Submission Worksheet

Submission Data

Course: IT114-450-M2025

Assignment: IT114 Java Problems

Student: Mariano R. (mr822)

Status: Submitted | **Worksheet Progress:** 100+%

Potential Grade: 11.00/10.00 (110.00%)

Received Grade: 0.00/10.00 (0.00%)

Started: 6/9/2025 5:02:54 PM

Updated: 6/9/2025 7:04:38 PM

Grading Link: https://learn.ethereallab.app/assignment/v3/IT114-450-M2025/it114-java-problems/grading/mr822

View Link: https://learn.ethereallab.app/assignment/v3/IT114-450-M2025/it114-java-problems/view/mr822

Instructions

- Overview Link: https://youtu.be/Mrahk6SFYao

1. Ensure you read all instructions and objectives before starting.
2. Create a new branch from `main` called `M2-Homework`
   1. `git checkout main` (ensure proper starting branch)
   2. `git pull origin main` (ensure history is up to date)
   3. `git checkout -b M2-Homework` (create and switch to branch)
3. Copy the template code from here: GitHub Repository - M2 Homework
   - It includes Problems 1-4 and a BaseClass. Put all into an `M2` folder or similar (adjust `package` reference at the top if you chose a different folder name).
   - Immediately record to history
     - `git add .`
     - `git commit -m "adding M2 HW baseline files"`
     - `git push origin M2-Homework`
     - Create a Pull Request from `M2-Homework` to `main` and keep it open
4. Fill out the below worksheet
   - Each Problem requires the following as you work
     - Ensure there's a comment with your UCID, date, and brief summary of how the problem was solved
     - Initial outline/plan of how you'll solve it via comments (add/commit after this stage)
     - Code solution (add/commit periodically as needed)
5. Once finished, click "Submit and Export"
6. Locally add the generated PDF to a folder of your choosing inside your repository folder and move it to Github
   1. `git add .`
   2. `git commit -m "adding PDF"`
   3. `git push origin M2-Homework`
   4. On Github merge the pull request from `M2-Homework` to `main`
7. Upload the same PDF to Canvas
8. Sync Local
   1. `git checkout main`

Section #1: (2 pts.) Problem 1 - Odds

Progress: 100%

☰ Task #1 (2 pts.) - Edit the `printOdds` method to output odd values of the array

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Challenge: Print odd values only in a single line separated by commas
- Step 1: sketch out plan using comments (include ucid and date)
- Step 2: Add/commit your outline of comments (required for full credit)
- Step 3: Add code to solve the problem (add/commit as needed)

🖼 Part 1:

Progress: 100%

Details:

Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program

Problem 1 Code


```
mramos2001@DESKTOP-US058AO:~/IT114$ javac M2/Problem1.java
mramos2001@DESKTOP-US058AO:~/IT114$ java M2.Problem1
Running Problem 1 for [mr822] [2025-06-09T17:01:48.9491/9282]
Objective: Print out only odd values in a single line separate by commas
Problem 1: Original Array: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
Output Array: 1, 3, 5, 7, 9

Problem 2: Original Array: [9, 8, 7, 6, 5, 4, 3, 2, 1, 0]
Output Array: 9, 7, 5, 3, 1

Problem 3: Original Array: [0, 0, 1, 1, 2, 2, 3, 3, 4, 4, 5, 5, 6, 6, 7, 7, 8, 8, 9, 9]
Output Array: 1, 1, 3, 3, 5, 5, 7, 7, 9, 9

Problem 4: Original Array: [9, 9, 8, 8, 7, 7, 6, 6, 5, 5, 4, 4, 3, 3, 2, 2, 1, 1, 0, 0]
Output Array: 9, 9, 7, 7, 5, 5, 3, 3, 1, 1

Completed Problem 1 for [mr822] [2025-06-09T17:01:48.964134252]
mramos2001@DESKTOP-US058AO:~/IT114$
```

Output Problem 1

💾 Saved: 6/9/2025 5:11:37 PM

🔗 Part 2:

Details:

Direct link to the file in the homework related branch from Github (should end in `.java`)

URL #1

https://github.com/mramos822/mr822-
IT114-N450-M2-Homework/M2/Problem1.java

👍

URL
https://github.com/mramos822/n

💾 Saved: 6/9/2025 5:11:37 PM

≣, Part 3:

Progress: 100%

Details:

Briefly explain how the code solves the challenge (note: this isn't the same as what the code does)

Your Response:

The code solves the problem by using a loop to check each number in the array and determine whether it is odd. A conditional checks value % 2 != 0, and if true, the value is printed.

💾 Saved: 6/9/2025 5:11:37 PM

Section #2: (2 pts.) Problem 2 - Sum

Progress: 100%

≣ Task #1 (2 pts.) - Edit the `sumValues` method to sum the array values and present them in a format with exactly two decimal places

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Challenge 1: Sum all the values of the passed in array and assign to `total`
- Challenge 2: Have the sum be represented as a number with exactly 2 decimal
- Example: 0.1 would be shown as 0.10, 1 would be shown as 1.00, etc
- Step 1: sketch out plan using comments (include ucid and date)
- Step 2: Add/commit your outline of comments (required for full credit)
- Step 3: Add code to solve the problem (add/commit as needed)

🖼 Part 1:

Progress: 100%

Details:

Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program

Problem 2 Code

Problem 2 Output

💾 Saved: 6/9/2025 5:34:42 PM

🔗 Part 2:

Progress: 100%

Details:

Direct link to the file in the homework related branch from Github (should end in `.java`)

URL #1

https://github.com/mramos822/mr822-IT114N450M2-Homework/M2/Problem2.java

👍

URL
https://github.com/mramos822/n

💾 Saved: 6/9/2025 5:34:42 PM

≡ᵧ Part 3:

Progress: 100%

Details:

Briefly explain how the code solves the challenges (note: this isn't the same as what the code does)

Your Response:

The code loops through the array to calculate the total. It uses a for each loop to accumulate the

The code loops through the array to calculate the total. It uses a for-each loop to accumulate the sum in a double variable. Then, it formats that total to exactly two decimal places.

💾 Saved: 6/9/2025 5:34:42 PM

Section #3: (2 pts.) Problem 3 - Conversion
Progress: 100%

≣ Task #1 (2 pts.) - Edit the `bePositive` method to make each value positive, convert it back to the orginal data type, and set it to the proper slot in the `output` array
Progress: 100%

Details:

- Only make edits where noted via provided comments
- Challenge 1: Make each value positive
- Challenge 2: Convert the values back to their original data type and assign it to the proper slot of the `output` array
- Step 1: sketch out plan using comments (include ucid and date)
- Step 2: Add/commit your outline of comments (required for full credit)
- Step 3: Add code to solve the problem (add/commit as needed)

🖾 Part 1:
Progress: 100%

Details:
Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program

Problem 3 Code

Problem 3 Output

💾 Saved: 6/9/2025 5:51:51 PM

🔗 Part 2:

Progress: 100%

Details:

Direct link to the file in the homework related branch from Github (should end in `.java`)

URL #1

https://github.com/mramos822/mr822-IT114-450-M2-Homework/M2/Problem3.java

👍

URL
https://github.com/mramos822/n

💾 Saved: 6/9/2025 5:51:51 PM

✏️ Part 3:

Progress: 100%

Details:

Briefly explain how the code solves the challenges (note: this isn't the same as what the code does)

Your Response:

The method uses instanceof to detect each input's type and applies Math.abs() to make values positive. Strings are parsed to numbers and returned as formatted strings. It ensures each element is processed and returned in its original type or an appropriate string version.

💾 Saved: 6/9/2025 5:51:51 PM

Section #4: (2 pts.) Problem 4 - Strings

Progress: 100%

☰ Task #1 (2 pts.) - Edit the `transformText` method to solve the challenges

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Challenge 1: Remove non-alphanumeric characters except spaces
- Challenge 2: Convert text to Title Case
- Challenge 3: Trim leading/trailing spaces and remove duplicate spaces

- Result 1-3: Assign final phrase to `placeholderForModifiedPhrase`
- Step 1: sketch out plan using comments (include ucid and date)
- Step 2: Add/commit your outline of comments (required for full credit)
- Step 3: Add code to solve the problem (add/commit as needed)

🖼 Part 1:

Progress: 100%

Details:

Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program

Problem 4 Output

Problem 4 Code

💾 Saved: 6/9/2025 6:10:27 PM

🔗 Part 2:

Progress: 100%

Details:

Direct link to the file in the homework related branch from Github (should end in `.java`)

URL #1

https://github.com/mramos822/mr822-
IT114-450/M2-Homework/M2/Problem4.java

👍 URL
https://github.com/mramos822/n

💾 Saved: 6/9/2025 6:10:27 PM

≡, Part 3:

Details:

Briefly explain how the code solves the challenges (note: this isn't the same as what the code does)

Your Response:

The method processes each input string by removing special characters, trimming extra spaces, and converting to Title Case. The cleaned result is stored. If the cleaned phrase has at least 3 characters, it extracts 3 middle characters; otherwise, it returns a fallback message.

💾 Saved: 6/9/2025 6:10:27 PM

☰ Task #2 (+ 1.11 pts.) - Edit the `transformText` method to solve the extra credit challenge (challenge 4)

Progress: 100%

Details:

- Only make edits where noted via provided comments
- Challenge 4: Extract middle 3 characters (beginning starts at middle of phrase)
- Assign result to 'placeholderForMiddleCharacters'
- If not enough characters assign "Not enough characters"
- Step 1: sketch out plan using comments (include ucid and date)
- Step 2: Add/commit your outline of comments (required for full credit)
- Step 3: Add code to solve the problem (add/commit as needed)

🖼 Part 1:

Progress: 100%

Details:

Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program

Extra Credit Code

Extra Credit Output

💾 Saved: 6/9/2025 6:16:10 PM

≡✎ Part 2:

Progress: 100%

Details:

Briefly explain how the code solves the extra credit challenge (note: this isn't the same as what the code does)

Your Response:

The code ensures the phrase is cleaned and spaces are removed before attempting the middle extraction. By checking the length first, it avoids errors and ensures only meaningful phrases are processed.

💾 Saved: 6/9/2025 6:16:10 PM

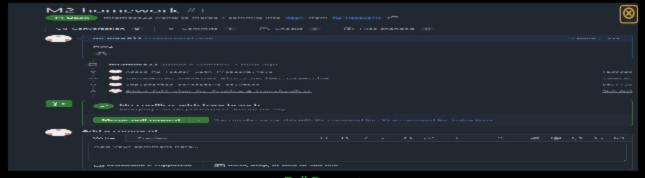# Section #5: (2 pts.) Misc

Progress: 100%

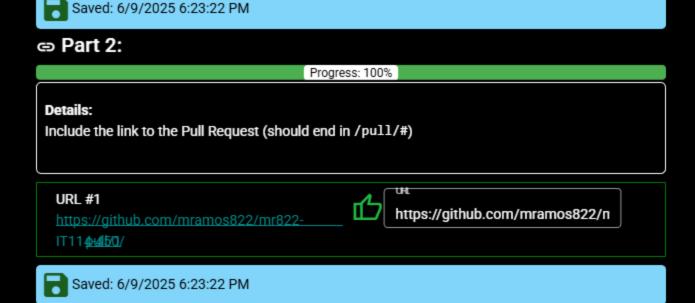## ≣ Task #1 (0.67 pts.) - Github Details

Progress: 100%

🖼 Part 1:

Progress: 100%

Details:

From the Commits tab of the Pull Request screenshot the commit history Following minimum should be present

Pull Request

🔗 Part 2:

Progress: 100%

Details:

Include the link to the Pull Request (should end in `/pull/#`)

URL #1

https://github.com/mramos822/mr822-IT114450/ 👍

URL
https://github.com/mramos822/n

🖼 Task #2 (0.67 pts.) - WakaTime - Activity

Progress: 100%

Details:

- Visit the WakaTime.com Dashboard
- Click `Projects` and find your repository
- Capture the overall time at the top that includes the repository name
- Capture the individual time at the bottom that includes the file time
- Note: The duration isn't relevant for the grade and the visual graphs aren't necessary

WakaTime

☰ Task #3 (0.67 pts.) - Reflection

Progress: 100%

✏ Task #1 (0.33 pts.) - What did you learn?

Progress: 100%

Details:

Briefly answer the question (at least a few decent sentences)

Your Response:

Through this assignment, I learned how to manipulate arrays and strings in Java to solve practical problems. I learned how to integrate version control with Git and track coding activity using WakaTime, which helped me understand how professional tools support software development workflows.

💾 Saved: 6/9/2025 7:03:41 PM

≡✏ Task #2 (0.33 pts.) - What was the easiest part of the assignment?

Progress: 100%

Details:
Briefly answer the question (at least a few decent sentences)

Your Response:

The easiest part of the assignment was implementing the logic for printing odd numbers in Problem 1. Using for loops and if statements felt intuitive, and the expected output was straightforward to test and confirm.

💾 Saved: 6/9/2025 7:04:03 PM

≡✏ Task #3 (0.33 pts.) - What was the hardest part of the assignment?

Progress: 100%

Details:
Briefly answer the question (at least a few decent sentences)

Your Response:

The hardest part was setting up and troubleshooting WakaTime with VS Code in a WSL environment. Since WakaTime runs differently in remote contexts, it required manual CLI installation and debugging configuration issues.

💾 Saved: 6/9/2025 7:04:38 PM