



JavaScript and jQuery Course

Session 07 Arrays

Arrays

- ▶ special type of variables / object
- ▶ stores **multiple values** using a single variable name
- ▶ for a list or collection

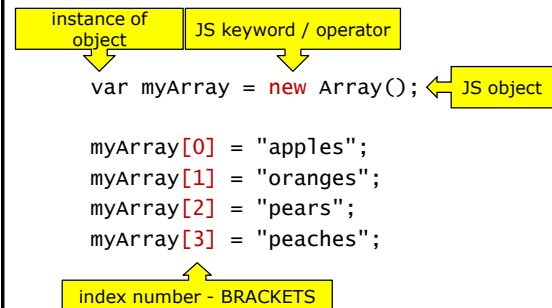
▶ 2

Arrays

- ▶ **Array()** **object** - uppercase A
- ▶ arrays always begin with **0 (zero) index**
- ▶ each element in the array has an **index** value
- ▶ an array has a **length** property

▶ 3

Create an Array – Alternative 1



▶ 4

Create an Array – Alternative 2

JS keyword / operator

```
var myArray = new Array("apples", "oranges", "pears");
```

Define contents of array
when declaring the variable

▶ 5

Create an Array – Alternative 3

```
var myArray = ["apples", "oranges", "pears"];
```

new operator not used
JS knows it is an array by the []

▶ 6

Create an Array – Alternative 4

```
var myArray = [ ];
```

Declare the array
then define contents later

```
myArray[0] = "apples";  
myArray[1] = "oranges";  
myArray[2] = "pears";
```

▶ 7

My Choice

```
var myArray = new Array();
```

```
myArray[0] = "apples";  
myArray[1] = "oranges";  
myArray[2] = "pears";  
myArray[3] = "peaches";
```

▶ 8

Retrieve element from an array – need to use brackets

```
var myArray = new Array();
```

```
myArray[0] = "apples";  
myArray[1] = "oranges";  
myArray[2] = "pears";  
myArray[3] = "peaches";
```

```
alert("myArray[0] is " + myArray[0]);
```



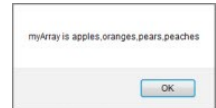
▶ 9

Retrieve all elements from an array

```
var myArray = new Array();
```

```
myArray[0] = "apples";  
myArray[1] = "oranges";  
myArray[2] = "pears";  
myArray[3] = "peaches";
```

```
alert("myArray is " + myArray);
```



▶ 10

Length property of an Array

```
var myArray = new Array();
```

```
myArray[0] = "apples";  
myArray[1] = "oranges";  
myArray[2] = "pears";  
myArray[3] = "peaches";
```

```
alert("myArray.length is " + myArray.length);
```

▶ 11

Loop through an array

```
var myArray = new Array();
```

```
myArray[0] = "apples";  
myArray[1] = "oranges";  
myArray[2] = "pears";  
myArray[3] = "peaches";
```

```
for (var i = 0; i < myArray.length; i++)  
{  
    alert("myArray[i] is " +  
          myArray[i]      +  
          "and i is "     + i);  
}
```



▶ 12

Modify an array

```
var myArray = new Array();
```

```
myArray[0] = "apples";
myArray[1] = "oranges";
myArray[2] = "pears";
myArray[3] = "peaches";
```

```
myArray[3] = "ripe peaches";
```

new value for element

► 13

Add an element to an array

```
var myArray = new Array();
```

```
myArray[0] = "apples";
myArray[1] = "oranges";
myArray[2] = "pears";
myArray[3] = "peaches";
```

```
myArray[myArray.length] = "bananas";
```

last element = [4]

► 14

Create empty array – add elements

```
var numbers = [];
```

```
for (var i = 0; i < 10; i++) {
    numbers[i] = i + 1;
}
```

Code that displays the numbers array created above

```
var numbersString = "";
```

```
for (var i = 0; i < numbers.length; i++) {
    numbersString += numbers[i] + " ";
}
```

```
alert (numbersString);
```

12345678910

OK

► 15

Loop through an array – using for loop

```
var myArray = new Array();
```

```
myArray[0] = "apples";
myArray[1] = "oranges";
myArray[2] = "pears";
myArray[3] = "peaches";
```

```
for (var i = 0; i < myArray.length; i++)
{
    alert("myArray[i] is " +
        myArray[i] +
        "and i is " + i);
}
```

► 16

Loop through an array – using **for-in** loop

```
var myArray = new Array();
```

```
myArray[0] = "apples";
myArray[1] = "oranges";
myArray[2] = "pears";
myArray[3] = "peaches";
```

The syntax of a for-in loop

```
for (var elementIndex in arrayName) {
    // statements that access the elements
}
```

```
for (var i in myArray)
{
    alert("myArray[i] is " +
        myArray[i] +
        "and i is " + i);
}
```

▶ 17

Loop through an array – using **for** loop and **innerHTML**

```
var myArray = new Array();
```

```
myArray[0] = "apples";
myArray[1] = "oranges";
myArray[2] = "pears";
myArray[3] = "peaches";
```

```
apples
oranges
pears
peaches
```

```
window.onload = function(){
    for (var i = 0; i < myArray.length; i++) {
        document.getElementById("array").innerHTML
            += myArray[i] + "<br>";
    }
} // end onload

<body><p id="array"> </p></body>
```

▶ 18

Loop through an array – using **for** loop and **innerHTML**

```
var myArray = new Array();
```

```
myArray[0] = "apples";
myArray[1] = "oranges";
myArray[2] = "pears";
myArray[3] = "peaches";
```

The fruits are:

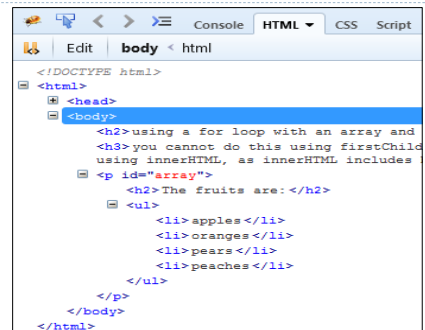
- apples
- oranges
- pears
- peaches

```
window.onload = function(){
    var str = "<h2>The fruits are:</h2>";
    var str2 = " ";
    for (var i = 0; i < myArray.length; i++)
    {
        str2 += "<li>" + myArray[i] + "</li>";
    }
    document.getElementById("array").innerHTML =
        str + "<ul>" + str2 + "</ul>";
} // end onload

<body><p id="array"> </p></body>
```

▶ 19

Loop through an array – test using **browser dev tools**



```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    <h2>using a for loop with an array and
    <h3>you cannot do this using firstChild
    using innerHTML, as innerHTML includes
    <p id="array">
      <h2>The fruits are:</h2>
      <ul>
        <li>apples</li>
        <li>oranges</li>
        <li>pears</li>
        <li>peaches</li>
      </ul>
    </p>
  </body>
</html>
```

▶ 20

Manipulate an Array

<code>join()</code>	Joins elements into a string
<code>concat()</code>	Merges arrays
<code>indexOf()</code>	Searches for an element
<code>pop()</code>	Removes last element
<code>push()</code>	Add an element to the end
<code>shift()</code>	Removes first element
<code>unshift()</code>	Add an element to the beginning

► 21

DATES - Methods of `Date()` Object

<code>var today = new Date();</code>	Define/initialize variable named <code>today</code>
<code>var dayNum = today.getDate();</code>	Returns day of month, from 1 to 31
<code>var dayWeek = today.getDay();</code>	Returns day-of-week, from 0 to 6 Sunday=0, Monday=1, ... Saturday=6
<code>var monthNum = today.getMonth();</code>	Returns month number, from 0 to 11 January=0, February=2, ... December=11
<code>var year = today.getFullYear();</code>	Returns the 4-digit year Use this instead of <code>getYear()</code>

► 22

Return day of the week using an **Array**

```
var today = new Date();
var theDay = today.getDay(); //returns a number from 0-6

//use that number as the array index to get its value
var days = new Array();

days[0] = "Sunday";
days[1] = "Monday";
days[2] = "Tuesday";
days[3] = "Wednesday";
days[4] = "Thursday";
days[5] = "Friday";
days[6] = "Saturday";

alert("The day is " + days[theDay] );
```

► 23

Return day of the week using **if..else if**

```
var today = new Date();
var theDay = today.getDay(); //returns a number from 0-6

var day;

if (theDay == 0) { day = "Sunday"; }
else if(theDay == 1) { day = "Monday"; }
else if(theDay == 2) { day = "Tuesday"; }
else if(theDay == 3) { day = "Wednesday"; }
else if(theDay == 4) { day = "Thursday"; }
else if(theDay == 5) { day = "Friday"; }
else if(theDay == 6) { day = "Saturday"; }

alert("The day is " + day );
```

► 24