learntoprogram.tv

Javascript Specialist Designation Program

Lab Exercise #13

Lab Instructions

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

2) Key in the basic HTML document structure and set **Lab #13** as the content for the title tag.

3) In the body of the HTML document add a heading 2 tag with the content, "Using the XMLHttpRequest object".

4) We're going to create a paragraph tag with a few words. Let's start by opening a <p> tag and giving it an id of "demo".

For the paragraphs content let's use, "This text will change"

Let's create a button. Using a button tag, placed after the previous <p> tag, add a "Change text" message.

Lastly, give the button an id of "httpButton". Your code should look something like this:

```
<h2>Using the XMLHttpRequest object</h2>

<button id='httpButton'>Change
text</button>

<p id="demo">This text will change</p>
```

5) We need to create a text document that will hold the new text on click. Create a separate .txt file and quickly save it as **text.txt**.

Inside the file, add :

```
This is the new text.
```

Remember we don't need quotes in a .txt file.

6) 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>
```

7) Open your **main.js** file and create a window.onload function.

Inside the onload function, search the document for an element with the id name "httpButton" then add an event listener to run a function "renderText" on click.

```
window.onload = function() {
document.getElementById('httpButton').add
EventListener('click', renderText)
};
```

8) Since the callback function, "renderText()" has not been created yet, let's go ahead and do that.

Inside the renderText function we're going to declare the variable "xhttp" and initialize it with a new instance of XMLHttpRequest().

```
var xhttp = new XMLHttpRequest();
```

9) Now let's create an "onreadystatechange" function and attach it to xhttp. Inside this function, search the document for an element with the id name "demo" and make its innerHTML equal to xhttp's responseText. "responseText" is a key inside the XML object.

```
function renderText() {

var xhttp = new XMLHttpRequest();

  xhttp.onreadystatechange = function() {
  document.getElementById("demo").innerHT
  ML = xhttp.responseText;
  }

};
```

10) Great! Lastly, still inside the renderText() function let's open a text document. Set the xhttp open() function and pass in the 3 parameters. The first one will be the action "GET", the second parameter will be the name

of the file, "text.txt", and the third will be true.

Send the xhttp request!

Your final code for "renderText()" should look like this:

```
function renderText() {

  var xhttp = new XMLHttpRequest();

  xhttp.onreadystatechange = function() {
  document.getElementById("demo").innerHT
  ML = xhttp.responseText;

  };

  xhttp.open("GET", "text.txt", true);

  xhttp.send();

};
```

11) 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.

Click the button and watch the text change from your request!

Challenge Yourself

Enhance your skills by attempting the exercises below.

1) Explore POST request and try to change the text in the text.txt file.

2) Put your site on the web. We've arranged a special deal with Blue Host. Visit **http://www.bluehost.com/track/learntoprogra m/** 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.