

Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT114-006-S2024/it114-m2-java-problems/grade/ms75>

IT114-006-S2024 - [IT114] M2 Java Problems

Submissions:

Submission Selection

1 Submission [active] 2/5/2024 12:33:47 AM

Instructions

^ COLLAPSE ^

Guide:

- 1 .Make sure you're in the main branch locally and ``git pull origin main`` 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/fdd2b37fa79a06ace9dd259ac82728b6>
- 4 .Create individual Java files for each problem and save the files inside a subfolder of your choice
 - 1 .The should end with the file extension in lowercase .java
- 5 .Move the unedited template files to github
 - 1 ``git add .``
 - 2 ``git commit -m "adding template files``
 - 3 ``git push origin <homework branch>`` (see below and don't include the `< >`)
 - 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``
 - 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
 - 1 .*This step is new since GitHub renders the PDF as an image the links aren't clickable so this method works better
 - 2 .*Remember, the github process of these files are encouragement for your tracking of your progress

Tasks: 8 Points: 10.00



Problem 1 (3 pts.)

^ COLLAPSE ^



v EXPAND v

Task #1 - Points: 1

Text: Screenshot of the Problem 1 Solved Code and Output



v EXPAND v

Task #2 - Points: 1

Text: Explain your solution



Problem 2 (3 pts.)

^ COLLAPSE ^



^ COLLAPSE ^

Task #1 - Points: 1

Text: Screenshot of the Problem 2 Solved Code and Output

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 terminal)

Checklist

*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Edits were done only in the getTotal() method and original template code/comments remain untouched (unless noted)
<input type="checkbox"/> #2	1	Only arr is used (no direct usage of a1, a2, a3, a4)
<input type="checkbox"/> #3	5	Passed in array's values get summed AND rounded to two decimal places like currency (i.e., 0.00, 0.10, 1.10)
<input type="checkbox"/> #4	1	Includes code comments with student's ucid and date
<input type="checkbox"/> #5	1	Terminal output is fully visible

Task Screenshots:



Large Gallery



Checklist Items (0)

Problem 2



Task #2 - Points: 1

Text: Explain your solution

Checklist

*The checkboxes are for your own tracking

#	Points	Details
#1	1	Clearly explains how the code/logic solves the problem (mentions both how the values get summed and how the rounding is solved correctly)

Response:

```
for (double num : arr) {  
    total += num;  
}
```

This code snippet is a for loop which goes through each element in the array and adds it to the variable "total".

```
totalOutput = String.format("%.2f", total);
```

This line of code takes the previously summed variable "total" is put through the String.format function to display the total with 2 decimal places. It then stores this new formatted total under the variable named "



Problem 3 (3 pts.)



Task #1 - Points: 1

Text: Screenshot of the Problem 2 Solved Code and Output



Task #2 - Points: 1

Text: Explain your solution



Reflection (1 pt.)



Task #1 - Points: 1

Text: Reflect on your experience



EXPAND

Task #2 - Points: 1

Text: Include the pull request link for this branch