

Announcements

- Quiz #6 on Friday
 - Topic: Two-dimensional arrays
- You cannot use regular expressions for Project #6.
- JavaScript Lint is extremely helpful when writing your JavaScript. Use it 😊

One-Dimensional Arrays

- Let's review and see some additional information about arrays.
- **Array** – ordered collection of values.
- **Indexing** – first element associated with index 0.
- An element of an array can be of any type and an array can hold different types of elements.
- **Initialization of arrays**
 - Via array literal – comma separated list of elements within square brackets
 - `var a = [2, 3, 5];`
 - `var b = []; // empty array`
 - Specified in the Array constructor
 - `var c = new Array();`
 - `var d = new Array(2, 3, 5); // initializes array with 2, 3, 5`
 - `var e = new Array(4); // defines array of size 4`
- You can print array contents with `alert`

One-Dimensional Arrays

- You can change the number of elements of an array at any time.

- Example

```
var data=[]; // initially zero elements  
data[0]=10;  
data[1]=20; // two elements by now
```

- **length property**

- Read/write value
 - Can be used to expand/truncate array
 - Example (truncating)

```
var data=[];  
data[0]=10;  
data[1]=20;  
data.length = 1;
```

- **Example:** [GetFilteredData.html](#)

Converting Between Arrays and Strings

- **From array to String via join:**

```
var a = [5, 9, 10];  
var aStr = a1.join();           // aStr → 5, 9, 10  
var aStr2 = a1.join("<br />"); // aStr2 → 5<br />9<br />10<br />
```

- **From String to array via split:**

```
var b = "30, 40, 50";  
var bArray = b.split(",");  
for (var idx=0; idx < bArray.length; idx++) // loop that prints 30, 40, 50  
    alert(bArray[idx]);
```

Combining and Dividing Arrays

- **concat** – join arrays together returning an array with the result. It does not modify the original arrays. You can pass additional array as comma-delimited parameters.

- **Example:**

```
var data1 = ["Mary", "June"];  
var data2 = ["Lynn", "Kim", "Tim"];  
var concatenated = data1.concat(data2); // concatenated will have  
// ["Mary", "June", "Lynn", "Kim", "Tim"]
```

- **splice** – removes a segment of elements from the array. First parameter represents starting index; second number of elements to retrieve from that index on.

- **Example:**

```
var ages = [-1, 4, 5, 6, 8, 10, 12, 14];  
var segment = ages.splice(2, 3);  
// ages now has the values -1,4,10,12,14  
// segment has the values 5,6,8
```

Two-Dimensional Arrays

- JavaScript does not support actual two-dimensional arrays
- You can simulate two-dimensional arrays by using arrays of arrays.
- About two-dimensional arrays
 - You can pass them and return them from functions like one-dimensional arrays
 - Any modifications in the function will be permanent
 - You can have ragged arrays
- **Example:** TwoDimensionalArrays.html