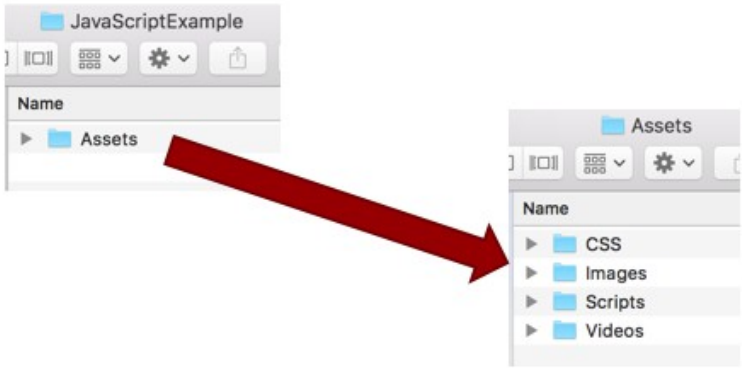
Lab 9 – Introduction to JavaScript

Please use the following site layout for your example:

Step 1 – Create a new project folder

- Create a new project folder named JavaScriptExample
- Add the following folders to the project folder:



Step 2: Add a html page to the project

- index.html

index.html

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <meta charset="utf-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1">
6    <title>My JavaScript Example</title>
7  </head>
8  <body>
9
10 </body>
11 </html>
```

Step 3: Add a CSS file to the page

Step 4 – add a JavaScript file to the project

- In the Assets/Scripts folder, create a file named site.js
- To add the JavaScript file to the page, add the following to the head section of the index.html page:
- <script src="Assets/Scripts/site.js"></script>

NB: If you are cutting and pasting from the lab document, make sure everything pastes as expected. Double quotes usually get modified and the page won't display as expected!!

Problem 1 - Variables

Create a html page and add the following JavaScript to the page to store a persons firstname and surname:

The above has created two new variables and stored a firstname and lastname for the person.

To display the firstname and lastname we can use document.write();

This will delete everything from the current page and display out new information if it is called after the page has loaded.

Add document.write(firstname + " " + lastname); to the JavaScript file (after the variable declarations!!)

Reload the page, the name should be showing on the page!!

Problem 2 – Displaying Info

Create a html page and add the following JavaScript to the page to store a persons firstname and surname (same as above):

For this example we will create a function to display out the users name when a button is clicked.

Add the following <button> tag to the page:

<button>Display Name</button>

Next add the following function to the site.js file:

The function is named displayName.

It will display the information stored in the firstname and lastname variables.

The function contains a local variable named message, this variable is only available for use within this function. If you try to access it outside the function we will get an error. It is known as a **local variable**.

A variable named fullname has been declared in the function. This will store the firstname a space and the lastname. Fullname is a local variable.

The fullname has been concatenated to the message string. (the strings have been joined together).

Document.write(message) will write the contents on the message variable string to the page when the function displayName is called!!

Update the button to call the function displayName when clicked:

```
<button onclick="displayName();">Display Name</button>
```

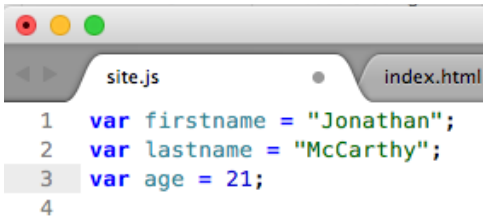
Save all open files and reload the page, the message should appear on the page when the button is pressed.

Eg.

When button is pressed:

Problem 3 – If/Else Statements

Create a html page and add the following JavaScript to the page to store a persons firstname and surname and age (similar to the above example):



If the persons age is 18 or over, we will welcome them to the niteclub, else we will them them not tonite!! As we are making a decision we will use an if else clause.

We will create a new function named niteClubEntry()

This function will perform the logic to check if the person is allowed into the niteclub or not!! Add the following function to the site.js page:

```
function niteClubEntry() {
    if (age > 17) {
        document.write("Welcome!!");
    } else {
        document.write("Not tonite");
    }
}
```

When the function is called, it will perform a check on the persons age. If they are over 17 display welcome, else display not tonite.

Add a button to the page to call the function when pressed:

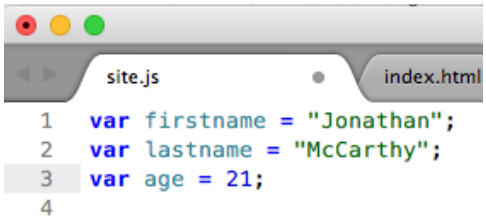
<button onclick="niteClubEntry();">Allowed In?</button>

Reload the page and test it is working correctly.

Try make some improvements to the operation of the example.

Problem 4 – For Loops

Create a html page and add the following JavaScript to the page to store a persons firstname and surname and age (similar to the above example):



For this problem we want to display the persons age on the page the same number of times as their age.

We will create a new function to perform this task: displayMultipleNames()

Update the button:

<button onclick="displayMultipleNames();">Allowed In?</button>

Save all open files and test the page!!

To do: Create another example and use a while loop!!

Problem 5 – Arrays

Create a html page and add the following JavaScript to the page:

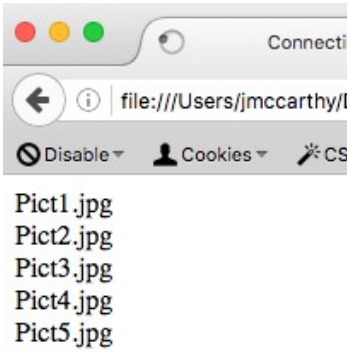
site.js

site.js × index.html × site.cs

```
1  pictures = new Array(5);
2  pictures[0] = "Pict1.jpg";
3  pictures[1] = "Pict2.jpg";
4  pictures[2] = "Pict3.jpg";
5  pictures[3] = "Pict4.jpg";
6  pictures[4] = "Pict5.jpg";
7
8  function displayPictureNames() {
9
10     for (i=0; i<pictures.length; i++) {
11         document.write(pictures[i] + "<br>");
12     } // close for loop
13
14 } // close function
```

Add a button to the page:

<button onclick="displayPictureNames();">Allowed In?</button>



The page should display the above when the button is clicked!!

Problem 6 – Arrays

Create a html page and add an image to the page:

The page will display 5 images, get 5 images of a simalay size and store them in the images folder in your project as follows:

Assets/Images/

- Pict1.jpg
- Pict2.jpg
- Pict3.jpg
- Pict4.jpg
- Pict5.jpg

Add the following html img element to the page.

```
<img id="myImg" src="Assets/Images/Pict1.jpg" alt="Pict 1" />

<button onclick="changePicture();">Next Pict</button>
```

Next create an Array of JavaScript Images (same as the previous example).

Add the following javaScript to your site.js file:

The changePicture function is using document.getElementById("myImg") to find an element on the page with an id = myImg (ie. Our img element).

When we have found the img element we can use JavaScript to change the picture it is displaying. We can do this be changing the src attribute to point at a different image.

We have an array of images, we can pick a different image from the array to display, eg. Pictures[1].

The path to the image needs to be the same as the one we used to display the image in HTML. ie. Assets/Images/Pict1.jpg

When we are preparing the new image to display the page to the image and the image name must be correct for the image to display on the page.

```
var nextPict = "Assets/Images/" + pictures[1];

document.getElementById("myImg").src = nextPict;
```

The above tells the page to get the filename from the pictures array and get this image to display on the page when the Next button is clicked.

Create your page example based on the screenshots above.

To do: can you make the Next button loop through displaying all the images in the array?

Problem 7 – Changing text on the page

Add a div to a new html page with an id of name

```
<div id="name"></div>
```

Give the div a unique identifier.

We will use JavaScript to access the div to change its content.

Using JavaScript we will add a name to the div when a button is pressed.

HTML

JavaScript

document.getElementById("name").innerHTML = "Johnny";

document.getElementById() is used in JavaScript to access an element on the HTML page with a specific id.

If JavaScript can locate the element on the page, it can make changes to it.

document.getElementById("name").innerHTML = "Johnny";

Description	JavaScript
Identify the element on the page	document.getElementById("name")
Access the element content	.innerHTML
Change the content	= "Johnny";

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" type="text/css" href="Assets/CSS/site.css">
  <script src="Assets/Scripts/site.js"></script>
  <title>My JavaScript Example</title>
</head>

<body>

<div id="name"></div>
<button onclick="changeName();">Press Me</button>

</body>
</html>
```


```
function changeName() {
    document.getElementById("name").innerHTML = "Jon McCarthy";
}
```

Create your page example using the following.

When the button is clicked the mage is added to the element for display.

Additional Work:

Question 1

Write a html page (timestable.html) with JavaScript code to show the 5 times tables:

Example:

5 times 1 = 5
5 times 2 = 10
5 times 5 = 15
etc… (up to 12 times 5 = 60)

Question 2

Write a html page (numbertest.html) with JavaScript code to perform the following functionality:

Check that the number stored in a variable is numeric data. The webpage should then show a message using appropriate tags indicating if the number input is

- greater than zero
  or
- less than or equal to zero
