Submission Worksheet

CLICK TO GRADE

https://learn.ethereallab.app/assignment/IT202-008-S2024/it202-m2-php-problems/grade/tlj3

IT202-008-S2024 - [IT202] M2 PHP Problems

Submissions:

Submission Selection

1 Submission [active] 2/3/2024 3:08:50 PM

Instructions

COLLAPSE

Guide:

1. Make sure you're in the main branch locally and `git pull origin main` any pending changes
2. Make a new branch per the recommended branch name below (git checkout -b ...)
3. Grab the template code from https://gist.github.com/MattToegel/48b48377eaa1937c886b7840c449750a
4. Create individual PHP files for each problem and save the files inside your public_html folder in a subfolder of your choice
    1. If you don't have this folder yet, refer to the setup lessons (you'll need a few files for the deployment to work)
5. Move the unedited template files to github
    1. `git add .`
    2. `git commit -m "adding template files`
    3. `git push origin <homework branch>` (see below and don't include the < >)
    4. Create and open a pull request from the homework branch to main (leave it open until later steps)
6. Note: As you work, it's recommended to add/commit at least after each solution is done (i.e., 3+ times in this case)
    1. Make sure the files are saved before doing this
7. Fill in the items in the worksheet below (save as often as necessary)
8. Once finished, export the worksheet
9. Add the output file to any location of your choice in your repository folder (i.e., a Module2 folder)
10. Check that git sees it via `git status`
11. If everything is good, continue to submit
    1. Track the file(s) via `git add`
    2. Commit the changes via `git commit` (don't forget the commit message)
    3. Push the changes to GitHub via `git push` (don't forget to refer to the proper branch)
    4. Create a pull request from the homework related branch to main (i.e., main <- "homework branch")
    5. Open and complete the merge of the pull request (it should turn purple)
    6. Locally checkout main and pull the latest changes (to prepare for future work)
12. Take the same output file and upload it to Canvas
    1. *This step is new since GitHub renders the PDF as an image the links aren't clickable so this method works better
    2. *Remember, the github process of these files are encouragement for your tracking of your progress

Branch name: M2-PHP-Problems

Tasks: 11 Points: 10.00

COLLAPSE

Problem 1 (3 pts.)

COLLAPSE

Task #1 - Points: 1

Text: Screenshot of the Problem 1 Solved Code and Output

Details:

Only make edits where the template code mentions.

Solution should ensure that any passed in array will have only the odd values output.
Requires at least 2 screenshots (code + output from heroku dev)

Checklist

*The checkboxes are for your own tracking

#	Points	Details
#1	1	Edits were done only in the processArray() function and original template code/comments remain untouched
#2	1	Only $arr is used (no direct usage of $a1, $a2, $a3, $a4)
#3	5	Only odd values output (not odd indexes/keys)
#4	1	Includes code comments with student's ucid and date
#5	1	Output of code includes heroku dev URL with student's ucid visible

Task Screenshots:

Large Gallery

Checklist Items (0)

Checklist Items (0)

Completed code for Problem 1

Heroku dev URL for Problem 1

COLLAPSE

Task #2 - Points: 1

Text: Explain your solution

Checklist

*The checkboxes are for your own tracking

#	Points	Details
#1	1	Clearly explains how the code/logic solves the problem (mentions how the odd values are determined)

Response:

For problem 1, since what I had to do was echo only odd values from the arrays, I decided on using an if statement for each value in the array. The if statement included something I learned in Java and Python, ($value % 2 == 1) echo $value . "
";

This is the equivalent of saying (value % 2 == 1) print(value). That code basically tells the program that if any value is an odd number, echo it.

COLLAPSE

Task #3 - Points: 1

Text: Link to Problem 1 file from Production

Details:

Recommended: Use the notes tool on the right to store the base URL for your heroku production URL

Paste the production URL below, append the path to the file from your dev URL.
Note: This link won't be active until you merge the pull request, so it's expected to not exist at the time of filling this task

URL #1

https://tlj3-prod-5077e065f26e.herokuapp.com/hw2/problem1.php

COLLAPSE

Problem 2 (3 pts.)

COLLAPSE

Task #1 - Points: 1

Text: Screenshot of the Problem 2 Solved Code and Output

Details:

Only make edits where the template code mentions.

Solution should ensure that any passed in array will have the numerical values summed and rounded to two decimals (similar to currency)
Requires at least 2 screenshots (code + output from heroku dev)

Checklist *The checkboxes are for your own tracking

#	Points	Details
#1	1	Edits were done only in the getTotal() function and original template code/comments remain untouched (unless noted)
#2	1	Only $arr is used (no direct usage of $a1, $a2, $a3, $a4)
#3	5	Passed in array's values get summed AND rounded to two decimal places like currency (i.e., 0.00, 0.10, 1.10)
#4	1	Includes code comments with student's ucid and date
#5	1	Output of code includes heroku dev URL with student's ucid visible

Task Screenshots:

Large Gallery

Checklist Items (0)

Completed code for Problem 2

Checklist Items (0)

Heroku dev URL for Problem 2

COLLAPSE

Task #2 - Points: 1

Text: Explain your solution

Checklist *The checkboxes are for your own tracking

#	Points	Details
#1	1	Clearly explains how the code/logic solves the problem (mentions both how the values get summed and how the rounding is solved correctly)

Response:

In problem 2, I used the array_sum command so that it could take every number in each array and add them all together in its own array. I, then, used the round command to round the total of each array, and I put 2 in those parenthesis along with $total so that it could be rounded to the 2nd decimal place.

COLLAPSE

Task #3 - Points: 1

Text: Link to Problem 2 file from Production

Details:

Recommended: Use the notes tool on the right to store the base URL for your heroku production URL

Paste the production URL below, append the path to the file from your dev URL.
Note: This link won't be active until you merge the pull request, so it's expected to not exist at the time of filling this task

URL #1

https://tlj3-prod-5077e065f26e.herokuapp.com/hw2/problem2.php

COLLAPSE

Problem 3 (3 pts.)

COLLAPSE

Task #1 - Points: 1

Text: Screenshot of the Problem 2 Solved Code and Output

Details:

Only make edits where the template code mentions.

Solution should ensure that any passed in array will have its values converted to a positive version of the value AND converted back to the original data type.
Requires at least 2 screenshots (code + output from heroku dev)

Checklist

*The checkboxes are for your own tracking

#	Points	Details
#1	1	Edits were done only in the bePositive() function and original template code/comments remain untouched
#2	1	Only $arr is used (no direct usage of $a1, $a2, $a3, $a4)
#3	5	Passed in array's values will get converted to a positive version AND converted back to the original data type
#4	1	Includes code comments with student's ucid and date
#5	1	Output of code includes heroku dev URL with student's ucid visible

Task Screenshots:

Large Gallery

Checklist Items (0)

Checklist Items (0)


Completed code for Problem 3	Heroku dev URL for Problem 3

COLLAPSE

Task #2 - Points: 1

Text: Explain your solution

Checklist *The checkboxes are for your own tracking

#	Points	Details
#1	1	Clearly explains how the code/logic solves the problem (mentions both the conversion to positive and conversion to original data type)

Response:

In problem 3, the array_map is applied to abs() and to each value in the array. This is to ensure that the output comes out as positive. Then var_dump displays the modified array with their data types.

COLLAPSE

Task #3 - Points: 1

Text: Link to Problem 3 file from Production

Details:

Recommended: Use the notes tool on the right to store the base URL for your heroku production URL

Paste the production URL below, append the path to the file from your dev URL.
Note: This link won't be active until you merge the pull request, so it's expected to note exist at the time of filling this task

URL #1

https://tlj3-prod-5077e065f26e.herokuapp.com/hw2/problem3.php

COLLAPSE

Reflection (1 pt.)

COLLAPSE

Task #1 - Points: 1

Text: Reflect on your experience

Details:

Talk about any issues you had, how you resolved them, and anything you learned during this process.

Provide concrete details/examples.

Response:

The overall assignment was pretty easy because my previous knowledge of Python and Java helped me out, but pretty challenging at the same time since I'm not familiar with PHP and I'm a beginner with Git and Github. I definitely had some bumps along the way, but with a bit of research and some failed attempts, I successfully solved each problem and they run properly on heroku and the localhost.

COLLAPSE

Task #2 - Points: 1

Text: Include the pull request link for this branch

Details:

The correct link will end with /pull/ and a number.