

Submission Worksheet

Submission Data

Course: IT202-450-M2025

Assignment: IT202 PHP Problems

Student: undefined (may23)

Status: Submitted | **Worksheet Progress:** 100+%

Potential Grade: 10.93/10.00 (109.30%)

Received Grade: 0.00/10.00 (0.00%)

Started: 6/9/2025 5:31:10 PM

Updated: 6/9/2025 6:54:56 PM

Grading Link: <https://learn.ethereallab.app/assignment/v3/IT202-450-M2025/it202-php-problems/grading/may23>

View Link: <https://learn.ethereallab.app/assignment/v3/IT202-450-M2025/it202-php-problems/view/may23>

Instructions

- Overview Link: <https://youtu.be/iwA6vyRPhoM>
- 1. Ensure you read all instructions and objectives before starting.
- 2. Create a new branch from dev called M2-Homework
 - 1. `git checkout dev` (ensure proper starting branch)
 - 2. `git pull origin dev` (ensure history is up to date)
 - 3. `git checkout -b M2-Homework` (create and switch to branch)
- 3. Copy the template code from here: [GitHub Repository - M2 Homework](#)
 - It includes Problems 1-4 and `base.php`. Put all into an M2 folder or similar inside your `public_html`
 - Immediately record to history
 - `git add public_html`
 - `git commit -m "adding M2 HW baseline files"`
 - `git push origin M2-Homework`
 - Create a Pull Request from M2-Homework to dev and keep it open
- 4. Fill out the below worksheet
 - Each Problem requires the following as you work
 - Ensure there's a comment with your UCID, date, and brief summary of how the problem was solved
 - Initial outline/plan of how you'll solve it via comments (add/commit after this stage)
 - Code solution (add/commit periodically as needed)
- 5. Once finished, click "Submit and Export"
- 6. Locally add the generated PDF to a folder of your choosing inside your repository folder and move it to Github
 - 1. `git add .`
 - 2. `git commit -m "adding PDF"`
 - 3. `git push origin M2-Homework`
 - 4. On Github merge the pull request from M2-Homework to dev
 - 5. On Github create a pull request from dev to prod and immediately merge. (This will trigger the prod deploy to make the heroku prod links work)
- 7. Upload the same PDF to Canvas
- 8. Sync Local

1. git checkout dev
2. git pull origin dev

Section #1: (2 pts.) Problem 1 - Odds

Progress: 100%

≡ Task #1 (2 pts.) - Edit the `printOdds` function to output odd values of the array

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Challenge: Print odd values only in a single line separated by commas
- Step 1: sketch out plan using comments (include ucid and date)
- Step 2: Add/commit your outline of comments (required for full credit)
- Step 3: Add code to solve the problem (add/commit as needed)

Part 1:

Progress: 100%

Details:

Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program (visit the proper file on Heroku dev after a manual deploy)

```
#!/bin/bash
# Task #1: Print odd values of an array
# UCID: may23, Date: 6/9/2025
# Plan:
# - Loop through the array
# - Check if each number is odd (i.e. number % 2 != 0)
# - Store odd numbers in a new array
# - Join the odd numbers with commas and print them
// Start Solution Edit
isOdd() {
    local number=$1
    if [ $(($number % 2)) != 0 ]; then
        return 0
    else
        return 1
    fi
}

# Loop through the array
for number in "${array[@]}"; do
    if isOdd $number; then
        oddNumbers+=("$number")
    fi
done

# Print the odd numbers with commas
IFS=,
echo "${oddNumbers[@]}"
```

code1

```
Completed Problem 1 for [may23] [2025-06-09 22:51:05]
Challenge: Print out only odd values in a single line separated by commas
Problem 1
Original Array: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Current Array: 1 3 5 7 9 11 13 15 17 19
Problem 2
Original Array: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Current Array: 1 3 5 7 9 11 13 15 17 19
Problem 3
Original Array: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
Current Array: 1 3 5 7 9 11 13 15 17 19
```

problem1

Part 2:

Progress: 100%

Details:

- Direct link to the file in the homework related branch from Github (should end in `.php`)
- Direct link to the file on Heroku Prod (Just grab the base prod url and manually enter the path to the file)

URL #1

<https://may23-it202-450-dev-70e1264ec6c5.herokuapp.com/problem1.php>



URL

<https://may23-it202-450-dev-70e1>



Saved: 6/9/2025 6:51:22 PM

Part 3:

Progress: 100%

Details:

Briefly explain `how` the code solves the challenge (note: this isn't the same as `what` the code does)

Your Response:

The code loops through each number in the array and uses % sign to check whether it's odd or not.



Saved: 6/9/2025 6:51:22 PM

Section #2: (2 pts.) Problem 2 - Sum

Progress: 100%

Task #1 (2 pts.) - Edit the `sumValues` function to sum the array values and present them in a format with exactly two decimal places

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Challenge 1: Sum all the values of the passed in array and assign to `total`
- Challenge 2: Have the sum be represented as a number with exactly 2 decimal
- Example: 0.1 would be shown as 0.10, 1 would be shown as 1.00, etc
- Step 1: sketch out plan using comments (include uid and date)
- Step 2: Add/commit your outline of comments (required for full credit)
- Step 3: Add code to solve the problem (add/commit as needed)

Step 6: Add code to solve the problem (add, commit as needed).

Details:

Briefly explain **how** the code solves the challenges (note: this isn't the same as **what** the code does)

Your Response:

The code iterates through each number in the array using a foreach loop and adds each value to a running total (\$total). After computing the total, the code calls number_format(\$total, 2) to produce a string representation of the number with exactly 2 decimal places.



Saved: 6/9/2025 6:50:50 PM

Section #3: (2 pts.) Problem 3 - Conversion

Progress: 100%

≡ Task #1 (2 pts.) - Edit the `bePositive` function to make each value positive, convert it back to the original data type, and set it to the proper slot in the `output` array

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Challenge 1: Make each value positive
- Challenge 2: Convert the values back to their original data type and assign it to the proper slot of the **output** array
- Step 1: sketch out plan using comments (include ucid and date)
- Step 2: Add/commit your outline of comments (required for full credit)
- Step 3: Add code to solve the problem (add/commit as needed)

Part 1:

Progress: 100%

Details:

Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program (visit the proper file on Heroku dev after a manual deploy)



problem3

code3

Saved: 6/9/2025 6:50:24 PM

Progress: 100%

- Direct link to the file in the homework related branch from Github (should end in `.php`)
- Direct link to the file on Heroku Prod (Just grab the base prod url and manually enter the path to the file)

<https://may23-it202-450-dev-70e1>

Saved: 6/9/2025 6:50:24 PM

Progress: 100%

Briefly explain **how** the code solves the challenges (note: this isn't the same as **what** the code does)

The code uses `abs($val)` to convert each element to its absolute value. This works correctly whether the value is a number or a numeric string. To ensure the transformed value has the same type as the original, the code uses `gettype()` to detect whether the original input was a string, integer, or float. It then explicitly casts the positive value back to that type before storing it.

Saved: 6/9/2025 6:50:24 PM

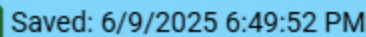
Progress: 98%

Progress: 97%

Progress: 91%

code4

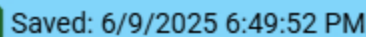
Missing Caption



Progress: 100%

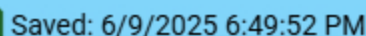
- Direct link to the file in the homework related branch from Github (should end in `.php`)
- Direct link to the file on Heroku Prod (visit the proper file on Heroku dev after a manual deploy)

<https://may23-it202-450-dev-70e1>



Progress: 100%

The code uses a regular expression with `preg_replace()` to eliminate all characters that are not letters, numbers, or spaces. The cleaned string is lowercased and then passed to `ucwords()`, which capitalizes the first letter of each word. The code uses `trim()` to remove leading/trailing whitespace and another regex to collapse multiple spaces into a single space.



- Only make edits where noted via provided comments

Progress: 100%

Details:

Briefly explain **how** the code solves the extra credit challenge (note: this isn't the same as **what** the code does)

Your Response:

The code removes the first and last characters of the cleaned phrase and then uses `substr()` to extract the 3 characters around the center – if the remaining string is long enough.



Saved: 6/9/2025 6:49:18 PM

Section #5: (2 pts.) Misc

Progress: 100%

☰ Task #1 (0.67 pts.) - Github Details

Progress: 100%

Part 1:

Progress: 100%

Details:

From the Commits tab of the Pull Request screenshot the commit history Following minimum should be present



Saved: 6/9/2025 6:45:30 PM

🔗 Part 2:

Progress: 100%

Details:

Include the link to the Pull Request (should end in `/pull/#`)

URL #1

<https://github.com/Mark-5555/may23-IT202x458/>



URL

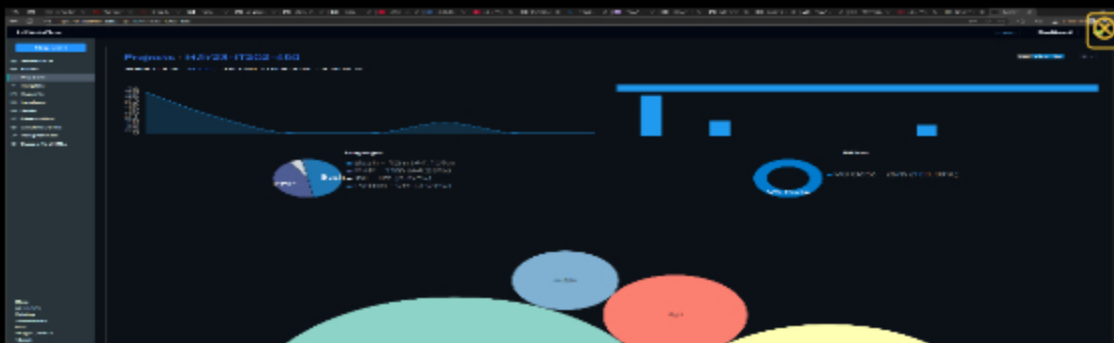
<https://github.com/Mark-5555/m>

Task #2 (0.67 pts.) - WakaTime - Activity

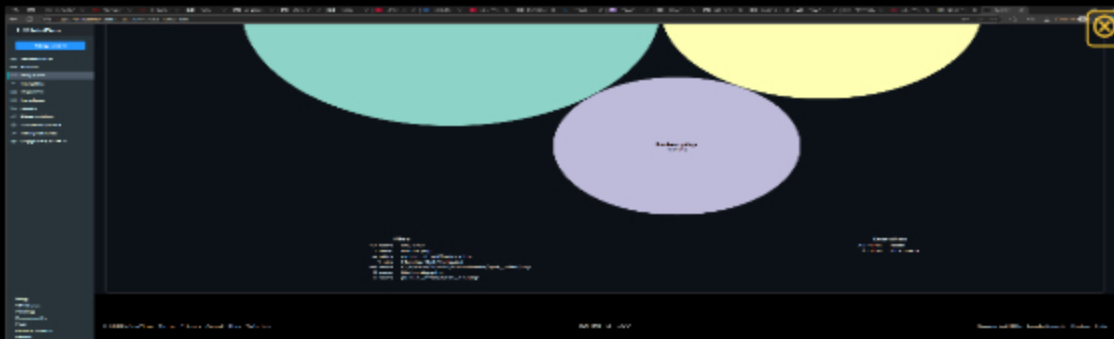
Progress: 100%

Details:

- Visit the WakaTime.com Dashboard
- Click **Projects** and find your repository
- Capture the overall time at the top that includes the repository name
- Capture the individual time at the bottom that includes the file time
- Note: The duration isn't relevant for the grade and the visual graphs aren't necessary



waka1



waka2

Saved: 6/9/2025 6:52:39 PM

Task #3 (0.67 pts.) - Reflection

Progress: 100%

Task #1 (0.33 pts.) - What did you learn?

Progress: 100%

Details:

Briefly answer the question (at least a few decent sentences)

Your Response:

I learnt how to write a php script and render that code on the browser, as well as

I learnt how to write a php script and render that code on the browser, as well as working in a github branch and merging that branch as needed.



Saved: 6/9/2025 6:53:50 PM

⇒ Task #2 (0.33 pts.) - What was the easiest part of the assignment?

Progress: 100%

Details:

Briefly answer the question (at least a few decent sentences)

Your Response:

Frankly it was typing the code.



Saved: 6/9/2025 6:54:10 PM

⇒ Task #3 (0.33 pts.) - What was the hardest part of the assignment?

Progress: 100%

Details:

Briefly answer the question (at least a few decent sentences)

Your Response:

Understanding how to render the code properly and merge branches, commit changes ...etc.



Saved: 6/9/2025 6:54:56 PM