Submission Worksheet

Submission Data

Course: IT114-450-M2025
Assignment: IT114 Module 3 User Input Challenges
Student: Matt T. (mt85)
Status: Submitted | **Worksheet Progress:** 100%
Potential Grade: 10.00/10.00 (100.00%)
Received Grade: 0.00/10.00 (0.00%)
Started: 6/8/2025 10:47:32 PM
Updated: 6/8/2025 11:00:27 PM
Grading Link: https://localhost:8080/assignment/v3/IT114-450-M2025/it114-module-3-user-input-challenges/grading/mt85
View Link: https://localhost:8080/assignment/v3/IT114-450-M2025/it114-module-3-user-input-challenges/view/mt85

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)

Progress: 100%

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

Progress: 100%

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)

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 (Capture 5 variations of tests)

```
26      try {
27          System.out.println(x:"Calculating result...");
28          // extract the equation (format is <num1> <operator> <num2>)
29
30          // check if operator is addition or subtraction
31
32          // check the type of each number and choose appropriate parsing
33
34          // generate the equation result (Important: ensure decimals display as the
35          // longest decimal passed)
36          // i.e., 0.1 + 0.2 would show as one decimal place (0.3), 0.11 + 0.2 would shows
37          // as two (0.31), etc
38          // mt85 08-08-2025      You, 1 second ago • Uncommitted changes
39          System.out.println(x:"The answer is pi");
40
41      } catch (Exception e) {
42          System.out.println(x:"Invalid input. Please ensure correct format and valid numbers.");
43      }
44
```

example code

```
Matt@EtherealLab MINGW64 /d/projects/NJIT/2025/summer/IT114-450-M2025 (M3-Homework)
$ java M3.CommandLineCalculator food + food
Running Problem 1 for [mt85] [2025-06-08T22:46:23.869853400]
 Objective: Implement a calculator using command-line arguments.
Calculating result...
The answer is pi
Completed Problem 1 for [mt85] [2025-06-08T22:46:23.880874]
```

example output

Saved: 6/8/2025 10:48:18 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/MattToegel/IT114-450-M2025/blob/M3-Homework/M3/CommandLineCalculator.java

URL
https://github.com/MattToegel/IT

Saved: 6/8/2025 10:48:18 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:

it uses xyz to do abc

Saved: 6/8/2025 10:48:18 PM

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

Progress: 100%

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

Progress: 100%

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`
  - `"/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)

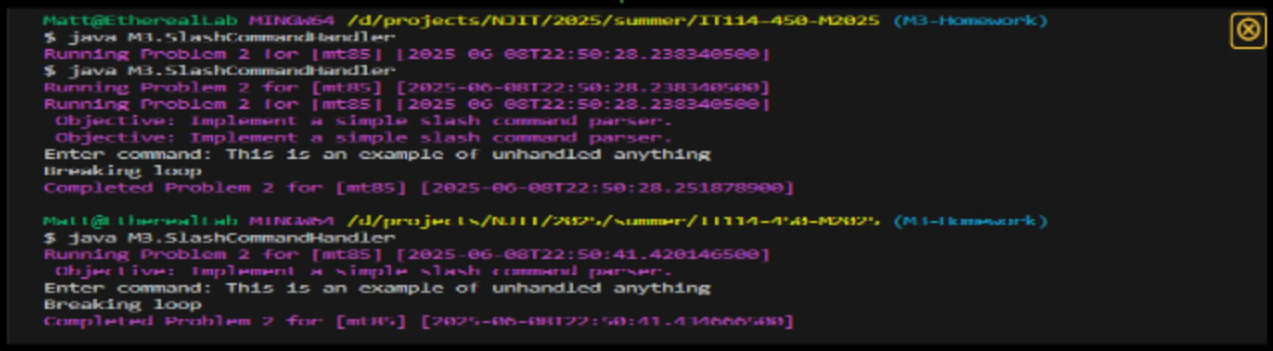
Part 1:

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 (Capture 3 variations of each command except "/quit")

example code



example output

Saved: 6/8/2025 10:51:58 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/MattToegel/IT114-450-M2025/blob/M3-Homework/M3/SlashCommandHandler.java

URL
https://github.com/MattToegel/IT

Saved: 6/8/2025 10:51:58 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:

utilized abc to solve each

Saved: 6/8/2025 10:51:58 PM

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

Progress: 100%

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

Progress: 100%

Details:

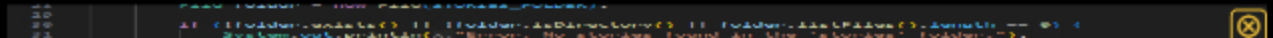
- Don't adjust the give code unless noted
- Ensure you have the `stories` folder with the 5 stories
- Challenge 1: Load a **random** story from the "stories" folder
- Challenge 2: Extract **each line** into a collection (i.e., ArrayList)
- Challenge 3: Prompts user for each placeholder (i.e., `<adjective>`)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

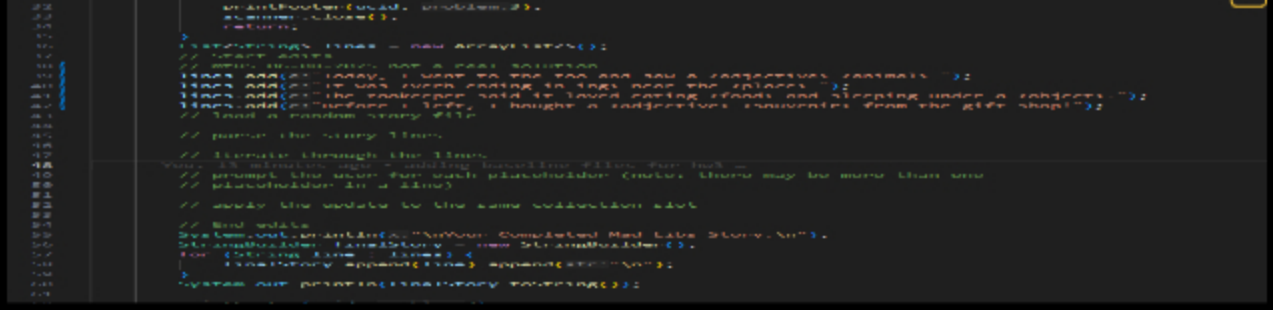
Part 1:

Progress: 100%

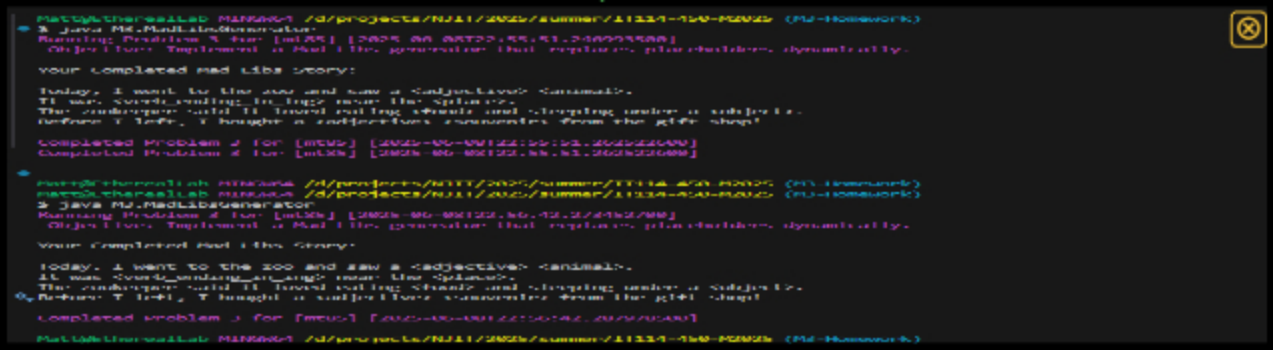
Details:
Two screenshots are expected

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





example solution



example output

Saved: 6/8/2025 10:57:52 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://localhost/assignment/v3/IT114-450-M2025/package%20M3;%20%20import%20java.io.File%20change%20to%20your%20ucid%20%20%20%20%20%20public%20static%20void%20main(Strir

URL #2
https://github.com/MattToegel/IT114-450-M2025/blob/M3-Homework/M3/MadLibsGenerator.java

URL
https://github.com/MattToegel/IT

Saved: 6/8/2025 10:57:52 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:

I used ... steps to solve each thing

Saved: 6/8/2025 10:57:52 PM

Section #4: (1 pt.) Misc

Progress: 100%

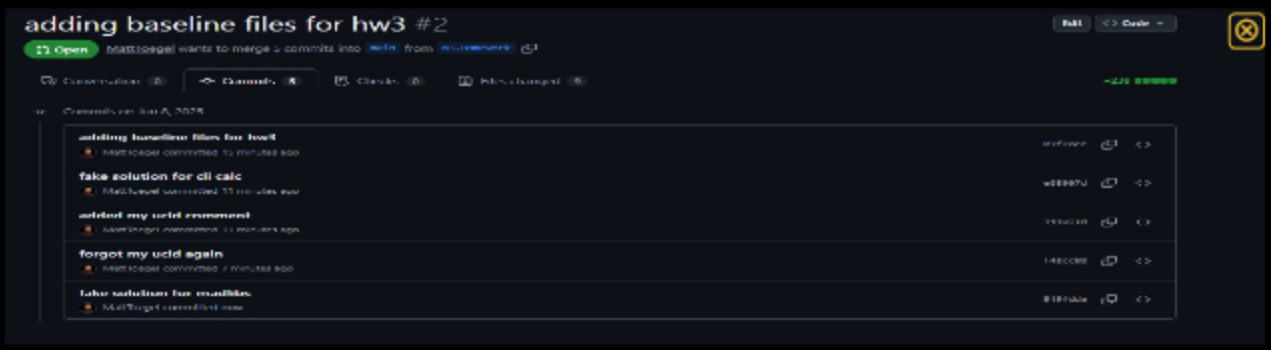
Task #1 (0.33 pts.) - Github Details

Progress: 100%

Part 1:

Progress: 100%

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



commit history for hw3

Saved: 6/8/2025 10:58:55 PM

Part 2:

Progress: 100%

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

URL #1
https://github.com/MattToegel/IT114-450-M2025/pull/2/commits

URL
https://github.com/MattToegel/IT

Saved: 6/8/2025 10:58:55 PM

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

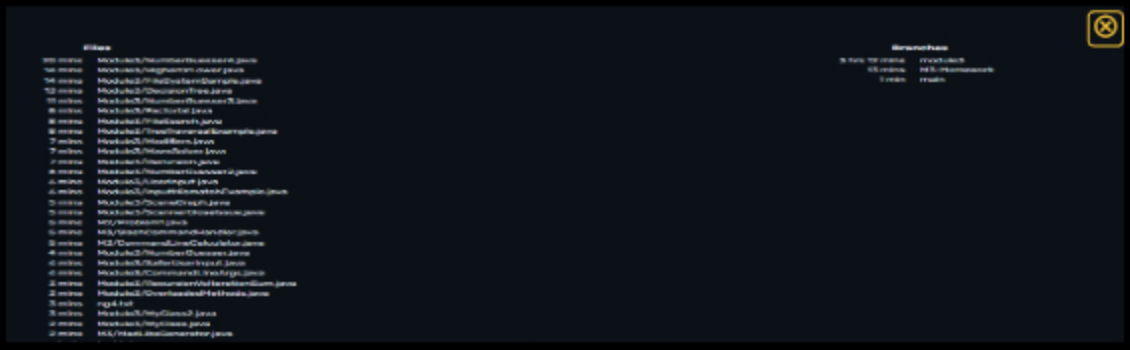
Progress: 100%

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 overview



wakatime specific

Saved: 6/8/2025 11:00:04 PM

Task #3 (0.33 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:

I learned ...

Saved: 6/8/2025 11:00:13 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 was ...

Saved: 6/8/2025 11:00:19 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 ..

Saved: 6/8/2025 11:00:27 PM