

# Internet Technologies

## jQuery

Mahesh Kumar

[[maheshkumar@andc.du.ac.in](mailto:maheshkumar@andc.du.ac.in)]

Course Web Page

[[www.mkbhandari.com/mkwiki](http://www.mkbhandari.com/mkwiki)]

# Outline

---

- 1 What is jQuery?
- 2 Finding elements
- 3 Doing things with your Selection

# Introduction to jQuery

---

- jQuery is a lightweight, "write less, do more", JavaScript library.
- The purpose of jQuery is to make it much easier to use JavaScript on your website.
- jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.
- jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

# Introduction to jQuery

---

- The jQuery library contains the following **features**:
  - 1 HTML/DOM manipulation
  - 2 CSS Manipulation
  - 3 HTML event methods
  - 4 Effects and Animation
  - 5 AJAX
  - 6 Utilities
- In addition, jQuery has **plugins** for almost any task out there.

# What is a jQuery?

---

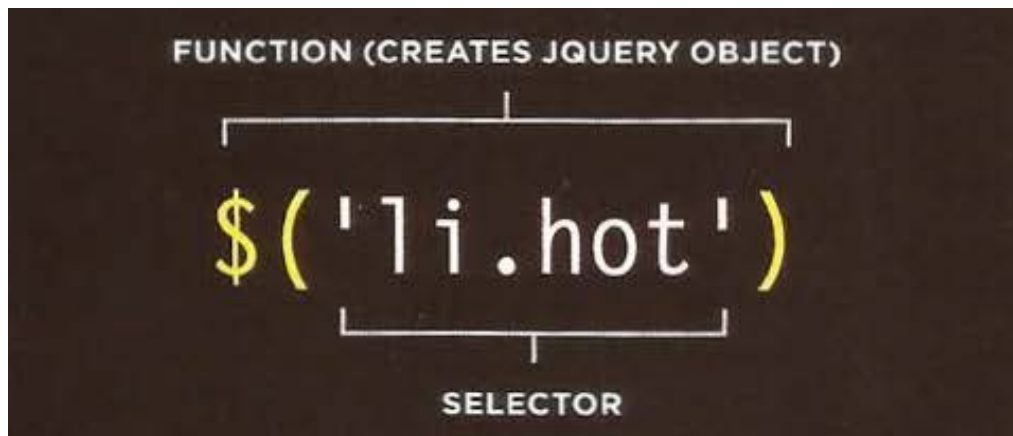
- A JavaScript file that you include in your web pages.
- It lets you:
  - 1 Find elements using [CSS-Style selectors](#), and then
  - 2 Do something with the elements using [jQuery methods](#).
- The jQuery object has many methods that you can use to work with the elements you select.
- The methods represent tasks that commonly need to perform with elements.

# What is a jQuery?

---

## 1 Find elements using CSS-Style selectors

- **jQuery()** function lets you find one or more elements in the page.
- It creates an object called **jQuery** which hold references to those elements.
- **\$()** is often used as shorthand for **jQuery()** as shown here:

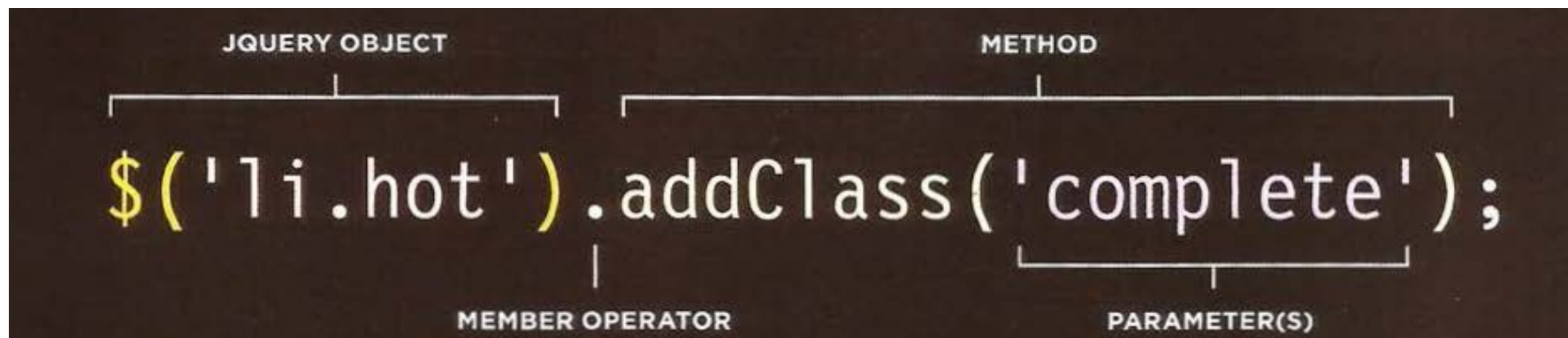


- **jQuery** function has one parameter: a CSS-style selector.
- Selector finds all of the `<li>` elements with a **class of hot**.

# What is a jQuery?

---

- 2 **Do something with the elements** using **jQuery methods**.
- A **jQuery object** is created by the **jQuery() function**.
  - The object and the elements it contains is referred to as a **matched set** or a **jQuery selection**.
  - use the **methods** of the **jQuery object** to update the elements that it contains.



# What is a jQuery?

---

- Similarities to DOM

- **jQuery** selectors perform a similar task to traditional DOM queries, but the syntax is much simpler.
- store the **jQuery** object in a variable, just as you can with DOM nodes.
- use **jQuery** methods and properties (like DOM methods and properties) to manipulate the DOM nodes that you select



# What is a jQuery?

---

- Key Differences from DOM

- It's **cross-browser**, and there is no need to write **fallback code**.
- **Selecting element is simpler**(because it uses CSS-style syntax) and is **more accurate**.
- **Event handling is simpler as it uses one method** that works in all major browsers.
- Methods affect all the selected elements without the need to loop through each one.
- Additional methods are provided for popular required tasks such as animation.
- Once you have made a selection, you can apply multiple methods to it.

# A Basic jQuery Example

---

- This example will use jQuery to update the content of the page.

```
<body>
  <div id="page">
    <h1 id="header">List</h1>
    <h2>Buy groceries</h2>
    <ul>
      <li id="one" class="hot"><em>fresh</em> figs</li>
      <li id="two" class="hot">pine nuts</li>
      <li id="three" class="hot">honey</li>
      <li id="four">balsamic vinegar</li>
    </ul>
  </div>
  ① <script src="js/jquery-1.11.0.js"></script>
  ② <script src="js/basic-example.js"></script>
</body>
```

# A Basic jQuery Example

---

- This example will use jQuery to update the content of the page.

- 1 Include the jQuery script in your page.
- 2 included JavaScript file that uses jQuery selectors and methods to update the content of the HTML page.

```
<body>
  <div id="page">
    <h1 id="header">List</h1>
    <h2>Buy groceries</h2>
    <ul>
      <li id="one" class="hot"><em>fresh</em> figs</li>
      <li id="two" class="hot">pine nuts</li>
      <li id="three" class="hot">honey</li>
      <li id="four">balsamic vinegar</li>
    </ul>
  </div>
  ① <script src="js/jquery-1.11.0.js"></script>
  ② <script src="js/basic-example.js"></script>
</body>
```

# A Basic jQuery Example

---

- This example will use jQuery to update the content of the page.

- ① Selects all of the <h1> - <h6> headings, and adds a value of headline to their class attributes.
- ② Selects the first three list items and does two things:
  - *The elements are hidden (in order to allow the next step).*
  - *The elements fade into view.*
- ③ Set an event listener on each of the <li> elements. When a user clicks on one, it triggers an anonymous function to remove that element from the page.

## JAVASCRIPT

```
① $(':header').addClass('headline');  
② $('li:lt(3)').hide().fadeIn(1500);  
③ {  
    $('li').on('click', function() {  
        $(this).remove();  
    });  
}
```

# A Basic jQuery Example

---

- There are several ways to start using jQuery on your web site. You can:
  - 1 **Download the jQuery library from jQuery.com**
    - *Place the downloaded file in the same directory as the pages where you wish to use it.*
  - 2 **Include jQuery from a CDN, like Google**

# Finding Elements

---

- Using jQuery, you usually select elements using CSS-style selectors.
- It also offers some **extra selectors**, noted below with a 'jQ'.

1

## Basic Selectors

*	All elements
element	All elements with that element name
#id	Elements whose id attribute has the value specified
.class	Elements whose class attribute has the value specified
selector1, selector2	Elements that match more than one selector

# Finding Elements

---

- Using jQuery, you usually select elements using CSS-style selectors.
- It also offers some **extra selectors**, noted below with a 'jQ'.

## 2 Hierarchy

ancestor descendant	An element that is a descendant of another element (e.g., <b>li a</b> )
parent > child	An element that is a direct child of another element ( use * in the place of the child to select all child elements of the specified parent)
previous + next	Adjacent sibling selector only selects elements that are immediately followed by the previous element
previous - siblings	Sibling selector will select any elements that are a sibling of the previous element

# Finding Elements

---

- Using jQuery, you usually select elements using CSS-style selectors.
- It also offers some **extra selectors**, noted below with a 'jQ'.

## 3 Basic Filters

:not(selector)		All elements except the one in the selector (e.g., <b>div: not ('#summary')</b> )
:first	jQ	The first element from the selection
:last	jQ	The last element from the selection
:even	jQ	Elements with an even index number in the selection
:odd	jQ	Elements with an odd index number in the selection
:eq(index)	jQ	Elements with an index number equal to the one in the parameter
:gt(index)	jQ	Elements with an index number greater than the parameter
:lt(index)	jQ	Elements with an index number less than the parameter
:header	jQ	All <b>&lt;h1&gt;</b> - <b>&lt;h6&gt;</b> elements
:animated	jQ	Elements that are currently being animated
:focus		The element that currently has focus



# Finding Elements

---

- Using jQuery, you usually select elements using CSS-style selectors.
- It also offers some [extra selectors](#), noted below with a 'jQ'.

## 4 Content Filters

:contains ('text')		Elements that contain the specified text as a paramete
:empty		All elements that have no children
:parent	jQ	All elements that have a child node (can be text or element)
:has (selector)	jQ	Elements that contain at least one element that matches the selector (e.g., <b>div:has(p)</b> matches all <b>div</b> elements that contain a <b>&lt;p&gt;</b> element)

# Finding Elements

---

- Using jQuery, you usually select elements using CSS-style selectors.
- It also offers some [extra selectors](#), noted below with a 'jQ'.

5

## Visibility Filters

:hidden	jQ	All elements that are hidden
:visible	jQ	All elements that consume space in the layout of the page Not selected <b>if: display: none; height/width:0;</b> ancestor is hidden Selected <b>if: visibility: hidden; opacity : 0</b> because they would take up space in layout

# Finding Elements

---

- Using jQuery, you usually select elements using CSS-style selectors.
- It also offers some **extra selectors**, noted below with a 'jQ'.

## 6 Child Filters

:nth-child (expr)	The value here is not zero-based e.g. <b>ul li:nth-child (2)</b>
:first-child	First child from the current selection
:last-child	Last child from the current selection
:only-child	When there is only one child of the element ( <b>div p:only-child</b> )

# Finding Elements

---

- Using jQuery, you usually select elements using CSS-style selectors.
- It also offers some **extra selectors**, noted below with a 'jQ'.

## 7 Attribute Filters

[attribute]		Elements that carry the specified attribute (with any value)
[attribute='value']		Elements that carry the specified attribute with the specified value
[attribute!='value']	jQ	Elements that carry the specified attribute but not the specified value
[attribute^='value']		The value of the attribute begins with this value
[attribute\$='value']		The value of the attribute ends with this value
[attribute*='value']		The value should appear somewhere in the attribute value
[attribute ='value']		Equal to given string, or starting with string and followed by a hyphen
[attribute~='value']		The value should be one of the values in a space separated list
[attribute] [attribute2]		Elements that match all of the selectors

# Finding Elements

- Using jQuery, you usually select elements using CSS-style selectors.
- It also offers some **extra selectors**, noted below with a 'jQ'.

8

## Form

:input	jQ	All input elements
:text	jQ	All text inputs
:password	jQ	All password inputs
:radio	jQ	All radio buttons
:checkbox	jQ	All checkboxes
:submit	jQ	All submit buttons
:image	jQ	All <b>&lt;img&gt;</b> elements
:reset	jQ	All reset buttons
:button	jQ	All <b>&lt;button&gt;</b> elements
:file	jQ	All file inputs
:selected	jQ	All selected items from drop-down lists
:enabled		All enabled form elements (the default for all form elements)
:disabled		All disabled form elements (using the CSS disabled property)
:checked		All checked radio buttons or checkboxes

## Doing things with your Selection

---

- Once you have selected the elements you want to work with (and they are in a jQuery object), the **jQuery methods** perform tasks on those elements.
- The jQuery objects created after the selection has a property called **length**, which will return the number of elements in the object.
- If the jQuery selection did not find any matching elements, you will not get an error by calling any of these methods - they just won't do or return anything.
- The jQuery method names written starting with a period (.) before the name.
  - *jQuery methods are easily distinguishable from built-in JavaScript methods, or methods of custom objects.*

# Doing things with your Selection

---

## 1 **Content Filters** - Get or change content of elements, attributes, text nodes

GET/ CHANGE CONTENT	<code>.html()</code> <code>.text()</code> <code>.replaceWith()</code> <code>.remove()</code>
ELEMENTS	<code>.before()</code> , <code>.after()</code> , <code>.prepend()</code> , <code>.append()</code> , <code>.remove()</code> , <code>.clone()</code> , <code>.unwrap()</code> , <code>.detach()</code> , <code>.empty()</code> , <code>.add()</code>
ATTRIBUTES	<code>.attr()</code> , <code>.removeAttr()</code> , <code>.addClass()</code> , <code>.removeClass()</code> , <code>.css()</code>
FORM VALUES	<code>.val()</code> <code>.isNumeric</code>

# Doing things with your Selection

---

## 2 Finding Elements - Find and select elements to work with & traverse the DOM

GENERAL	<code>.find()</code> , <code>.closest()</code> , <code>.parent()</code> , <code>.parents()</code> , <code>.children()</code> , <code>.siblings()</code> , <code>.next()</code> , <code>.nextAll()</code> , <code>.prev()</code> , <code>.prevAll()</code>
FILTER/ TEST	<code>.filter()</code> <code>.not()</code> <code>.has()</code> <code>.is()</code> <code>:contains()</code>
ORDER IN SELECTION	<code>.eq()</code> <code>.lt()</code> <code>.gt()</code>



# Doing things with your Selection

---

- 3 **Dimension/Position** - Get or update the dimensions or position of a box

DIMENSION	<code>.height(), .width(), .innerHeight(), .innerWidth() .outerHeight(), .outerWidth() \$(document).height() \$(document).width() \$(window).height() \$(window).width()</code>
POSITION	<code>.offset() .position() .scrollLeft() .scrollTop()</code>

# Doing things with your Selection

---

## 4 **Effects & Animation** - Add effects and animation to parts of the page

BASIC	<code>.show()</code> <code>.hide()</code> <code>.toggle()</code>
FADING	<code>.fadeIn()</code> <code>.fadeOut()</code> <code>.fadeTo()</code> <code>.fadeToggle()</code>
SLIDING	<code>.slideDown()</code> <code>.slideUp()</code> <code>.slideToggle()</code>
CUSTOM	<code>.delay()</code> , <code>.stop()</code> , <code>.animate()</code>

# Doing things with your Selection

---

- 5 **Events** - Create event listeners for each element in the selection

DOCUMENT/ FILE	<code>.ready()</code> <code>.load()</code>
USER INTERACTION	<code>.on()</code>

**NOTE:** Earlier there were methods for individual types of event, such as `.click()` , `.hover()` , `.submit()` , However, these have been dropped in favour of the `.on()` method to handle events

# References

---

- 1 JavaScript and JQuery – Interactive Front-end Web Development, (Jon Duckett), John Wiley and Sons, Inc.
- 2 <https://developer.mozilla.org/en-US/docs/Web/JavaScript>