Submission Worksheet

CLICK TO GRADE

https://learn.ethereallab.app/assignment/IT202-008-S2024/it202-m2-php-problems/grade/jn426

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

Submissions:

Submission Selection

1 Submission [active] 2/5/2024 2:16:15 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.)

EXPAND

Task #1 - Points: 1

Text: Screenshot of the Problem 1 Solved Code and Output

EXPAND

Task #2 - Points: 1

Text: Explain your solution

EXPAND

Task #3 - Points: 1

Text: Link to Problem 1 file from Production

COLLAPSE

Problem 2 (3 pts.)

EXPAND

Task #1 - Points: 1

Text: Screenshot of the Problem 2 Solved Code and Output

EXPAND

Task #2 - Points: 1

Text: Explain your solution

EXPAND

Task #3 - Points: 1

Text: Link to Problem 2 file from Production

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)

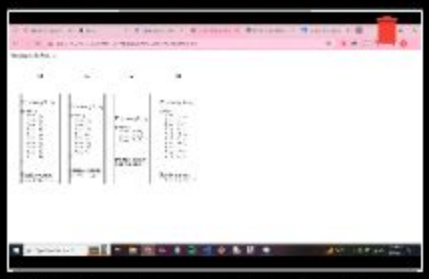
Checklist

*The checkboxes are for your own tracking

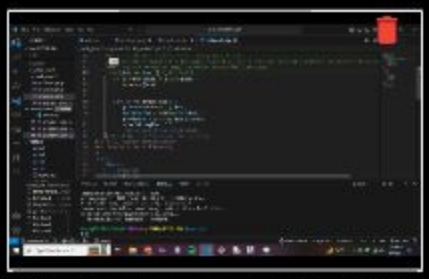
#	Points	Details
#1	1	Edits were done only in the bePositive() function and original template code/comments remain untouched
#2	1	Only $arr is used (no direct usage of $a1, $a2, $a3, $a4)
#3	5	Passed in array's values will get converted to a positive version AND converted back to the original data type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

5

1-4

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:

For floats and integers, I converted to positive using the abs() method, this method did not change the original data type so I did not have to worry about that. For the strings, I cast the string into integers, then found the absolute value using abs() to convert them to positive, and then casted it back to being a string.

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://jn426-it202-prod-a94514875a20.herokuapp.com/assignment2/problem3.php

COLLAPSE

Reflection (1 pt.)

EXPAND

Task #1 - Points: 1

Text: Reflect on your experience

EXPAND

Task #2 - Points: 1

Text: Include the pull request link for this branch