



By Vijay Shivakumar



By Vijay Shivakumar

Software required

IDE : Aptana Studio / Visual Studio / Eclipse / Netbeans

Web Server : Tomcat 5.5 or IIS or WAMP

Browsers : IE , Firefox with Firebug / Chrome,
Optional : Safari Opera Etc...



By Vijay Shivakumar

About Me

Who am I

Why am I here

Vijay Shivakumar
Designer | Developer | Trainer



CERTIFIED EXPERT
Flex® with AIR

Training on web technologies and Adobe products from past 11 years



By Vijay Shivakumar

About You

Who you are

Why you are here



By Vijay Shivakumar

About jQuery

- Created by Jhon Resig (<http://ejohn.org>)
- Started in 2005
- Uses CSS3 selectors
- Lightweight
- Compact
- 1000+ plug-ins available till date



By Vijay Shivakumar

What is jQuery to you

- Free
- Open source
- Write less do more
- Simplifies the task of creating complex / interactive websites
- Works across browsers
- Abstracts browser specific features

Contd..



By Vijay Shivakumar

About jQuery

- Allows developers to concentrate on desired output
- Manages page content



By Vijay Shivakumar

Compatibility

Works with modern browser model



6.0 or Greater
Issues with 1-5



3.0 or Greater
Issues with 1-2.1



9.0 or Greater
Issues with 1-8



2.0 or Greater
Issues with 1.0.X



1.0 or Greater
N/A



By Vijay Shivakumar

Common Tasks

Page load

- Performs a bunch of tasks on page load on each element

Select

- Dig for the desired element and cycle through it

Event

1. Retrieve page content
2. Manipulates
3. Puts content back to page



By Vijay Shivakumar

Download and Install



Download jQuery
v1.9.1 or v2.0.0



By Vijay Shivakumar

Download From CDN

Google Ajax API CDN

<https://ajax.googleapis.com/ajax/libs/jquery/1.5.1/jquery.min.js>

Microsoft CDN

<http://ajax.aspnetcdn.com/ajax/jQuery/jquery-1.5.1.min.js>

jQuery CDN (via Media Temple)

<http://code.jquery.com/jquery-1.5.1.min.js> Minified version

<http://code.jquery.com/jquery-1.5.1.js> Source version



By Vijay Shivakumar

CDN advantage

- Script Files Caching
- Supports both http and https
- Regional servers delay response latency
- Browser parallelism



By Vijay Shivakumar

Lets start using jQuery

- Include library
- Listen for events
- Execute some task upon event



By Vijay Shivakumar

jQuery Features

Core Functionality

- Common jQuery functions and utilities

Selection and Transversal

- Provides functions to find contents, elements

Manipulation of CSS

- Has functionalities to adding, removing and editing CSS

Events

- Simplify handling DOM events

Contd..



By Vijay Shivakumar

jQuery Features

Effects

- Animation such as hiding, showing, moving etc,

Ajax

- Loading content from pages and working with JSON

UI

- Commonly used widgets

Extensibility

- Adds to existing library with more functionalities



By Vijay Shivakumar

Selectors and Filters

jQuery selectors will return an array of objects that match your criteria.

Filters will further refine the returned array

The Array returned is not an array of DOM elements, but it is an collection of jQuery objects wrapped around it.



By Vijay Shivakumar

Basic jQuery Selectors

jQuery selectors work the same way like CSS selectors do.

SELECTOR	PURPOSE
tagName	Finds all elements that are of tagName
#identifier	Finds element with that ID
.className	Finds all elements with class sent to .className
tag#id.class	Gets elements the tag with #id with .class
*	Gets all elements



By Vijay Shivakumar

Compare DOM vs. jQuery

```
<html>
  <head>
    <title>DOM V/S jQuery</title>
  </head>
  <body>
    <ul id="list1">
      <li class="a">item1</li>
      <li class="b">item2</li>
      <li class="c">item3</li>
      <li class="d">item4</li>
    </ul>
    <p>Paragraph 1</p>
    <p>Paragraph 2</p>
  </body>
</html>
```

Get all <p>tags

```
document.getElementsByTagName("p");
$("p");
```

Get all tags with id "list1"

```
document.getElementById("list1");
$("#list1");
```

Get all tags with class "a"

```
Hmmm.. Pretty complex
$("li.a");
```



By Vijay Shivakumar

Hierarchical selectors

SELECTOR	PURPOSE
selector, selector	To select more than one type of element
.class.class	Elements with both classes assigned
parent > child	Finds all direct children of parent
ancestor descendent	All decedents of ancestor
tag1 + tag2	Finds the first tag2 sibling of tag1
tag1 ~ tag2	Finds all tag2 siblings of tag1



By Vijay Shivakumar

Selectors | More

SELECTOR	PURPOSE
.children()	Fetch the direct children of the selector
.find("selector")	Fetch all children of selector



By Vijay Shivakumar

Filters



By Vijay Shivakumar

jQuery Filters

Basic	Filters like first last even odd etc...
Content	Filters on content if it contain certain string
Visibility	Filters on elements based on their visibility settings
Attribute	Filter on elements with certain attributes
Child	Filters on elements on parent child relationship
Form	Specialized filters that work on form elements



Basic Filters

By Vijay Shivakumar

:first	Selects the first element of returned set
:last	Selects the last element of returned set
:even	Selects all even elements of returned set
:odd	Selects all odd elements of returned set
:eq(n)	Selects if the index given by you
:gt(n)	Selects all the index more than provided by you
:lt(n)	Selects index less than provided
:header	Selects all header elements like h1 h2 h3 etc
:animated	Selects all elements that are animated in some way
:not(selector)	Nagation tag – selects all other than provided



By Vijay Shivakumar

Content Filters

<code>:contains(text)</code>	Select all elements containing specified text.
<code>:empty</code>	Select all elements that have no children That has no nodes or elements
<code>:has(selector)</code>	Selects elements which contain the specified selector.
<code>:parent</code>	All elements that has a child node or element (if the element is a parent)



By Vijay Shivakumar

Visibility Filters

:hidden	Selects the hidden elements
:visible	Selects the visible elements



By Vijay Shivakumar

Attribute Filters

[attribute]	Elements with specified attribute
[attribute = value]	Specified attribute with a value
[attribute != value]	Elements that do not have attribute value set by us
[attribute ^= value]	Elements that have attribute value specified in the start
[attribute \$= value]	Elements that have attribute value specified in the end
[attribute *= value]	Elements that have attribute value specified anywhere
[attFilter1] [attfilter2]	Elements that match all the attribute filters specified by us should be



By Vijay Shivakumar

Keep this image in mind





By Vijay Shivakumar

Child Filters

:first-child	Selects the first child of their parent
:last-child	Selects the last child of their parent
:only-child	Selects the only child of their parent
:nth-child(child num)	Selects the child at the given child count
:nth-child(even)	Selects even children
:nth-child(odd)	Selects odd children
:nth-child(equation)	Equations can be like $2n$, $3n$, etc..



By Vijay Shivakumar

Form Filters

:input	Selects all input elements
:text	Selects all text input elements
:button	Selects all buttons
:checked	Selects is a checkbox or radio is checked
:radio	Selects all radio buttons
:submit	Finds and selects all submit elements
:image	Selects all image elements
:password	Selects all password elements
:checkbox	Selects all checkboxes



By Vijay Shivakumar

Traversing



By Vijay Shivakumar

Traversing

<code>size()</code>	Returns the count/number of matched elements
<code>get()</code>	Gets all matching objects as DOM elements
<code>get(index)</code>	Gets a single DOM element with specified index from the matching objects index can be -1 etc
<code>find(expression)</code>	Searches for descendants that match expression
<code>each(fn)</code>	Executes a function for every result element
<code>children()</code>	Gets all direct matching children of the selector
<code>filter()</code>	Returns all the matching elements



By Vijay Shivakumar

Traversing Advanced



By Vijay Shivakumar

Traversing More...

<code>first()</code>	Will select the first matching selector
<code>last()</code>	Will select the last matching selector
<code>eq()</code>	Will select the matching selector in the index
<code>parent() , parents()</code>	Will select the parent or parents in the same hierarchy
<code>closest()</code>	Will find the first matching parents



By Vijay Shivakumar

Chaining

next()	Gets the next element of the selection
prev()	Gets the previous element of the selection
nextAll()	Gets all the next elements of the selection
prevAll()	Gets all the previous elements of the selection
end()	Returns to initial selection
nextUntil()	
prevUntil()	



By Vijay Shivakumar

Statement Chaining

```
$(selector).fn1().fn2().fn3();
```



By Vijay Shivakumar

Inserting contents

<code>append(content)</code>	Adds the content at the end of selector
<code>appendTo(selector)</code>	Adds content from one selector to the end of another selector
<code>prepend(content)</code>	Adds content at the beginning of selector
<code>prependTo(selector)</code>	Adds content from one selector to the beginning of another selector
<code>after(content)</code>	Adds content after the selected element
<code>before(content)</code>	Adds content before the selected element
<code>insertAfter(selector)</code>	Will insert the element after the selection
<code>insertBefore(selector)</code>	Will insert the element before the selection



By Vijay Shivakumar

Wrap - Remove

<code>wrap(html)</code>	Wraps each matching element with specified html
<code>wrap(selector)</code>	Wraps each matching element with specified selector
<code>wrapAll(html)</code>	Wraps all matching element with specified html
<code>wrapAll(selector)</code>	Wraps all matching element with specified selector
<code>wrapInner(html)</code>	Wraps the inner tags with specified html
<code>wrapInner(selector)</code>	Wraps the inner tags with specified selector
<code>empty()</code>	Empties the selection
<code>remove()</code>	Removes the selection
<code>unwrap()</code>	Will remove its current parent
<code>detach()</code>	Will remove the elements but not its events



By Vijay Shivakumar

Create, Edit and Update

Create html by:-

```
var newHeader = $("<h1>My New Header</h1>");
```

Or by:-

```
var myString = "<h1>My New Header</h1>";
```

```
var myHeader = $(myString);
```

Or by :

```
 $("<div/>",{class:'boxer', click:function(){  
    alert("hello from new element");  
}})
```



By Vijay Shivakumar

Create, Edit and Update

html()	Gives the content of first matched element
html(content)	Sets the content of every matched html element
text()	Gives the text content of first matched element
text(content)	Sets the text to all matched elements
val()	Gets the value of a form element
val(new value)	Sets the value of a form element



By Vijay Shivakumar

Modify attributes

<code>attr(name)</code>	Gets the value of first matched attribute, if the attribute is not existing it returns undefined
<code>attr(key,value)</code>	Sets a single attribute on matching element
<code>attr(properties)</code>	You can use an object notation syntax to set a large number or attributes. <code>\$("#img").attr({src:"images/photo1.jpg", alt:"its me"})</code>
<code>attr(key,fn)</code>	Sets attribute to a computed value
<code>removeAttr(name)</code>	Removes specified attributes to all matches



By Vijay Shivakumar

CSS in jQuery



CSS in jQuery

By Vijay Shivakumar

<code>css(name)</code>	Returns the value of the first matched selector
<code>css(property,value)</code>	A single value that can be assigned to the selector
<code>css(properties)</code>	Sets the properties to the selector Eg: <code>var cssObj = { 'background-color' : '#cecece', 'color' : '#ddd' }; (this).css(cssObj);</code>
<code>addClass(class)</code>	Will add a class to the selection
<code>hasClass(class)</code>	Returns a boolean value if the class is existing
<code>removeClass(class)</code>	Removes the class if existing
<code>toggleClass(class)</code>	If the class exists it removes else adds
<code>toggleClass(class,switch)</code>	Use a switch if returns true adds class else removes



By Vijay Shivakumar

CSS positions in jQuery

<code>offset()</code>	Returns offset of the element relative to document
<code>position()</code>	Gets top and left position of the element
<code>scrollTop()</code>	Gets the scroll top value of the element
<code>scrollLeft()</code>	Gets the scroll left value of the element
<code>height()</code>	Returns height of an element
<code>height(val)</code>	Sets the height value to specified
<code>width()</code>	Returns width of an element
<code>width(val)</code>	Sets the width value to specified



By Vijay Shivakumar

Events in jQuery



By Vijay Shivakumar

Working with events

Few Mouse Events

<code>.click()</code>	When you click on an object
<code>.dblclick()</code>	When you double click on an object
<code>.mouseover()</code>	When the mouse cursor is over an object
<code>.hover()</code>	Similar to mouse over but can take 2 functions
<code>.mousemove()</code>	When the mouse cursor is moving
<code>.mouseup()</code>	When the mouse is released
<code>.mousedown()</code>	When the mouse is pressed on the target object



By Vijay Shivakumar

Working with events

Few Document Events

`.ready()`

Once the DOM is ready even before the elements of page are not loaded

`.load()`

When all elements are completely loaded

`.unload()`

When the user navigates away from the document

`.error()`

When any elements are not loaded



By Vijay Shivakumar

Working with events

Few Event Handlers

`.bind()`

Binds an event to an element

`.unbind()`

Removes event from an element

`.one()`

Binds event and allows event to happen only once,

`.trigger()`

Will simulate the event as if user created it . bubbles

`.triggerHandler()`

Will simulate the event but wont bubble to DOM
If target doesn't handle the event the event does nothing



By Vijay Shivakumar

Working with events

New Event Handlers

`on()`

Adds an event listener

`off()`

Removes the event from an element



By Vijay Shivakumar

jQuery | Ajax



By Vijay Shivakumar

jQuery | Ajax features

Allows parts of the page to be updated

Cross browser support

Simple API

Supports both GET and POST

Loads HTML, XML, Text, JSON and even JavaScripts



By Vijay Shivakumar

jQuery Ajax tool box

`$(selector).load()`; loads data from server to selector

`$.get()` and `$.post()`; gets raw data from server

`$.getJSON()`; uses GET and returns JSON

`$.getScript()`; uses GET and returns javascript

`$.ajax()`; provides core functionality

All the above can consume REST based web services, XML and more



Lets \$.load some HTML

By Vijay Shivakumar

```
$(selector).load(url, data, callback);
```

Objective is to load html content from server in to the selector

```
$(function(){  
    $("#btn").click(function(){  
        $("#myDiv").load('details.html');  
    });  
});
```



By Vijay Shivakumar

No.. I need a part of html

If you need only a part of the html then..

```
$(selector).load(url, data, callback);
```

Will download the entire page and filters the data you need



By Vijay Shivakumar

Sending data to server

```
$(selector).load(url, data, callback);
```

```
$("myDiv").load("details.php", {pageSize:10});
```



Send the data as an object



By Vijay Shivakumar

Adding a callback fn

```
$(selector).load(url, data, callback);
```

```
$("#myDiv").load('notfound.html',  
    function(response,status,xhr){  
        if(status == "error"){  
            alert(xhr.statusText)  
        }  
    });  
});
```

Response : Data coming back
Status: can be error or success
xhr : is XMLHttpRequest object



By Vijay Shivakumar

Using \$.get();

```
$.get(url,data,callback,datatype);
```

The \$.get, \$.post, \$.getJSON \$.ajax are global functions

```
$.get("details.htm", function(data){  
    $("#myDiv").html(data);  
});
```




By Vijay Shivakumar

Using \$.post();

```
$.post(url,data,callback,datatype);
```

```
$.post("details.htm", function(data){  
    $("#myDiv").html(data);  
});
```



By Vijay Shivakumar

Using \$.getJSON();

```
$.getJSON(url,{data},callback)
```

```
$.getJSON("names.json",function(data){  
    alert(data.fName);  
});
```



Using \$.ajax()

By Vijay Shivakumar

```
$.ajax({  
  url:'names.json',  
  data:{"user":"vijay","pass":"password"},  
  dataType:"json",  
  success: function(data,status,xhr){  
    alert(data.fName);  
    timeout:8000,  
    cache:false,  
    complete:function  
  },  
  error:function(xhr, status, error){  
    alert('Error :' status);  
  }  
})
```



By Vijay Shivakumar

Utilities | Type testing

`$.isArray(array)` tells us if what we passed is an array

`$.isFunction(function)` tells us if what we passed is a function

`$.isEmptyObject(object)` tells us if what we passed is an empty object

`$.isPlainObject(object)` tells us if what we passed is plain object

`$.isXmlDoc(xml)` tells us if what we passed is an xml doc or node

`$.isNumeric(number)` tells us if what we passed is a number

`$.isWindow(window)` tells us if what we passed is a window

`$.type(obj)` will tell if boolean, string, number, function, regex, object, undefined, null, array or date was passed



By Vijay Shivakumar

jQuery | Effects



By Vijay Shivakumar

Animations | Hide/Show

<code>show()</code>	Reveals the selector if hidden
<code>show(speed,callback)</code>	Reveals the selector with specified speed and call a fn
<code>hide()</code>	Hides the selector if shown
<code>hide(speed,callback)</code>	Hides the selector with specified speed and call a fn
<code>toggle()</code>	Toggles between hide and show
<code>toggle(switch)</code>	Toggles with a computed logic
<code>toggle(speed,callback)</code>	Toggles with speed and call a function



By Vijay Shivakumar

Animations | Fade

<code>fadeIn(speed, callback)</code>	Reveals the selector if hidden
<code>fadeOut(speed, callback)</code>	Hides the selector with specified speed and call a function
<code>fadeTo(speed, opacity, callback)</code>	Hides the selector to specific opacity (between 0 and 1) and calls a function



By Vijay Shivakumar

Animations | Slide

<code>slideUp(speed, callback)</code>	Slides up the selector to hide
<code>slideDown(speed, callback)</code>	Slides down the selector to show
<code>slideToggle(speed, callback)</code>	Slides up if shown and slides down if hidden



By Vijay Shivakumar

Animations | Custom

<code>animate(param,duration,easing, callback)</code>	Param: properties to animate Duration : in milliseconds Easing: type of easing (linear, swing) Callback : a function set to call after anim
<code>animate(param,options)</code>	Param: properties to animate Options: set of options for animation to occur
<code>anim.stop()</code>	Stops all animation happening on page with that name



By Vijay Shivakumar

tricks

<http://james.padolsey.com/jquery/>



By Vijay Shivakumar

Thank you !

vijay.shivu@gmail.com