


In-Class Exercise/Demo

In-Class Exercise / Demo



- In today's demo exercise, we will continue use  to test our JavaScript codes.
- Before we begin our demo exercises to test jQuery codes today, head to GAP and download the **url.txt** file on your desktop for easy access. Open this file in Notepad, Wordpad or TextEdit.

In-Class Exercise / Demo



- Similar to the last class set up, for each exercise, you will be given:
 1. A link to a CodePen exercise. **Fork** this exercise so it's saved on to your own account. *What is fork? Fork means clone, make a copy, duplicate.*
 2. Codes to enter on to your forked CodePen exercise.

Note: Do not work or make any changes to the original codepen exercise. You should only work on the forked pen that you'd saved.

In-Class Exercise / Demo



Exercise 1: Using jQuery Selectors (tag) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/qBaexPB>

- HTML markup is already set up for this exercise (see HTML window). And on the Live View (white window), you should see the content printed from the HTML. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$("#p").css("background-color", "yellow");
```
- On live view, the result is display automatically:

This is a paragraph.

This is second paragraph.

This is third paragraph.

In-Class Exercise / Demo



Exercise 2: Using jQuery Selectors (#id) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/zYKgRPQ>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
/* This would select second division only*/  
$("#div2").css("background-color", "yellow");
```

- On live view, the result is display automatically:

This is first division of the DOM.

This is second division of the DOM.

This is third division of the DOM

In-Class Exercise / Demo



Exercise 3: Using jQuery Selectors (.class) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/MWjNQQz>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
/* This would select second division only*/  
$(".big").css("background-color", "yellow");
```

- On live view, the result is display automatically:

This is first division of the DOM.

This is second division of the DOM.

This is third division of the DOM

In-Class Exercise / Demo



Exercise 4: Using jQuery Selectors (*) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/abmeqqe>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
/* This would select all the elements */  
$("*").css("background-color", "yellow");
```

- On live view, the result is display automatically:

This is first division of the DOM.

This is second division of the DOM.

This is third division of the DOM

In-Class Exercise / Demo



Exercise 5: Using jQuery Selectors (combination) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/dypxdmG>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$(".big, #div3").css("background-color", "yellow");
```

- On live view, the result is display automatically:

This is first division of the DOM.

This is second division of the DOM.

This is third division of the DOM

In-Class Exercise / Demo



Exercise 6: Using jQuery Attributes (title) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/bGwXLxv>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
var title = $("em").attr("title");  
$("#divid").text(title);
```

- On live view, the result is display automatically:

This is first paragraph.

This is second paragraph.



This is first paragraph.

This is second paragraph.

Bold and Brave

In-Class Exercise / Demo

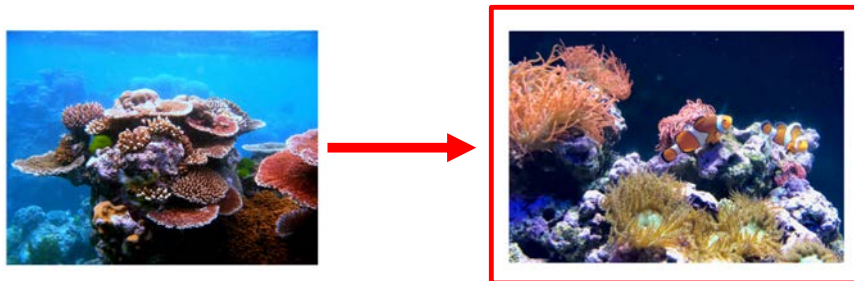


Exercise 7: Using jQuery Attributes (src) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/rNMXJqg>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function. Copy the url from **url.txt** file:

```
$("#myimg").attr("src",  
"https://upload.wikimedia.org/wikipedia/commons/5/5c/Clown_anemone_fish  
%2C_Amphiprion_ocellaris%2C_Lisbon_Oceanarium.jpg");
```
- On live view, the result is display automatically:



In-Class Exercise / Demo



Exercise 8: Using jQuery Attributes (add class attribute) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/zYKgRmg>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$("em").addClass("selected");  
$("#myid").addClass("highlight");
```

- On live view, the result is display automatically:

This is first paragraph.

This is second paragraph.

In-Class Exercise / Demo



Exercise 9: Using jQuery Attributes (change attribute values) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/YzGmeRp>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function. Copy the url from **url.txt** file:

```
$("#img").attr({  
  src: "https://upload.wikimedia.org/wikipedia/commons/5/51/JQuery_UI-Logo.svg",  
  title: "jQuery UI",  
  alt: "jQuery UI Logo"  
});
```

- On live view, the result is display automatically:

Following is the logo of jQuery



In-Class Exercise / Demo



Exercise 10: Example 1

Using jQuery Attributes (set attribute value with function) to modify HTML content

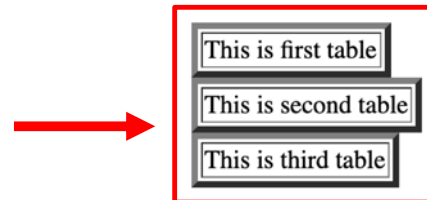
Fork this pen: <https://codepen.io/rloke/pen/BaLXYGJ>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function.

```
//Example 1:  
$("table").attr("border", function(index) {  
    return "4px";  
})
```

- On live view, the result is display automatically:

This is first table
This is second table
This is third table



In-Class Exercise / Demo



Exercise 10: Example 2

Using jQuery Attributes (set attribute value with function) to modify HTML content

Fork this pen: Not require. Continuing on the same pen.

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script below the previous script within the callback function. Copy the url from **url.txt** file:

```
//Example 2:  
$("img").attr("src", function() {  
    return "https://upload.wikimedia.org/wikipedia/commons/c/c4/" + this.title;  
})
```

- On live view, the result is display automatically:

 MINI_Cooper_S_Convertible_2005.jpg



In-Class Exercise / Demo



Exercise 11: Using jQuery Attributes (remove class attribute) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/KKgOQOJ>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$("#p#pid1").removeClass("red");
```

- On live view, the result is display automatically:

This is first paragraph.

This is second paragraph.



This is first paragraph.

This is second paragraph.

In-Class Exercise / Demo



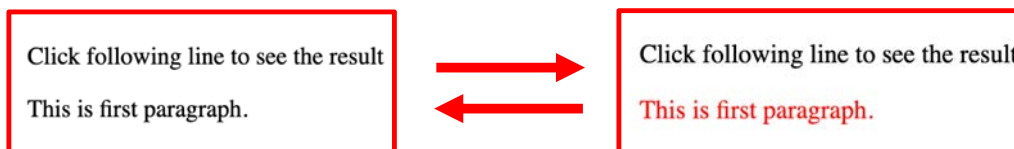
Exercise 12: Using jQuery Attributes (toggle class) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/MWjNQNd>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$("#p#pid").click(function () {  
    $(this).toggleClass("red");  
});
```

- On live view, the result is display when click on text:



In-Class Exercise / Demo



Exercise 13: Using jQuery DOM Traversing (find elements by index) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/BaLXrBx>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$("li").eq(2).addClass("selected");
```

- On live view, the result is display automatically:

- list item 1
- list item 2
- **list item 3**
- list item 4
- list item 5
- list item 6

In-Class Exercise / Demo



Exercise 14: Using jQuery DOM Traversing (filtering out elements) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/VwKoXLb>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$("li").filter(".middle").addClass("selected");
```

- On live view, the result is display automatically:

- list item 1
- list item 2
- list item 3
- list item 4
- list item 5
- list item 6

In-Class Exercise / Demo



Exercise 15: Using jQuery DOM Traversing (locate descendent elements) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/gOwVepv>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$("p").find("span").addClass("selected");
```

- On live view, the result is display automatically:

This is 1st paragraph and **THIS IS RED**

This is 2nd paragraph and **THIS IS ALSO RED**

In-Class Exercise / Demo



Exercise 16: Using jQuery DOM Traversing (children selector method) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/mdrNxPp>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$("div").children(".selected").addClass("blue");
```

- On live view, the result is display automatically:

Hello

Hello Again

And Again

And One Last Time

In-Class Exercise / Demo



Exercise 17: Using jQuery DOM Traversing (closest selector method to find match) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/OJRKvNE>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$(document).bind("click", function (e) {  
    $(e.target).closest("li").toggleClass("highlight");  
});
```

- On live view, the result is display when toggle click on each item:

Click any item below to see the result:

- list item 1
- **list item 2**
- list item 3
- list item 4
- **list item 5**
- list item 6

In-Class Exercise / Demo



Exercise 18: Using jQuery CSS (set element width & height) to modify HTML content

Fork this pen: <https://codepen.io/rloke/pen/OJbJJJj>

- HTML markup is already set up for this exercise. And the content can be seen on the Live View. We will add JavaScript to interact with HTML.
- In the JS window, enter the following script within the callback function:

```
$("#div:first").width(150);  
$("#div:first").css("background-color", "green");
```

- On live view, the result is display when toggle click on each item:



In-Class Exercise / Demo



Submission:

- Post all 18 of your Codepen exercise links on the message box in the dropbox at GAP Week 5 Day 1.
- Post at the end of this class session.