>
$( "div, span, p.myClass" ).css( "border", "3px solid red" );
</script>

</body>
</html>
```

div

p class="myClass"

p
class="notMyClass"

span

Hierarchy Selectors

`$(“table td”)` *// descendants*

`$(“tr > td”)` *// children*

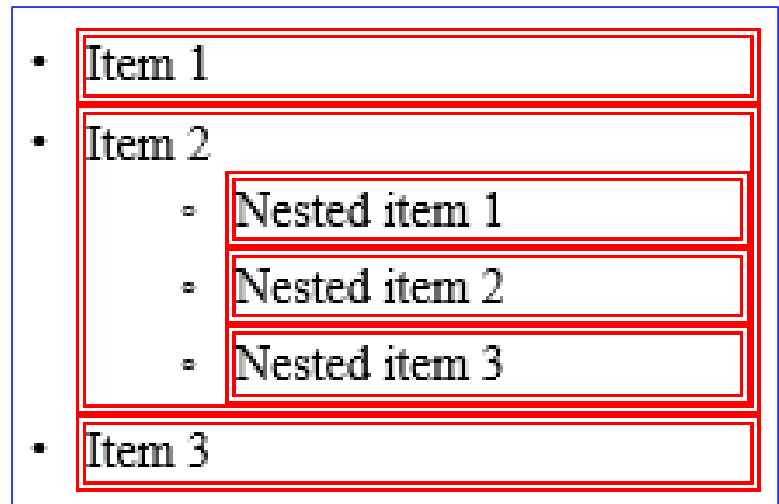
`$(“label + input”)` *// next*

`$(“#content ~ div”)` *// siblings*

Decendents

```
<html lang="en">
<head>
<title>child demo</title>
  <style>
    body {font-size: 14px;}
  </style>
  <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
</head>
<body>
<ul class="topnav">

  <li>Item 1</li>
  <li>Item 2
    <ul>
      <li class="1">Nested item 1</li>
      <li class="2">Nested item 2</li>
      <li class="3">Nested item 3</li>
    </ul>
  </li>
  <li>Item 3</li>
</ul>
<script>
$( "ul li" ).css( "border", "3px double red" );
</script>
```

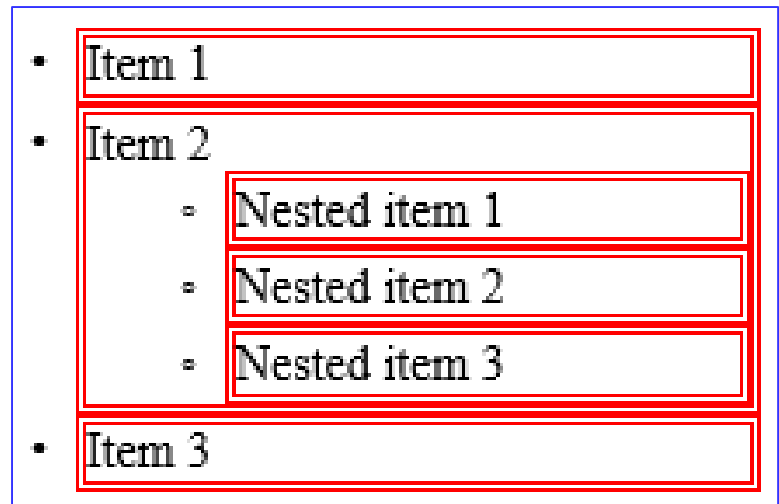


Children

```
<html lang="en">
<head>
  <title>child demo</title>
  <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
</head>
<body>
<ul class="topnav">
  <li>Item 1</li>
  <li>Item 2
    <ul>
      <li class="1">Nested item 1</li>
      <li class="2">Nested item 2</li>
      <li class="3">Nested item 3</li>
    </ul>
  </li>
  <li>Item 3</li>
</ul>

<script>
  $( "ul > li" ).css( "border", "3px double red" );
</script>

</body>
</html>
```



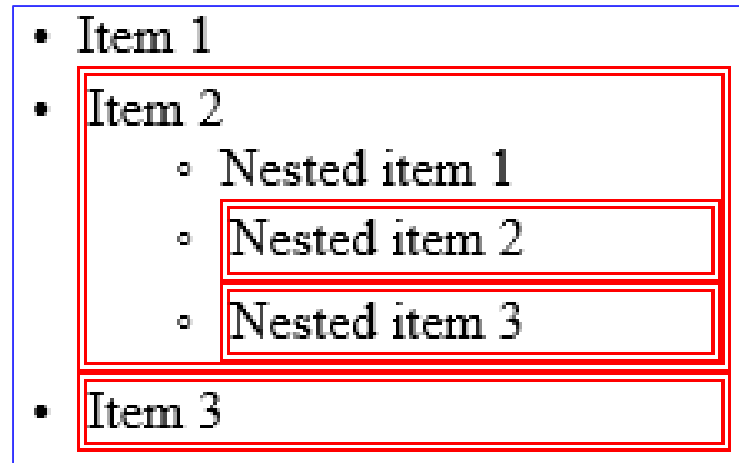
Next

```
<html lang="en">
<head>
  <title>child demo</title>
  <script src="https://code.jquery.com/jquery-1.10.2.js"></script>
</head>
<body>
<ul class="topnav">

  <li>Item 1</li>
  <li>Item 2
    <ul>
      <li class="1">Nested item 1</li>
      <li class="2">Nested item 2</li>
      <li class="3">Nested item 3</li>
    </ul>
  </li>
  <li>Item 3</li>
</ul>

<script>
$( "li + li" ).css( "border", "3px double red" );
</script>

</body>
</html>
```



```

<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>next adjacent demo</title>
  <script src="https://code.jquery.com/jquery-3.5.0.js"></script>
</head>
<body>

<form>
  <label for="name">Name:</label>
  <input name="name" id="name">
  <fieldset>
    <label for="newsletter">Newsletter:</label>
    <input name="newsletter" id="newsletter">
  </fieldset>
</form>
<input name="none">

<script>
$( "label + input" ).css( "color", "blue" ).val( "Labeled!" );
</script>

</body>
</html>

```

Name:

Newsletter:

Siblings (prev ~ siblings)

```
<body>

<div>div (doesn't match since before #prev)</div>
<span id="prev">span#prev</span>
<div>div sibling</div>
<div>div sibling <div id="small">div niece</div></div>
<span>span sibling (not div)</span>
<div>div sibling</div>

<script>
$( "#prev ~ div" ).css( "border", "3px groove blue" );
</script>

</body>
</html>
```

div (doesn't
match since
before
#prev)

span#prev

div sibling

div sibling
div niece

span sibling
(not div)

div sibling

Selects all sibling elements that follow after the “prev” element, have the same parent, and match the filtering “siblings” selector.

Selection Index Filters

<code>\$("tr:first")</code>	<i>// first element</i>
<code>\$("tr:last")</code>	<i>// last element</i>
<code>\$("tr:lt(2)")</code>	<i>// index less than</i>
<code>\$("tr:gt(2)")</code>	<i>// index gr. than</i>
<code>\$("tr:eq(2)")</code>	<i>// index equals</i>

Visibility Filters

`$(“div:visible”)`

- *Anything that consume space*

`$(“div:hidden”)`

- *Anything that do not consume space*

<http://api.jquery.com/visible-selector/>

Attribute Filters

```
$(“div[id]”) // has attribute
$(“div[id=‘firstname’]”) // equals to
$(“div[id^=‘first’]”) // starts with
$(“div[id$=‘name’]”) // ends with
$(“a[href*=‘ntu’]”) // contains
```

```
<div id=‘firstname’ .... ></div>
<a href=“wkwsci.ntu.edu.sg” ...></a>
```


Forms Selectors

`$(“input:checkbox”)` *// checkboxes*

`$(“input:radio”)` *// radio buttons*

`$(“:button”)` *// buttons*

`$(“:text”)` *// text inputs*

Forms Filters

```
$(“input:checked”)    // checked  
$(“input:selected”)  // selected  
$(“input:enabled”)    // enabled  
$(“input:disabled”)  // disabled
```

Find Dropdown Selected Item

```
<select name="cities">  
  <option value="1">Tel-Aviv</option>  
  <option value="2" selected="selected">Yavne</option>  
  <option value="3">Raanana</option>  
</select>
```

```
$( "select[name='cities'] option:selected" ).val()
```

Effects

Showing or Hiding Element

// just show

```
$(“div”).show();
```

// reveal slowly, slow=600ms

```
$(“div”).show(“slow”);
```

// hide fast, fast=200ms

```
$(“div”).hide(“fast”);
```

// hide or show in 100ms

```
$(“div”).toggle(100);
```

Sliding Elements

```
$(“div”).slideUp();
```

```
$(“div”).slideDown(“fast”);
```

```
$(“div”).slideToggle(1000);
```

Find in selected

```
// select paragraph and then find  
// elements with class 'header' inside  
$("p").find(".header").show();
```

```
// equivalent to:  
$(".header", $("p")).show();
```

CSS – change color

```
<!DOCTYPE html>
<html>
<head>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js">
</script>
<script>
$(document).ready(function(){
    $("button").click(function(){
        $("body").css("background-color","yellow");
        $("h2").css("color","red");
    });
});
</script>
</head>
<body>
<h2>Hello</h2>
<button>Set background-color to yellow</button>
</body>
</html>
```


DatePicker

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>The DatePicker</title>
<link rel="stylesheet" href="//code.jquery.com/ui/1.10.4/themes/smoothness/jquery-ui.css">
<script src="//code.jquery.com/jquery-1.9.1.js"></script>
<script src="//code.jquery.com/ui/1.10.4/jquery-ui.js"></script>
<script>
    $(function() {
        $( "#datepicker" ).datepicker();
    });
</script>
</head>
<body>
<p>Date: <input type="text" id="datepicker"></p>
</body>
</html>
```

<http://jqueryui.com/datepicker/>

jQuery UI (<http://jqueryui.com/>)

- ❑ Accordion (<http://jqueryui.com/accordion/>)
- ❑ Menu (<http://jqueryui.com/selectmenu/>)
- ❑ Effects
- ❑ Form Validation
(<http://jqueryvalidation.org/files/demo/>)
 - [Validator cheat sheet](#)

Accordion

▼ Section 1

Section 1

▶ Section 2

▶ Section 3

▶ Section 4

Accordion

```
<html lang="en">
<head>
<link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/themes/base/jquery-ui.css">
  <link rel="stylesheet" href="/resources/demos/style.css">
  <script src="https://code.jquery.com/jquery-1.12.4.js"></script>
  <script src="https://code.jquery.com/ui/1.12.1/jquery-ui.js"></script>

  <script>
    $( function() {
      $( "#accordion" ).accordion();
    } );
  </script>

</head>
```

Others

Widgets

- ❑ Menu, Selectmenu
- ❑ Datepicker
- ❑ Dialog
- ❑ Tabs
- ❑ CheckboxRadio

Effects

- ❑ Show,hide,effect,toggle,animation

Form validation (confirm password)

```
$("#submitBtn").click(function () {  
    var passwd = $("#passwd").val();  
    var cfmpwd = $("#cfmpwd").val();  
  
    if (passwd == cfmpwd) {  
        ...  
    } else {  
        //notify failure to match  
    }  
})
```

Server side validation

□ Pros

- More secure and not dependent on browser settings.
- User not able to bypass.

□ Cons

- Computationally more heavy as the request needs to be sent to the server for processing.

Client side validation

□ Pros

- Faster – do not need to send to server for processing.

□ Cons

- May fail – users turn off scripting

jQuery tutorials

□ Code Academy

- http://www.codecademy.com/courses/you-and-jquery/0?curriculum_id=4fc3018f74258b0003001f0f#!/exercises/0

□ Code School:

- <http://www.codeschool.com/courses/jquery-air-first-flight>

□ Hide and Show

- http://www.w3schools.com/jquery/tryit.asp?filename=tryjquery_hide_show

□ Slide Up and Down/ In and Out

- http://w3schools.com/jquery/tryit.asp?filename=tryjquery_slide_down

□ jQuery UI

- <http://jqueryui.com/>