Course: IT114-002-S2025

Assignment: IT114 Java Problems

Student: Hitarth P. (hp627)

Status: Submitted | Worksheet Progress: 110.00%

Potential Grade: 11.00/10.00 (110.00%) Received Grade: 0.00/10.00 (0.00%)

Grading Link: https://learn.ethereallab.app/assignment/v3/IT114-002-S2025/it114-java-problems/grading/hp627

Instructions

- 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)
 - 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	oush origin M2-Homework
Crea	a Pull Peguest from M2_Homework to main and keep it one

- 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)
- 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
 - 2. git pull origin main

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

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

Combo Task:

Weight: 100%

Objective: Edit the 'printOdds' method to output odd values of the array

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)

Item:#1

Weight: 40%

Details:

Two screenshots are expected

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

Image Prompt



Snippet of relevant code showing solution





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

Task #1 (2 pts.) - Edit the `sumValues` method to sum the array val

Combo Task:

Weight: 100%

Objective: Edit the `sumValues` method to sum the array values and present them in a format with exactly two decimal places

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)

Item:#1

Weight: 40%

Details:

Two screenshots are expected

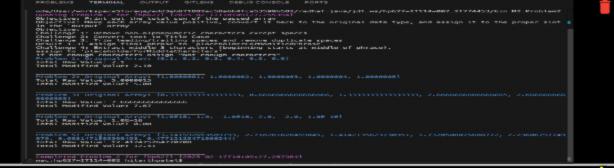
- 1. Snippet of relevant code showing solution (with ucid/date comment)
- 2. Full output of executing the program

Image Prompt



Snippet of relevant code





Output of executing





Saved: 2/17/2025 11:45:59 AM

Item:#2

Weight: 20%

Details:

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

■ Url Prompt

URL #1



https://github.com/hitarthpat/hp62

https://github.com/hitarthpat/hp627-IT114h002/nain/M2/Problem2.java

Saved: 2/17/2025 11:45:59 AM

Item:#3

Weight: 40%

Details:

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

Text Prompt

Your Response:

In order to solve the challenge, it needs to add all of the values provided to it. It then formats the number to the desired format. It does it by checking if each number has a remainder of 1 when divided by 2.



Saved: 2/17/2025 11:45:59 AM

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

Task #1 (2 pts.) - Edit the 'bePositive' method to make each value

Combo Task:

Weight: 100%

Objective: 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

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)

Item:#1

Weight: 40%

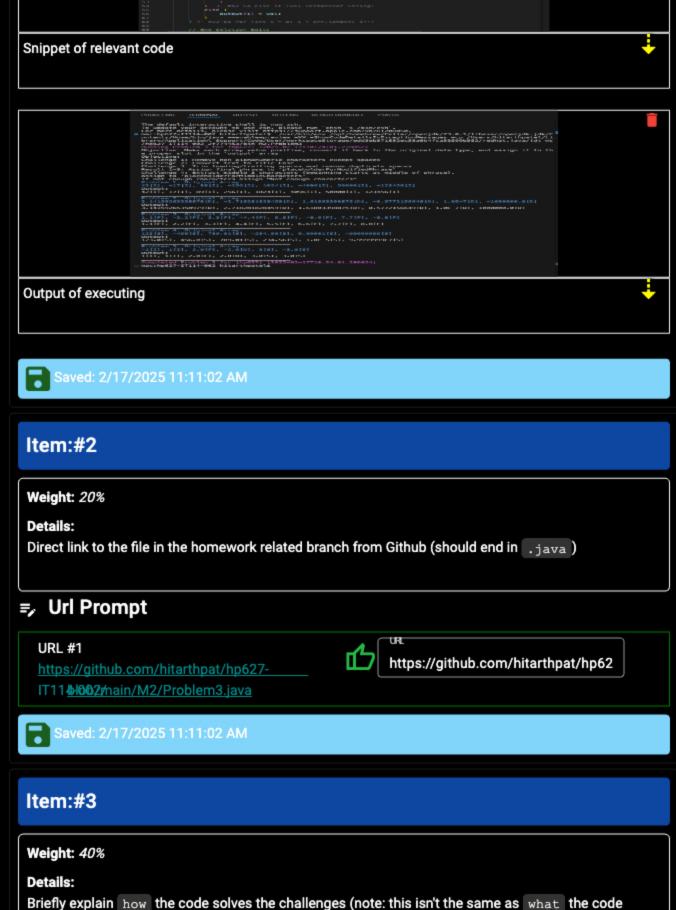
Details:

Two screenshots are expected

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

≡ Image Prompt





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

Text Prompt

Your Response:

The data provided is in many different data types, so the code needs to be able to identify the data

types and do the proper conversion in order to make it positive if it is negative. When the code identifies the data type, it does an operation specific to that data type in order to make it compatible with the Math.abs() method that returns a positive value.



Saved: 2/17/2025 11:11:02 AM

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

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

Combo Task:

Weight: 100%

Objective: Edit the `transformText` method to solve the challenges

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)

Item:#1

Weight: 40%

Details:

Two screenshots are expected

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

≡ Image Prompt

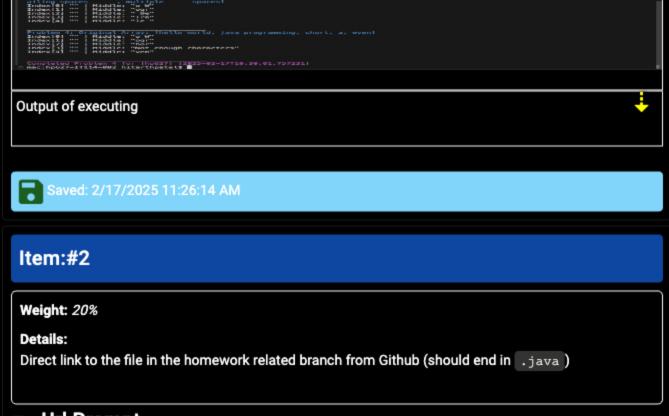
```
Snippet of relevant code

Snippet of relevant code

Snippet of relevant code
```

String placeholder ForModifiedFirese





≡, Url Prompt



https://github.com/hitarthpat/hp627-IT114h002/nain/M2/Problem4.java



https://github.com/hitarthpat/hp62

Saved: 2/17/2025 11:26:14 AM

Item:#3

Weight: 40%

Details:

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

Text Prompt

Your Response:

The code checks through every character in the words/phrases provided, and it checks if it is an alphanumeric character or space. If it is not, it does not include it in the final word that is constructed. It also checks for duplicate spaces and does not include them in the final string. To convert to titlecase, it checks for words after spaces and capitalizes them for the conversion.



Saved: 2/17/2025 11:26:14 AM

Combo Task:

Weight: 45%

Objective: Edit the `transformText` method to solve the extra credit challenge (challenge 4)

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)

Item:#1

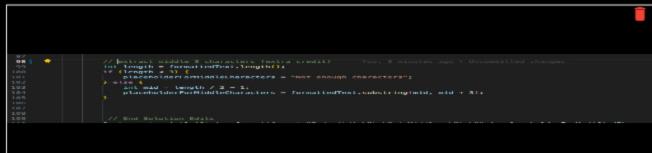
Weight: 40%

Details:

Two screenshots are expected

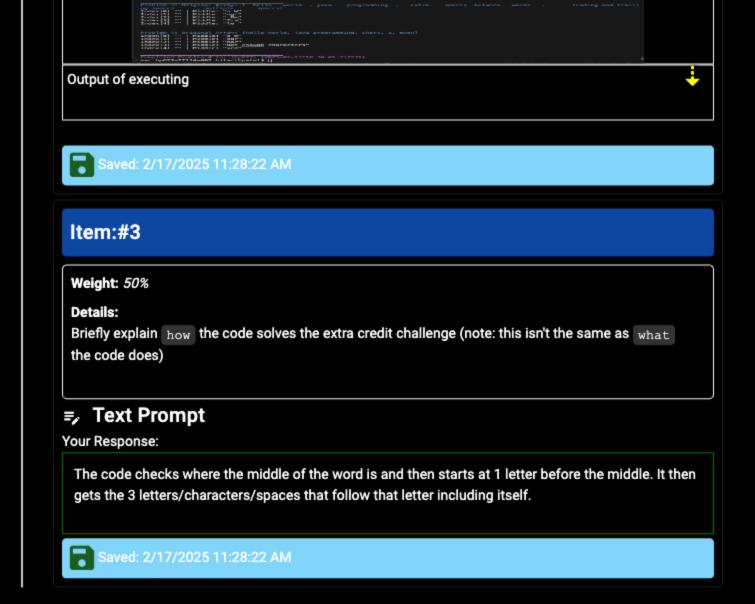
- 1. Snippet of relevant code showing solution (with ucid/date comment)
- 2. Full output of executing the program

Image Prompt



Snippet of relevant code





Section #5: (2 pts.) Misc

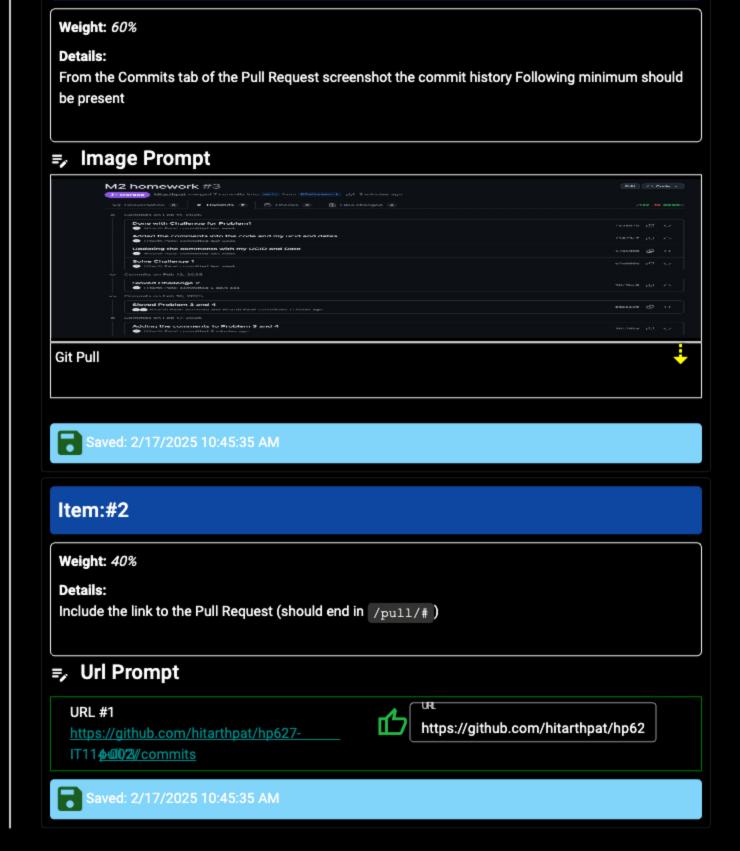
Task #1 (0.67 pts.) - Github Details

Combo Task:

Weight: 33.33%

Objective: Github Details

Item:#1



Task #2 (0.00 / 0.67 pts.) - WakaTime - Activity

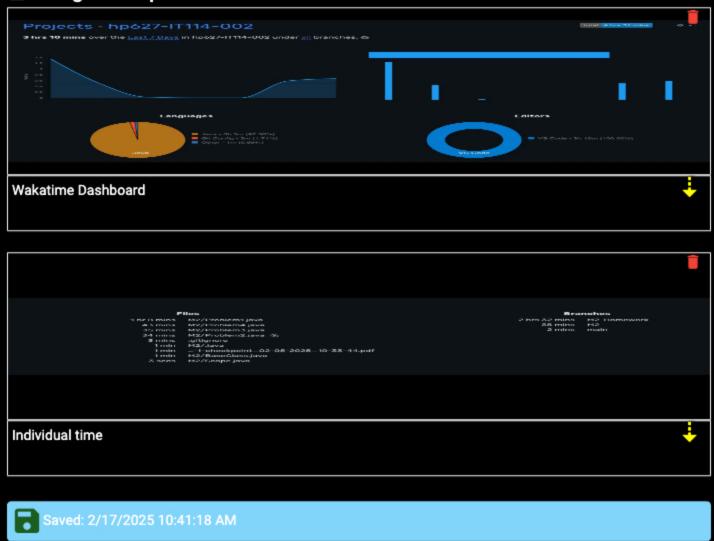
Weight: 33.33%

Objective: WakaTime - Activity

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

Image Prompt



Task #3 (0.00 / 0.67 pts.) - Reflection

Sub-Tasks:

Task #1 (0.00 / 0.33 pts.) - What did you learn?

Weight: *33.33*%

Objective: What did you learn?

Details:

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

Text Prompt

Your Response:

I learned how to manipulate strings and different methods. I learned how to iterate though an array while using a for-each loop code. Use of modulus operator (%) to check for odd numbers. Removing non-alphanumeric characters (\$%#&) while having spaces.



Saved: 2/17/2025 11:46:30 AM

Task #2 (0.00 / 0.33 pts.) - What was the easiest part of the a

Weight: 33.33%

Objective: What was the easiest part of the assignment?

Details:

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

Text Prompt

Your Response:

The easiest part of the assignment was seeing if the array had a odd Problem 1. All I had to do was check if the number was odd. While the code goes through each number in the array to check if its odd.



Saved: 2/17/2025 11:39:58 AM

Task #3 (0.00 / 0.33 pts.) - What was the hardest part of the

Weight: 33.33%

Objective: What was the hardest part of the assignment?

Details:

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

= Text Prompt

Your Response:

The hardest part of the assignment was the manipulation for problem 4. It required multiple text transformations in a certain pattern making it more intricate. Having to manually filtering out the special characters. Having to capitalize the first letter of each word while making everything else lowercase.



Saved: 2/17/2025 11:46:37 AM