**Homework 6: PHP/JSON, Form validation.**

**Total: 45 Points.**

**Due: Tuesday, April 30th. NO extension possible.**

1. PhP/JSON

(20pts) Write the code for a web service **books.php** that outputs JSON data about books available at a book store in a particular category. This service should take a GET parameter named category and search a data file for all books in that category.

Your PHP code will read a data file named books.txt. Each line in that file matches the following format, with each token of information separated by a single vertical bar: name|author|category|date|price

All tokens of data are guaranteed not to contain any vertical bars in them. You can assume all data in the file is valid although it may contain extra blank lines and these shouldn’t crash your program or alter your output.

The JSON that is output should contain a number labeled “count”. This should represent a count of all books at the book store. It should also contain a list called “books” that contains a list for each book in the passed in category. The list for each book should contain the year labeled as “year”, the price labeled as “price”, the title labeled as “title”, and the author labeled as “author”. I

f the category parameter is not passed return a 400 status and stop execution immediately. If no books match the category you should send back the same data as usual, just leave the course array empty.

An example input file: 2

1 Burgers for the 21st Century|Stuart Reges|cooking|2010|75.00 Harry Potter and the Sorceror's Stone|J.K. Rowling|children|1998|19.99 Breakfast for Dinner|Amanda Camp|cooking|2009|22.00

Output using the above input file and books.php?category=cooking

{ "count":3,

"books":[ {"year" : "2010",

"price" : 75.00,

"title" : "21 Burgers for the 21st Century",

"author" : "Stuart Reges" },

{"year" : "2009",

"price" : 22.00,

"title" : "Breakfast for Dinner",

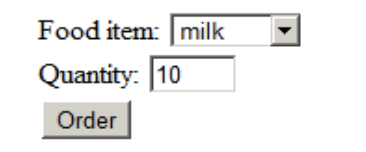
"author" : "Amanda Camp" } ] }

}

1. (25pts) Form Validation.

Download the folder the image files and .txt from blackboard.

Write the PHP code for two partial web pages for ordering food from an online store. The first page you will write is a form named order.php that allows the user to choose what kind of food to buy and how many to buy. In your form, include a drop-down menu of food items available. Include a text box for the user to enter a quantity of the item to purchase (2-characters-wide), and an "Order" button to submit the form. The form should look like this:



The food items to list should be based on what JPG food images (you can Google them yourself or use the ones I provided) are available in the current directory. For example, if the current directory contains apple.jpg and steak.jpg, then "apple" and "steak" appear in the drop-down list. Write the portion of order.php that would appear between <body> and </body>. **You don't need to write any CSS.**

**Hint: use <form action=”order-submit.php” method = “post”>**

The second page you will write is named order-submit.php. The form in order.php submits its data as a POST request to order-submit.php. The output of order-submit.php is an HTML page fragment, a single paragraph indicating information about the order as described below. (Do not use print or echo.) The store's current inventory is stored in a file named inventory.txt on the server in the current directory. Each line of the file represents one item available in the store, its quantity available, and its price per unit, separated by tabs. Assume that the file exists, that its contents are valid, and that there are no duplicates.

Here is an example inventory:

apple 4 1.00

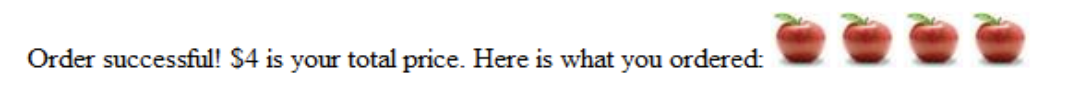
chicken 1 3.25

cookie 38 0.25

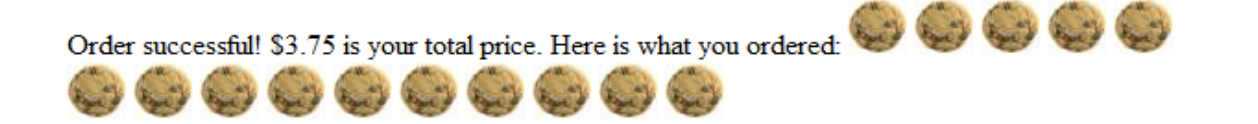
milk 9 4.50

tomato 27 0.50

In general your task is to look up the price per unit of the item the user is ordering, and use this price to compute the total order cost. For example, if the item costs $0.50 and the user orders 7 of them, the total order is 7 \* 0.50 = $3.50. The page's output in the general successful case is to inform the user that the order was successful, display the total order cost (you do not need to round it), and show the user a series of images representing what was ordered. For example, if the user orders 4 apples, your output should display 4 copies of apple.jpg.



Here is another output from ordering 15 of the item 'cookie'. (the output wraps to the next line in the browser)



The store is only able to complete an order if that food item is in the inventory and the store has enough of that item in stock. For example, if the inventory matches the above text file (which has 9 'milk' in stock), and the user tries to order 10 of 'milk', the following error should be displayed:

The items in this inventory may overlap with the images that were listed previously, but there might be some images that don't have representation in the inventory text file and vice versa. If an item is not present in the inventory file, assume that its quantity available in stock is 0 and display the same sort of error message.

You do not need to modify the inventory.txt file or update its quantity of the item being ordered. If the food item or quantity parameters are not passed, issue an HTTP 400 error. If they are present, you may assume that the food item value passed is a string and the quantity value passed is a positive integer.

***Grading schemes:***

100% To receive full credits. The code must run on server without errors.

90% Successfully implement both .php but without reporting HTTP 400 ("HTTP/1.1 400 Invalid Request") error if food item or quantity parameter are not passed.

Bonus points might be considered for better user/interface with CSS.