Submission Worksheet

CLICK TO GRADE

https://learn.ethereallab.app/assignment/IT202-007-F2024/generic-module-5-multi-dimension-php-problems/grade/ass89

Course: IT202-007-F2024
Assigment: [Generic] Module 5 Multi-Dimension PHP Problems
Student: Alanor S. (ass89)

Submissions:

Submission Selection

1 Submission [submitted] 10/21/2024 6:53:12 PM

Instructions

COLLAPSE

Overview video: https://youtu.be/lp568G93Noo

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/f7b0489fb0d8cee615d6626056ac5de2
4. Create individual PHP files for each problem and save the files inside your `public_html` folder in a subfolder of your choice.
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 `Module5` 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., dev <- "homework branch").
5. Open and complete the merge of the pull request (it should turn purple).
6. Locally checkout dev and pull the latest changes (to prepare for future work).

12. Take the same output file and upload it to Canvas.

Branch name: M5-MD-PHP-Problems

Group

100%

Group: Problem 1
Tasks: 1
Points: 3

COLLAPSE

Task

100%

Group: Problem 1
Task #1: Problem 1 Evidence
Weight: ~100%
Points: ~3.00

COLLAPSE

Details:

Only make edits where the template code mentions.

Solution should add logic to create a new array with only name, color, and region (subset of the original data)

Columns: 1

Sub-Task

100%

Group: Problem 1
Task #1: Problem 1 Evidence
Sub Task #1: Show the output from heroku dev (url must be visible)

Task Screenshots

Gallery Style: 2 Columns

4 2 1

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

Caption(s) (required) ✓

Caption Hint: *Describe/highlight what's being shown*

Task URLs

URL #1

https://asa89-it202-007-dev-bef7e2e4f18c.herokuapp.com/Module5HW/Prob1.php

URL

https://asa89-it202-007-dev-bef7e2e4f18c.hero

Sub-Task

100%

Group: Problem 1

Task #1: Problem 1 Evidence

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

Task Screenshots

Gallery Style: 2 Columns

4 2 1

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

Caption(s) (required) ✓

Caption Hint: *Describe/highlight what's being shown*

Task Response Prompt

Explain in concise steps how this logically works

Response:

We're given a bunch of arrays representing different bird species and their attributes. Each bird is represented as an associative array with several attributes. We're also given an empty array called subset. The goal was to extract data (name, color, and region) from each bird and put it into the empty array. To do that, I iterated through each bird in the input array $birds using the foreach loop. Once the information was extracted, it was added into the subset array keys.

End of Task 1

End of Group: Problem 1

Task Status: 1/1

Group

100%

Group: Problem 2
Tasks: 1
Points: 3

COLLAPSE

Task

100%

Group: Problem 2
Task #1: Problem 2 Evidence
Weight: ~100%
Points: ~3.00

COLLAPSE

Details:
Only make edits where the template code mentions.

Solution should add logic to create a new array with original properties plus age and isClassic (extra data)

Columns: 1

Sub-Task

100%

Group: Problem 2
Task #1: Problem 2 Evidence
Sub Task #1: Show the output from heroku dev (url must be visible)

Task Screenshots

Gallery Style: 2 Columns

4 2 1

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

Caption(s) (required) ✓
Caption Hint: *Describe/highlight what's being shown*

Task URLs

URL #1
https://asa89-it202-007-dev-bef7e2e4f18c.herokuapp.com/Module5HW/Prob2.php

URL
https://asa89-it202-007-dev-bef7e2e4f18c.hero

bef7c2c4f10c.herokuapp.com/Module5HW/Prob2.php

Sub-Task

100%

Group: Problem 2
Task #1: Problem 2 Evidence
Sub Task #2: Show the code solution (ucid/date as comment must be present)

Task Screenshots

Gallery Style: 2 Columns

4 2 1

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

Caption(s) (required) ✓

Caption Hint: *Describe/highlight what's being shown*

Task Response Prompt

Explain in concise steps how this logically works

Response:

This one is basically the same as the first problem, except it build on it a bit more. We are once again given a bunch of arrays and the goal is to extract the data (make, model,and year) and to add the data(age, isClassic) to an empty array. We had to find the current year dynamically, so for this, I took advantage of PHP's date function and formatted it to the year (24). I then iterated through each car in the input array $cars. Once again, each car us an associative array with properties. The age had to be calculated and I did that by subtracting the car's manufacturing year, accessed by $car['year'] from the $currentYear variable. I then check to see if the car is considered a classic car (older than 25 years old). To do that, I compared its age(which was just calculated) to the $classic_age. It will return boolean logic based on the comparison results. Now that we have the data to input into the new array (besides the information to be gotten from iterating through the array), it is time to build a new associative array, called $processedCars. For each car, the information (make,model, year) along with the 2 additional properties (age, isClassic) is added to the $processedCars array.

End of Task 1

End of Group: Problem 2
Task Status: 1/1

Group

100%

Group: Problem 3
Tasks: 1
Points: 3

COLLAPSE

Task

100%

Group: Problem 3
Task #1: Problem 3 Evidence
Weight: ~100%
Points: ~3.00

COLLAPSE

Details:

Only make edits where the template code mentions.
Solution should add logic to join the arrays on userId
Requires at least 2 screenshots (code + output from heroku dev)
Live URL must be Herokue Prod

Columns: 1

Sub-Task

100%

Group: Problem 3
Task #1: Problem 3 Evidence
Sub Task #1: Show the output from heroku dev (url must be visible)

Task Screenshots

Gallery Style: 2 Columns

4 2 1

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

Caption(s) (required) ✓

Caption Hint: *Describe/highlight what's being shown*

Task URLs

URL #1

https://asa89-it202-007-dev-bef7e2e4f18c.herokuapp.com/Module5HW/Prob3.php

URL

https://asa89-it202-007-dev-bef7e2e4f18c.hero

Sub-Task

100%

Group: Problem 3
Task #1: Problem 3 Evidence
Sub Task #2: Show the code solution (ucid/date as comment must be present)

Task Screenshots

Gallery Style: 2 Columns

4 2 1

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

Caption(s) (required) ✓

Caption Hint: *Describe/highlight what's being shown*

Task Response Prompt

Explain in concise steps how this logically works

Response:

This one is slighly more complicated, in theory. We're given two sets of arrays. One set ($users) contains the user information(userId,name,age) and the second set ($activities) contains activities associated with each user, also tied to the userId. The goal here is to merge the information from these two sets of arays based on the common key userId. I first created an empty associative array called $activityArray, where the userId serves as the key and the corresponding activity serves as the value (essentially created a map for easy lookup). I then looped through the $users array to extract each user's userId. For each user, I did a lookup in the $activityArray to find the associated activity using the userId. I then merged the user data with the corresponding activity, creating a combined associative array for each user. Using the array_merge function, I merged the user's data (name,age, etc) with their activity. This merged array was then added to the $joined array.

End of Task 1

End of Group: Problem 3
Task Status: 1/1

Group

100%

Group: Reflection
Tasks: 3
Points: 1

COLLAPSE

Task

100%

Group: Reflection
Task #1: Reflect on your experience
Weight: ~33%

Points: ~0.33

COLLAPSE

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.

Task Response Prompt

Response:

I think the hardest part was figuring out the logic for the last question. My initial train of thought was to make nested loops, which was more complicated than the result I ended up with. The key, quite literally, was to create a map. I also wanted to make sure I wasn't overwriting information from the arrays when I merged them, which was another issue I had. By explicitly adding the activity as a key value pair (['activity' => $activityArray[$userId]]), that issue was resolved.

End of Task 1

Task

100%

Group: Reflection
Task #2: Include the pull request link for this branch
Weight: ~33%
Points: ~0.33

COLLAPSE

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

Task URLs

URL #1
https://github.com/AlCSS442/asa89-IT202-007/pull/11

URL
https://github.com/AlCSS442/asa89-IT202-007/p

End of Task 2

Task

100%

Group: Reflection
Task #3: Add Screenshot of Wakatime
Weight: ~33%
Points: ~0.33

COLLAPSE

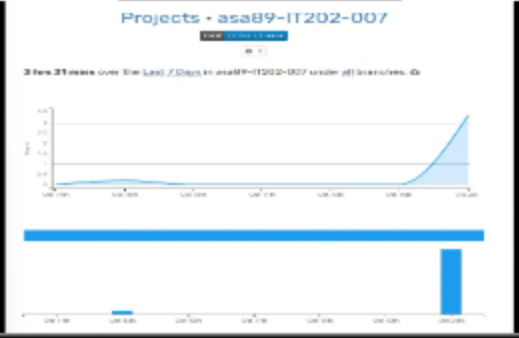
Details:

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: 2 Columns

4 2 1



Add Screenshot of Wakatime

wakatime pt2

End of Task 3

End of Group: Reflection
Task Status: 3/3

End of Assignment