Week 4

4.1 Strings

- JavaScript strings are used for storing and manipulating text.

1. Strings inside quotes

- A JavaScript string is zero or more characters written inside quotes.
- You can use single or double quotes:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Strings</h2>

<p> Strings are written inside quotes. You can use
single or double quotes:</p>

<p id="demo"> </p>

<script>
var carName1 = "Volvo XC60"; // Double quotes
var carName2 = 'Volvo XC60'; // Single quotes

document.getElementById("demo").innerHTML =
carName1 + "" +carName2;
</script>

</body>
</html>
```

JavaScript Strings

Strings are written inside quotes. You can use single or double quotes:

Volvo XC60 Volvo XC60

---

2. String Length

- To find the length of a string, use the built-in length property:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript String Properties</h2>

<p> The length property returns the length of a
string:</p>

<p id="demo"> </p>

<script>
var txt = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
var sln = txt.length;
document.getElementById("demo").innerHTML = sln;
</script>

</body>
</html>
```

JavaScript String Properties

The length property returns the length of a string:

26

3. Escape Character

- Because strings must be written within quotes, JavaScript will misunderstand this string:

```
var x = "We are the so-called "Vikings" from the north.";
```

- The string will be chopped to "We are the so-called ".
- The solution to avoid this problem, is to use the backslash escape character.
- The backslash (\) escape character turns special characters into string characters:

Code	Result	Description
\’	‘	Single quote
\’’	“	Double quote
\\	\	Backlash

a. \”

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Strings</h2>

<p> The escape sequence \" inserts a double quote in a
string.</p>

<p id="demo"> </p>

<script>
var x = "We are the so-called "Vikings" from the
north.";
document.getElementById("demo”).innerHTML = x;
</script>

</body>
</html>
```

JavaScript String Properties

The escape sequence \" inserts a double quote in a string.

We are the so-called "Vikings" from the north.

b. \’

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Strings</h2>

<p> The escape sequence \’ inserts a double quote in a
string.</p>

<p id="demo"> </p>

<script>
var x = 'It\'s alright.';
document.getElementById("demo”).innerHTML = x;
</script>

</body>
</html>
```

JavaScript String Properties

The escape sequence \' inserts a double quote in a string.

It's alright.

c. \ \

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Strings</h2>

<p> The escape sequence \\ inserts a double quote in a
string.</p>

<p id="demo"> </p>

<script>
var x = "The character \\ is called backslash.";
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>
```

JavaScript String Properties

The escape sequence \\ inserts a double quote in a string.

The character \ is called backslash.

4.2 Arrays

- An array is a special variable, which can hold more than one value at a time.
- An array can hold many values under a single name, and you can access the values by referring to an index number.

1. Creating an Array

a. Using an array literal is the easiest way to create a JavaScript Array.

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Arrays</h2>

<p id="demo"></p>

<script>
var cars = ["Saab", "Volvo", "BMW"];

document.getElementById("demo").innerHTML = cars;
</script>

</body>
</html>
```

JavaScript Arrays

Saab,Volvo,BMW

b. Using the JavaScript Keyword new.

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Arrays</h2>

<p id="demo"></p>

<script>
var cars = new Array ["Saab", "Volvo", "BMW"];

document.getElementById("demo").innerHTML = cars;
</script>

</body>
</html>
```

JavaScript Arrays

Saab,Volvo,BMW

2. Access the Elements of an Array

- You access an array element by referring to the index number.

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Arrays</h2>

<p>JavaScript array elements are accessed using
numeric indexes (starting from 0).</p>

<p id="demo"></p>

<script>
var cars = ["Saab", "Volvo", "BMW"];

document.getElementById("demo").innerHTML = cars[0];
</script>

</body>
</html>
```

JavaScript Arrays

JavaScript array elements are accessed using numeric indexes (starting from 0).

Saab

3. Changing an Array Element

- This statement changes the value of the first element in `cars`:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Arrays</h2>

<p>JavaScript array elements are accessed using
numeric indexes (starting from 0).</p>

<p id="demo"></p>

<script>
var cars = ["Saab", "Volvo", "BMW"];
cars[0] = "Opel";

document.getElementById("demo").innerHTML = cars[0];
</script>

</body>
</html>
```

JavaScript Arrays

JavaScript array elements are accessed using numeric indexes (starting from 0).

Opel,Volvo,BMW

4. Access the Full Array

- With JavaScript, the full array can be accessed by referring to the array name:

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Arrays</h2>

<p id="demo"></p>

<script>
var cars = ["Saab", "Volvo", "BMW"];

document.getElementById("demo").innerHTML = cars;
</script>

</body>
</html>
```

JavaScript Arrays

Saab,Volvo,BMW

5. The Length Property

The length property of an array returns the length of an array (the number of array elements).

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Arrays</h2>

<p>The length property returns the length of an
array.</p>

<p id="demo"></p>

<script>
var fruits = ["Banana", "Orange", "Apple", "Mango"];
document.getElementById("demo").innerHTML =
fruits.length;
</script>

</body>
</html>
```

JavaScript Arrays

The length property returns the length of an array.

4

5. Accessing the First Array Element

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Arrays</h2>

<p> JavaScript array elements are accesses using
numeric indexes (starting from 0).</p>

<p id="demo"></p>

<script>
var fruits = ["Banana", "Orange", "Apple", "Mango"];
var fruits = fruits[0];
document.getElementById("demo").innerHTML =
fruits.length;
</script>

</body>
</html>
```

JavaScript Arrays

JavaScript array elements are accesses using numeric indexes (starting from 0).

Banana

6. Accessing the Last Array Element

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Arrays</h2>

<p> JavaScript array elements are accesses using
numeric indexes (starting from 0).</p>

<p id="demo"></p>

<script>
var fruits = ["Banana", "Orange", "Apple", "Mango"];
var fruits = fruits[fruits.length-1];
document.getElementById("demo").innerHTML =
fruits.length;
</script>

</body>
</html>
```

JavaScript Arrays

JavaScript array elements are accesses using numeric indexes (starting from 0).

Mango

Source:

JavaScript Arrays. (n.d.). w3schools. Retrieved on February 8, 2021 from https://www.w3schools.com/js/js_arrays.asp

JavaScript Strings. (n.d.). w3schools. Retrieved on February 8, 2021 from https://www.w3schools.com/js/js_strings.asp