Course: IT114-006-S2025

Assignment: IT114 Module 3 User Input Challenges

Student: Muhammad K. (muk)

Status: Submitted | Worksheet Progress: 96.30%

Potential Grade: 9.00/10.00 (90.00%) Received Grade: 0.00/10.00 (0.00%)

Grading Link: https://learn.ethereallab.app/assignment/v3/IT114-006-S2025/it114-module-3-user-input-

<u>challenges/grading/muk</u>

Instructions

- Ensure you read all instructions and objectives before starting.
- Create a new branch from main called M3-Homework
 - 1. git checkout main (ensure proper starting branch)
 - 2. git pull origin main (ensure history is up to date)
 - 3. git checkout -b M3-Homework (create and switch to branch)
- 3. Copy the template code from here: GitHub Repository M3 Homework
 - It includes CommandLineCalculator, SlashCommandHandler, MadLibsGenerator, a BaseClass and a stories folder with 5 stories (used for MadLibsGenerator). Put all into an M3 folder or similar (adjust package reference at the top if you chose a different folder name).
 - Immediately record to history
 - \supset git add .
 - □ git commit -m "adding M3 HW baseline files"
 - □ git push origin M3-Homework
 - ☐ Create a Pull Request from M3 Homework to main and keep it open
- 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.
 - □ Update the ucid variable
 - 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 M3-Homework
 - 4. On Github merge the pull request from M3 Homework to main
- 7. Upload the same PDF to Canvas
- 8. Sync Local
 - 1. git checkout main
 - 2. git pull origin main

Section #1: (3 pts.) Challenge 1 - Command Line Calculator (Add/sub)

Task #1 (3 pts.) - Edit the `main` method to solve the requirements

Combo Task:

Weight: 100%

Objective: Edit the 'main' method to solve the requirements

Details:

- Don't adjust the give code unless noted
- Challenge 1: Accept two numbers and an operator as command-line arguments (+ and -)
- Challenge 2: Allow integer and floating-point numbers
 - Ensure correct decimal places in output based on input (e.g., 0.1 + 0.2 → 1 decimal place)
- · Display an error for invalid inputs or unsupported operators
- 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/datecomment)
- Full output of executing the program (Capture 5 variations of tests)

Image Prompt

Snippet of code







Saved: 2/27/2025 12:33:16 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/MuhammadKhan621/muk-IT114-006/blob/M3-



https://github.com/MuhammadKha

Homework/M3/CommandLineCalculator.java



Saved: 2/27/2025 12:33:16 AM

Item:#3

Weight: 40%

Details:

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

Text Prompt

Your Response:

The code ensures that three arguments are provided: the first (a number), the second (either + or -), and the third (another number). If the user does not enter the correct format, they will be given a message. Depending on whether the user enters a + or -, the corresponding operation will be executed in the command line when the user types in the correct format. At the end, DecimalFormat is used to ensure the total is displayed with the correct number of decimal places.

Section #2: (3 pts.) Challenge 2 - Slash Command Handler

Task #1 (3 pts.) - Edit the `main` method to solve the requirements

Combo Task:

Weight: 100%

Objective: Edit the 'main' method to solve the requirements

Details:

- Don't adjust the give code unless noted
- Challenge 1: Accept user input as slash commands (Commands are case-insensitive)
 - "/greet <name>" → Prints "Hello, <name>!"
 - "/roll <num>d<sides>" → Roll <num> dice with <sides> and returns a single out
 - "/echo <message>" → Prints the message back
 - "/quit" → Exits the program
- Challenge 2: Print an error for unrecognized commands
- Challenge 3: Print errors for invalid command formats (when applicable)
- 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 (Capture 3 variations of each command except "/quit")

Image Prompt



Snipped of unfinished Code

Snipped of unfinished Code



Item:#2

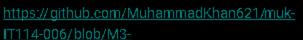
Weight: 20%

Details:

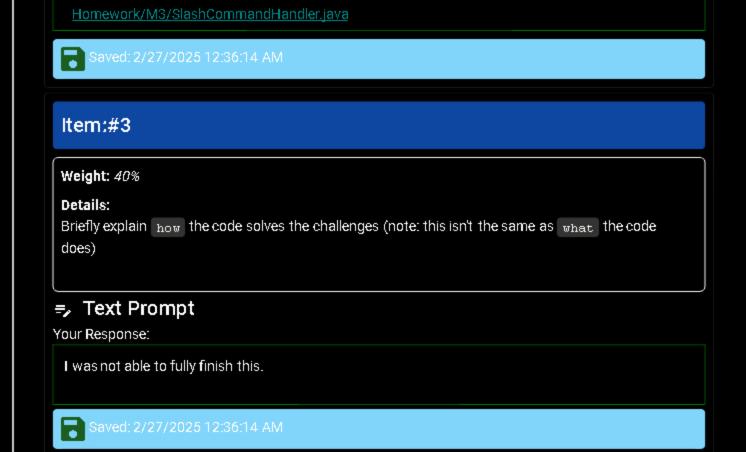
Direct link to the file in the homework related branch from Github (should end in .java)

Url Prompt

URL #1







Section #3: (3 pts.) Challenge 3 - Mad Libs Generator

Task #1 (3 pts.) - Edit the `main` method to solve the challenges

Combo Task:

Weight: 100%

Objective: Edit the 'main' method to solve the challenges

Details:

- · Don't adjust the give code unless noted
- Ensure you have the stories folder with the 5 stories
- Challenge 1: Load a random story from the "stories" folder
- Challenge 2: Extract each line into a collection (i.e., ArrayList)
- Challenge 3: Prompts user for each placeholder (i.e., <adjective>)

- Any word the user types is acceptable, no need to verify it it matches the placeholder type.
- · Any placeholder with underscores should display with spaces instead
- Challenge 4: Replace placeholders with user input (assign back to original slot in collection)
- 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 (Capture the process for at least 2 stories)

Image Prompt



Missing Caption





Saved: 2/27/2025 12:36:45 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/MuhammadKhan621/muk-



https://github.com/MuhammadKha

IT114-006/blob/M3-

Homework/M3/MadLibsGenerator.java



Saved: 2/27/2025 12:36:45 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:

I was not able to finish this

Saved: 2/27/2025 12:36:45 AM

Section #4: (1 pt.) Misc

Task #1 (0.33 pts.) - Github Details

Combo Task:

Weight: 33.33%

Objective: Github Details

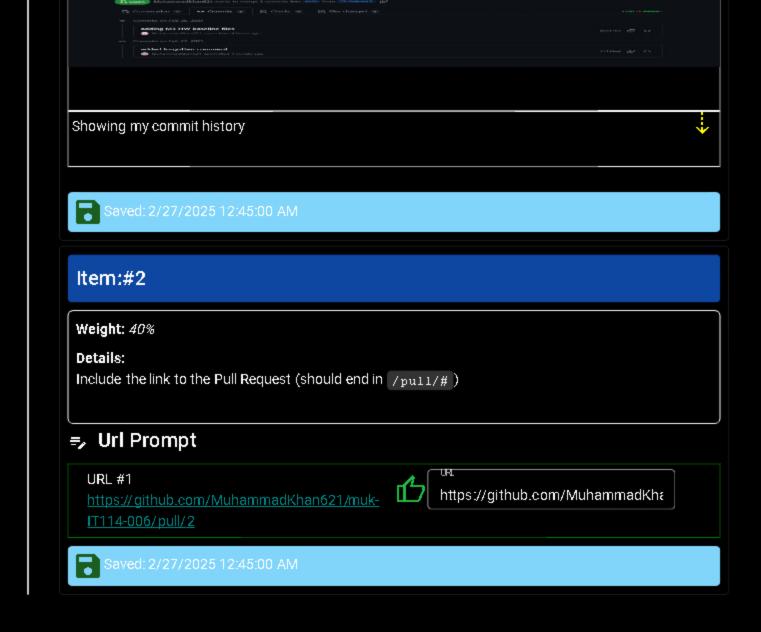
Item:#1

Weight: 60%

Details:

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

Image Prompt



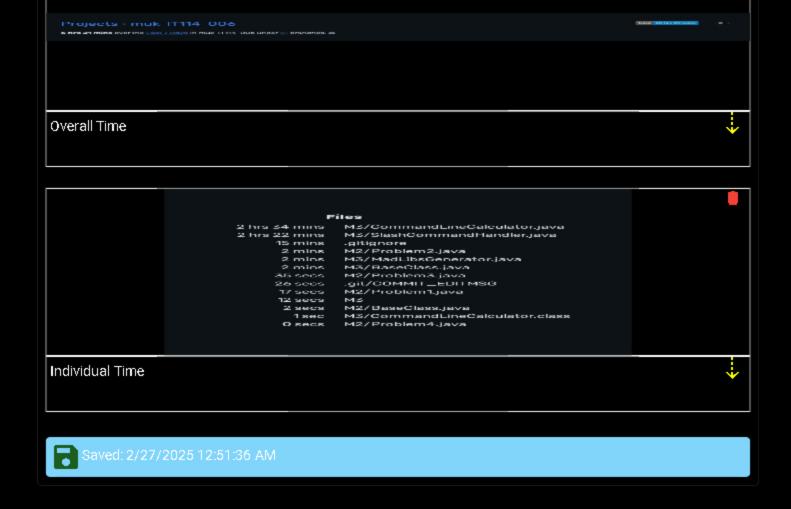
Task #2 (0.00 / 0.33 pts.) - WakaTime - Activity

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 / 0.33 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 take input from the command line when running the code rather than using a scanner to ask users for input. I learned how difficult coding can be.

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:

Only the CommandLineCalucaltor assignment was all right because it was straightforward for me.

Saved: 2/27/2025 12:56:28 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:

I'd say the SlashCommandHandler and MadLibsGenerator were the hardest even though I was not able to try doing the Madlibs assignment in time. I got a little done for the SlashCommand assignment, but I got stuck on some of it and did not finish all.



Saved: 2/27/2025 1:00:28 AM