

Week 5: Arrays, (Strings) & Slideshows

Arrays

- An **array** is an **object** that contains a **set of related information** and is also a **data type** that holds, contains and **stores values in a numbered sequence**
- An **array** is set up by **using the array() constructor**
- Each value in that array is called an **“element”** (of that array)
- Each “element” (of an array) is assigned within the array starting with the first element being positioned (indexed) at zero

Array Properties

- The **“length”** property is set to the number of element within the array plus one element
- Yes, I know that setting the “length” property this way seems lame
- To access an element from an array from a function or method you will need to use the [] operator

Slideshow – p. 89, Script 5.5:

```
myPix = new Array("images/pathfinder.gif","images/surveyor.gif","images/surveyor98.gif")
thisPic = 0
imgCt = myPix.length - 1

// We need to start out by declaring our variables that we will be using throughout the script
// The first variable that we need to declare is the “myPix” variable, which we will assign the value of our
“new Array” that will contain the images of our slideshow
// Remember, that the “new” operator is used when declaring a new object to be used, which in this case
is the “Array” of images
// The second variable we need is the “thisPic” variable, which we will give the value of “zero” and will be
used in a counter later in the script
// The third variable we need is the “imgCt” variable which we will give the value of “myPix.length – 1”
// By assigning the “imgCt” variable as “myPix.length – 1” we are making sure that the value of “imgCt”
will math the values in the array constructor

function processPrevious() {
    if (document.images && thisPic > 0) {
        thisPic--
        document.myPicture.src=myPix[thisPic]
    }
}

// We then need to define two functions that will enable the user to move back and fourth through the
elements of the array via an event handler (or link), thus allowing the user to view our slides
```

```
// In the first function we need to use a conditional statement to make sure that the browser understand
"document.images" syntax

// We also need to ask if "thisPic" is greater than zero as well by using the "&&" (double ampersand) and
">" (greater than) comparisons – p. 11

// If our test gets passed, then the " - " operator will subtract one from "thisPic" when the function
"processPrevious()" searches through the array

// In order to be able to search through the array we need assign "document.myPicture.src" to
"myPix[thisPic]" using the "[]" (double bracket) operator

// Once again, notice that "document.myPicture.src" works only if the image is named "myPicture" with in
the image tag
```

```
function processNext() {
    if (document.images && thisPic < imgCt) {
        thisPic++
        document.myPicture.src=myPix[thisPic]
    }
}
```

```
// If our test gets passed, then the " ++" operator will add one from "thisPic" when the function
"processNext()" searches through the array
```

```

<a href="javascript:processPrevious()">&lt;&lt; Previous</a>
<a href="javascript:processNext()">Next &gt;&gt;</a>
```

```
// By now, you all should know how this part works
```

Wraparound Slideshow – Script 5.6:

```
myPix = new Array("images/callisto.jpg","images/europa.jpg","images/io.jpg","images/ganymede.jpg")
thisPic = 0
imgCt = myPix.length - 1
```

```
// Again, We need to start out by declaring our variables that we will be using throughout the script
```

```
function chgSlide(direction) {
    if (document.images) {
        thisPic = thisPic + direction
        if (thisPic > imgCt) {
            thisPic = 0
        }
        if (thisPic < 0) {
```

```

        thisPic = imgCt
    }
    document.myPicture.src=myPix[thisPic]
}
}

```

// In this example, we will be using only one function with compound “if” statements instead of two different functions

// The function “chgSlide()” gets declared with the argument/parameter “direction” that will get a value set within the function called from the user event/link

// We then need to use a conditional statement to make sure that the browser understands “document.images” syntax

// If it does then we declare another value to the variable “thisPic”

// We assign the value “thisPic + direction” which essentially says that we add our direction parameter to the variable “thisPic”

// If the browser understand “document.images” syntax we need to then state if “thisPic > imgCt” (thisPic is greater than imgCt) set “thisPic” to “zero”

// If the browser understand “document.images” syntax we need to, also, state if “thisPic < 0” (thisPic is less than 0) set “thisPic” to “imgCt”

// Again, in order to be able to search through the array we need assign “document.myPicture.src” to “myPix[thisPic]” using the “[]” (double bracket) operator

// Once again, notice that “document.myPicture.src” works only if the image is named “myPicture” with in the image tag

```


<a href="javascript:chgSlide(-1)">&lt;&lt; Previous</a>
<a href="javascript:chgSlide(1)">Next &gt;&gt;</a>

```

// Here in the event handler/link when we call the function “chgSlide(direction)” and set our values of “direction” to either next one or previous one