

# Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT202-452-M2024/it202-module-2-php-problems/grade/sha38>

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

## Submissions:

Submission Selection

1 Submission [active] 5/31/2024 4:28:59 PM

## Instructions

^ COLLAPSE ^

Overview video: [https://youtu.be/Vvx\\_cfdLqgM](https://youtu.be/Vvx_cfdLqgM)

## 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`.
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: 6 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 Herokue Prod

#1) Show the output from heroku dev (url must be visible)



sha38-dev-2a73c56b3a2.herokuapp.com/M2/problem1.php

Problem 1: Odd Output

A1	A2	A3	A4
Processing Array: array (0 => 0, 1 => 1, 2 => 2, 3 => 3, 4 => 4, 5 => 5, 6 => 6, 7 => 7, 8 => 8, 9 => 9, 10 => 10) Odds output: 1 3 5 7 9	Processing Array: array (0 => 0, 1 => 1, 2 => 2, 3 => 3, 4 => 4, 5 => 5, 6 => 6, 7 => 7, 8 => 8, 9 => 9, 10 => 10) Odds output: 1 3 5 7 9	Processing Array: array (0 => 0, 1 => 1, 2 => 2, 3 => 3, 4 => 4, 5 => 5, 6 => 6, 7 => 7, 8 => 8, 9 => 9, 10 => 10) Odds output: 1 3 5 7 9	Processing Array: array (0 => 0, 1 => 1, 2 => 2, 3 => 3, 4 => 4, 5 => 5, 6 => 6, 7 => 7, 8 => 8, 9 => 9, 10 => 10, 11 => 11, 12 => 12, 13 => 13, 14 => 14, 15 => 15, 16 => 16, 17 => 17, 18 => 18, 19 => 19, 20 => 20, 21 => 21) Odds output: 1 1 3 3 5 5 7 7 9 9

**Caption (required)** ✓

***Describe/highlight what's being shown***

**Shows output from heroku dev**

**URL (required)** ✓

*URL must be Heroku prod*

<https://sha38-prod-ae798a1dca98.herokuapp.com/M2/problem1.php>

#2) Show the code solution (ucid/date as comment must be present)



**Caption (required)** ✓

***Describe/highlight what's being shown***

**Shows code solution around line 13**

**Explanation (required)** ✓

***Explain in concise steps how this logically works***

 **EDIT RESPONSE**

We start by iterating over each number in the array. We are taking each number from the array and dividing by 2 and getting the remainder. The '!=0' check if the remainder is not equal to zero, if remainder is not zero, the number is odd. If the number is odd (true), it will print out the number.

### Problem 2 (3 pts.)

^ COLLAPSE ^

### Task #1 - Points: 1

### Text: Problem 2 Evidence

^ COLLAPSE ^

## Details:

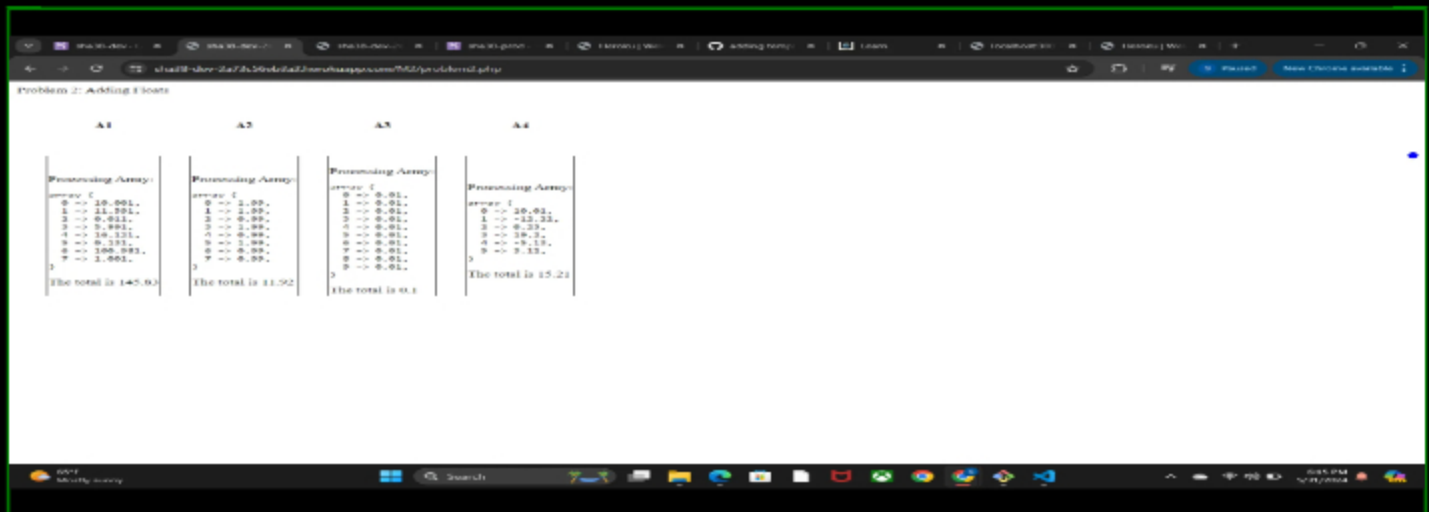
Only make edits where the template code mentions.

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 (url must be visible)



Caption (required) ✓

Describe/highlight what's being shown

Shows the output from heroku dev

URL (required) ✓

URL must be Heroku prod

<https://sha38-prod-ae798a1dca98.herokuapp.com/M2/problem2.php>

#2) Show the code solution (ucid/date as comment must be present)



```
$arr = [10.001, 1.002, 0.003, 0.004, 0.005, 0.006, 0.007, 0.008];
function getTotal($arr) {
    echo "<div>Processing Array:</div><br><div><pre></pre></div>";
    $total = 0.00;
    //start edits
    //note: use the $arr variable, don't directly touch $a1 $a4
    //TODO do adding here //you can mention, you can mention changes
    foreach ($arr as $value) {
        $total += $value;
    }
    //TODO do rounding stuff here (round to two decimals i.e., 0.10, 0.01, 0.00)
    $total = round($total, 2);
    //end edits
    echo "The total is $total";
}
echo "Problem 2: Adding Floats<br>";
//
<table>
<thead>
```

### Caption (required) ✓

Describe/highlight what's being shown

Shows code solution at line 13.

### Explanation (required) ✓

Explain in concise steps how this logically works

 EDIT RESPONSE

We start by going through each element in the array. We then take each element and add them all together. That value becomes our total. We then take the total and round it to two 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 (url must be visible)



cha38-dev-2a73c56eb3a2.herokuapp.com/M2/problem3.php

Problem 3: Re Positive

A1	A2	A3	A4
Processing Array: array ( 0 => -1, 1 => -2, 2 => -2, 3 => -5, 4 => 5, 5 => -6, 6 => -7, 7 => -8, 8 => -9, 9 => -10, )	Processing Array: array ( 0 => -1, 1 => -2, 2 => -2, 3 => -5, 4 => 5, 5 => -6, 6 => -7, 7 => -8, 8 => -9, 9 => -10, )	Processing Array: array ( 0 => -8.81, 1 => -0.0001, 2 => -0.12, )	Processing Array: array ( 0 => -1, 1 => -2, 2 => -2, 3 => -5, 4 => 5, 5 => -6, 6 => -7, 7 => -8, 8 => -9, 9 => -10, )
Positive output: 1 (D), 2 (D), 3 (D), 4 (D), 5 (D), 6 (D), 7 (D), 8 (D), 9 (D), 10 (D)	Positive output: 1 (D), 2 (D), 3 (D), 4 (D), 5 (D)	Positive output: 0.91 (D), 0.0001 (D), 0.15 (D)	Positive output: 1 (S), 2 (S), 3 (S), 4 (S), 5 (S), 6 (S), 7 (S), 8 (S), 9 (S), 10 (S)

### Caption (required) ✓

Describe/highlight what's being shown

Describe/highlight what's being shown  
Shows output from heroku dev

URL (required) ✓

URL must be Heroku prod

<https://sha38-prod-ae798a1dca98.herokuapp.com/M2/problem3.php>

#2) Show the code solution (ucid/date as comment must be present)



```
1 <?php
2 // Problem 3
3 // User: sha38
4 // Date: 2023-04-10
5
6 // Function to make an array of numbers positive
7 function makePositive($arr) {
8     // Create a new array to store the positive values
9     $positiveValues = [];
10
11     // Loop through the input array and calculate the absolute value of each element
12     foreach ($arr as $value) {
13         // Calculate the absolute value of the element
14         $positiveValue = abs($value);
15
16         // Add the positive value to the new array
17         $positiveValues[] = $positiveValue;
18     }
19
20     // Return the array of positive values
21     return $positiveValues;
22 }
23
24 // Test the function
25 $arr = [-1, 2, -3, 4, -5, 6, -7, 8, -9, 10];
26 $positiveValues = makePositive($arr);
27
28 // Display the output as a JSON array
29 echo json_encode($positiveValues);
30
31 // End of file
```

Caption (required) ✓

Describe/highlight what's being shown

Shows code solution at line 15

Explanation (required) ✓

Explain in concise steps how this logically works

[EDIT RESPONSE](#)

We go through every element in the array and turn those values into positive. If original value of element is a string, it will be cast back to a string before adding to output, but if it is not a string, it is directly added to output. Each value is then printed with it's type.

Reflection (1 pt.)

[COLLAPSE](#)

Task #1 - Points: 1

Text: Reflect on your experience

**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 didn't really have any issues other forgetting to save files so it can be updated on the localhost. I just really noticed how everything seem to connected one to another, like git to vscode, then to github and heroku.



^ COLLAPSE ^

Task #2 - Points: 1

Text: Include the pull request link for this branch

**i** Details:

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

URL #1

<https://github.com/ShahadatAnadil/sha38-it202-452/pull/4>



^ COLLAPSE ^

Task #3 - Points: 1

Text: Add Screenshot of Wakatime

**i** Details:

Note: The duration of time isn't directly related to the grade, the goal is to just make sure time is being tracked

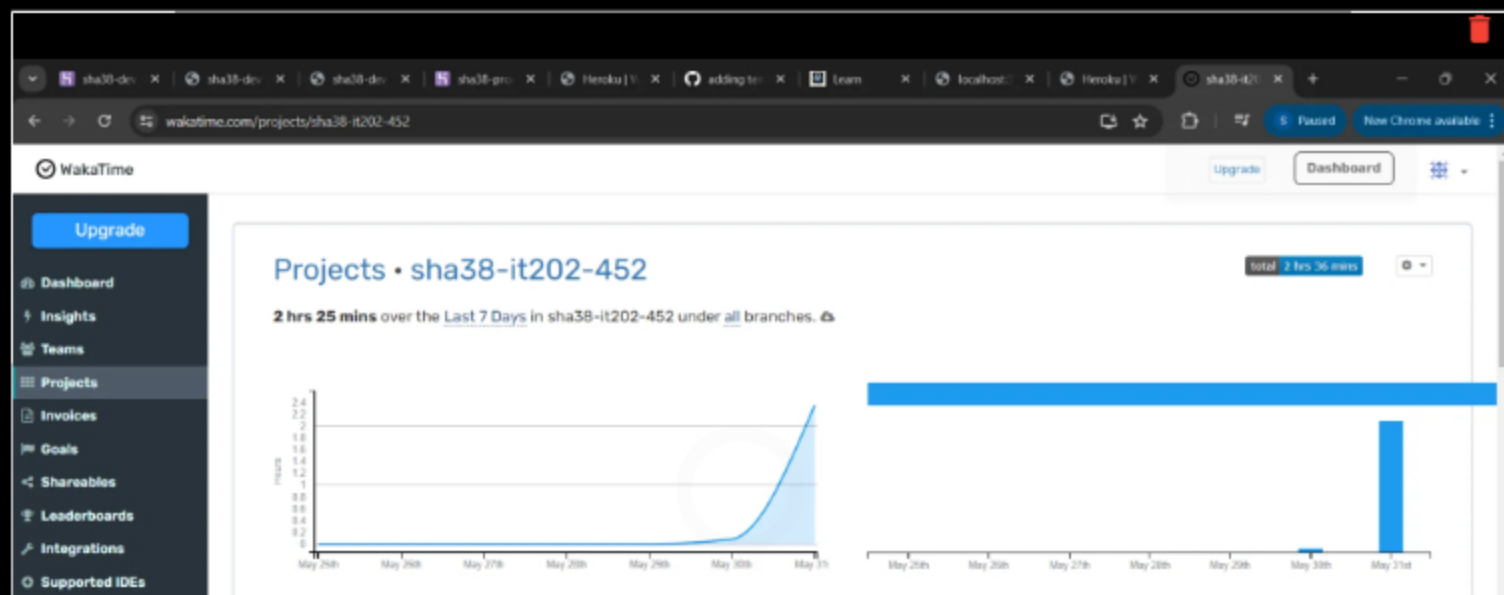
Task Screenshots:

Gallery Style: Large View

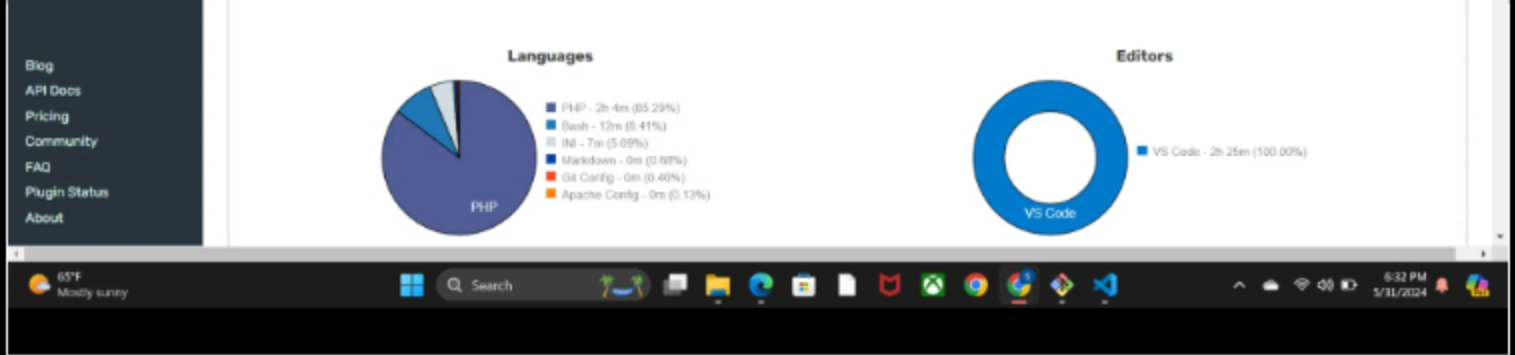
Small

Medium

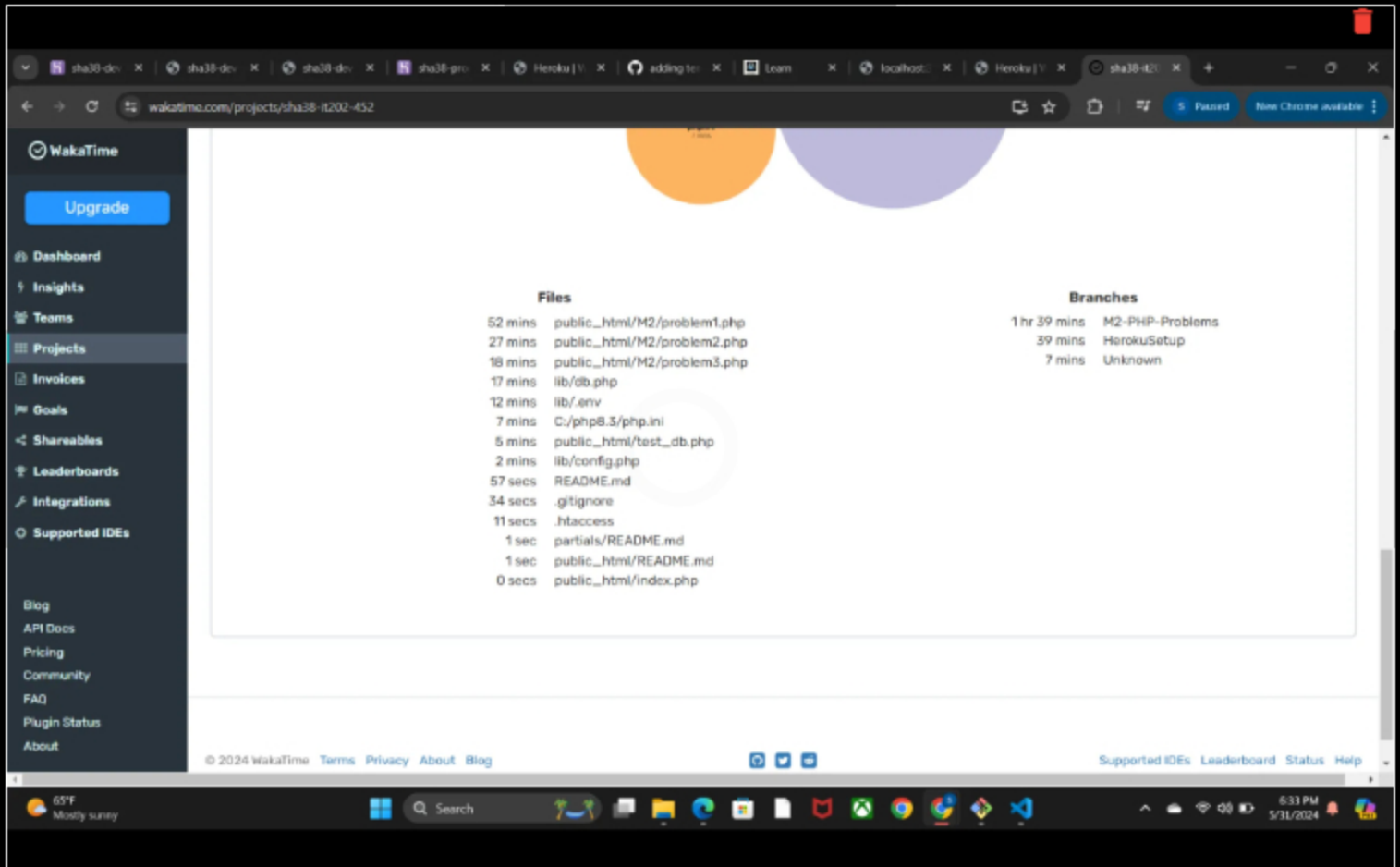
Large







showing overview time



showing specific time

End of Assignment