



COMP 4021

Internet Computing

jQuery Introduction

Dr. Kenneth LEUNG

Slides created by Prof. Dik Lun LEE

jQuery JavaScript Library

- You know a bit about JavaScript and DOM now and know it is not easy to write JavaScript to manipulate DOM
- JQuery is a JavaScript library (<http://jquery.com>)
 - And Like any other libraries, jQuery aims to implement popular operations in easy-to-use jQuery API, e.g., **DOM manipulation, event handling, client-server interaction**
- Free and open source; most popular JavaScript library in use today; means lots of support
- Reasonably small footprint
- Cross browser support
 - Other toolkits: Yahoo UI Library (YUI), Google Web Toolkit, etc.

What jQuery Does

- A “easy-to-use” API for DOM manipulation:
 - Select DOM elements using CSS-like selectors
 - Set properties of selected DOM elements
 - Create, delete, show, hide DOM elements
 - Defines event behavior (click, mouse movement, dynamic styles, animations, dynamic content)
 - While CSS separate content from style, jQuery separate behavior from structure (i.e., no need to worry about traversing a tree)
- You can do all of these using plain JavaScript, but jQuery is a lot easier to use. Why?
- Use CSS selector to **select a group of objects**, AND
- Apply an **operation to all objects**, OR
- **Iterate through each of them** to perform specialized action

Let's Begin: jQuery Ready Function

- Execute a function as soon as a page is fully loaded

```
<html> <head>
```

```
  <script src="jquery-2.1.4.js"></script>
```

```
  <script>
```

```
    $(document).ready(function() {
```

Your jQuery code here

```
    },
```

```
  </script>
```

```
</head><body>...</body></html>
```

Or abbreviated as:

```
$(function() {  
  ...  
});
```

What is the problem of executing a function before a page is fully loaded?

How Easy is jQuery?

```
$ (function() {  
  $("#greenbox").click(function() {  
    $("#greenbox").hide();  
    $("#redbox").show();  
  });  
  $("#redbox").click(function() {  
    $("#redbox").hide();  
    $("#greenbox").show();  
  });  
});
```

Try to read this code, and imagine do the same using JavaScript

Website Administration



Instructor: Harry Plantinga

Course objectives: With modern content management systems, complex, good-looking, and functional web sites can be constructed with little programming. This course presents an introduction to many of the topics needed for setting up and administering a Web site with a content management system.

Putting all this knowledge to good use, we will attempt to work with local non-profit organizations, setting up a website for them according to their specifications.

Website Administration

Instructor: Harry Plantinga

Course objectives: With modern content management systems, complex, good-looking, and functional web sites can be constructed with little programming. This course presents an introduction to many of the topics needed for setting up and administering a Web site with a content management system.

Putting all this knowledge to good use, we will attempt to work with local non-profit organizations, setting up a website for them according to their specifications.



Select Objects and Perform Action (1)

CSS Style

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

```
$("div").addClass("blue");
$("div").removeClass("blue");
```

- \$(selector).methodName();
- \$("div") is the selector that returns all DIV elements in DOM
- addClass() add the specified class to all the DIV elements:
<div class=**"blue"**> I have a new color! </div>

Select Objects and Perform Action (2)

`$(selector).methodName();`

<code>\$("p").click()</code>	Add click event (and event handler) to every <p> object
<code>\$("p").hide()</code>	Hide all <p> objects
<code>\$("p").show()</code>	Show all <p> objects
<code>\$("p").toggle()</code>	Toggle all <p> objects between "hide" and "show"

Example: jQuery in Action

```
<script type="text/javascript">
    $(function(){
        $('p').html("Hello World !! ");
    });
```

```
</script>
```

```
<body>
```

```
<p></p>
```

```
</body>
```

- What would be displayed?
- What if there are many <p>...</p> tags?
- What happens to <p>Existing text</p>
- How to achieve same result with pure JavaScript?

jQuery vs JavaScript

- Hide all DIVs

```
var divs = document.getElementsByTagName('div');  
for (i=0 ; i<divs.length; i++) {  
    divs[ i ].style.display = 'none';  
}
```

JavaScript

```
$('#div').hide();
```

jQuery

jQuery vs JavaScript

- Click a button to display or hide a DIV

```
function toggle_visibility(id) {  
    var e = document.getElementById(id);  
    if (e.style.display != 'none')  
        e.style.display = 'none';  
    else  
        e.style.display = 'block'; }  
... ..
```

```
<button id="mybutton"  
    onclick="toggle_visibility('foo');">  
    Click to toggle visibility</button>  
<div id="foo">Hi</div>
```

JavaScript

```
<button id="mybutton">Click</button>  
$(function(){  
    $("#mybutton").click(function(){  
        $("#foo").toggle();  
        return false;  
    });  
});
```

jQuery

Power of jQuery

- By now, you can see that jQuery's power comes from:
 - Powerful CSS selectors
 - Perform action on a whole group of tags
 - High-level, commonly used functions: `hide()`, `show()`, `toggle()`, etc.

Basic CSS Selectors (more to come next)

<code>\$('#id')</code>	element with a specified ID
<code>\$('p')</code>	all elements with the specified name
<code>\$('.class')</code>	all elements with the specified class
<code>\$('*')</code>	all elements