CS 3380 Lab Assignment 1

1 Directions

This assignment must be completed by **Sunday, February 1st at 11:59 PM**. You must upload your PHP code to Blackboard. The uploaded file should be named `lab1.php`. Your code must also be hosted on your Babbage account. The TAs will show you how to do this. Your lab 1 submission should be reachable and functional through the following URL:

`http://babbage.cs.missouri.edu/~<pawprint>/cs3380/lab1/lab1.php`

If that URL does not work you will lose points. To make the grading easier, please include the full URL for your page in the comments when you submit your file via Blackboad. **Late submissions will not be accepted.**
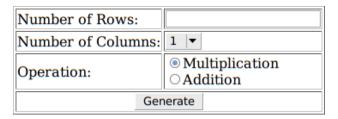
2 Goals

- Learn how to deploy web pages through personal Babbage accounts.

- Learn some basic HTML, including tables and forms.

- Learn how to generate HTML through a PHP script.

- Understand how data is sent from a clients browser to a servers PHP script.

3 Tasks

3.1 Download

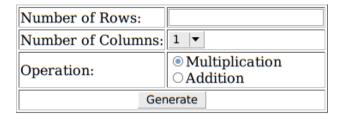For this lab, you will be provided with the HTML for the following page:

You can download a copy of my starter PHP file by executing the following commands in your terminal:

```
mkdir ~/public_html/cs3380/lab1
cd ~/public_html/cs3380/lab1
wget -O lab1.php http://babbage.cs.missouri.edu/~klaricm/ss15/cs3380/lab1/download.txt
```

Then, open the `lab1.php` file in your favorite text editor. Note that you might not be able to copy-paste the above commands. You may need to type them manually into your terminal.
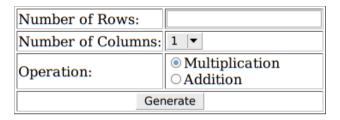
3.2 Implementation

You will be responsible for creating the PHP script required so that, when the user enters a number of rows and columns, the corresponding multiplication or addition table will be displayed. For example, here is the table produced for a 3-row, 8-column multiplication table:

Number of Rows:								
Number of Columns:	1 ▾							
Operation:	⦿ Multiplication ○ Addition							
Generate								

The 3 x 8 multiplication table.

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1	1	2	3	4	5	6	7	8
2	2	4	6	8	10	12	14	16
3	3	6	9	12	15	18	21	24

Notice that the row/column headings are shown in bold font. Your solution should do the same. Also, here's an example of a 5-row, 4-column addition table:

Number of Rows:				
Number of Columns:	1 ▾			
Operation:	⦿ Multiplication ○ Addition			
Generate				

The 5 x 4 addition table.

0	**1**	**2**	**3**	**4**
1	2	3	4	5
2	3	4	5	6
3	4	5	6	7
4	5	6	7	8
5	6	7	8	9

Notice that both multiplication and addition show a setnence above the table that indicates the dimensions of the table. Your solution should provide this as well.
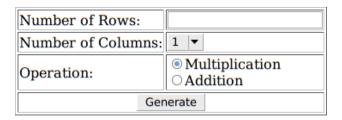
3.3 HTML Formatting

Use your browser's "view source" option to see the HTML produced by your PHP code. Is it well-organized and nicely indented? Or is it an unreadable mess? To receive full points, your code should be easy to read. Add formatting to the HTML produced by PHP by echoing the \n and \t escape sequences to output newlines and tabs.

3.4 Error Handling

In addition, you will be responsible for implementing modest of error checking:

- Check that numeric input was entered for the number of rows using `is_numeric`.

- Check that a positive number was entered for the number of rows

- Check that the "Generate" button was pressed using `isset`. (The TAs will elaborate on this in lab.)

For example, if a user enters "x" for the number of rows, the following error message should be displayed:

Number of Rows:	
Number of Columns:	1 ▼
Operation:	◉ Multiplication ○ Addition
Generate	

Invalid rows and/or columns parameters.

4   Comparison

You can compare your version with a working copy found at:
   `http://babbage.cs.missouri.edu/~klaricm/ss15/cs3380/lab1/lab1.php`