Introduction To jQuery

- John Teague
  - Co-Founder, Principal Consultant, Next Level Technology
    - Custom Software Development
    - Agile Coaching & Training
    - IT Guidance

http://johnteague.lostechies.com

Before We Get Started

- Using a Different Library?
  - – Let's not get into a Library war. If you are using a different library and you are happy with it, Good!!!
- Have Your own Custom Library
  - – Just Stop!! Pick an existing library and stick with it. If it doesn't do something exactly the way you want, customize and contribute to the comummunity.

jQuery

- JavaScript library to help perform common and / or difficult client-side tasks
- Advantages
  - Cross Browser
  - Light-weight
  - Easy DOM selection & manipulation
  - Pluggable architecture keeps file size down and easily extensible

Features to Cover

- DOM Element Selectors
- Event Handling
- HTML Manipulation and Traversal
- AJAX

Unobtrusive JavaScript

- Progressive Enhancement
  - Build site for least functionality, progressively add style and functionality
- Separation of structure, style and behavior
- HTML, CSS, and JavaScript is separate files

Unobtrusive JavaScript

- No in-line event handlers
- All code is contained in external .js files
- The site remains usable without JavaScript
- Existing links and forms are repurposed
- JavaScript dependent elements are dynamically added to the page

Obtrusive JavaScript

- Have you read our

```
<a href="javascript:window.open(
  'terms.html',
  'popup‘,
  'height=500,width=400,toolbar=no’);">
 terms and conditions</a>
```

Converted to Unobtrusive

- HTML
  - Have you read our<a href="terms.html” **class="sidenote”**>terms and conditions</a>
- CSS
  - CSS class sidenote
- JavaScript
  - Event Handler for the click event in a separate

Functions

- Named

```
function doSomething(arg1,arg2){
   //code goes here
}
```

- Inline or Anonymous

```
function(){//code goes here}
```

Callbacks

- A Callback is a parameter to a function that points to another function
  - It can be a named function or an anonymous function
  - Example, click handler:

```
$(‘a’).click(function(){//click logic goes
  here})
$(‘a’).click(clickFunction);
```

Objects

- Named

  ```
  function UserObject(param){
    this.param1 = param;
  }
  var userObj = new UserObject(‘x’);
  ```

- Anonymous
  - Open and closed with brackets, properties defined with : and separated with ,

  -
    ```
    {property1: “value”,
     property2: 123,
     function1: function(){}//or the name of
                                  //a function
    }
    ```

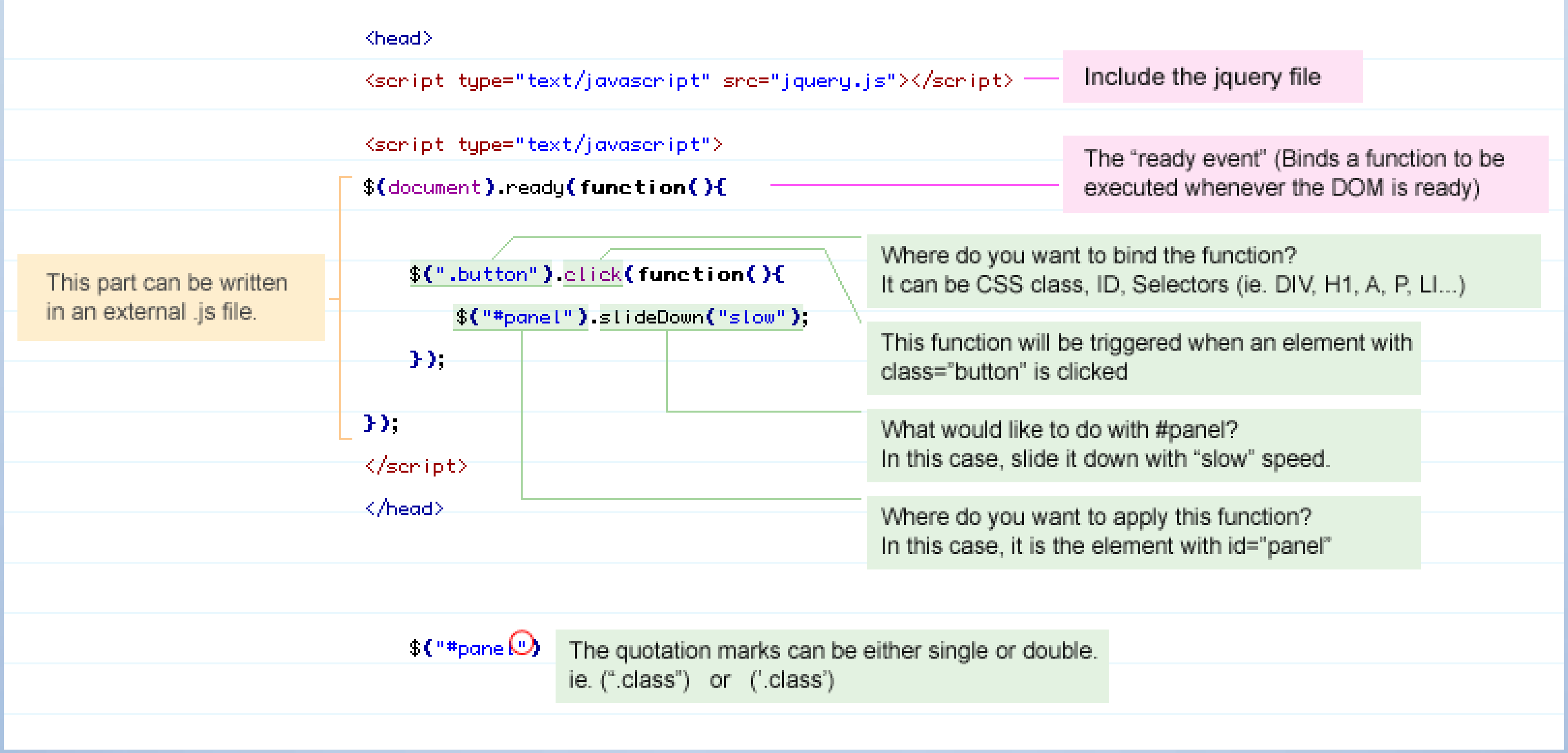
jQuery Object

```
jQuery(‘selector’) or $(“selector”)
```

- Returns a jQuery object with an array or selected elements and helper methods.
- When we perform an action on the jQuery object, it will be performed on all DOM Elements encapsulated in it.

jQuery

Anatomy of a jQuery Page



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

<script type="text/javascript">
$(document).ready(function(){

    $(".button").click(function(){
        $("#panel").slideDown("slow");
    });

});
</script>
</head>
```

Include the jquery file

The “ready event” (Binds a function to be executed whenever the DOM is ready)

This part can be written in an external .js file.

Where do you want to bind the function? It can be CSS class, ID, Selectors (ie. DIV, H1, A, P, LI...)

This function will be triggered when an element with class="button" is clicked

What would like to do with #panel? In this case, slide it down with “slow” speed.

Where do you want to apply this function? In this case, it is the element with id="panel"

```
$("#panel")
```

The quotation marks can be either single or double. ie. (“.class”) or (‘.class’)

http://www.webdesignerwall.com/tutorials/jquery-tutorials-for-designers/

Starting Code When the Page is loaded

- Standard approach
  - document.onload
    - Does Not fire until entire page is loaded, including images
- With jQuery
  - `$(document).ready(function);`
  - shortcut method:
    - `$.ready(function);`
  - Function parameter can be either a named function or a anonymous function

```
$(document).ready(function(){
    //Your Code Goes Here
});
```

Selectors

- Supports most CSS 1-3 selectors and xquery selectors
  - Simple Examples:
    - Select all elements: `$(‘*’)`
    - Select all div elements: `$(‘div’)`
    - Select element by id: `$(‘#id’)`
    - Select all elements with class: `$(‘.class’)`
    - Select a List of elements: `$(‘h1,h2,h3’)` `//there is an easier way`

Hierarchy Selectors

- Ancestor Descendant Selectors
  - Select all paragraphs inside and element

    `$(‘#wrapper p’)`
  - Select all input elements on a form
    - `$(‘form input’)`
- Parent Child Selectors
  - Find all child elements of a selector
    - `$(‘#element > *’)`
- Previous / Next Selectors
  - Find all inputs that are next to a label
    - `$(‘label + input’)`

Pseudo Class Filters

- Convenience selectors modeled after CSS pseudo selectors
  - :first :last
  - :even :odd
  - :eq(index) :gt(index) :lt(index)
  - :header :hidden :visible

Attribute Filters

- Filtering elements based on values of their attributes
  - find input with name attribute = value
    - `$('input[name=??]') –`
  - Find anchor tags that start with mailto:
    - `$('a[href^=mailto]')`
  - Find anchor tags that end with 'pdf'
    - `$('a[href$=pdf]')`

Traversing

- The jQuery object has a list of dom elements,
- Several helper methods are available when you need to find or modify the elements selected.
  - – find(selector)
  - – filter(selector)
  - – children(selector)
    - $(‘div’).children().addClass(‘class’);

Manipulation

- Changing Contents
  - All manipulation methods have an overload that sets the value
    - `Html()`
    - `Text()`
    - `Val() – form fields`
- Insert Inside, Outside, and Around selected elements
- Replace and Remove Contents of selected elements

CSS

- Get the Style Property
  - $(‘abc’).css(‘style-name’);
  - $(‘abc’).css(‘style-name’,value);
  - $(‘abc’).css({backgoundColor:’green’,color:’blue’);
- Positioning, Height & Width
  - $(‘abc’).offset()
    - Returns object with left & top properties
- Adding and Removing css classes
  - $(‘abc’).addClass(‘class-name’);
  - $(‘abc’).removeClass(‘class-name’);
  - $(‘abc’).toggleClass(‘class-name’);

Events

- Can attach events and trigger events
- Use bind() method to bind to any event
- Helper methods
  - Click(function) change(function)
  - Mouse events
- Interaction event handlers
  - Hover(mouseover, mouseout)
  - Toggle(fn1,fn2,…)

Effects

- Simple methods to show, hide and simple visual effects
  - `show(), show(‘fast’|’slow’)`
  - `hide(), hide(‘fast’|’slow’)`
  - `toggle();`
- Similar functions to fade in/out, slide in/out
- Custom animations
- More effects in the jQuery.UI library

Ajax

- Provides a base ajax method and several specialized methods
  - `$.ajax(options)`
    - `$.ajax({type:’post’,url:’remote.aspx’})`
  - `$.get(url,data,callback);`
  - `$.post(url,data,callback);`
  - `$.load(url,data,callback);//loads html`
  - `$.getJSON(url,data,callback);`