**Maria Padilla**

**COMP266**

**Unit 4 – Script Use and Augmentation**

**Learning Diary**

In this unit, we worked with scripts made with Javascript. The purpose was for us to start getting familiar with Javascript, look for some scripts made by others, try to understand them, incorporate them into our own code and finally give our comments about them.

In order to do so, I started by watching Javascript tutorials, such as the one found in the following youtube link: <https://www.youtube.com/watch?v=W6NZfCO5SIk>, as well as the ones found in w3schools. I have never done any coding before; therefore, I really needed to learn at least the very basics in order to understand what the scripts meant and how they were doing the things they were programmed to do.

After watching/reading some tutorials, I decided to use a Javascript for a slideshow and incorporate it into my own website. I took such script from the w3schools website, specifically from: <https://www.w3schools.com/howto/tryit.asp?filename=tryhow_js_slideshow>. I made some adjustments to my code, so that it worked with my own images and I also made very few changes, so that it looked how I wanted (all html, css, and javascript codes with modifications will be included in .zip file).

I found this script was useful because I wanted to incorporate a slideshow from the very start. I believe it is a good addition to the website, not only because it improves its appearance, but also because it could be useful for the personas and scenarios I have discussed in previous units.

The slideshow would give a preview of the paintings sold on the website. So, the clients could easily see right from the start what kind of products they could get in this site. If something catches their eye in the homepage with the help of the slideshow, I think that would increase the chances of them taking the time to go the gallery and explore all the options there, thus increasing the chances of them buying something.

The script was applied only to the Homepage, so this is the only code that was changed. The code for the rest of the pages remained the same.

**What went well?**

I liked that we had an “extra unit” simply to start getting familiar with Javascript before we start using it to create scripts from scratch. I think it was a good thing to see scripts already finished and working, to understand better how they work.

**What didn’t go well?**

For reasons beyond my power, I had to pause this course and work in other projects. That meant I didn’t practice nor worked on anything related to Javascript or any of the other topics we’ve learned. That complicated things for me a lot because once I came back, I had to re-check many things and it was a bit hard to get back on track.

**Original Code:**

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width, initial-scale=1">

<style>

\* {box-sizing: border-box}

body {font-family: Verdana, sans-serif; margin:0}

.mySlides {display: none}

img {vertical-align: middle;}

/\* Slideshow container \*/

.slideshow-container {

 max-width: 1000px;

 position: relative;

 margin: auto;

}

/\* Next & previous buttons \*/

.prev, .next {

 cursor: pointer;

 position: absolute;

 top: 50%;

 width: auto;

 padding: 16px;

 margin-top: -22px;

 color: white;

 font-weight: bold;

 font-size: 18px;

 transition: 0.6s ease;

 border-radius: 0 3px 3px 0;

 user-select: none;

}

/\* Position the "next button" to the right \*/

.next {

 right: 0;

 border-radius: 3px 0 0 3px;

}

/\* On hover, add a black background color with a little bit see-through \*/

.prev:hover, .next:hover {

 background-color: rgba(0,0,0,0.8);

}

/\* Caption text \*/

.text {

 color: #f2f2f2;

 font-size: 15px;

 padding: 8px 12px;

 position: absolute;

 bottom: 8px;

 width: 100%;

 text-align: center;

}

/\* Number text (1/3 etc) \*/

.numbertext {

 color: #f2f2f2;

 font-size: 12px;

 padding: 8px 12px;

 position: absolute;

 top: 0;

}

/\* The dots/bullets/indicators \*/

.dot {

 cursor: pointer;

 height: 15px;

 width: 15px;

 margin: 0 2px;

 background-color: #bbb;

 border-radius: 50%;

 display: inline-block;

 transition: background-color 0.6s ease;

}

.active, .dot:hover {

 background-color: #717171;

}

/\* Fading animation \*/

.fade {

 -webkit-animation-name: fade;

 -webkit-animation-duration: 1.5s;

 animation-name: fade;

 animation-duration: 1.5s;

}

@-webkit-keyframes fade {

 from {opacity: .4}

 to {opacity: 1}

}

@keyframes fade {

 from {opacity: .4}

 to {opacity: 1}

}

/\* On smaller screens, decrease text size \*/

@media only screen and (max-width: 300px) {

 .prev, .next,.text {font-size: 11px}

}

</style>

</head>

<body>

<div class="slideshow-container">

<div class="mySlides fade">

 <div class="numbertext">1 / 3</div>

 <img src="img\_nature\_wide.jpg" style="width:100%">

 <div class="text">Caption Text</div>

</div>

<div class="mySlides fade">

 <div class="numbertext">2 / 3</div>

 <img src="img\_snow\_wide.jpg" style="width:100%">

 <div class="text">Caption Two</div>

</div>

<div class="mySlides fade">

 <div class="numbertext">3 / 3</div>

 <img src="img\_mountains\_wide.jpg" style="width:100%">

 <div class="text">Caption Three</div>

</div>

<a class="prev" onclick="plusSlides(-1)">&#10094;</a>

<a class="next" onclick="plusSlides(1)">&#10095;</a>

</div>

<br>

<div style="text-align:center">

 <span class="dot" onclick="currentSlide(1)"></span>

 <span class="dot" onclick="currentSlide(2)"></span>

 <span class="dot" onclick="currentSlide(3)"></span>

</div>

<script>

var slideIndex = 1;

showSlides(slideIndex);

function plusSlides(n) {

 showSlides(slideIndex += n);

}

function currentSlide(n) {

 showSlides(slideIndex = n);

}

function showSlides(n) {

 var i;

 var slides = document.getElementsByClassName("mySlides");

 var dots = document.getElementsByClassName("dot");

 if (n > slides.length) {slideIndex = 1}

 if (n < 1) {slideIndex = slides.length}

 for (i = 0; i < slides.length; i++) {

 slides[i].style.display = "none";

 }

 for (i = 0; i < dots.length; i++) {

 dots[i].className = dots[i].className.replace(" active", "");

 }

 slides[slideIndex-1].style.display = "block";

 dots[slideIndex-1].className += " active";

}

</script>

</body>

</html>

**Comments made for the Javascript**

var slideIndex = 1;

//this part is to state a variable, which in this case is called slideIndex and its value is 1

showSlides(slideIndex);

//This part is invoking the result of the function showSlides with the paramenter being the variable slideIndex

function plusSlides(n) {

 showSlides(slideIndex += n);

}

// This part invoked the showSlides function, to show the slideIndex plus n, whose value is defined in the html code. The purpose is to make the next botton show the following slide

function currentSlide(n) {

 showSlides(slideIndex = n);

}

//This function is used for the little dots that appear under the images. It invokes the showSlide function to show the slide of the n value we defined on the html code.

function showSlides(n) {

 var i; //declaration of variable i, which is left undefined

 var slides = document.getElementsByClassName("mySlides"); //In this section, a new variable "slides" is determined and its value will be given by the method document.getElementsByClassName will return all the elements in the document with the Class Name "mySlides" as a collection of nodes

 var dots = document.getElementsByClassName("dot"); //In this section the previous method is used again but for a new variable "dots" and the class name will be "dot"

 if (n > slides.length) {slideIndex = 1} //In this section, the length of the collection of nodes is used. If n is larger than the value of the slides variable, the value for slideIndex will be 1

 if (n < 1) {slideIndex = slides.length} //If n is smaller than 1, the value for slideIndex will be the length of the variable “slides”

 for (i = 0; i < slides.length; i++) {

 slides[i].style.display = "none";

 }

//This is a loop. It works as long that the value of variable "i" is smaller than the length of the variable "slides", and increases a value every time that the loop is repeated. Every time these conditions are met, the slides[i] will be hidden. This means that all slides will be hidden.

 for (i = 0; i < dots.length; i++) {

 dots[i].className = dots[i].className.replace(" active", "");

 }

//The same method is used for the variable "dots", except in this case the dots will be unactive for all cases when the conditions are met.

 slides[slideIndex-1].style.display = "block";

//This part says that only the slide with the value slideIndex-1 will be displayed.

 dots[slideIndex-1].className += " active";

}

//This part says only the dot for the same value will be active.

**Other comments:**

The names of the variables are meaningful, which makes it easy to infer what is their purpose in the script.

Also, all variables and functions follow the cammel notation, which is the recommended notation in Javascript.

I found the script was well organized and looked clean; however, it is worth mentioning that it was very short, so it is not as easy to get lost or confused as it is with a very long code.