

Javascript Specialist Designation Program

Lab Exercise #10

"I'm late, I'm late, for a very important date!"

Lab Instructions

1) Run your Brackets development environment, create a new file and quickly save it under the file name **tenth_javascript_lab.html**

2) Key in the basic HTML document structure and set **Lab #10** as the content for the title tag.

3) In the body of the HTML document add a heading 1 tag with the content, "Check to see if it's a weekend".

4) We're going to need an input box for the user to input a date and a button to submit.

Start with an input tag, give it a type equal to text and an id of "inputDay".

Next, open a button tag, give it an id of "checkDay" and "CHECK" as its content.

```
<h1>Check to see if it's a weekend</h1>
```

```
<input type='text' id='inputDay' />
```

```
<button id='checkDay'> CHECK </button>
```

5) Now for some Javascript. Create a separate Javascript file and quickly save it as **main.js**.

Link the HTML file to the Javascript file.

```
<script src="main.js"><script>
```

6) Open your **main.js** file and create the window.onload function.

Inside the onload function, we're going to add an event listener for the button. Search the document for an element with the id "checkDay" then add a "click" event listener and run a function name "runIsWeekend".

```
window.onload = function() {  
  document.getElementById('checkDay').addEventListener("click", runIsWeekend);  
};
```

7) The response function "runIsWeekend" will be responsible for checking the date the user input and calling a function while passing in the input information.

Inside the "runIsWeekend" let's declare a variable called "dateToCheck" and initialize with the input box's value.

```
var dateToCheck =  
document.getElementById('inputDay').value;
```

Afterwards, we will call the function that will check to see if it is a weekend. Let's call it "isWeekend" and pass in "dateToCheck" as a parameter.

Your code should look something like this:

```
function runIsWeekend() {  
  var dateToCheck =  
  document.getElementById('inputDay').value;  
  isWeekend(dateToCheck);  
};
```

8) Now for the main course. The "isWeekend" function will take in a parameter, let's call it "userDate"

```
function isWeekend(userDate) {}
```

Inside the isWeekend function we will declare a variable called "d" then initialize it by creating a new Date object and passing in "userDate" as a parameter.

```
var d = new Date(userDate);
```

9) Below the variable create an if statement and check to see if the day of "d" is equal to 6 OR 0 (since Saturday and Sunday fall on these

numbers). To do this, you need to use the `getDay()` function and attach it to `"d"`. If true, return an alert saying "It's a weekend!". Else, alert, "Not a weekend!"

Your code should look something like this:

```
function isWeekend(userDate) {  
    var d = new Date(userDate);  
    if(d.getDay() == 6 || d.getDay() == 0) {  
        return alert("It's a weekend!");  
    }  
    alert('Not a weekend!');  
};
```

10) Click File Save to save the current version of your document. Navigate to the HTML document using your operating system and double click it. The document should open in your default browser.

The date in your input box should be written in the following format:

"July 13, 2017" or "Dec 20, 2017"

Click submit and check to see if it's a weekend!

Challenge Yourself

Enhance your skills by attempting the exercises below.

1) Write a Javascript function to get the number of days in a month then push them into an array!

2) Put your site on the web. We've arranged a special deal with Blue Host. Visit <http://www.bluehost.com/track/learntoprogram/> and click "Get Stated Now." You will be able to access web hosting plans as low as \$3.49 a month. Once Blue Host takes you through the process of creating your domain and web server upload your lab and post the URL for the others in the class to see.