

JavaScript: Calling Functions

Calculating Average & Creating Forms

By: Rogelio Rios

CIS 2336

Wednesday 5:30 pm – 7pm

Create webpage to Calculate Average

- The goal of this assignment is to create a simple webpage with a button that prompts the user with 2 dialogue boxes.
- First dialogue box should display a statement of introduction and what is going to happen when user clicks ok.
- Second dialogue box should display the arithmetic and the result of it using JavaScript.
- To start create a JS and HTML file where your going to write your website and function.

1. We create webpage

Internet Application Development CIS 2336. Natalia Fofanova

If you click button below, it will call Alert box, from which you will call JS function to calculate Average

Submit

After clicking "Submit" you will get alert box

heyuhnem.com says

Hello from: Natalia Fofanova let's calculate the average.

OK

After Clicking OK you will get alert with Average

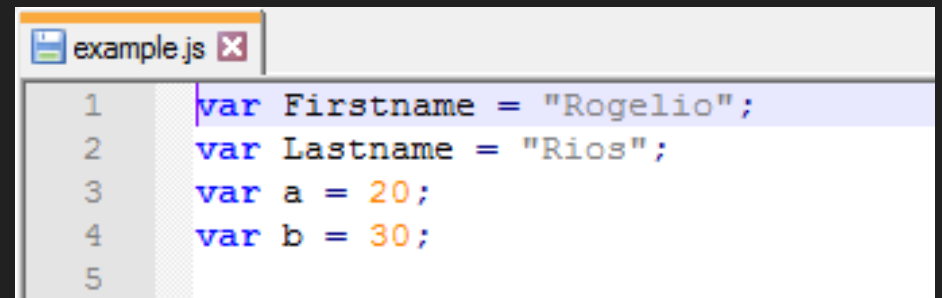
heyuhnem.com says

The average of 20 & 10 = 15

OK

Start with JavaScript

- Its best to start with JavaScript because we can make an average function easy.
- Call all variables that will be needed for this assignment (Firstname, Lastname, a and b)
- "a and b" can be any number



```
example.js
1  var Firstname = "Rogelio";
2  var Lastname = "Rios";
3  var a = 20;
4  var b = 30;
5
```


Creating Our First Function

- Once all variables have been called we make our first function to solve the arithmetic side.
- Since we are making a simple average function we will use this equation to get out average.
- Note that if we needed to get user input we would have to use arrays in our equation.

```
function getAverage(a,b) {  
    return (a+b) / 2;  
}
```

Creating Alert boxes

- We create another function that can be named as you see fit
- Use `alert()`; and add info required as shown in the image.

```
function myFunction() {  
    alert("Hello from: " + Firstname + " " + Lastname + " " + "let's calculate the average.");  
    alert("The average of " + a + " & " + b + " = " + getAverage(a, b));  
}
```

Moving to HTML

- Set it up as a normal HTML page first
- Then add `<script>` tags inside the `<head>` tag
- In the opening `<script>` tag write type and src of the .js file.
- There is two script tags because without it when you call the function it wont work for some reason.

```
<script type="text/javascript" src="example.js"></script>  
<script>  
</script>
```

Making Our Button

- Once the location of the .js file has been determined in HTML its time for creating our button.
- Creating <button> tag is the next thing to do (inside body tag).
- Just make the onClick = "myFunction()"
- And that's all it take to make a webpage with a button to find average.

```
<button type="button" onclick="myFunction()">Click me!</button>
```


Creating A Form in HTML

- Start off by making a html and setting up the basics
- In the body use <form> tags to create a form
- Add desired info and create input tag as shown in picture
- Once its done close it

```
<html>
<head>
  <link rel="stylesheet" href="FormStyle.css">
</head>
<body>

  <form>
    First name: <input type="text" name="fname" required pattern = "[A-Za-z]+"><br>
    Last name: <input type="text" name="lname" required><br>
    Email: <input type="text" name="email" required><br>
    Phone: <input type="text" name="phone" size = 12
           required pattern = "[0-9]{3}[ -][0-9]{4}"><br>

    <button type="submit" value="Submit">Submit</button>
    <button type="reset" value="Reset">Reset</button>
    <p id="ErrMsg"></p>

  </form>

</script>
</body>

</html>
```


Styling Our Form

- Using CSS you can style the form any way you would like
- Or you can leave them default
- For my case I styled them as shown in the picture

```
/* Style inputs with type="text", select elements and textareas */
input[type=text], select, textarea {
  width: 30%; /* Full width */
  padding: 12px; /* Some padding */
  border: 1px solid #ccc; /* Gray border */
  border-radius: 4px; /* Rounded borders */
  box-sizing: border-box; /* Make sure that padding and width */
  margin-top: 6px; /* Add a top margin */
  margin-bottom: 16px; /* Bottom margin */
  resize: vertical; /* Allow the user to vertically resize the */
}

/* Style the submit button with a specific background color etc. */
input[type=submit] {
  background-color: #4CAF50;
  color: white;
  padding: 12px 20px;
  border: none;
  border-radius: 4px;
  cursor: pointer;
}

/* When moving the mouse over the submit button, add a darker background color */
input[type=submit]:hover {
  background-color: #45a049;
}

/* Add a background color and some padding around the form */
.container {
  border-radius: 5px;
  background-color: #f2f2f2;
  padding: 20px;
}
```

First name:

Last name:

Email:

Phone: