

# LEARN. DO. EARN

ACADGILD



## FRONT END DEVELOPMENT (WITH ANGULARJS)



Website : <http://www.acadgild.com>  
LinkedIn : <https://www.linkedin.com/company/acadgild>  
Facebook : <https://www.facebook.com/acadgild>

© copyright ACADGILD



# Session 5 – jQuery





# Agenda – jQuery

Sl No	Agenda Title
1	What is jQuery?
2	Setting up jQuery
3	jQuery Syntax & Selector
4	jQuery Example
5	jQuery hide, show, toggle
6	Filtering Elements
7	Attribute Selector
8	jQuery Traversing
9	jQuery Traversing Parent
10	jQuery Traversing Child
11	jQuery Traversing Sibling
12	jQuery DOM Manipulation
13	jQuery DOM Manipulation - Example





# What is jQuery?

- **jQuery** is a lightweight JavaScript library.
- **jQuery** is the most popular, and also the most extendable library build on JavaScript
- **Purpose of jQuery** is to make it much easier to use JavaScript on your website
- The **jQuery** library contains the following features:
  - HTML/DOM manipulation
  - CSS manipulation
  - HTML event methods
  - Effects and animations
  - AJAX
  - Utilities





# Setting up jQuery

- The best way to include jQuery is to include it from a CDN (Content Distribution Network) like Google CDN or Microsoft CDN as shown below:
- Here we have included it from Google CDN

```
<head>  
<script  
src="http://ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js">  
</script>  
</head>
```

- **Note:** In case you are not connected to internet, you can download jQuery and keep it on your local server.
- Go to <http://jquery.com/download/> and download the jQuery 2.x uncompressed development version.





# jQuery Syntax & Selector

- The jQuery syntax is tailor made for selecting HTML elements and performing some action on the element(s).
- Basic syntax is: **`$(selector).action`**
  - A \$ sign to define/access jQuery
  - A (selector) to "query (or find)" HTML elements
  - A jQuery action() to be performed on the element(s)

## Examples:

- `$(this).hide()` - hides the current element.
- `$("p").hide()` - hides all <p> elements.
- `$(".test").hide()` - hides all elements with class="test".
- `$("#test").hide()` - hides the element with id="test".





# jQuery Example

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

<body>
  <div id = "message-box"><Hello, Welcome to Acad Gild!<div>
  <p class = "para">Welcome to the section</p>
  <script>
    var ref = $("#message-box");
    ref_class = $(".para");
    var ref_tag = $("h1");
    alert(msg);
  </script>
</body>
```





# jQuery hide, show, toggle

- With jQuery, you can hide and show HTML elements
  - `hide()` - to hide an element
  - `Show()` – to show an element

## Syntax:

- `$(selector).hide(speed,callback);`
- `$(selector).show(speed,callback);`

## jQuery toggle()

- With jQuery, you can toggle between the `hide()` and `show()` methods with the `toggle()` method.

## Syntax:

- `$(selector).toggle(speed,callback);`







# Filtering Elements

- The three most basic filtering methods are **first()**, **last()** and **eq()** which allow you to select a specific element based on its position in a group of elements.
- Other filtering methods, like **filter()** and **not()** allow you to select elements that match or do not match a certain criteria.

## Example:

```
$(document).ready(function()  
{  
  $("div p").first().css("background-color", "yellow");  
});
```





# Attribute Selector

- The [attribute] selector selects each element with the specified attribute.

**`$("[attribute]")`**

- To select all the tags with value as `http://google.com` we can write

`$("a[href='http://google.com']")`

- To select all link tags that have url value beginning with Google we use

`$("a[href^='google' ]")`

- To select all link tags that have end url value as com we use

`$("a[href$='google' ]")`

- To Select all tags link tags that have href attribute value use

`$("a[href ]")`





- **Parents**

- **Children**

- **Siblings**

There are also a few other methods that build onto these basic methods: `.nextAll()`, `.nextUntil()`, `.prevAll()` and `.prevUntil()`.





# jQuery Traversing Parent

// Selecting an element's direct parent:

// returns [ div.child ]

`$( "span.subchild" ).parent();`

```
<div class="grandparent">
  <div class="parent">
    <div class="child">
      <span class="subchild"></span>
    </div>
  </div>
  <div class="surrogateParent1"></div>
  <div class="surrogateParent2"></div>
</div>
```





# jQuery Traversing Child

// Selecting an element's direct child:

// returns [ div.parent, div.surrogateParent1, div.surrogateParent2 ]

`$( "div.grandparent" ).children( "div" );`

```
<div class="grandparent">
  <div class="parent">
    <div class="child">
      <span class="subchild"></span>
    </div>
  </div>
  <div class="surrogateParent1"></div>
  <div class="surrogateParent2"></div>
</div>
```





# jQuery Traversing Sibling

// Selecting a next sibling of the selectors:

// returns [ div.surrogateParent1 ]

— \$( "div.parent" ).next();

```
<div class="grandparent">
  <div class="parent">
    <div class="child">
      <span class="subchild"></span>
    </div>
  </div>
  <div class="surrogateParent1"></div>
  <div class="surrogateParent2"></div>
</div>
```





# jQuery DOM Manipulation

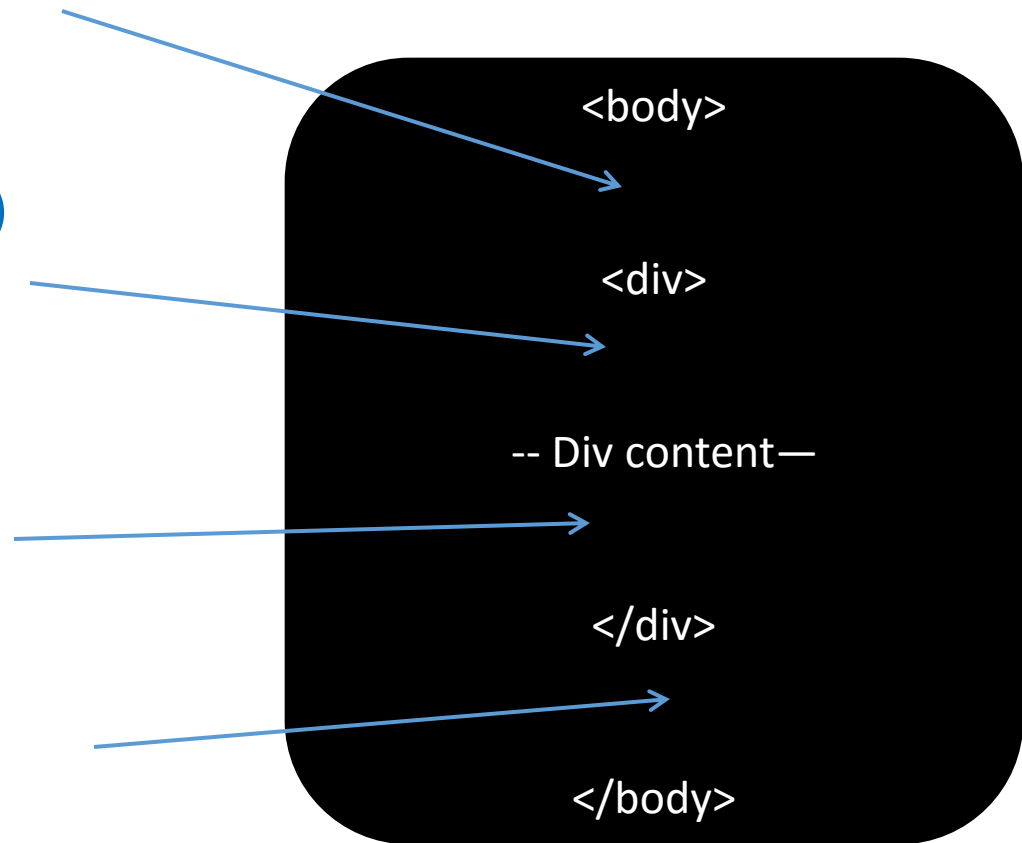
- There are many ways to change an existing element.
- Among the most common tasks is changing the inner HTML or attribute of an element.
- jQuery offers simple, cross-browser methods for these sorts of manipulations.
- They are:
  - **append**: Appends after HTML to the selected element.
  - **prepend**: Prepends before HTML to the selected element.
  - **after**: Adds HTML after the element.
  - **before**: Adds HTML before the element.
  - **html**: Replaces an existing HTML inside an element and adds the provided HTML.
  - **remove**: Removes an element from the DOM.
  - **addClass**: Adds CSS class to an element
  - **removeClass**: removes CSS class from an element
  - **toggleClass**: Toggles between adding/removing classes
  - **Css**: Sets or returns the style attribute





# jQuery DOM Manipulation Example

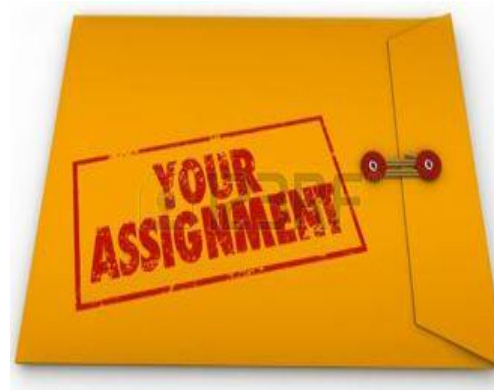
- `$("#target").before("add data")`
  - adds content before the div
- `$("#target").append("add data")`
  - adds before the div content
- `$("#target").prepend("add data")`
  - adds after the div content
- `$("#target").after("add data")`
  - adds content here after the div







# Lets Discuss Assignments



**Assignment**





# Get in Touch with ACADGILD

## Contact Info:

- Website : <http://www.acadgild.com>
- LinkedIn : <https://www.linkedin.com/company/acadgild>
- Facebook : <https://www.facebook.com/acadgild>
- Support: [support@acadgild.com](mailto:support@acadgild.com)

