Working on Assignment

Making a Request

Continue Later | Finish | Instructions

(worth 5 points)

Objectives: Use GDownloadURL() to send a request to, and handle the response from, the server

Make a drop down list with dynamic data

Practice with JavaScript Strings and Arrays

Material From: Week 2 Module

Chapter 3 in "Pragmatic Ajax"

Data files that are attached to this assignment

~~~~~~

Create a web page that allows the user to browse Foothill's CTIS course offerings. This page will show course descriptions for many courses in each of three departments in the CTIS division. Each department has a list of course numbers and each course has a short description.

The user will start by selecting the department from one drop down list. Once the user has chosen a department, your page will update a second drop down list to make it contain only course numbers for courses in that department. When the user chooses a course number, your page will show the description for that course.

Your program will get the data (department and course names, numbers, and descriptions) from an external file. These data files are attachd to this assignment. You must use GDownloadUrl() to generate an XMLHttpRequest object and generate a callback in order to get the data into your program. To use GDownloadUrl() from your own computer, you must have a Google Maps API key, and you must put the key in a script tag in your html file. See documentation here:

http://code.google.com/apis/maps/documentation/

When you move your source file (and the data) to a webserver for submission, you have to change the key to match the server you are using. For the ctislab server, use the following key:

ABQIAAAAKfSIpqvYvv06ct-cn8YFARQd22Grf9IKmPGUd3sckVmPHWTBWxTzscbD3Epe9l1_9t0HqDRPqFNTpw

Your source code can contain no department names or course names. In this way, your source code will need no modification when the courses change, only the data files will need to be modified. Because all of this functionality must be implemented in JavaScript, the body of your html page will contain only the following:

```
<body>

<div id = "browseCatalog">

<select id = "selectDepartment">

<option> </option>

</select>

<select id = "selectCourse">

<option> </option>

</select>

<p id = "courseDescription">

Choose a department and a course number to view a course description

</p>

</div>

</body>
```

Steps:

1. Before you start coding, make sure you understand the data files, and that you know exactly how your completed web page will function from the user's point of view.

2. Write the JavaScript that builds the web page by creating DOM elements, assigning event handlers, and testing - one element at a time. For example, first see if you can create a dropdown list of departments from the file "departments.txt". Then make your program pop up an
~~~~~~

alert with the chosen department name when the user chooses a department from the dropdown list. Do not proceed any further until your program has these features.

3. Thoroughly test your page, and review your code to make sure that it follows the Program Guidelines for this course, and that it meets the requirements for this assignment.

4. Upload your pages to the losaltos webserver and test it from ctislab. In the Assignments tool in etudes, paste a link to your html document and submit.

Hints:

1. Refer to and use predefined JavaScript string manipulation methods like "split()".

2. You will probably need to define a global variable or two, so that your whole program has access to it. Use no more than 2 global variables in order to receive full credit.

3. Use the attribute "selectedIndex" of a select node to find which option in its dropdown list is currently selected.

4. You may find the following JavaScript functions useful for this assignment:

```
// Creates a new node of type "element", appends it to "parent".  If "text" is

// nonempty, this also creates a text node with "text" in it and appends it to the

// new node. Sets the name and id attributes to "nameId".

function addNode(parent, element, nameId, text)

{     var childEl = document.createElement(element);

      childEl.setAttribute("id", nameId);

      childEl.setAttribute("name", nameId);

      if (text != "")

      {    textNode = document.createTextNode(text);

            childEl.appendChild(textNode);

      }

      parent.appendChild(childEl);

      return childEl;

}

// Modifies the dropdown list that has a select node with "id". Adds options to it

// from "stringArray". The string in each element of "stringArray" becomes one

// option in the dropdown list.

function makeDropDown(id, stringArray)

{     var selectNode = document.getElementById(id);

      selectNode.options.length = 0;  // get rid of old stuff in the drop down

      for ( i = 0; i < stringArray.length; ++i)

      {      addNode (selectNode, "option", stringArray[i], stringArray[i]);

      }

}
```

- castcourses.txt
- ciscourses.txt
- coincourses.txt
- departments.txt

Answer

Font family Font size

Continue Later Finish Instructions