Deliverable 1:


```php
<?php
$a1 =[0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15];
$a2 =[-3,-2,-1,0,1,2,3,4,5,6,7,8,9,10];
$a3 =[15,14,13,12,11,10,9,8,7,6,5,4,3,2,1,0];
$a4 =[0,0,1,1,2,2,3,3,4,4,5,5,6,6,7,7,8,8,9,9,10,10];
function processArray($arr) {
    //use the $arr variable to iterate over
    echo "<br>Processing Array: <br> <pre>". var_export($arr, True) . "</pre>";
    echo "<br>Divisible By 3 Output:<br>";
    //TODO add logic here to echo out any values divisible by 3
    foreach($arr as $num){
        if ($num % 2 != 0){
            echo $num, ", ";       You, 2 days ago • completed hw2 …
        }
    } <- #11-15 foreach($arr as $num)
} <- #6-16 function processArray($arr)
?>
<h2> Problem 1: Divisible By 3 Output</h2>
<table>
    <thead>
        <th>a1</th>
        <th>a2</th>
        <th>a3</th>
        <th>a4</th>
    </thead>
    <tbody>
        <tr>
            <td>
                <?php processArray($a1); ?>
```

Problem 1: Divisible By 3 Output

In the screenshot above, the screenshot is my code to select and print out only odd numbers from each array. I used a "foreach" loop to iterate through each element in the array. Each element in the array is then checked if it is odd using "%" 2. If remainder is NOT 0, then the number is odd.

Deliverable 2:

```php
<?php
$a1 =[10.001,11.101,0.011,3.991,16.121,8.131,100.231,1.001];
$a2 =[1.99,1.99, 0.99, 1.973, 0.99,1.91, 0.91, 0.99];
$a3 =[0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 01.01];
$a4 =[10.01, -12.22, 0.23, 29.20, -5.13, 2.12];
function getTotal($arr) {
    //use the $arr variable to iterate over
    echo "<br>Processing Array: <br> <pre>". var_export($arr, True) . "</pre>";
    $total = 0.00;          You, last week • adding template baselines …
    //TODO do adding here
    foreach ($arr as $num){
        $total += $num;
    }
    //TODO do rounding stuff here.
    $total = round($total, 2);
    //TODO Extra Credit: show 2 point precision even if last number is a 0
    echo "<br>The total is ".var_export($total, true);
} <- #6-18 function getTotal($arr)
```

Problem 2: Sum the values and display the total

A1	A2	A3	A4
Processing Array:	Processing Array:	Processing Array:	
		array (	Processing Array:
array (	array (	0 => 0.01,	
0 => 10.001,	0 => 1.99,	1 => 0.01,	array (
1 => 11.101,	1 => 1.99,	2 => 0.01,	0 => 10.01,
2 => 0.011,	2 => 0.99,	3 => 0.01,	1 => -12.22,
3 => 3.991,	3 => 1.973,	4 => 0.01,	2 => 0.23,
4 => 16.121,	4 => 0.99,	5 => 0.01,	3 => 29.2,
5 => 8.131,	5 => 1.91,	6 => 0.01,	4 => -5.13,
6 => 100.231,	6 => 0.91,	7 => 0.01,	5 => 2.12,
7 => 1.001,	7 => 0.99,	8 => 0.01,	)
)	)	9 => 1.01,	
		)	
			The total is 24.21
The total is 150.59	The total is 11.74		
		The total is 1.1	

The source code above utilizes a "foreach" loop to iterate through each element in the array. Each element of the array is then added to the "$total" variable. The total is then echo out to show the total on the screen.

Deliverable 3:

```php
IT202-007 > M2 > ☁ problem3.php > ⓥ bePositive
You, 2 days ago | 1 author (You)
1   <?php
2   $A1 =[-1, -2, -3, -4, -5, -6, -7, -8, -9, -10];
3   $B2 =[-1, 1, -2, 2, 3, -3, -4, 115];
4   $C3 =[-0.03, 0.0002, -.11];
5   $D4 =['-1',"hello","2","-3","-4","5","-6","6",'-7','71',"71","a"];
6   function bePositive($arr) {
7       //use the $arr variable to iterate over
8       echo "<br>Processing Array: <br> <pre>". var_export($arr, True) . "</pre>";
9       echo "<br>Positive Output";
10      // TODO echo to output any value that positive (i.e., ignore negatives and characters)
11      foreach ($arr as $val){
12          if (is_numeric($val) && floatval($val > 0)){
13              echo ' ', floatval($val), ',';
14          }
15      } <- #11-15 foreach ($arr as $val)
16  } <- #6-16 function bePositive($arr)        You, last week · adding template baselines
```
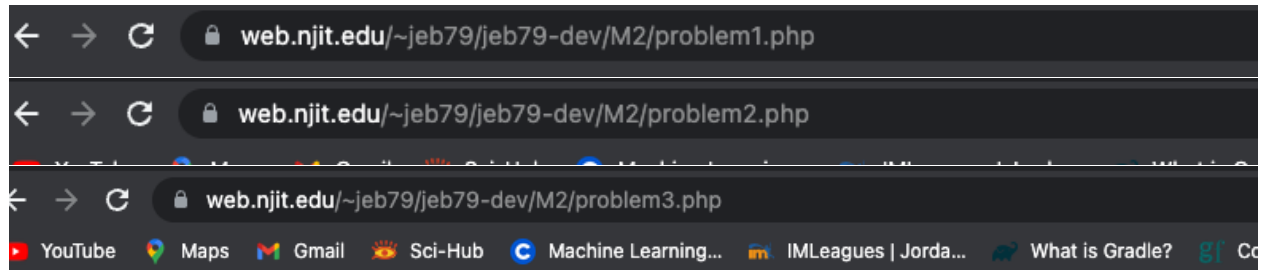
← → C ⓘ localhost:3000/M2/problem3.php

▶ YouTube   📍 Maps   M Gmail   🔆 Sci-Hub   Ⓒ Machine Learning...   🎽 IMLeagues | Jorda...   🐗 What is G

Problem 3: Be Positive

A1	B2	C3	D4

```
Processing Array:

array (
  0 => -1,
  1 => -2,
  2 => -3,
  3 => -4,
  4 => -5,
  5 => -6,
  6 => -7,
  7 => -8,
  8 => -9,
  9 => -10,
)

Positive Output
```

```
Processing Array:

array (
  0 => -1,
  1 => 1,
  2 => -2,
  3 => 2,
  4 => 3,
  5 => -3,
  6 => -4,
  7 => 115,
)

Positive Output 1, 2, 3, 115,
```

```
Processing Array:

array (
  0 => -0.03,
  1 => 0.0002,
  2 => -0.11,
)

Positive Output 0.0002,
```

```
Processing Array:

array (
  0 => '-1',
  1 => 'hello',
  2 => '2',
  3 => '-3',
  4 => '-4',
  5 => '5',
  6 => '-6',
  7 => '6',
  8 => '-7',
  9 => '71',
  10 => '71',
  11 => 'a',
)

Positive Output 2, 5, 6, 71, 71,
```

Again, a "foreach" loop was used to iterate through the elements in the array. An "if" statement is used to check each element is a numeric value and the float value is greater than 0. If both statements result in true, then the code inside the if statement is executed. The code inside the if statement will then "echo" the element to console.

Deliverable 4:

Pull request URL:
https://github.com/j0rdanbruce/IT202-007/pull/3