Homework Turnin

Name:	Stephen Hung
Email:	hungs3@uw.edu
Student ID:	1536327
Section:	AD
Course:	CSE 154 16au
Assignment:	hw6
Receipt ID:	e13089390b0cbec788c4d7037eb04c35

```
no compiler configured for web
```

Turnin Successful!

The following file(s) were received:

bestreads.js (4502 bytes)

```
/*
    Stephen Hung
    CSE 154 AD
    Javascript file for bestreads.html.
    This provides the functionality to load all the books onto the home page,
    display the book's information once it is clicked, and go to the home page
    when the home button is clicked.
*/
// Anonymous function following the module pattern.
(function(){
    "use strict";
    // module-global constant variable for the php URL.
    var URL = "https://webster.cs.washington.edu/students/hungs3/hw6/bestreads.php";

    // Anonymous function for when the page loads.
    // This function loads the home page and sets the home button's onclick.
    window.onload = function(){
        loadHome();
        var homeButton = document.getElementById('back');
        homeButton.onclick = loadHome;
    };

    // Lodas the home page by querying the server for the books and
    // setting the single book div to hidden.
    function loadHome(){
        var singleBookDiv = document.getElementById('singlebook');
        singleBookDiv.style.display = "none";
        queryServer("?mode=books",loadBookXML);
    }

    // Loads the information on a specific book as a response
    // to an AJAX request.
    function loadInfo(){
        var jsonText = JSON.parse(this.responseText);
        document.getElementById('title').innerText = jsonText.title;
        document.getElementById('author').innerText = jsonText.author;
        document.getElementById('stars').innerText = jsonText.stars;
    }

    // Loads the description of a book as a response to an AJAX request.
    function loadDescription(){
        var descriptionText = this.responseText;
        document.getElementById('description').innerText = descriptionText;
    }

    // Loads the reviews of a book as a response to an AJAX request.
    function loadReviews(){
```

```
        var reviewsHTML = this.responseText;
        document.getElementById('reviews').innerHTML = reviewsHTML;
    }

    // Loads the cover image for a book when needed.
    function loadCoverImage(folderName){
        document.getElementById('cover').src = "https://webster.cs.washington.
        edu/students/hungs3/hw6/books/" + folderName + "/cover.jpg";
    }

    // Loads the XML doc for each book as a response to an AJAX request.
    // Then it creates divs for each of the books and shows their cover
    // and title in it.
    function loadBookXML(){
        var allBooksDiv = document.getElementById('allbooks');
        allBooksDiv.style.display = "";
        allBooksDiv.innerHTML = "";
        var node = this.responseXML;
        var bookArray = node.querySelectorAll('book');
        for(var i = 0; i < bookArray.length; i++){
            // For each book in the XML doc, create a div with the cover image and title.

            var bookURL = "https://webster.cs.washington.edu/students/hungs3/hw6/books/";
            var bookDiv = document.createElement('div');
            bookDiv.onclick = bookClick;
            bookDiv.id = bookArray[i].querySelector('folder').textContent;
            var bookTitle = document.createElement('p');
            var bookImage = document.createElement('img');
            bookImage.setAttribute("alt","cover image");
            var srcPath = bookURL + bookArray[i].querySelector('folder').textContent +
            "/cover.jpg";
            bookImage.setAttribute("src",srcPath);
            bookTitle.innerText = (bookArray[i].querySelector('title').textContent);
            bookDiv.appendChild(bookImage);
            bookDiv.appendChild(bookTitle);
            document.getElementById('allbooks').appendChild(bookDiv);
        }
    }

    // Event Handler for when a book is clicked.
    // This function querys the server for the information, description, and reviews
    // once the book is clicked.
    function bookClick(){
        document.getElementById('allbooks').style.display = "none";
        document.getElementById('singlebook').style.display = "";
        queryServer("?mode=info&title=" + this.id,loadInfo);
        queryServer("?mode=description&title=" + this.id,loadDescription);
        queryServer("?mode=reviews&title=" + this.id,loadReviews);
        loadCoverImage(this.id);
    }

    // Querys the server for given params & loads a given function once
    // the request is done loading.
    function queryServer(params,onloadFunction){
        var ajax = new XMLHttpRequest();
        ajax.onload = onloadFunction;
        ajax.open("GET",URL + params,true);
        ajax.send();
    }

})();
```

bestreads.php (4346 bytes)

```
<?php
/*
        Stephen Hung
        CSE 154 AD
        PHP page for bestreads.html, this php page handles requests
        for different types of modes (info, description, revies, and books).
        It also takes in an additional parameter in the form of title.
        This PHP page returns information such an XML doc containing all the books,
        a JSON response with the book's title, review author, and review, information
        about the book, and a description of the book.
    */

    // Default directory for hw6.
    $dir = "/www/html/students/hungs3/hw6";

    // If the mode & title parameters are in the request.
```

```php
    if(isset($_GET["mode"]) && isset($_GET["title"])){
        // If/Else statement for the different types of modes.
        if($_GET["mode"] == "info"){
            infoMode($dir,$_GET["title"]);
        }else if($_GET["mode"] == "description"){
            descriptionMode($dir,$_GET["title"]);
        }else if($_GET["mode"] == "reviews"){
            reviewsMode($dir,$_GET["title"]);
        }
    }else if( isset($_GET["mode"]) ){
        // If there is only a mode in the request.
        if ($_GET["mode"] == "books"){
            // Check if the mode is "book".
            bookMode($dir);
        }
    }

    // Create and print/return a JSON file for the specified book.
    // The JSON file contains information such as the title, author, and rating.
    // This information is taken from info.txt.
    function infoMode($dir,$title){
        $dir = $dir . "/books/" . $title . "/info.txt";
        list($title,$author,$stars) = file($dir);
        // Takes in the title, author, and # of stars and creates a JSON
        // file to print.
        $jsonBookInfo = new stdClass();
        $jsonBookInfo->title = $title;
        $jsonBookInfo->author = $author;
        $jsonBookInfo->stars = $stars;
        print json_encode($jsonBookInfo);
    }

    // Prints/Returns the description of the book from description.txt
    // in the book's folder.
    function descriptionMode($dir,$title){
        $dir = $dir . "/books/" . $title . "/description.txt";
        $descriptioncontent = file_get_contents($dir);
        print $descriptioncontent;
    }

    // Prints/Returns all reviews for the specified book.
    // This function returns it in HTML format such that it follows the assignment's
    // guidelines.
    function reviewsMode($dir,$title){
        $dir = $dir . "/books/" . $title . "/review*.txt";
        foreach(glob($dir) as $review){
            list($author,$score,$review) = file($review);
            // Prints HTML formatted info (author, score, and review).
            print "<h3>".$author."<span>".$score."</span></h3>";
            print "<p>".$review."</p>";
        }
    }

    // Prints/Returns an XML sheet that contains all the books in the hw6 folder.
    // The tags for each book contain their title & the folder they are in.
    function bookMode($dir){
        $dir = $dir . "/books/*";
        $xmldoc = new DOMDocument();
        $books_tag = $xmldoc->createElement('books');
        $xmldoc->appendChild($books_tag);
        foreach(glob($dir) as $file){
            // For each book folder, generate an XML tag for that book and
            // append it to the XML sheet.
            $book_tag = generateBookXML($file,$xmldoc);
            $books_tag->appendChild($book_tag);
        }
        header("Content-type: text/xml");
        print $xmldoc->saveXML();
    }


    // Create an XML tag for each book specified and returns it.
    // The XML tag contains the title & folder for each book.
    function generateBookXML($file,$xmldoc){
        $book_tag = $xmldoc->createElement('book');
        // Creates tags for the title & folder and adds it to the
        // general book tag.

        $file_content = file($file."/info.txt");
        $title_tag = $xmldoc->createElement('title');
        $title_text = $xmldoc->createTextNode($file_content[0]);
```

```
        $title_tag->appendChild($title_text);

        $folder_tag = $xmldoc->createElement('folder');
        $foldername = substr($file, strrpos($file,"/")+1);
        $folder_text = $xmldoc->createTextNode($foldername);
        $folder_tag->appendChild($folder_text);

        $book_tag->appendChild($title_tag);
        $book_tag->appendChild($folder_tag);
        return $book_tag;
    }
?>
```