

Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT202-452-M2024/generic-module-5-multi-dimension-php-problems/grade/mrs43>

IT202-452-M2024 - [Generic] Module 5 Multi-Dimension PHP Problems

Submissions:

Submission Selection

1 Submission [active] 6/24/2024 6:42:48 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

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

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)



A1	A2	A3	A4
Processing Array: <pre>array (0 => array ('id' => 1, 'name' => 'Sparrow', 'size' => 'small', 'color' => 'brown', 'region' => 'North America',), 1 => array ('id' => 2, 'name' => 'Robin', 'size' => 'small', 'color' => 'red', 'region' => 'Europe',),)</pre> Subset output: <pre>array (0 => array ('name' => 'Sparrow', 'color' => 'brown', 'region' => 'North America',), 1 => array ('name' => 'Robin', 'color' => 'red', 'region' => 'Europe',),)</pre>	Processing Array: <pre>array (0 => array ('id' => 3, 'name' => 'Eagle', 'size' => 'large', 'color' => 'brown', 'region' => 'Worldwide',), 1 => array ('id' => 4, 'name' => 'Parrot', 'size' => 'medium', 'color' => 'green', 'region' => 'Tropical',),)</pre> Subset output: <pre>array (0 => array ('name' => 'Eagle', 'color' => 'brown', 'region' => 'Worldwide',), 1 => array ('name' => 'Parrot', 'color' => 'green', 'region' => 'Tropical',),)</pre>	Processing Array: <pre>array (0 => array ('id' => 5, 'name' => 'Penguin', 'size' => 'medium', 'color' => 'black and white', 'region' => 'Antarctica',), 1 => array ('id' => 6, 'name' => 'Flamingo', 'size' => 'large', 'color' => 'pink', 'region' => 'Africa',),)</pre> Subset output: <pre>array (0 => array ('name' => 'Penguin', 'color' => 'black and white', 'region' => 'Antarctica',), 1 => array ('name' => 'Flamingo', 'color' => 'pink', 'region' => 'Africa',),)</pre>	Processing Array: <pre>array (0 => array ('id' => 7, 'name' => 'Owl', 'size' => 'medium', 'color' => 'white', 'region' => 'Worldwide',), 1 => array ('id' => 8, 'name' => 'Hummingbird', 'size' => 'small', 'color' => 'varied', 'region' => 'Americas',),)</pre> Subset output: <pre>array (0 => array ('name' => 'Owl', 'color' => 'white', 'region' => 'Worldwide',), 1 => array ('name' => 'Hummingbird', 'color' => 'varied', 'region' => 'Americas',),)</pre>

Caption (required) ✓

Describe/highlight what's being shown

Screenshot of problem 1 from heroku dev

URL (required) ✓

URL must be Heroku prod

<https://mrs43-prod-5f0e0e79560e.herokuapp.com/M5/problem1.php>

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



```
// Note: use the $birds variable to iterate over, don't directly touch $a1-$a4
// TODO add logic here to create a new array with only name, color, and region
$subset = []; // result array
// Start edits
//mrs43 6-24-2024
foreach($birds as $bird){
    foreach($bird as $key => $value){
        if($key == "name" or $key == "color" or $key == "region"){
            $bird[$key] = $value;
        } else {
            unset($bird[$key]);
        }
    }
    $subset[] = $bird;
}
// End edits
```

Caption (required) ✓

Describe/highlight what's being shown

Screenshot of code solution for problem 1 from VS Code

Explanation (required) ✓

Explain in concise steps how this logically works

PREVIEW RESPONSE

There is first a loop that iterates through each inner array. There is another loop that iterates through each key value pair inside the inner array. An if statement is then used to add a key value pair to the result array if the key is "name", "color", or "region". If the key is not one of the desired keys, then the key value pair is deleted.

Problem 2 (3 pts.)

^COLLAPSE ^



^COLLAPSE ^

Task #1 - Points: 1

Text: Problem 2 Evidence

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)

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)



```
Processing Array:
array (
  0 =>
    array (
      'id' => 1,
      'make' => 'Toyota',
      'model' => 'Camry',
      'year' => 2010,
    )
  1 =>
    array (
      'id' => 2,
      'make' => 'Honda',
      'model' => 'Civic',
      'year' => 2005,
    )
)

New Properties Output:
array (
  0 =>
    array (
      'id' => 1,
      'make' => 'Toyota',
      'model' => 'Camry',
      'year' => 2010,
      'age' => 14,
      'isClassic' => false,
    )
  1 =>
    array (
      'id' => 2,
      'make' => 'Honda',
      'model' => 'Civic',
      'year' => 2005,
      'age' => 19,
      'isClassic' => false,
    )
)

Processing Array:
array (
  0 =>
    array (
      'id' => 3,
      'make' => 'Ford',
      'model' => 'Mustang',
      'year' => 1995,
    )
  1 =>
    array (
      'id' => 4,
      'make' => 'Chevrolet',
      'model' => 'Impala',
      'year' => 2000,
    )
)

New Properties Output:
array (
  0 =>
    array (
      'id' => 3,
      'make' => 'Ford',
      'model' => 'Mustang',
      'year' => 1995,
      'age' => 29,
      'isClassic' => true,
    )
  1 =>
    array (
      'id' => 4,
      'make' => 'Chevrolet',
      'model' => 'Impala',
      'year' => 2000,
      'age' => 24,
      'isClassic' => false,
    )
)

Processing Array:
array (
  0 =>
    array (
      'id' => 5,
      'make' => 'Nissan',
      'model' => 'Altima',
      'year' => 2015,
      'age' => 9,
      'isClassic' => false,
    )
  1 =>
    array (
      'id' => 6,
      'make' => 'BMW',
      'model' => '3 Series',
      'year' => 2018,
      'age' => 6,
      'isClassic' => false,
    )
)

New Properties Output:
array (
  0 =>
    array (
      'id' => 5,
      'make' => 'Nissan',
      'model' => 'Altima',
      'year' => 2015,
      'age' => 9,
      'isClassic' => false,
    )
  1 =>
    array (
      'id' => 6,
      'make' => 'BMW',
      'model' => '3 Series',
      'year' => 2018,
      'age' => 6,
      'isClassic' => false,
    )
)

Processing Array:
array (
  0 =>
    array (
      'id' => 7,
      'make' => 'Mercedes',
      'model' => 'C Class',
      'year' => 2011,
      'age' => 13,
      'isClassic' => false,
    )
  1 =>
    array (
      'id' => 8,
      'make' => 'Audi',
      'model' => 'A4',
      'year' => 1990,
      'age' => 34,
      'isClassic' => true,
    )
)

New Properties Output:
array (
  0 =>
    array (
      'id' => 7,
      'make' => 'Mercedes',
      'model' => 'C Class',
      'year' => 2011,
      'age' => 13,
      'isClassic' => false,
    )
  1 =>
    array (
      'id' => 8,
      'make' => 'Audi',
      'model' => 'A4',
      'year' => 1990,
      'age' => 34,
      'isClassic' => true,
    )
)
```

Caption (required) ✓

Describe/highlight what's being shown

A screenshot of problem 2 from heroku dev

URL (required) ✓

URL must be Heroku prod

<https://mrs43-prod-5f0e0e79560e.herokuapp.com/M5/problem2.php>

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



```
// Start edits
// mrs43 6-24-2024
$isClassic = false;
foreach($cars as $car){
    foreach($car as $key => $value){
        if($key == "year"){
            $currentYear = date("Y");
            $age = $currentYear - $value;
            $car["age"] = $age;
            if($age >= $classic_age){
                $car["isClassic"] = true;
            } else{
                $car["isClassic"] = false;
            }
        }
    }
} <- #36-45 if($key == "year")
} <- #35-46 foreach($car as $key => $value)
$processedCars[] = $car;
} <- #34-48 foreach($cars as $car)

// End edits
```

Caption (required) ✓

Describe/highlight what's being shown

The code solution for problem 2 from VS Code

Explanation (required) ✓

Explain in concise steps how this logically works

PREVIEW RESPONSE

There is an outer loop that iterates through each inner array. There's another loop that iterates through each key value pair inside the inner array. An if statement sees if the key is "year". If the key is, then the current year and age are calculated. The key "age" is assigned the appropriate age value. There is another if statement that sees if the car's age is greater than or equal to the classic age (25 years). If it is, then the key "isClassic" is assigned the value "true". If not, it is assigned "false".

Problem 3 (3 pts.)

COLLAPSE

Task #1 - Points: 1

Text: Problem 3 Evidence

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 Heroku Prod

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



```
ms43-dev-605cd201cf1b.herokuapp.com/M5/problem3.php
Automa Dolphin Fa... Heroku THD Mobile THD Desktop Library Think Python Self Service OSLA Software Developm... Book...

//code: array (
    0 =>
        array (
            'userId' => 3,
            'name' => 'Alice',
            'age' => 20,
        )
    1 =>
        array (
            'userId' => 2,
            'name' => 'Bob',
            'age' => 14,
        )
    2 =>
        array (
            'userId' => 3,
            'name' => 'Charlie',
            'age' => 22,
        )
    3 =>
        array (
            'userId' => 4,
            'name' => 'Diana',
            'age' => 25,
        )
    4 =>
        array (
            'userId' => 5,
            'name' => 'Eve',
            'age' => 23,
        )
    5 =>
        array (
            'userId' => 6,
            'name' => 'Frank',
            'age' => 26,
        )
    6 =>
        array (
            'userId' => 7,
            'name' => 'Grace',
            'age' => 25,
        )
    7 =>
        array (
            'userId' => 8,
            'name' => 'Hank',
            'age' => 30,
        )
    8 =>
        array (
            'userId' => 9,
            'name' => 'Ivy',
            'age' => 27,
        )
    9 =>
        array (
            'userId' => 10,
            'name' => 'Jack',
            'age' => 28,
        )
    10 =>
        array (
            'userId' => 11,
            'name' => 'Karen',
            'age' => 29,
        )
    11 =>
        array (
            'userId' => 12,
            'name' => 'Leo',
            'age' => 31,
        )
    12 =>
        array (
            'userId' => 13,
            'name' => 'Mia',
            'age' => 32,
        )
    13 =>
        array (
            'userId' => 14,
            'name' => 'Noah',
            'age' => 33,
        )
    14 =>
        array (
            'userId' => 15,
            'name' => 'Olivia',
            'age' => 34,
        )
    15 =>
        array (
            'userId' => 16,
            'name' => 'Peter',
            'age' => 35,
        )
    16 =>
        array (
            'userId' => 17,
            'name' => 'Quinn',
            'age' => 36,
        )
    17 =>
        array (
            'userId' => 18,
            'name' => 'Ryan',
            'age' => 37,
        )
    18 =>
        array (
            'userId' => 19,
            'name' => 'Sara',
            'age' => 38,
        )
    19 =>
        array (
            'userId' => 20,
            'name' => 'Tina',
            'age' => 39,
        )
    20 =>
        array (
            'userId' => 21,
            'name' => 'Uma',
            'age' => 40,
        )
    21 =>
        array (
            'userId' => 22,
            'name' => 'Victor',
            'age' => 41,
        )
    22 =>
        array (
            'userId' => 23,
            'name' => 'Wendy',
            'age' => 42,
        )
    23 =>
        array (
            'userId' => 24,
            'name' => 'Xavier',
            'age' => 43,
        )
    24 =>
        array (
            'userId' => 25,
            'name' => 'Yara',
            'age' => 44,
        )
    25 =>
        array (
            'userId' => 26,
            'name' => 'Zoe',
            'age' => 45,
        )
)

//joined output:
array (
    0 =>
        array (
            'userId' => 3,
            'name' => 'Alice',
            'age' => 20,
            'activity' => 'Running',
        )
    1 =>
        array (
            'userId' => 2,
            'name' => 'Bob',
            'age' => 14,
            'activity' => 'Swimming',
        )
    2 =>
        array (
            'userId' => 3,
            'name' => 'Charlie',
            'age' => 22,
            'activity' => 'Cycling',
        )
    3 =>
        array (
            'userId' => 4,
            'name' => 'Diana',
            'age' => 25,
            'activity' => 'Hiking',
        )
    4 =>
        array (
            'userId' => 5,
            'name' => 'Eve',
            'age' => 23,
            'activity' => 'Climbing',
        )
    5 =>
        array (
            'userId' => 6,
            'name' => 'Frank',
            'age' => 26,
            'activity' => 'Shooting',
        )
    6 =>
        array (
            'userId' => 7,
            'name' => 'Grace',
            'age' => 25,
            'activity' => 'Dancing',
        )
    7 =>
        array (
            'userId' => 8,
            'name' => 'Hank',
            'age' => 30,
            'activity' => 'Surfing',
        )
    8 =>
        array (
            'userId' => 9,
            'name' => 'Ivy',
            'age' => 27,
            'activity' => 'Golfing',
        )
    9 =>
        array (
            'userId' => 10,
            'name' => 'Jack',
            'age' => 28,
            'activity' => 'Fishing',
        )
    10 =>
        array (
            'userId' => 11,
            'name' => 'Karen',
            'age' => 29,
            'activity' => 'Reading',
        )
    11 =>
        array (
            'userId' => 12,
            'name' => 'Leo',
            'age' => 31,
            'activity' => 'Gardening',
        )
    12 =>
        array (
            'userId' => 13,
            'name' => 'Mia',
            'age' => 32,
            'activity' => 'Cooking',
        )
    13 =>
        array (
            'userId' => 14,
            'name' => 'Noah',
            'age' => 33,
            'activity' => 'Painting',
        )
    14 =>
        array (
            'userId' => 15,
            'name' => 'Olivia',
            'age' => 34,
            'activity' => 'Skiing',
        )
    15 =>
        array (
            'userId' => 16,
            'name' => 'Peter',
            'age' => 35,
            'activity' => 'Judo',
        )
    16 =>
        array (
            'userId' => 17,
            'name' => 'Quinn',
            'age' => 36,
            'activity' => 'Taekwondo',
        )
    17 =>
        array (
            'userId' => 18,
            'name' => 'Ryan',
            'age' => 37,
            'activity' => 'Volleyball',
        )
    18 =>
        array (
            'userId' => 19,
            'name' => 'Sara',
            'age' => 38,
            'activity' => 'Baseball',
        )
    19 =>
        array (
            'userId' => 20,
            'name' => 'Tina',
            'age' => 39,
            'activity' => 'Basketball',
        )
    20 =>
        array (
            'userId' => 21,
            'name' => 'Uma',
            'age' => 40,
            'activity' => 'Softball',
        )
    21 =>
        array (
            'userId' => 22,
            'name' => 'Victor',
            'age' => 41,
            'activity' => 'Baseball',
        )
    22 =>
        array (
            'userId' => 23,
            'name' => 'Wendy',
            'age' => 42,
            'activity' => 'Basketball',
        )
    23 =>
        array (
            'userId' => 24,
            'name' => 'Xavier',
            'age' => 43,
            'activity' => 'Softball',
        )
    24 =>
        array (
            'userId' => 25,
            'name' => 'Yara',
            'age' => 44,
            'activity' => 'Baseball',
        )
    25 =>
        array (
            'userId' => 26,
            'name' => 'Zoe',
            'age' => 45,
            'activity' => 'Basketball',
        )
)
```

Caption (required) ✓

Describe/highlight what's being shown

URL (required) ✓

URL must be Heroku prod

<https://mrs43-prod-5f0e0e79560e.herokuapp.com/M5/problem3.php>

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



```
// Start edits
// mrs43 6-24-2024
foreach($users as $user){
    foreach($user as $key => $value){
        $user[$key] = $value;
        if($key == "userId"){
            foreach($activities as $activity){
                if($activity[$key] == $value){
                    $user["activity"] = $activity["activity"];
                }
            }
        }
    }
}
$joined[] = $user;
}
// End edits
```

Caption (required) ✓

Describe/highlight what's being shown

A screenshot of the code solution for problem 3 from VS Code

Explanation (required) ✓

Explain in concise steps how this logically works

PREVIEW RESPONSE

A outer loop first iterates through each inner array of the "users" array. Another loop then iterates through each key value pair in that array. If the key is "userId", then another loop is used to iterate through the "activities" array to assign the "activity" key the appropriate value based on the "userId".



Reflection (1 pt.)

^COLLAPSE ^



Task #1 - Points: 1

Text: Reflect on your experience

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:

The only issue I ran into was figuring out how to join the two arrays in problem 3. I first was trying to do two separate loops, which did not work. So I thought to focus on one and then iterate through the other one to find the matching user. I was on the right track, but I accidentally assigned the "activity" key to the entire activity array on my first try. I then fixed this to only assign the correct activity value.



^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/m-sansone/mrs43-IT202-452/pull/31>



^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

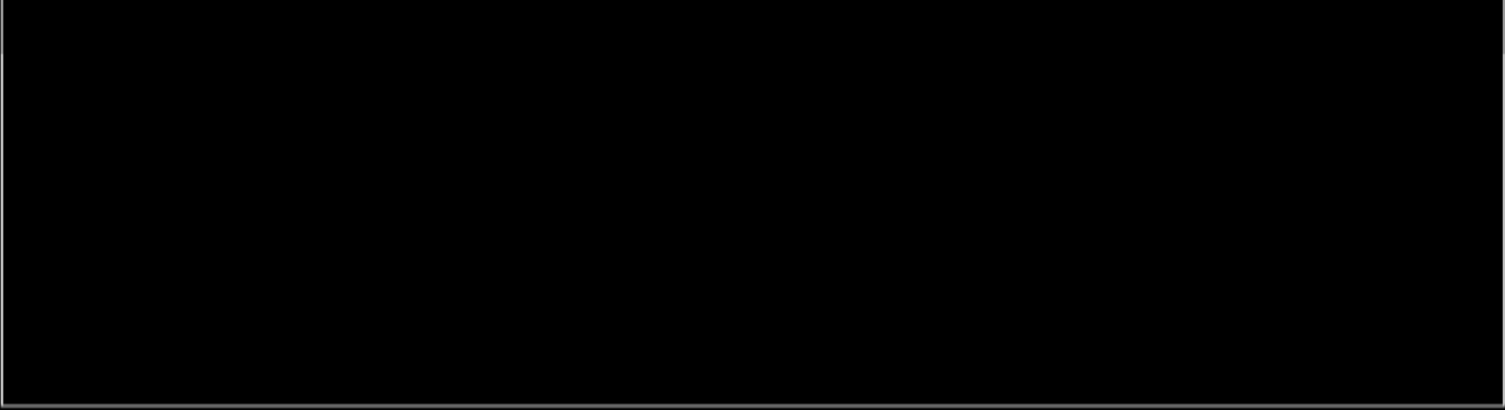
Task Screenshots:

Gallery Style: Large View

Small

Medium

Large



A screenshot of Wakatime extension in VS Code

End of Assignment