

WEB DESIGN

Javascript & JQuery

Topics

- Javascript & JQuery
- JQuery UI

Dynamic HTML (DHTML)

- Manipulating the web page's structure is essential for creating a highly responsive UI
- Two main approaches
 - Manipulate page via plain JS
 - Manipulate page using JS + library (e.g., jQuery)

Document Object Model (DOM)

- Fancy name for the web page's structure
- Web page is basically a tree structure
 - One node per HTML element
 - Each node can have attributes

Welcome to jQuery

- jQuery is one of many available libraries that
 - Provide functions for manipulating the web page
 - With fairly good performance
 - Help to keep your JS code clean
 - Indirectly help to protect security (somewhat)
- Those are the benefits of using such a library
- The downside is that you have an extra dependency and need to learn a new library

Examples of things you can do with jQuery

- Read the contents of DOM nodes (tag)
- Modify the contents of DOM nodes
- Modify the appearance of DOM nodes
- Create and attach new DOM nodes
- Remove DOM nodes
- Run a function right when the page is ready
- Add and remove event handlers
- Retrieve content from a web server
- Send content to a web server

Aspects of the DOM and jQuery

- **Identification:** how do I obtain a reference to the node that I want.
- **Traversal:** how do I move around the DOM tree.
- **Node Manipulation:** how do I get or set aspects of a DOM node.
- **Tree Manipulation:** how do I change the structure of the page.

Warning: You need clean HTML

- If you want jQuery to perform reliably...
 - Always include `<html></html>` tag
 - Always put this line before your `<html>` tag
`<!DOCTYPE html>`
 - This tells the browser to operate in "standards" mode.
 - Always include `"` around your attribute values
`blah blah`

Getting started with jQuery

- Download a copy of the jquery JS file and store it on your hard drive
- Reference the JS file in your HTML (online method or offline method)
 - What are the differences?
 - Which one is better?
- Access the jQuery functions via the \$ object

Online Method: CDN Hosted JQuery

- A number of large enterprises provide hosted copies of jQuery on existing CDN networks that are available for public use.
 - jQuery CDN
 - <http://code.jquery.com/jquery-1.4.4.min.js>
 - Google Ajax API CDN
 - Microsoft CDN
 - <http://ajax.microsoft.com/ajax/jquery/jquery-1.4.4.min.js>

Offline Method

- Download your jquery.js file through www.jquery.com

Referencing JQuery: Online vs Offline

- In development mode, you may use the offline method. You won't need internet access
- In production mode, well I myself prefer the online method rather than the offline method. Users are using internet access anyway to reach my website. So, I don't want to maintain the JQuery file.

Getting Started

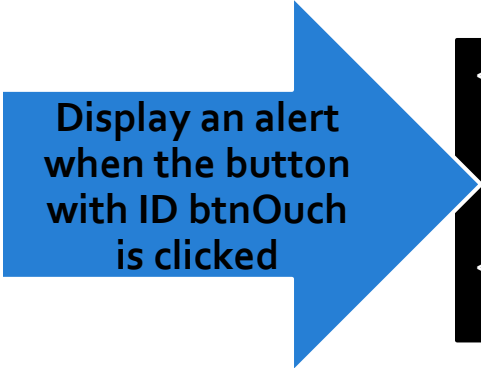
```
<!DOCTYPE html>
<html>
  <head>
    <title>Fun with jQuery</title>
  </head>

  <body>
    <h2>Hello, jQuery!</h2>
    <button id='btnOuch'>Say Ouch</button>
    <script src='http://code.jquery.com/jquery.js'></script>
    <script>
      $("#btnOuch").click(function(){
        alert("Ouch! That hurt.");
      });
    </script>
  </body>
</html>
```

- Include jQuery in the source file
- Define jQuery functions
- Save this file as index.htm
- Try it!

Example 1: A Closer Look

- `$("#btnOuch")` selects the element with ID `btnOuch`
- `click()` binds a click event to selected element
- The function executes when the click event is fired



**Display an alert
when the button
with ID `btnOuch`
is clicked**

```
<script>
    $("#btnOuch").click(function(){
        alert("Ouch! That hurt.");
    });
</script>
```

How jQuery Works

- The jQuery syntax is used to **select** HTML elements and perform some **action** on those element(s).
- Basic syntax: **\$(selector).action()**
 - A dollar sign to define jQuery
 - A (selector) to find HTML elements
 - An action() to be performed on the element(s)

jQuery Selectors

Syntax	Description
<code>\$(this)</code>	Current HTML element
<code>\$("p")</code>	All <code><p></code> elements
<code>\$("p.intro")</code>	All <code><p></code> elements with <code>class="intro"</code>
<code>\$(".intro")</code>	All elements with <code>class="intro"</code>
<code>\$("#intro")</code>	The first element with <code>id="intro"</code>
<code>\$("ul li:first")</code>	The first <code></code> element of each <code></code>
<code>\$("[href\$='.jpg']")</code>	All elements with an <code>href</code> attribute that ends with <code>".jpg"</code>
<code>\$("div#intro .head")</code>	All elements with <code>class="head"</code> inside a <code><div></code> element with <code>id="intro"</code>

Some jQuery Events

Event Method	Description
<code>\$(selector).click(function)</code>	Invokes a function when the selected elements are clicked
<code>\$(selector).dblclick(function)</code>	Invokes a function when the selected elements are double-clicked
<code>\$(selector).focus(function)</code>	Invokes a function when the selected elements receive the focus
<code>\$(selector).mouseover(function)</code>	Invokes a function when the mouse is over the selected elements
<code>\$(selector).keypress(function)</code>	Invokes a function when a key is pressed inside the selected elements

Example 2 :

Getting a value from a textbox

- Put a textbox in your page
- Also put a button below the textbox
- When you click the button,
 - Show the value of the textbox in an alert box
`alert($("#myTextBox").val());`
 - In case the textbox is empty, show alert message "The textbox is empty"

Example 2 :

Putting a value to a textbox

- Put two textboxes in your page
- Also put a button below the textbox
- When you click the button,
 - Put the value from 1st textbox into the 2nd textbox
`$("#my2ndTextBox").val($("#my1stTextBox").val());`
 - In case the 1st textbox is empty, show alert message "The textbox is empty"

Example 3

Getting value from attribute

```
<script>
    $("#btnReplace").click(function(){
        alert ($("#mytextbox").attr("myid"));
    });
</script>
```

Consider you have this textbox in your page:

```
<input type="textbox" name="mytextbox" id="mytextbox" myid="123" value="">
```

Example 4

Putting value to an attribute

```
<script>
    $("#btnReplace").click(function(){
        alert($("#mytextbox").attr("myid")); // old value
        $("#mytextbox").attr("myid", "new value");
        alert($("#mytextbox").attr("myid")); // new one
    });
</script>
```

Consider you have this textbox in your page:

```
<input type="textbox" name="mytextbox" id="mytextbox"
      myid=" ...??...." value="...">
```

Exercise

- You have a textbox, a button and a hyperlink
- You type a certain domain address on the textbox and then click the button
- So, the destination of the link will be change
- Example:
 - You type <http://www.ubaya.ac.id>, click the button, so when you click the hyperlink, it will redirect you to <http://www.ubaya.ac.id>

Manipulating CSS

CSS Properties	Description
<code>\$(selector).css(propertyName)</code>	Get the style property value of the first selected element
<code>\$(selector).css(propertyName,value)</code>	Set the value of one style property for selected elements
<code>\$(selector).css({properties})</code>	Set multiple style properties for selected elements
<code>\$(selector).addClass(class)</code>	Apply style <i>class</i> to the selected elements
<code>\$(selector).removeClass(class)</code>	Remove style <i>class</i> to the selected elements

Example 5

Change color of paragraph lemon when btnColor is clicked

```
<script>
  $("#btnColor").click(function(){
    $("#lemon").addClass("blue");
  });
</script>
```

```
<style type="text/css">
  .red{
    color:red;
  }
  .blue{
    color:blue;
  }
</style>
```


Example 6

```
<script>
    $("#btnColorCheck").click(function(){
        alert($("#lemon").css("color"));
    });
</script>
```

Display the color of the paragraph lemon when btnColorCheck is clicked.

Example 7

Highlight (background-color = yellow) any paragraph that is double-clicked

```
<script>
    $("p").dblclick(function(){
        $(this).css("background-color",
"yellow");
    });
</script>
```

EXERCISE

1. Create these css class in your html file:

```
<style>
```

```
.red-color { background: red; }
```

```
.yellow-color { background: yellow; }
```

```
.green-color { background: green; }
```

```
div { margin: 0 auto; width: 150px; height: 150px; }
```

```
</style>
```

EXERCISE

2. Write your another html :

```
<body>
```

```
<div id='mybox' class='red-color' ></div>
```

```
<p style='text-align: center; '>
```

```
  <button id='buttonred'>Change Color to Red</button>
```

```
  <button id='buttonyellow'>Change Color to Yellow</button>
```

```
  <button id='buttongreen'>Change Color to Green</button>
```

```
</p>
```

```
<script>
```

```
  //.... Your script should be here
```

```
  // when you click buttonred button, the div will be backgrounded into red
```

```
  // when you click buttonyellow button, the div will be backgrounded into yellow
```

```
  // when you click buttongreen button, the div will be backgrounded into green
```

```
</script>
```

```
</body>
```

EXERCISE

3. Write your another html :

```
<body>
<div id='mybox' class='red-color' current='red'></div>
<p style='text-align: center;'>
    <button id='mybutton'>Change Color</button>
</p>
<script>
    //.... Your script should be here
    // if current div color is red and you click the button, it will change into yellow
    // if current div color is yellow and you click the button, it will change into green
    // if current div color is gree and you click the buttonn, it will change into red
</script>
</body>
```

Some jQuery Effects

Function	Description
<code>\$(selector).hide()</code>	Hide selected elements
<code>\$(selector).show()</code>	Show selected elements
<code>\$(selector).toggle()</code>	Toggle (between hide and show) selected elements
<code>\$(selector).slideDown()</code>	Slide-down (show) selected elements
<code>\$(selector).slideUp()</code>	Slide-up (hide) selected elements
<code>\$(selector).slideToggle()</code>	Toggle slide-up and slide-down of selected elements
<code>\$(selector).fadeIn()</code>	Fade in selected elements
<code>\$(selector).fadeOut()</code>	Fade out selected elements
<code>\$(selector).fadeTo()</code>	Fade out selected elements to a given opacity
<code>\$(selector).fadeToggle()</code>	Toggle between fade in and fade out

Example 8

```
<script>
    $("h2").click(function(){
        $(this).hide("slow");
    });
</script>
```

What will this do?

What happens if you
have more than one h2?

Try it!

Example 9

Hide all blue paragraphs when
btnHideBlue is clicked

```
<script>
  $("#btnHideBlue").click(function(){
    $("p.blue").hide("slow");
  });
</script>
```

```
<button id='btnHideBlue'>Hide Blue</button>
```


Example 10

Create a toggle button that shows/hides paragraph lemon.

```
<script>
  $("#btnToggle").click(function(){
    $("#lemon").slideToggle("slow");
  });
</script>
```

Example 11

```
<script>
    $("#btnFade").click(function(){
        $("#lemon").fadeTo("slow", 0.5);
    });
</script>
```

Fade paragraph lemon
to 50% opacity when
btnFade is clicked.

Exercise

- Put this code in your HTML BODY

```
<body>
```

```
<select id="cbochoice">
```

```
<option value="">Pick One Fruit...</option>
```

```
<option value="1">Mango</option>
```

```
<option value="2">Orange</option>
```

```
<option value="3">Melon</option>
```

```
<option value="4">Other Fruit</option>
```

```
</select>
```

```
<input type="text" id="txtfruit" style="display: none;">
```

```
</body>
```

Exercise

- At the first time, the textbox will be hidden.
But when you choose "Other fruit" at the combobox, the textbox will be appeared and put the cursor in the textbox (using focus())
- What is the JQuery code?

Exercise

- Answer:

```
<script>
```

```
$(document).ready(function(){  
    $("#cbochoice").change(function() {  
        var choice = $(this).val();  
        if(choice=='4') {  
            $("#txtfruit").show();  
            $("#txtfruit").focus();  
        } else {  
            $("#txtfruit").hide();  
        }  
    });  
});  
</script>
```

Manipulating HTML

Function	Description
<code>\$(selector).html(content)</code>	Changes the (inner) HTML of selected elements
<code>\$(selector).append(content)</code>	Appends content to the (inner) HTML of selected elements
<code>\$(selector).after(content)</code>	Adds HTML after selected elements

Example 12

```
<script>
  $("#btnReplace").click(function(){
    $("#lemon").html("Lollipop soufflé ice cream tootsie roll donut...");
  });
</script>
```

Replace text in
paragraph lemon when
btnReplace is clicked.

Example 13

Append text to paragraph
lemon on mouseover

```
<script>
  $("#lemon").mouseover(function(){
    $(this).append(" Cookie! ");
  });
</script>
```

```
<p id='lemon'>Lemon drops biscuit chocolate...</p>
```


Do This Exercise!!!

- You have the HTML code:

```
<label>Photo:</label><br>
<div id='container-photo'>
  <input type='file' name='photo[]' id='photo1'>
</div>
<input type='hidden' id='hid_photo_counter' name='hid_photo_counter'
value='1'>
<input type='button' id='btn_photo'>
```

- Whenever you click the 'btn_photo', it will add another input-file object and add the value of 'hid_photo_counter'

```
<div id='container-photo'>
  <input type='file' name='photo[]' id='photo1'>
  <input type='file' name='photo[]' id='photo2'>
</div>
<input type='hidden' id='hid_photo_counter' name='hid_photo_counter'
value='2'>
```

Another Exercise

- Let say you have these html:

```
<div class="container" style="height: 200px;">  
  <div id="mybox" style="width: 100px; height: 100px;color: red">  
    </div>  
</div>  
<button id="btntopleft">Top Left</button>  
<button id="btntopright">Top Right</button>  
<button id="btnbottomleft">Bottom Left</button>  
<button id="btnbottomright">Bottom Right</button>
```

- The 'div' element with id="mybox" will be moved to top left, or top right, or bottom left or bottom right whenever the button is clicked

jQuery UI

- jQuery UI is a collection of GUI widgets, animated visual effects, and themes implemented with jQuery (a JavaScript library), Cascading Style Sheets, and HTML.
- All of jQuery UI's widgets are fully themeable using a consolidated, coordinated theme mechanism

jQuery UI Widgets

- Accordion – Accordion containers
- Autocomplete – Auto-complete boxes based on what the user types
- Button – Enhanced button appearance, turn radio buttons and checkboxes into pushbuttons
- Datepicker – Advanced date-picker
- Dialog – Show dialog boxes on top of other content, easily and robustly
- Menu – Show a Menu
- Progressbar – Progress bars, both animated and not
- Slider – Fully customizable sliders^[11]
- Spinner – Show a Number Spinner
- Tabs – Tabbed user interface handling, with both inline and demand-loaded content
- Tooltip – Show a Tooltip

How to Use

- Same with jquery but you'll need to add some more script
- Ex:
 - `<script src="//code.jquery.com/ui/1.11.4/jquery-ui.js"></script>`

Example 14

```
<!-- Datepicker -->  
  
<script>  
  $(function() {  
    $( "#datepicker" ).datepicker();  
  });  
</script>
```

Show a popup calendar
when it's clicked.

Example 15

```
<!-- Datepicker -->  
  
<script>  
  $(function() {  
    $( "#datepicker" ).datepicker({dateFormat: "dd-mm-yy"});  
  });  
</script>
```

Show a popup calendar
when it's clicked and
output with requested
format