



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...



Who am I Why am I here

> Vijay Shivakumar Designer | Developer | Trainer Adobe



Training on web technologies and Adobe products from past 11 years



Who you are Why you are here



About jQuery

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



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



About jQuery

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



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



Common Tasks

By Vijay Shivakumar

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



Download and Install





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



CDN advantage

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



Lets start using jQuery

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



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..



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



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.



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 |



Compare DOM vs. jQuery

```
<html>
 <head>
  <title>DOM V/S jQuery</title>
 </head>
<body>
ul id="list1">
    item1
    item2
    item3
    item4
Paragraph 1
Paragraph 2
</body>
</html>
```

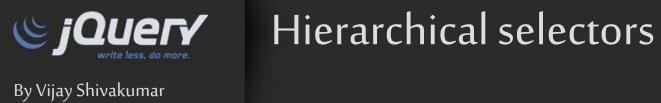
```
Get all tags
document.getElementsByTagName("p");
$("p");
```

```
Get all tags with id "list1" document.getElementByld("list1"); $("#list1");
```

```
Get all tags with class "a"

Hmmm.. Pretty complex

$("li.a");
```



| 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 |
| | |



Selectors | More

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



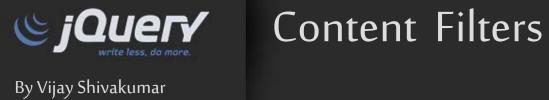
Filters



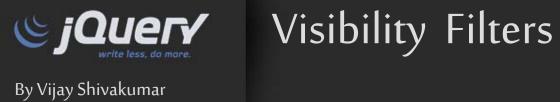
| Basic | Filters like first last even odd etc |
|------------|--|
| Content | Filters on content if it contain certain string |
| Visibility | Filters on elements based on their visibility settings |
| Attribute | Filter on elements with certain attributes |
| Child | Filters on elements on parent child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



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



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



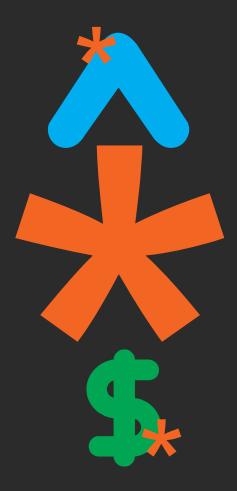
| :hidden | Selects the hidden elements |
|----------|------------------------------|
| :visible | Selects the visible elements |



| [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 |



Wite less, do more. Keep this image in mind





| :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 |



| :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 |



Traversing



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



Traversing Advanced





Traversing More...

| first() | Will select the first matching selector |
|----------------------|---|
| last() | Will select the last matching selector |
| eq() | Will select the matching selector in the index |
| parent() , parents() | Will select the parent or parents in the same hierarchy |
| closest() | Will find the first matching parents |
| | |



| 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 |
| prevAII() | Gets all the previous elements of the selection |
| end() | Returns to initial selection |
| nextUntil() | |
| prevUntil() | |
| | |
| | |



Statement Chaining

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



Inserting contents

By Vijay Shivakumar

append(content)

appendTo(selector)

prepend(content)

prependTo(selector)

after(content)

before(content)

insertAfter(selector)

insertBefore(selector)

Adds the content at the end of selector

Adds content from one selector to the end of another selector

Adds content at the beginning of selector

Adds content from one selector to the beginning of another selector

Adds content after the selected element

Adds content before the selected element

Will insert the element after the selection

Will insert the element before the selection



Wrap - Remove

wrap(html)

By Vijay Shivakumar

wrap(selector)

wrapAll(html)

wrapAll(selector)

wrapInner(html)

wrapInner(selector)

empty()

remove()

unwrap()

detach()

Wraps each matching element with specified html

Wraps each matching element with specified selector

Wraps all matching element with specified html

Wraps all matching element with specified selector

Wraps the inner tags with specified html

Wraps the inner tags with specified selector

Empties the selection

Removes the selection

Will remove its current parent

Will remove the elements but not its events



Create, Edit and Update

```
Create html by:-
var newHeadder = $("<h1>My New Headder</h1>");
Or by:-
var myString = "<h1>My New Headder</h1>";
var myHeader = $(myString);

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



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 |



Modify attributes

attr(name)

By Vijay Shivakumar

attr(key,value)

attr(properties)

attr(key,fn)

removeAttr(name)

Gets the value of first matched attribute, if the attribute is not existing it returns undefined

Sets a single attribute on matching element

You can use an object notation syntax to set a large number or attributes.

\$("img").attr({src:"images/photo1.jpg", alt:"its me"})

Sets attribute to a computed value

Removes specified attributes to all matches



By Vijay Shivakumar

CSS in jQuery



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



CSS positions in jQuery

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



By Vijay Shivakumar

Events in jQuery



By Vijay Shivakumar

Few Mouse Events

.click()

.dblclick()

.mouseover()

.hover()

.mousemove()

.mouseup()

.mousedown()

When you click on an object

When you double click on an object

When the mouse cursor is over an object

Similar to mouse over but can take 2 functions

When the mouse cursor is moving

When the mouse is released

When the mouse is pressed on the target object



By Vijay Shivakumar

Few Document Events

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

.load()
.unload()
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

Few Event Handlers

.bind()

.unbind()

.one()

.trigger()

.triggerHandler()

Binds an event to an element

Removes event from an element

Binds event and allows event to happen only once,

Will simulate the event as if user created it . bubbles

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



By Vijay Shivakumar

New Event Handlers

| on() | Adds an event listener |
|-------|-----------------------------------|
| off() | Removes the event from an element |
| | |
| | |
| | |
| | |
| | |



jQuery | Ajax



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



jQuery Ajax tool box

By Vijay Shivakumar

\$(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
 $(selectror).load(url, data, callback);
 Objective is to load html content from server in to the
   selector
 $(function(){
    $("#btn").click(function(){
        $("#myDiv").load('details.html');
     });
```



No.. I need a part of html

If you need only a part of the html then.. \$(selectror).load(url, data, callback);

Will download the entire page and filters the data you need



Sending data to server

\$(selectror).load(url, data, callback);

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



Send the data as an object



Adding a callback fn



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);
});
```



Using \$.getJSON();

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

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



Using \$.ajax()

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



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(number) 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



jQuery | Effects



Animations | Hide/Show

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



| fadeln(speed, callback) | Reveals the selector if hidden |
|--------------------------|---|
| fadeOut(speed, callback) | Hides the selector with specified speed and call a function |
| | Hides the selector to specific opacity (between 0 and 1) and calls a function |



| slideUp(speed, callback) | Slides up the selector to hide |
|------------------------------|--|
| slideDown(speed, callback) | Slides down the selector to show |
| slideToggle(speed, callback) | Slides up if shown and slides down if hidden |



Animations Custom

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



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



Thank you!

vijay.shivu@gmail.com