

# 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. Check the status via `git status`.
  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 ^



^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 (url must be visible)



**Caption (required)**

*Describe/highlight what's being shown*

Missing caption

**URL (required)**

*URL must be Heroku prod*

Missing URL

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



**Caption (required)**

*Describe/highlight what's being shown*

Missing caption

**Explanation (required)**

*Explain in concise steps how this logically works*

Missing text



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)



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*

Missing caption

**URL (required)**

*URL must be Heroku prod*

Missing URL

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



**Caption (required)**

*Describe/highlight what's being shown*

Missing caption

**Explanation (required)**

*Explain in concise steps how this logically works*

Missing text

**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 Herokuue Prod

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



**Caption (required)**

*Describe/highlight what's being shown*

Missing caption

**URL (required)**

*URL must be Heroku prod*

Missing URL





**Caption (required)**

*Describe/highlight what's being shown*

Missing caption

**Explanation (required)**

*Explain in concise steps how this logically works*

Missing text

● Reflection (1 pt.)

^COLLAPSE ^

● Task #1 - Points: 1

^COLLAPSE ^

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 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();

● Task #2 - Points: 1

^COLLAPSE ^

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/davidredziniak/dr475-IT202-452-M2024/pull/8>

End of Assignment