Course: IT114-010-S2025
Assignment: IT114 Module 3 User Input Challenges
Student: Hector M. (hem)
Status: Submitted | Worksheet Progress: 100.00%
Potential Grade: 10.00/10.00 (100.00%)
Received Grade: 0.00/10.00 (0.00%)
Grading Link: https://learn.ethereallab.app/assignment/v3/IT114-010-S2025/it114-module-3-user-input-challenges/grading/hem

Instructions

1. Ensure you read all instructions and objectives before starting.
2. 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
        - `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
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
        - Update the `ucid` variable
        - 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 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)

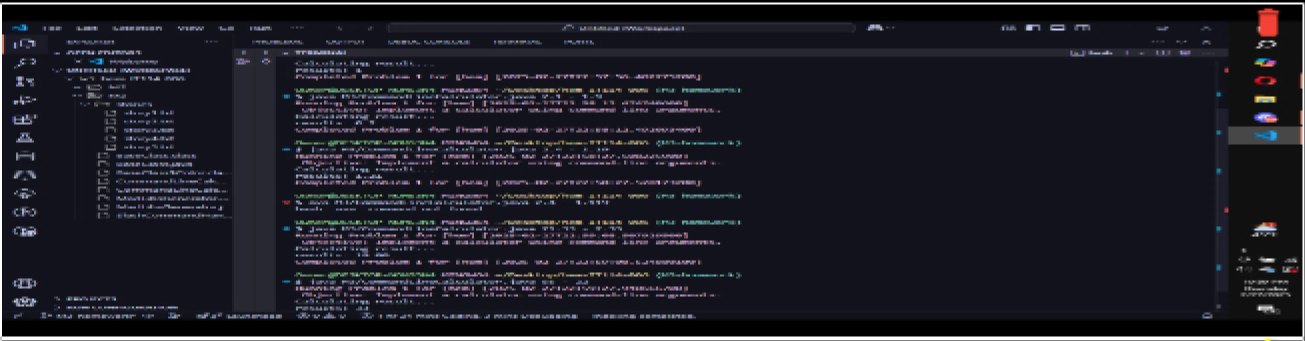
Item:#1

Weight: *40%*

Details:
Two screenshots are expected

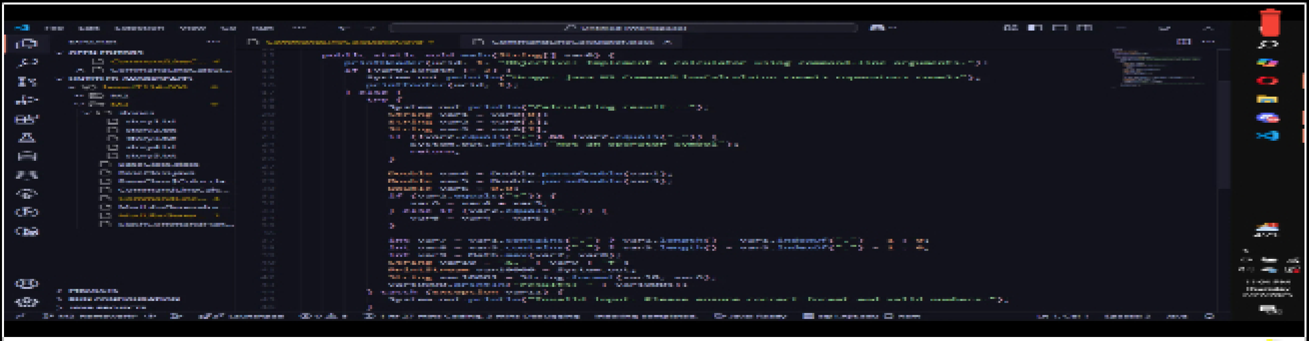
1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program (Capture 5 variations of tests)

Image Prompt



```
[illegible]
```

Output of the code with 5 different examples



code itself

Saved: 2/27/2025 11:10:51 PM

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://learn.ethereallab.app/assignment/v3/IT114-2025/it114-module-3-user-input-challenges/view/M3/CommandLineCalculator.java

URL
https://learn.ethereallab.app/assign

Saved: 2/27/2025 11:10:51 PM

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:

Well at first I tried taking the arguments num1 and num2 immediately as integers, but realized that wouldnt work. So i changed them to string1 and string2 and then changed them to intergers later in the code. I also took the operator varaible as a string and used a conditional statement to figure out whether the user wanted to add or subtract their two numbers. then the tricky part was really the decimal reformatting, at first i just tried finding the the index of the decimals of both num1 and num2, then subtracted the two to find the decimal of the output, that didnt work so i quickly googled some examples and found the math max function and implemented that. I then assigned that value to a variable and used that variable as the number when I formatted the ending strings decimal placements.

Saved: 2/27/2025 11:10:51 PM

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

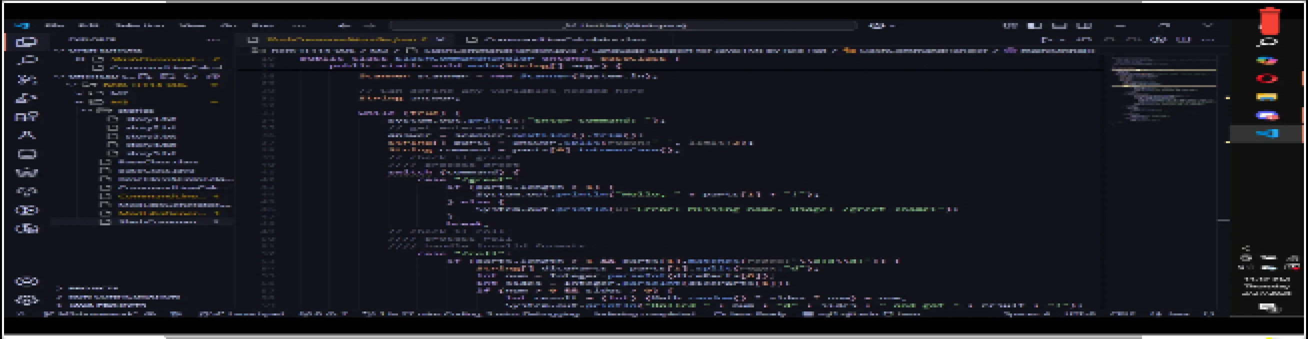
Weight: *40%*

Details:
Two screenshots are expected

1. Snippet of relevant code showing solution (with ucid/date comment)

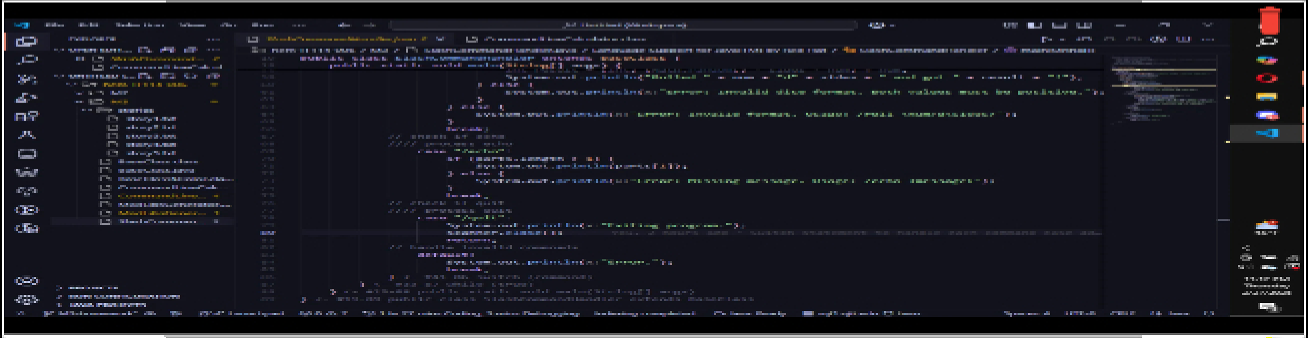
2. Full output of executing the program (Capture 3 variations of each command except "/quit")

Image Prompt



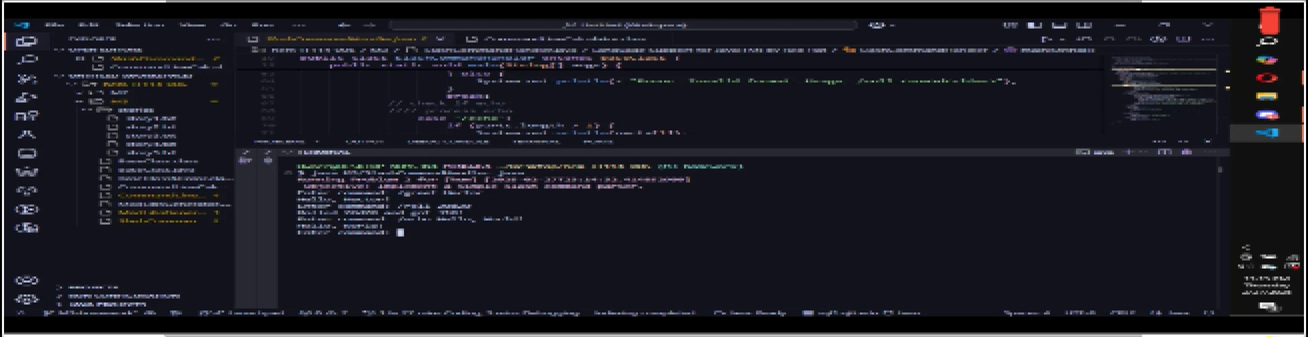
[illegible]

First part of the code



[illegible]

Part 2 of the code



[illegible]

3 different outputs

Saved: 2/27/2025 11:20:57 PM

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://learn.ethereallab.app/assignment/v3/IT114-S2025/it114-module-3-user-input-challenges/view/M3/SlashCommandHandler.java

URL
https://learn.ethereallab.app/assig

Saved: 2/27/2025 11:20:57 PM

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:

This one wasnt as hard of a struggle as the other two, it was a lot of trial and error. The first time I got the user input I had an issue with when I did something like "/greet" and had a space afterward I would get an error. I add the trim() method after NextLine() to ensure a cleaner input. Then the rest was simple trial and error with the switch statement making sure every case worked one by one. The biggest issue I had was implementing dice logic, I dont think it works properly still but the actually calling of the command does.

Saved: 2/27/2025 11:20:57 PM

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 if 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)

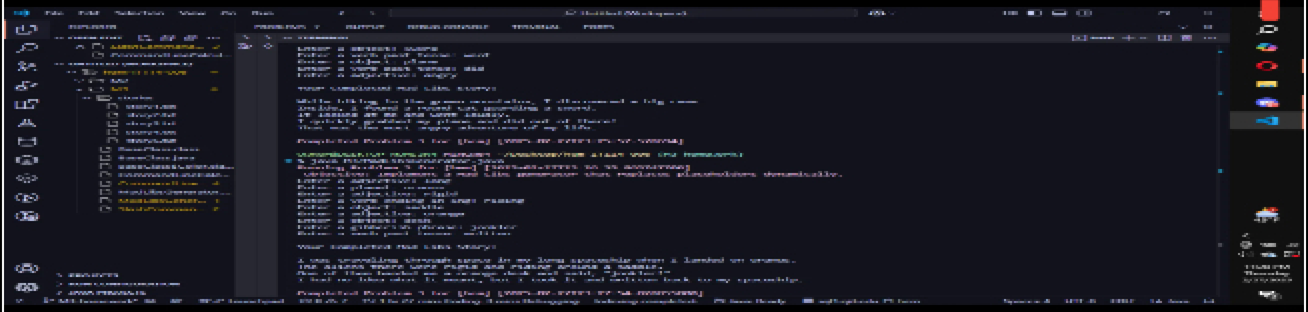
Item:#1

Weight: *40%*

Details:
Two screenshots are expected

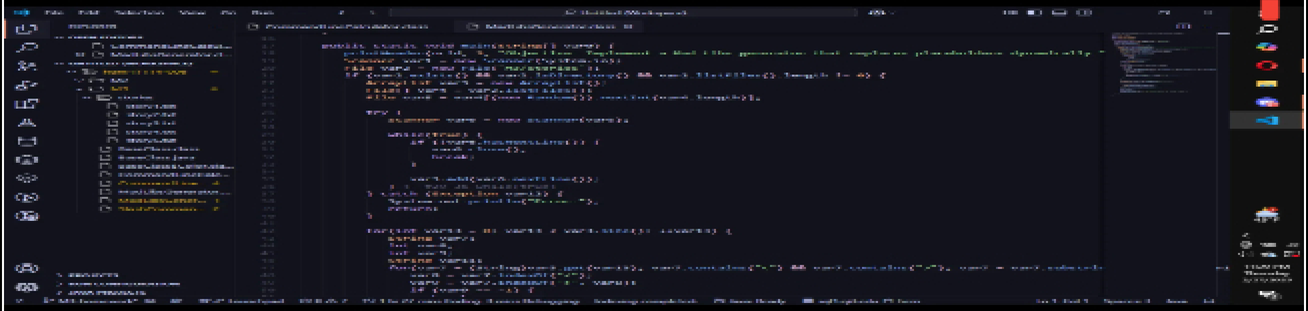
1. Snippet of relevant code showing solution (with ucid/date comment)
2. Full output of executing the program (Capture the process for at least 2 stories)

Image Prompt



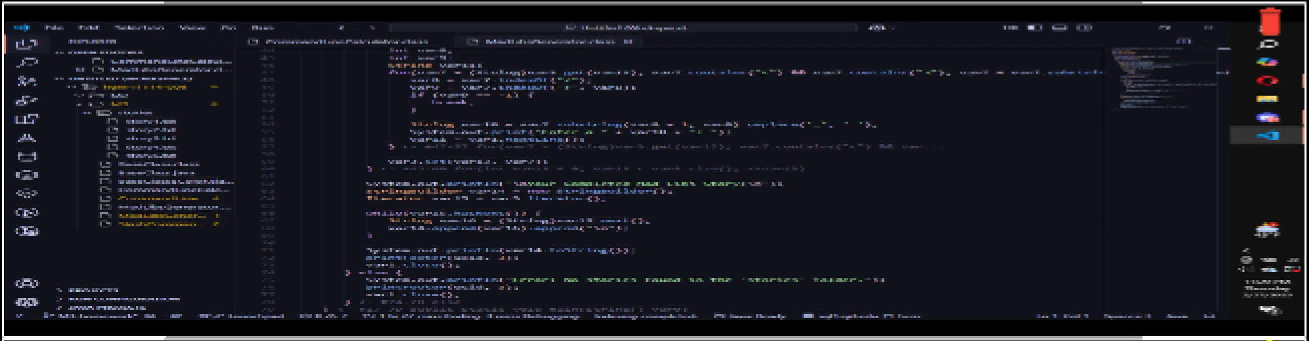
```
[illegible]
```

output for madlibs



```
[illegible]
```

code for madlibs


Code part 2

Saved: 2/27/2025 11:44:53 PM

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://learn.ethereallab.app/assignment/v3/IT114-2025/it114-module-3-user-input-challenges/view/M3/MadlibsGenerator.java

URL
https://learn.ethereallab.app/assigr

Saved: 2/27/2025 11:44:53 PM

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:

First to load a random story I made an array and put each file into said array. I then used math.Random and multiplied it by the length of the array to get a random array. The real issue I had was finding how I could single out the placeholder words and use them to let the user know what kind of word they had to type. I searched up a lot of references on stack overflow and found different code

I used as a framework to apply the challenge given. First I checked the line of the story if it contained a bracket (as in these <>), I ended up using things like index of and substring to find the placement of the brackets to find the word inside so that the user can know what they need to type. I then used the indexs to find the start and end of where the users input would be replaced, then replaced it. Finally using the string builder to finsh and present the whole story.

Saved: 2/27/2025 11:44:53 PM

Section #4: (1 pt.) Misc

Task #1 (0.33 pts.) - Github Details

Combo Task:

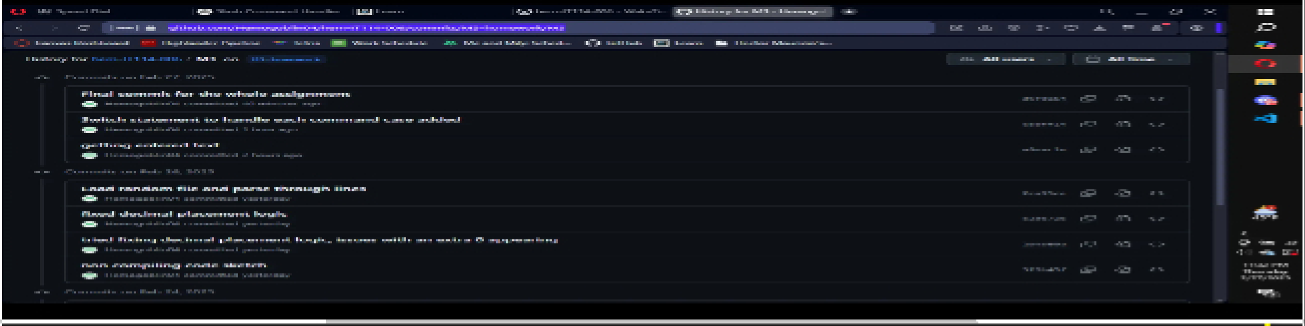
Weight: *33.33%*
Objective: *Github Details*

Item:#1

Weight: *60%*

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

Image Prompt



```
Commits on Feb 27, 2025
Final commit for the whole assignment
Switch statement to handle each command case added
getting entered text
Commits on Feb 26, 2025
Load random file and parse through lines
fixed decimal placement bugs
tried fixing decimal placement bugs, issues with an extra 0 appearing
non computing code sketch
Commits on Feb 24, 2025
```

Commit Histor

Saved: 2/27/2025 11:33:31 PM

Item:#2

Weight: *40%*

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

Url Prompt

URL #1
https://github.com/Hemogoblin04/hem-IT114-006 commits/M3-homework/M3

URL
https://github.com/Hemogoblin04/

Saved: 2/27/2025 11:33:31 PM

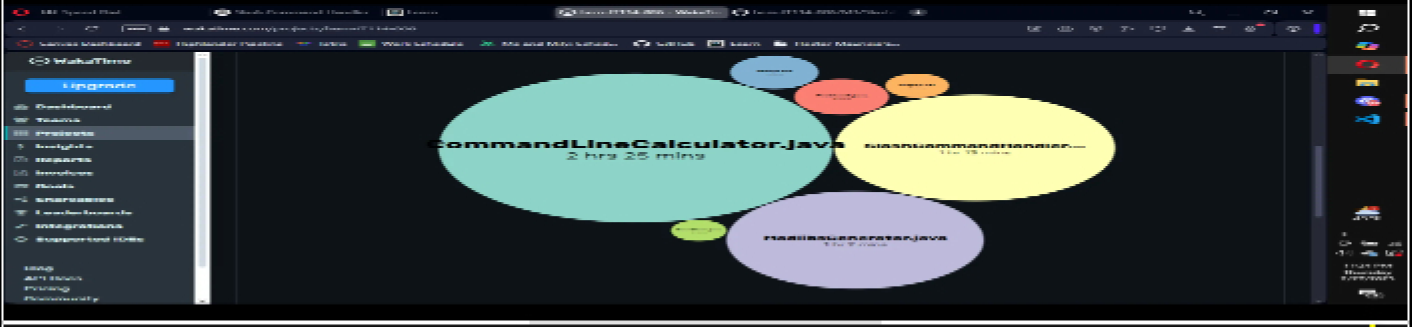
Task #2 (0.00 / 0.33 pts.) - WakaTime - Activity

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



wakatime

Saved: 2/27/2025 11:24:52 PM

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:

That I genuinely should pay more attention, I realized today in class as in on 2/27 my approach for the madlibs generator is grossly over complicated and that I need to do more sketching prior to coding because I did too much trial and error with my code since I didnt plan it out prior. Going forward I need to just plan a lot more and probably do my homework earlier.

Saved: 2/27/2025 11:35:25 PM

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 SlashCommandHandler was the easiest to me as it was really just a switch statement to handle the entire prompt and I didn't deal with any new hurdles per se just trial and error implementing a form of code I had already handled before.

Saved: 2/27/2025 11:22:26 PM

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 command line calculator genuienly made my headache when it came to the formatting of the decimals. It took me forever to figure out how I could format it since I didnt know where to find the proper commands to handle it.

Saved: 2/27/2025 11:23:40 PM