

ACAD 275

JQuery/Javascript Exam Review

April 2019

Exam Resources

[Study Guide](#)

[Notes on the Exam](#)

[JQuery vs. Javascript Review](#)

Javascript Concepts

[Styles](#)

[“If” and conditional logic](#)

[Objects](#)

[“For” and Conditional Loops](#)

[Variables](#)

[Data Validation](#)

[Methods](#)

[Data Transformation](#)

[Functions](#)

[Array loops](#)

[Arrays](#)

[Searching through arrays](#)

[Dynamic Documents](#)

Styles

CSS Properties define aspects of objects. When writing in HTML, use style blocks and property names are like "background-color" with declarations such as "width: 200px". When manipulating in JS, property names are like "backgroundColor" and declarations such as "width='200px'". When manipulating with jQuery use .css(). You should already be familiar with CSS and HTML. For the JS and jQuery you can review Editing style properties and the CSS block in Query review and JS

Objects

Generally created with HTML tags. Often have UNIQUE "id" properties. If you can "locate" an object with Javascript, you can then read a property, write/change a property, or otherwise manipulate it. See Intro to programming and JS

Variables

Can store (read and write) data such as strings, numbers, objects, functions.
(click for sample code)

```
var favcolor = "red";
```

```
var favnumber = 77;
```

```
var img2 = document.getElementById('image2');
```

See JS variables, expressions and functions.

Methods

Functionality or "mini-programs" already built into the language. Call/activate them by name. ALWAYS call with parenthesis. Sometimes call "off" an object.

Functions

"Mini-programs" written by YOU that are called/run by name. Can set up with arguments (so that data is passed INTO them when called). Always define AND call them with parentheses.

```
// Simple function with NO parameters that ALWAYS changes page to red
    var go_red = function(){
        document.querySelector("body").style.backgroundColor = 'red';
    }
// call/activate with
    go_red();
// Function with ONE parameter, color to change page to
    var change_bg = function(newcolor){
        document.querySelector("body").style.backgroundColor = newcolor;
    }
// call/activate with
    change_bg('teal');
// Function with TWO parameters, ID of image to change and new src
```

[Exam Resources](#) | [Javascript Concepts](#) | [jQuery Concepts](#) | [Functionality](#) | [Don't worry about](#)

Arrays

More complex type of variable that can store multiple rows of data (rather than just one piece). Created by a command such as "var favcolors = new Array

```
// Simple/single array
var favcolors = new Array;
favcolors[0] = "red";
favcolors[1] = "pink";
favcolors[2] = "yellow";
favcolors[3] = "purple";
favcolors[4] = "teal";
```

```
// also
favcolors = new Array;
favcolors.push("red");
favcolors.push("pink");
favcolors.push("yellow");
favcolors.push("purple");
favcolors.push("teal");
```

```
also
var favcolors = ["red", "pink", "yellow", "purple", "teal"];
```

[Exam Resources](#) | [Javascript Concepts](#) | [jQuery Concepts](#) | [Functionality](#) | [Don't worry about](#)

```
var imagepath = new Array;
```

```
var imagedescr = new Array;
```

Dynamic Documents

Populating containers (divs, body) with new content by new innerHTML (in JS) or using the html() method (in jQuery).

```
// Javascript innerHTML property of containers
<div id="box1"></div>
<script>
    document.querySelector('#box1').innerHTML = 'new content for div';
</script>

// jQuery html() function
$('#box1').html('new content for div');
```

“If” and conditional logic

Write a condition/expression that if "true" the following set of code is run. Can have else if and else clauses.

```
if(favcolor == "red") {  
    alert('i like the color red too!');  
}
```

```
if(document.querySelector('#emailbox').value.indexOf('usc.edu') != -1) {  
    alert('Go Trojans!');  
}
```

```
if(your_age >= 18) {  
    alert('You can vote in the U.S.!');  
}
```

“For” and Conditional Loops

Loop/repeat a set of code based on defined conditions.

```
//simple for loop with alert
  for(i=0;i<9;i++){
    alert('Current number is ' + i);
  }
```

Data Validation

Use methods such as `indexOf()` to determine if data matches desired content and/or formats.

Data Transformation

Use methods such as `charAt()`, `substring()`, and `replace()` and regular expressions to modify data into different formats or content.

Array loops

Building code that "loops" around the data of an array, such as to build dynamic page content.

```
newhtml = "";
for(i=0;i<dogimages.length;i++){
    newhtml += "<img src='" + dogimages[i] + "'> ";
}
document.querySelector("#content").innerHTML += newhtml;
```

Searching through arrays

Using if clauses inside a for loop, parse through array data to find/match desired information. See the searching through arrays portion of Searching through arrays and review your Bookmark/link assignment.

JQuery Concepts

[Set up](#)

[Selectors](#)

[\(document\), \(this\), \(window\)](#)

[on\(\)](#)

[Methods](#)

[Object modification](#)

[Command chain](#)

[Javascript + JQuery](#)

Set up

Load library and starter/ready block

```
<script src="http://code.jquery.com/jquery.js"></script>  
<script>  
$(document).ready(function(){  
  
});  
</script>
```

Selectors

Targeting objects through selectors

(document), (this), (window) objects

(document), (this) and (window) objects

on()

Using on() to delay code based on an event/action on an object

```
$('#button2').on("click", function(){  
    // do something when button is clicked  
});
```

Methods

Basic jQuery object methods like show and hide, toggle, fadeIn and fadeOut, slideDown and slideUp, etc. And using delay() to "pause" between those simple object animations.

Object modification

Altering objects with jQuery methods like `html()`, `css()` and `attr()`

```
$('#image2').attr("src", "image3.jpg");
```

```
$('body').css('background-color', 'red');
```

```
$('#box2').html('new div content');
```

Command chain

Chaining multiple commands off one object.

Javascript + JQuery

Using Javascript structural commands (if statements, for loops, variables, arrays and functions) inside jQuery.

Functionality in JQuery and Javascript

[Detect user interaction](#)

[Object manipulation](#)

[Variables](#)

[Parallel Arrays](#)

[Dynamic data](#)

[“workflow”](#)

[Page react to user interaction.](#)

[Data validation](#)

Detect user interaction

`<input type="button" id="button1">`

in Javascript

```
document.querySelector("#button1").addEventListener("click", function(){  
    // JS to execute when user clicks on #button1  
})
```

in jQuery

```
$("#button1").on("click", function(){  
    // jQuery to run when user clicks on #button1  
});
```

Object manipulation

Manipulate objects through their html/tag attributes and style properties

Javascript

```
document.querySelector('#image1').src = 'image2.jpg';  
document.querySelector("body").style.backgroundColor='red';  
document.querySelector('#box2').style.width = '200px';
```

jQuery

```
$("#image1").attr("src", "image2.jpg");  
$("body").css("background-color", "red");  
$("#box2").css("width", "200pox");
```

Variables

Store data/objects in variables

Javascript

```
var favcolor = "red";  
var favnumber = 77;  
var img2 = document.querySelector('#image2');
```

jQuery

```
var favcolor = "red";  
var favnumber = 77;  
var img2 = $("#image2");
```

Parallel Arrays

Store data in arrays/parallel arrays and randomly pick out information,
Use/output all data, and/or Match and extract one set of data

Dynamic data

Use data to dynamically create items on a page or edit objects

“workflow”

Create a "workflow" of events on a page

Page react to user interaction.

Through logic have a page operate or react differently based on user interaction and/or data

Data validation

Validate and/or change user data

Don't worry about... (NOT ON TEST)

animate()

Property-value maps

Scripting media

Synchronizing elements (Complex object sequences)