

Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT202-452-M2024/it202-module-2-php-problems/grade/dr475>

IT202-452-M2024 - [IT202] Module 2 PHP Problems

Submissions:

Submission Selection

1 Submission [active] 5/27/2024 12:29:16 PM

Instructions

^ COLLAPSE ^

Overview video: <https://youtu.be/aUn31pOxxJo> Guide:

1. Make sure you're in the dev branch locally and `git pull origin dev` 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 branch_name` (see below)
 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`

1. Push the changes 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.

Branch name: M2-PHP-Problems

Tasks: 5 Points: 10.00

Problem 1 (3 pts.)

^COLLAPSE ^

Task #1 - Points: 1

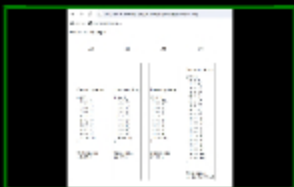
Text: Problem 1 Evidence

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)
Live URL must be Heroku Prod

#1) Show the output from heroku dev



Caption (required) ✓

Describe/highlight what's being shown
Heroku dev deployment showing problem 1 being solved

#2) Show the code solution



Caption (required) ✓

Describe/highlight what's being shown
The edited code to solve problem 1

Explanation (required)

URL (required) ✓

URL must be Heroku prod

<https://it202-dr475-prod-7559be2e2ad4.herokuapp.com/>

Explanation (required) ✓

Explain in concise steps how this logically works
I created a foreach loop to iterate over the passed array \$arr, assigning each value to the variable "\$item". \$item is only printed out if the value modulus 2 is not 0 (since if the value % 2 is 0, it is even).

Problem 2 (3 pts.)

^COLLAPSE ^

Task #1 - Points: 1

Text: Problem 2 Evidence

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)

Live URL must be Heroku Prod

#1) Show the output from heroku dev



id	name	age	weight
1	John	25	180
2	Jane	30	150
3	Bob	22	160
4	Alice	28	140
5	Charlie	35	170

Sum: 700

Caption (required) ✓

Describe/highlight what's being shown

Heroku dev deployment showing problem 2 being solved

URL (required) ✓

#2) Show the code solution



```
$arr = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100];  
$sum = 0;  
foreach ($arr as $item) {  
    if ($item % 2 != 0) {  
        $sum += $item;  
    }  
}  
echo "Sum: " . $sum . "  
";
```

Caption (required) ✓

Describe/highlight what's being shown

The edited code to solve problem 2

Explanation (required)

✓

URL must be Heroku prod
<https://it202-dr475-prod-7559be2e2ad4.herokuapp.com/>

Explain in concise steps how this logically works
I created a foreach loop to iterate over the passed array \$arr, assigning each value to the variable "\$item". Each iteration adds the \$item value to the \$total variable. After the for loop concludes, I updated the \$total variable by formatting the initial total to be rounded to only 2 decimal places.

Problem 3 (3 pts.)

^COLLAPSE ^

Task #1 - Points: 1

Text: Problem 3 Evidence

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)
Live URL must be Heroku Prod

#1) Show the output from heroku dev



Caption (required) ✓

Describe/highlight what's being shown
Heroku dev deployment showing problem 3

#2) Show the code solution



Caption (required) ✓

Describe/highlight what's being shown
The edited code to solve problem 3

being solved

URL (required) ✓

URL must be Heroku
prod

[https://it202-dr475-
prod-7559be2e2ad4.herokuapp.com/](https://it202-dr475-prod-7559be2e2ad4.herokuapp.com/)

Explanation (required)

✓

*Explain in concise steps
how this logically works*

I created a for loop to iterate over the length of the passed array \$arr, with each index valued at \$i starting at 0. I set two variables \$positive which finds the absolute value (positive value) of the array variable at index \$i, and \$type which gets the type ('string', 'decimal', 'integer') of the variable in the array. I then created a switch statement that has different logic for the type string, and type numerical value (double, integer). For string, \$output[\$i] would be the strval(\$positive) (sets it to a string type), and for numerical values \$output[\$i] would be \$positive (either a decimal or integer).



Reflection (1 pt.)

^COLLAPSE ^



Task #1 - Points: 1

Text: Reflect on your experience

^COLLAPSE ^

i Details:

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

Provide concrete details/examples. At least a few sentences.

Response:


I had no issues while developing other than remembering and re-learning PHP syntax and its helper functions. This includes: `number_format()`, `gettype()`, `strval()`, and `count()`;



^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.

URL #1

<https://github.com/davidredziniak/dr475-IT202-452-M2024/pull/4>

End of Assignment