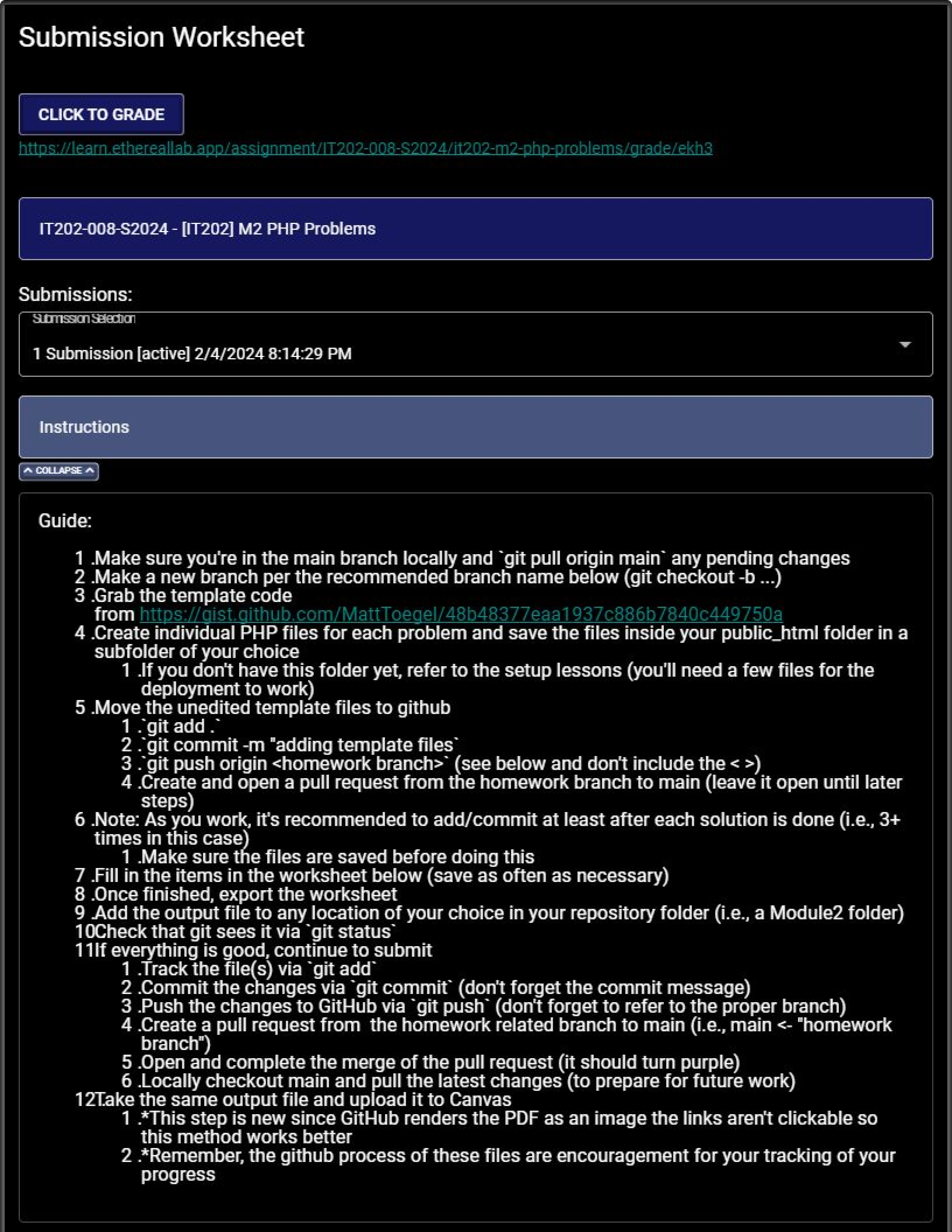


Submission Worksheet

https://learn.ethereallab.app/assignment/IT202-008-S2024/it202-m2-php-problems/grade/ekh3

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

Submissions:

Submission Selection

1 Submission [active] 2/4/2024 8:14:29 PM

Instructions

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)
10Check that git sees it via `git status`
11If 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)
12Take 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

● Problem 1 (3 pts.)

^ COLLAPSE ^

● ^ 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

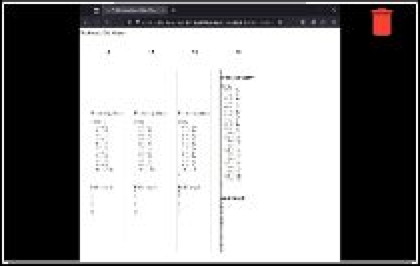
Task Screenshots:

Large Gallery

Checklist Items (3)

#1 Edits were done only in the processArray() function and original template code/comments remain untouched

#2 Only $arr is used (no direct usage of $a1, $a2, $a3, $a4)



Checklist Items (2)

#3 Only odd values output (not odd indexes/keys)

#5 Output of code includes heroku dev URL with student's ucid visible

Problem 1 code results with Heroku link

#4 Includes code comments with student's ucid and date

Problem 1 code.

● ^ 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:

To implement this problem, I used a for each loop to go through each element in the given array. Each element would be checked to see if it is used using the modulus operator, checking if the remainder is 1 where the division is 2, to see if they are odd. The formula would look something like if $element%2 == 1. If they do have a remainder of 1, then echo the specific element plus a <br> for cosmetics.

● ^ 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://ekh3-it202-008-prod-e9bbcd10cf36.herokuapp.com/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 (4)

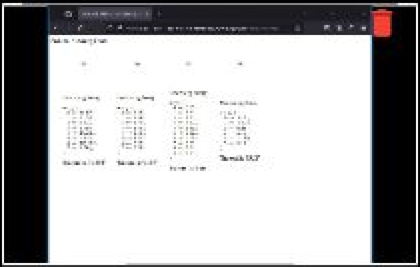
#1 Edits were done only in the getTotal() function and original template code/comments remain untouched (unless noted)

#2 Only $arr is used (no direct usage of $a1, $a2, $a3, $a4)

#3 Passed in array's values get summed AND rounded to two decimal places like currency (i.e., 0.00, 0.10, 1.10)

#4 Includes code comments with student's ucid and date

Problem 2 code containing comments and logic.



Checklist Items (1)

#5 Output of code includes heroku dev URL with student's ucid visible

Problem 2: Heroku results in dev site.

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, the goal is to cycle through all the values adding them to the total then after which round it. To sum up all the values, a for each loop is used to iterate through all the values and add them to the total variable provided. This is done by adding the current value of the total to the new element. Rounding would use the math function round() with the total and a precision of 2 to get the 2 decimal places, then using the format-number function, the output will be a 2 decimal place string. The result is saved back into the total to be displayed by the given echo.

● ^ 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://ekh3-it202-008-prod-e9bbcd10cf36.herokuapp.com/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)

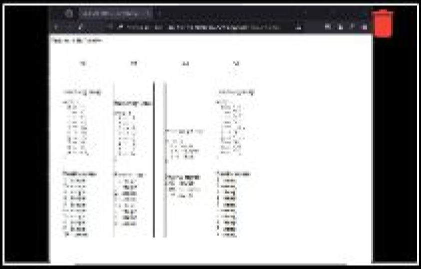
Checklist *The checkboxes are for your own tracking

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 (1)

#5 Output of code includes heroku dev URL with student's ucid visible

Question 3 results in Herokku dev

Checklist Items (4)

#1 Edits were done only in the bePositive() function and original template code/comments remain untouched

#2 Only $arr is used (no direct usage of $a1, $a2, $a3, $a4)

#3 Passed in array's values will get converted to a positive version AND converted back to the original data type

#4 Includes code comments with student's ucid and date

Problem 3: Code

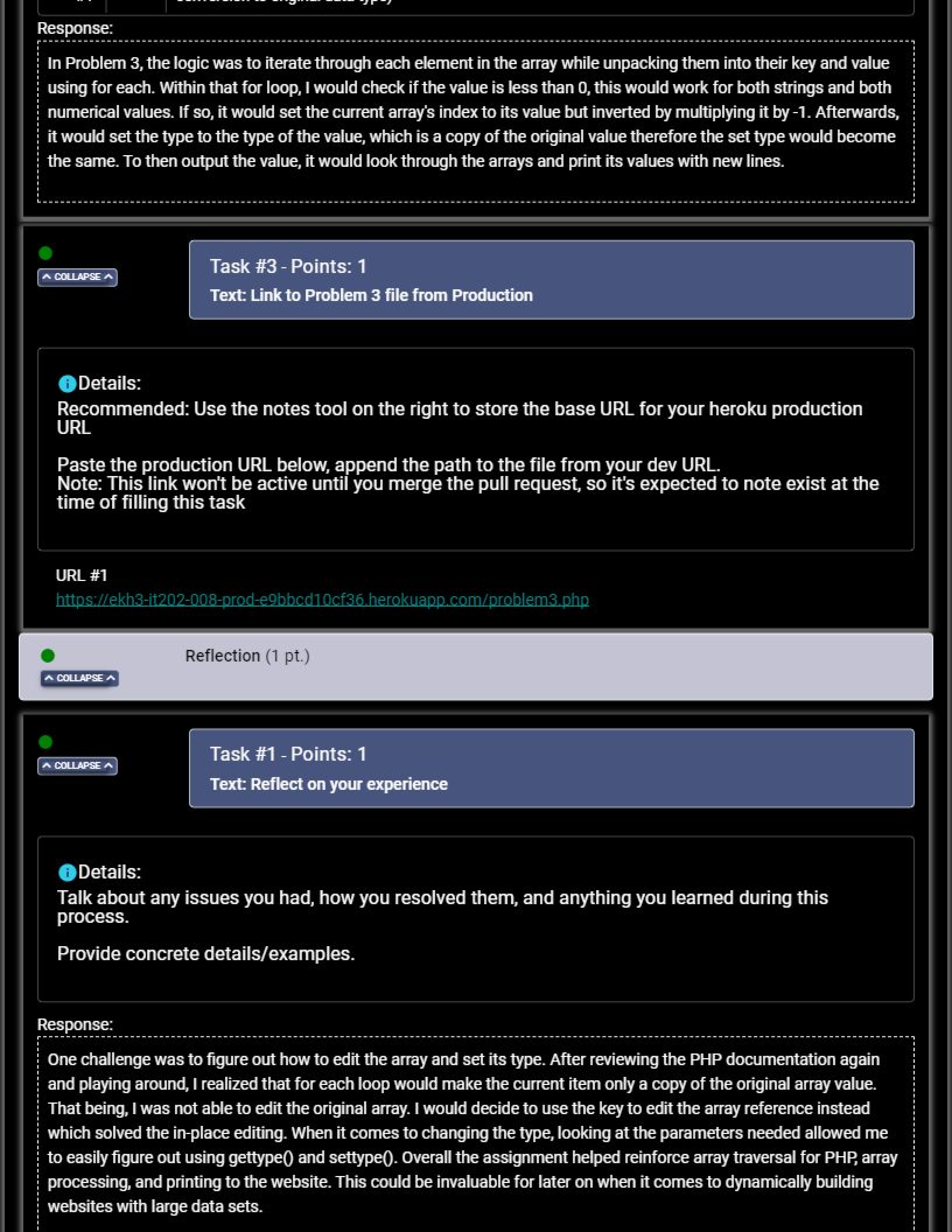
● COLLAPSE

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)



#4 conversion to original data type)

Response:

In Problem 3, the logic was to iterate through each element in the array while unpacking them into their key and value using for each. Within that for loop, I would check if the value is less than 0, this would work for both strings and both numerical values. If so, it would set the current array's index to its value but inverted by multiplying it by -1. Afterwards, it would set the type to the type of the value, which is a copy of the original value therefore the set type would become the same. To then output the value, it would look through the arrays and print its values with new lines.

● 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://ekh3-it202-008-prod-e9bbcd10cf36.herokuapp.com/problem3.php

● Reflection (1 pt.)

● Task #1 - Points: 1

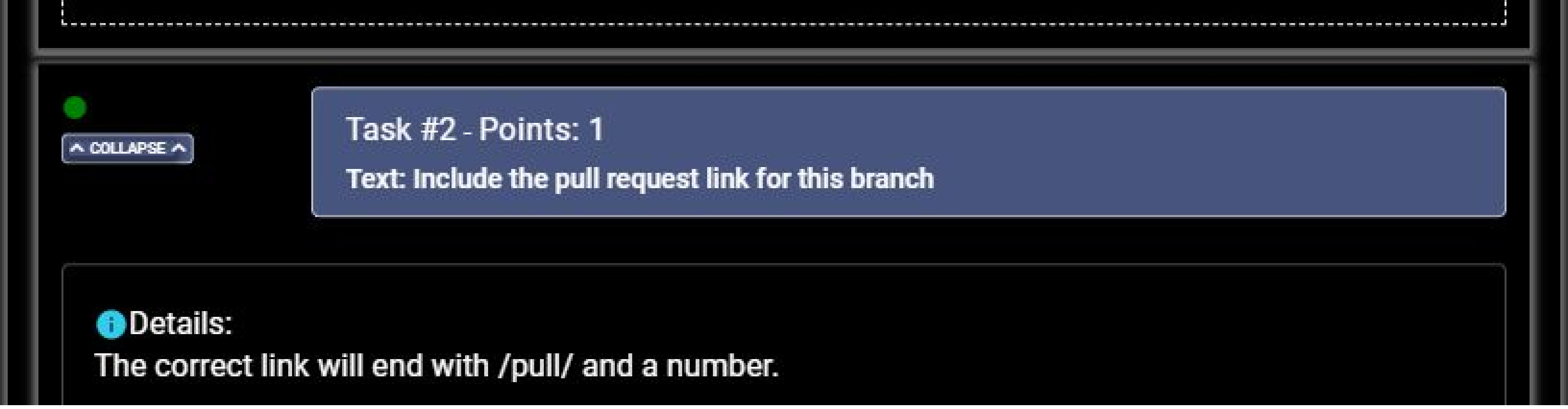
Text: Reflect on your experience

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

Provide concrete details/examples.

Response:

One challenge was to figure out how to edit the array and set its type. After reviewing the PHP documentation again and playing around, I realized that for each loop would make the current item only a copy of the original array value. That being, I was not able to edit the original array. I would decide to use the key to edit the array reference instead which solved the in-place editing. When it comes to changing the type, looking at the parameters needed allowed me to easily figure out using gettype() and settype(). Overall the assignment helped reinforce array traversal for PHP, array processing, and printing to the website. This could be invaluable for later on when it comes to dynamically building websites with large data sets.



Task #2 - Points: 1

Text: Include the pull request link for this branch

Details:

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