Submission Worksheet

CLICK TO GRADE

https://learn.ethereallab.app/assignment/IT114-005-F2024/it114-module-2-java-problems/grade/kr553

Course: IT114-005-F2024
Assigment: [IT114] Module 2 Java Problems
Student: Kush R. (kr553)

Submissions:

Submission Selection

1 Submission [submitted] 9/23/2024 10:10:09 AM

Instructions

^ COLLAPSE ^

Overview Video: https://youtu.be/4M8Di5jrcZQ

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. Create a folder in your local repo called `Module2`
4. Grab the template code from https://gist.github.com/MattToegel/fdd2b37fa79a06ace9dd259ac82728b6.
5. Create individual Java files for each problem and save the files inside the `Module2` folder.
    1. They should end with the file extension in lowercase `.java`.
6. 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).
7. 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.
    2. A file is unsaved if you see a white dot in the tab where the filename shows in VS Code
8. Fill in the items in the worksheet below (save as often as necessary).
9. Once finished, export the worksheet.
10. Add the output file to any location of your choice in your repository folder (i.e., a `Module2` folder).
11. Check that git sees it via `git status`.
12. If everything is good, continue to submit.

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

Branch name: M2-Java-Problems

Group

100%

Group: Problem 1
Tasks: 1
Points: 3

COLLAPSE

Task

100%

Group: Problem 1
Task #1: Screenshot of the Problem 1 Solved Code and Output
Weight: ~100%
Points: ~3.00

COLLAPSE

Details:
Only make edits where the template code mentions.

Solution should ensure that any passed in array will have only the odd values output.
Requires at least 2 screenshots (code + output from terminal)

Columns: 1

Sub-Task

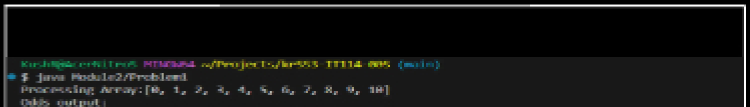
100%

Group: Problem 1
Task #1: Screenshot of the Problem 1 Solved Code and Output
Sub Task #1: Screenshot the output of the solved problem

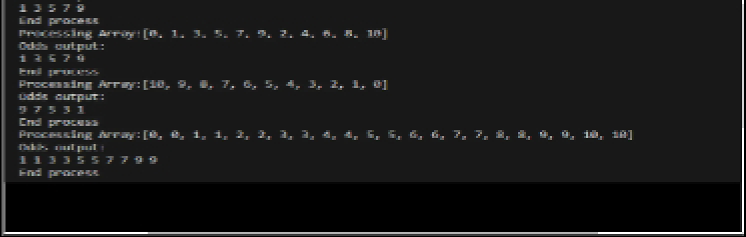
Task Screenshots

Gallery Style: 2 Columns

4 2 1



```
Kush@AcerNitro5 MINGW64 ~/Projects/kr553-IT114-005 (main)
$ java Module2/Problem1
Processing Array:[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
Odds output:
```



```
1 3 5 7 9
End process
Processing Array:[0, 1, 3, 5, 7, 9, 2, 4, 6, 8, 10]
Odds output:
1 3 5 7 9
End process
Processing Array:[10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
Odds output:
9 7 5 3 1
End process
Processing Array:[0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9, 10, 10]
Odds output:
1 1 3 3 5 5 7 7 9 9
End process
```

output

Caption(s) (required) ✓

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

Sub-Task

100%

Group: Problem 1

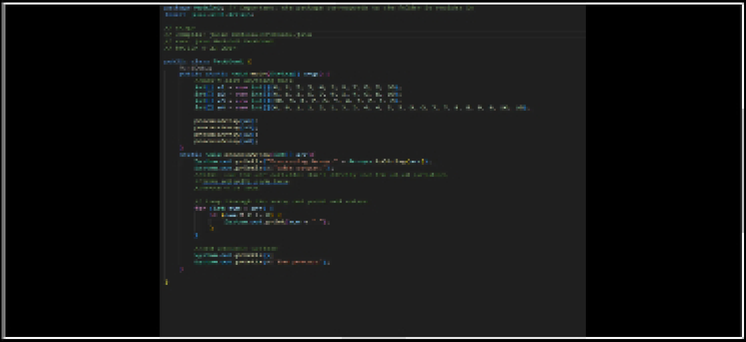
Task #1: Screenshot of the Problem 1 Solved Code and Output

Sub Task #2: Screenshot the code solution (ucid/date must be included as a comment)

Task Screenshots

Gallery Style: 2 Columns

4 2 1



[illegible]

Code

Caption(s) (required) ✓

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

Task Response Prompt

Explain in concise steps how this logically works

Response:

So first in the main funtion we define arrays and pass into the function which returns only odd output. In the process array function we are first printing the arrays which was given to us and later printing output statement followed by for loop which iterates over array and it checks if number is divided by 2 does not returns 0 then it prints the number else it proceeds to next number of array. so in the output we get all the odd values.

End of Task 1

End of Group: Problem 1

Task Status: 1/1

Group

100%

Group: Problem 2

Tasks: 1

Points: 3

Points: 3

^ COLLAPSE ^

Task

100%

Group: Problem 2
Task #1: Screenshot of the Problem 2 Solved Code and Output
Weight: ~100%
Points: ~3.00

^ COLLAPSE ^

Details:
Only make edits where the template code mentions.

Solution should ensure that any passed in array will have its values summed AND the final result converted to two decimal places (i.e., 0.10, 1.00, 1.01).

Columns: 1

Sub-Task

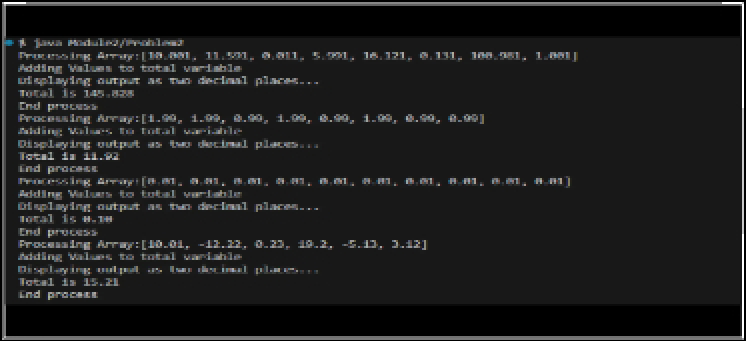
100%

Group: Problem 2
Task #1: Screenshot of the Problem 2 Solved Code and Output
Sub Task #1: Screenshot the output of the solved problem

Task Screenshots

Gallery Style: 2 Columns

4 2 1



```
$ java Module2/Problem2
Processing Array:[10.001, 11.591, 0.011, 5.991, 16.121, 0.131, 100.981, 1.001]
Adding Values to total variable
Displaying output as two decimal places...
Total is 145.828
End process
Processing Array:[1.99, 1.99, 0.99, 1.99, 0.99, 1.99, 0.99, 0.99]
Adding Values to total variable
Displaying output as two decimal places...
Total is 11.92
End process
Processing Array:[0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01, 0.01]
Adding Values to total variable
Displaying output as two decimal places...
Total is 0.10
End process
Processing Array:[10.01, -12.22, 0.23, 19.2, -5.13, 3.12]
Adding Values to total variable
Displaying output as two decimal places...
Total is 15.21
End process
```

Output

Caption(s) (required) ✓

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

Sub-Task

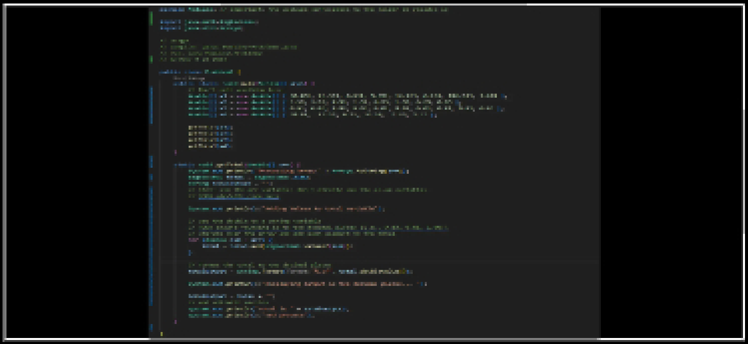
100%

Group: Problem 2
Task #1: Screenshot of the Problem 2 Solved Code and Output
Sub Task #2: Screenshot the code solution (ucid/date must be included as a comment)

Task Screenshots

Gallery Style: 2 Columns

4 2 1



code

Caption(s) (required) ✓

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

Task Response Prompt

Explain in concise steps how this logically works

Response:

So in the main function we declare arrays of doubles which has decimal points and pass arrays to funtion getTotal. in the get total function first we print the arrays we have been given to calculate the total. then we print a statement to show we are adding the values to variable, here i changed the total variable to Big decimal i searched on internet about the problem i was facing and i got to know about big decimal value which will avoid the rounding error i got at first time the third value i got was 0.0999999999 and after implementing bigDecimal i got correct rounded answer for that. the for loop here just adds the number and we set precision of the decimal numbers to 2 by %0.2f and we returns the double value.

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: Screenshot of the Problem 3 Solved Code and Output
Weight: ~100%
Points: ~3.00

COLLAPSE

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.

Columns: 1

Sub-Task

100%

Group: Problem 3

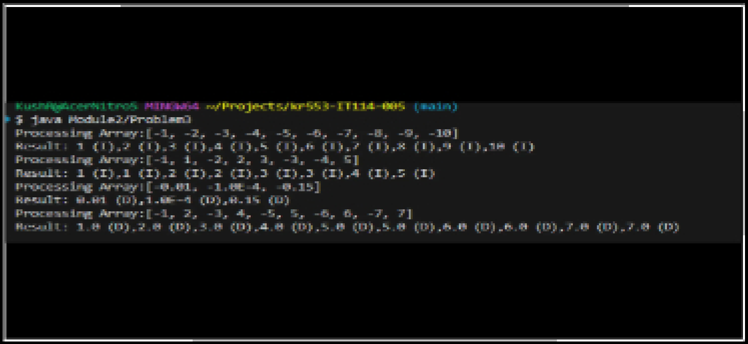
Task #1: Screenshot of the Problem 3 Solved Code and Output

Sub Task #1: Screenshot the output of the solved problem

Task Screenshots

Gallery Style: 2 Columns

4 2 1



```
KushA@AcerNitro5 MINGW64 ~/Projects/kr553-IT114-005 (main)
$ java Module2/Problem3
Processing Array:[-1, -2, -3, -4, -5, -6, -7, -8, -9, -10]
Result: 1 (I),2 (I),3 (I),4 (I),5 (I),6 (I),7 (I),8 (I),9 (I),10 (I)
Processing Array:[-1, 1, -2, 2, 3, -3, -4, 5]
Result: 1 (I),1 (I),2 (I),2 (I),3 (I),3 (I),4 (I),5 (I)
Processing Array:[-0.01, -1.0E-4, -0.15]
Result: 0.01 (D),1.0E-4 (D),0.15 (D)
Processing Array:[-1, 2, -3, 4, -5, 5, -6, 6, -7, 7]
Result: 1.0 (D),2.0 (D),3.0 (D),4.0 (D),5.0 (D),5.0 (D),6.0 (D),6.0 (D),7.0 (D),7.0 (D)
```

Output

Caption(s) (required) ✓

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

Sub-Task

100%

Group: Problem 3

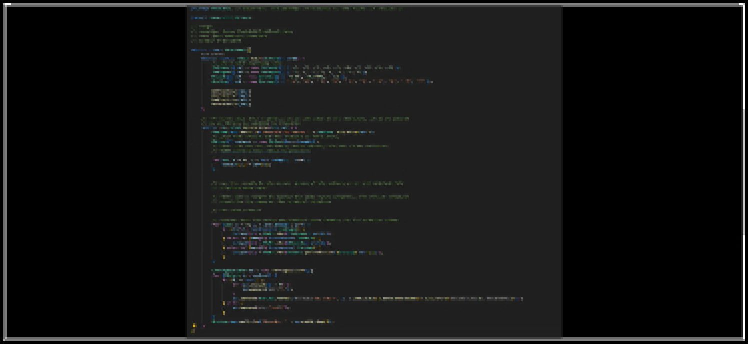
Task #1: Screenshot of the Problem 3 Solved Code and Output

Sub Task #2: Screenshot the code solution (ucid/date must be included as a comment)

Task Screenshots

Gallery Style: 2 Columns

4 2 1



Code

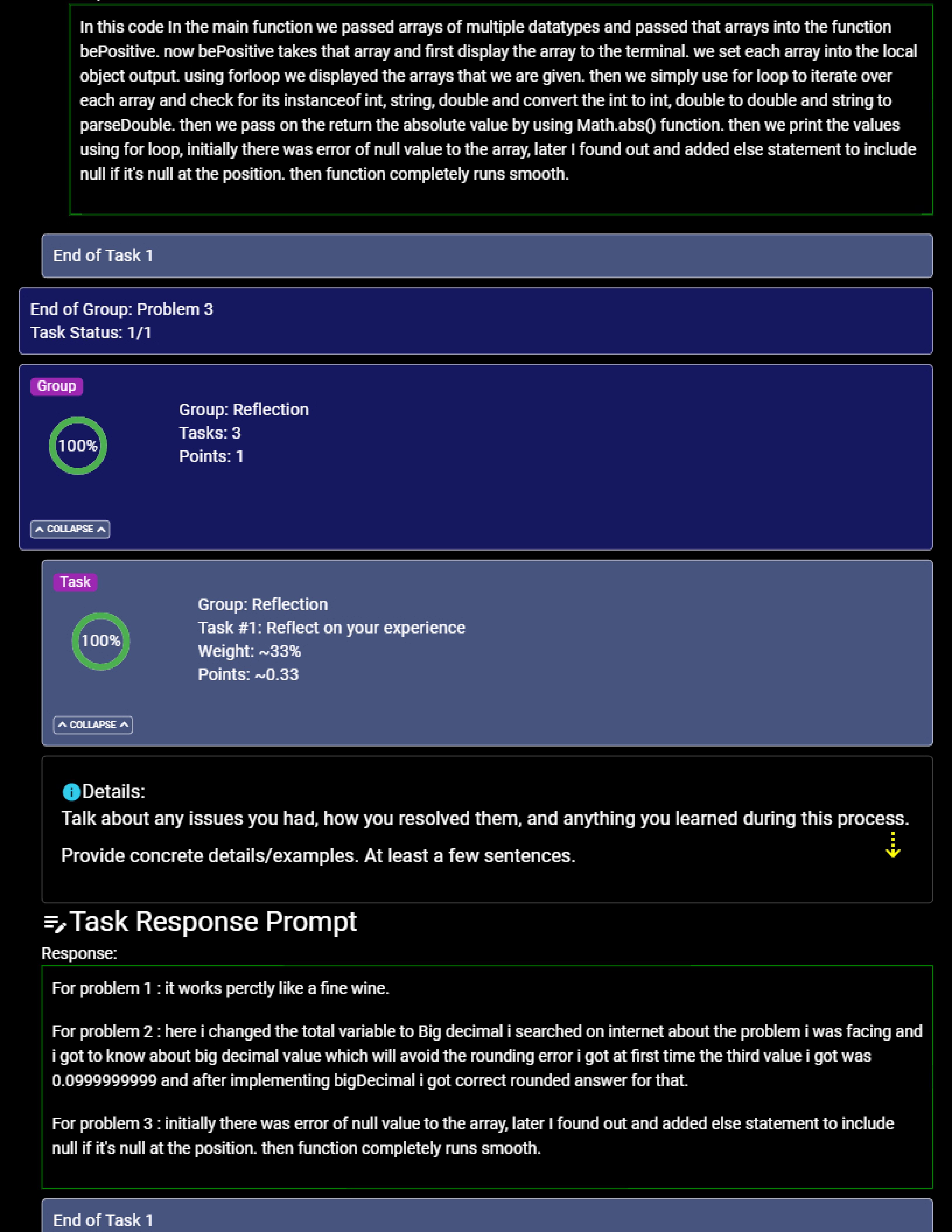
Caption(s) (required) ✓

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

Task Response Prompt

Explain in concise steps how this logically works

Response:



In this code In the main function we passed arrays of multiple datatypes and passed that arrays into the function bePositive. now bePositive takes that array and first display the array to the terminal. we set each array into the local object output. using forloop we displayed the arrays that we are given. then we simply use for loop to iterate over each array and check for its instanceof int, string, double and convert the int to int, double to double and string to parseDouble. then we pass on the return the absolute value by using Math.abs() function. then we print the values using for loop, initially there was error of null value to the array, later I found out and added else statement to include null if it's null at the position. then function completely runs smooth.

End of Task 1

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

Group: Reflection
Tasks: 3
Points: 1

Group: Reflection
Task #1: Reflect on your experience
Weight: ~33%
Points: ~0.33

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:

For problem 1 : it works perctly like a fine wine.

For problem 2 : here i changed the total variable to Big decimal i searched on internet about the problem i was facing and i got to know about big decimal value which will avoid the rounding error i got at first time the third value i got was 0.0999999999 and after implementing bigDecimal i got correct rounded answer for that.

For problem 3 : initially there was error of null value to the array, later I found out and added else statement to include null if it's null at the position. then function completely runs smooth.

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/KushDev19/kr553-IT114-005/pull/2

URL
https://github.com/KushDev19/kr553-IT114-005/

End of Task 2

Task

100%

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

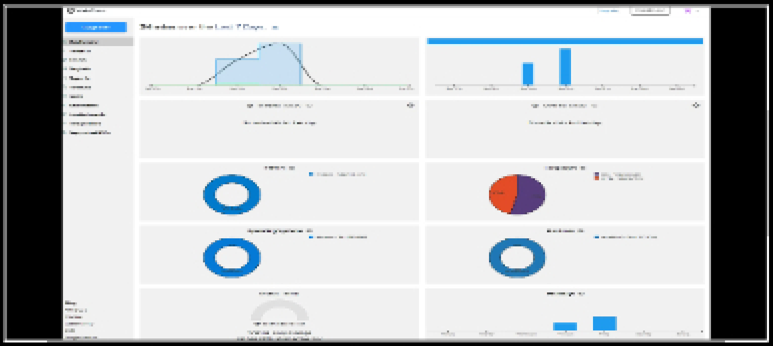
COLLAPSE

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



I just remembered i switched profile and i did not had waka time in another profile

End of Task 3

End of Group: Reflection
Task Status: 3/3

End of Assignment