

Submission Worksheet

Submission Data

Course: IT114-450-M2025

Assignment: IT114 Module 3 User Input Challenges

Student: Alex P. (ap2869)

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

Potential Grade: 10.00/10.00 (100.00%)

Received Grade: 0.00/10.00 (0.00%)

Started: 6/16/2025 9:33:05 PM

Updated: 6/16/2025 9:54:38 PM

Grading Link: <https://learn.ethereallab.app/assignment/v3/IT114-450-M2025/it114-module-3-user-input-challenges/grading/ap2869>

View Link: <https://learn.ethereallab.app/assignment/v3/IT114-450-M2025/it114-module-3-user-input-challenges/view/ap2869>

Instructions

- Overview Link: <https://youtu.be/iowHMCKuj5o>

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., $0.1 + 0.2 \rightarrow 1$ 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)

```
String numbers = args[0];
String operator = args[1];
String number1 = args[2];
String number2 = args[3];
double num1 = Double.parseDouble(number1);
double num2 = Double.parseDouble(number2);
double result = 0;

// OPERATOR CHECKING
if (operator.equals("+")) {
    // Addition
    result = num1 + num2;
} else if (operator.equals("-")) {
    // Subtraction
    result = num1 - num2;
} else {
    System.out.println("Invalid operator. Please use only + or -");
}

// OUTPUT
System.out.println("Result: " + result);

// Error handling
try {
    // Parse arguments
    if (args.length != 4) {
        System.out.println("Invalid input. Please ensure correct format and valid numbers.");
    }
} catch (Exception e) {
    System.out.println("Invalid input. Please ensure correct format and valid numbers.");
}
```

snippet of code challenge 1

```
1. Running Problem 1 for Exp2A691 [2025-09-16T21:30:50.414466]
2. Running Problem 2 for Exp2A691 [2025-09-16T21:30:50.414466]
3. Running Problem 3 for Exp2A691 [2025-09-16T21:30:50.414466]
4. Running Problem 4 for Exp2A691 [2025-09-16T21:30:50.414466]
5. Running Problem 5 for Exp2A691 [2025-09-16T21:30:50.414466]
6. Running Problem 6 for Exp2A691 [2025-09-16T21:30:50.414466]
7. Running Problem 7 for Exp2A691 [2025-09-16T21:30:50.414466]
8. Running Problem 8 for Exp2A691 [2025-09-16T21:30:50.414466]
9. Running Problem 9 for Exp2A691 [2025-09-16T21:30:50.414466]
10. Running Problem 10 for Exp2A691 [2025-09-16T21:30:50.414466]
11. Running Problem 11 for Exp2A691 [2025-09-16T21:30:50.414466]
12. Running Problem 12 for Exp2A691 [2025-09-16T21:30:50.414466]
13. Running Problem 13 for Exp2A691 [2025-09-16T21:30:50.414466]
14. Running Problem 14 for Exp2A691 [2025-09-16T21:30:50.414466]
15. Running Problem 15 for Exp2A691 [2025-09-16T21:30:50.414466]
16. Running Problem 16 for Exp2A691 [2025-09-16T21:30:50.414466]
17. Running Problem 17 for Exp2A691 [2025-09-16T21:30:50.414466]
18. Running Problem 18 for Exp2A691 [2025-09-16T21:30:50.414466]
19. Running Problem 19 for Exp2A691 [2025-09-16T21:30:50.414466]
20. Running Problem 20 for Exp2A691 [2025-09-16T21:30:50.414466]
```

[illegible]

 Saved: 6/16/2025 9:54:38 PM

⇒ Part 2:

Progress: 100%

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

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

URL #1  <https://github.com/ap2869/IT-111450M3-homework/M3/CommandLineCalculator.java>

URL #1  URL <https://github.com/ap2869/IT-1110450M3-homework/M3/CommandLineCalculator.java> <https://github.com/ap2869/IT-114>



URL #1  URL <https://github.com/ap2869/IT-1110450M3-homework/M3/CommandLineCalculator.java> <https://github.com/ap2869/IT-114>

URL #1  URL <https://github.com/ap2869/IT-1110450M3-homework/M3/CommandLineCalculator.java> <https://github.com/ap2869/IT-114>



 Saved: 6/16/2025 9:54:38 PM

⇒ Part 3:

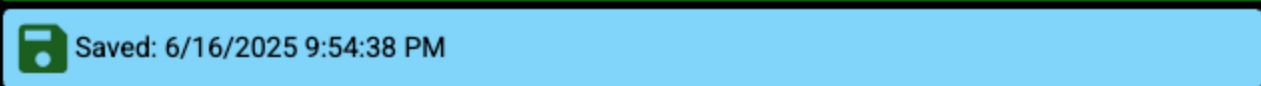
Progress: 100%


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

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

Your Response:

this one is simple. the user is asked to enter a number an operator and another number. but the operator can only be + or -. if neither of these operator is chosen an error occurs. the result then prints assuming its operator is - or +



 Saved: 6/16/2025 9:54:38 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)

- Details:**

 - Don't adjust the give code unless noted
 - Challenge 1: Accept user input as slash commands (Commands are case-insensitive)

Progress: 100%

Details:

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

URL #1

<https://github.com/ap2869/IT-114/blob/master/homework/M3/SlashCommandHandler.java>



URL

<https://github.com/ap2869/IT-114>



Saved: 6/16/2025 9:52:45 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:

/ if (command.startsWith("/echo")) for each slash command a / is assigned and once a slash command is chosen it goes to its assigned code. a if /echo is assigned a message is given if /greet is assigned a greeting is given if /dice is assigned a dice is rolled. the user inputs the side a and how many dice. and if /quit is assigned the program quits and gives a goodbye message



Saved: 6/16/2025 9:52:45 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

- Any placeholders 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 uid/date comment)
2. Full output of executing the program (Capture the process for at least 2 stories)

```
// MadLibs Generator
// Author: [Your Name]
// Date: [Current Date]

// Constants
const ADJECTIVES = ['happy', 'sad', 'tired', 'excited'];
const VERBS = ['run', 'jump', 'dance', 'sing'];
const NOUNS = ['cat', 'dog', 'bird', 'fish'];
const PLACEHOLDERS = ['_adjective_', '_verb_', '_noun_'];

// Function to pick a random element from an array
function getRandomElement(arr) {
  return arr[Math.floor(Math.random() * arr.length)];
}

// Function to generate a MadLibs story
function generateMadLibsStory() {
  const story = `
    One day, I was feeling ${getRandomElement(ADJECTIVES)}.
    I decided to go for a walk in the park.
    I saw a ${getRandomElement(NOUNS)} that was ${getRandomElement(ADJECTIVES)}.
    It was so ${getRandomElement(ADJECTIVES)} that I decided to pick it up.
    I took it home and it became my new ${getRandomElement(NOUNS)}.
    I was so ${getRandomElement(ADJECTIVES)} about it that I decided to share it with my friends.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the store.
    They bought a lot of ${getRandomElement(NOUNS)} and they were all so ${getRandomElement(ADJECTIVES)} that they decided to go home.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to bed.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the moon.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the stars.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the end of the world.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the beginning of time.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the center of the earth.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the top of the mountain.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the bottom of the ocean.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the middle of the desert.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the edge of the universe.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the heart of the matter.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the soul of the man.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the mind of the body.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the spirit of the world.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the love of the people.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the peace of the world.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the happiness of the people.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the joy of the world.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the love of the people.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the peace of the world.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the happiness of the people.
    They were all so ${getRandomElement(ADJECTIVES)} that they decided to go to the joy of the world.
  `;
  return story;
}
```

snippet of code challenge 3


```
> java -Dpath.separator=/ MadLibsGenerator
Running Problem 3 for [ap2869] [2025-06-16T21:20:07.219995]
Objective: Implement a Mad Libs generator that replaces placeholders dynamically.
Story Picked: story4.txt
Please write a adjective:
big
Please write a verb:
run
Please write a object:
rock
Please write a adjective:
little
Please write a object:
stone
Please write a food:
pizza
Please write a adjective:
heavy
Please write a animal:
dog
Please write a adjective:
ugly

Your Completed Mad Libs Story:

My best friend invented a big machine that can run.
All you need is a rock, and it will turn it into a little stone.
Yesterday, I put a pizza in it, and out came a heavy dog!
Now, everyone calls me the ugly scientist!

Completed Problem 3 for [ap2869] [2025-06-16T21:20:08.699259]
~/Desktop/811T/Summer/114/IT-114-458-PE3-summer25
```

output of challenge 3 code

 Saved: 6/16/2025 9:49:21 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/ap2869/
IT-114M3-
homework/M3/MadLibsGenerator.java](https://github.com/ap2869/IT-114M3-homework/M3/MadLibsGenerator.java)



URL

<https://github.com/ap2869/IT-114>

 Saved: 6/16/2025 9:49:21 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:

pretty simple the code gets a random file from the "stories" in an array list get the lines from the story that was chosen, and replaces and input the words that have < > with words that the user has to input. once that happened the word is replaced. the user is asked to enter an adajetive, verd, animal, etc. with that a mad lib is formed.



Saved: 6/16/2025 9:49:21 PM

Section #4: (1 pt.) Misc

Progress: 100%

☰ Task #1 (0.33 pts.) - Github Details

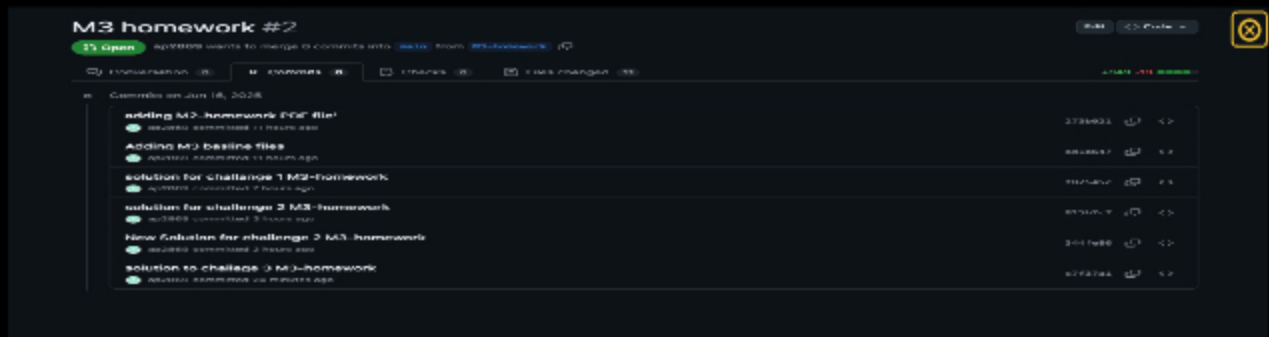
Progress: 100%

Part 1:

Progress: 100%

Details:

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



snippet of commit history



Saved: 6/16/2025 9:40:36 PM

Part 2:

Progress: 100%

Details:

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

URL #1

1

URL



Saved: 6/16/2025 9:40:36 PM

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

Progress: 100%

Details:

- Visit the [WakaTime.com](https://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

Projects • IT-114-450

7 hrs 57 mins over the Last 7 Days in IT-114-450 under all branches. ⓘ

snippet of waka time project

Files		Branches	
3 hrs 53 mins	M3/SlashCommandHandler.java	7 hrs 57 mins	M3 homework
2 hrs 6 mins	M3/CommandLineCalculator.java		
1 hr 42 mins	M3/ModLibGenerator.java		
8 mins	IT3Java.java		
5 mins	M4/CommandLineCalculator1.java		
1 min	M2/DasecClass.java		
57 secs	CommandLineCalculator.java		
33 secs	M3/SlashCommandHandler1.java		
19 secs	SlashCommandHandler.java		
17 secs	M2/Problem1.java		
16 secs	M4/ResetClass.java		
12 secs	CommandLineCalculator1.java ⓘ		
11 secs	M3/stories/story1.txt		
0 secs	M3/SlashCommandHandler.class		
7 secs	M3/test.java		
7 secs	SlashCommandHandler1.java		
6 secs	M3/stories/story2.txt		
5 secs	test.java		
4 secs	Java		

snippet of waka time files



Saved: 6/16/2025 9:42:22 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 alot from hw3. i actually didnt know about the catch and try method. it becamese very difficult in the beginning. i actually had to do some practice examples outside of the hw to finally get the hang of what was happening. but once i did get the hang of it it was simple. kinda



Saved: 6/16/2025 9:46:36 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 again the thought process of how i was going to do each assignment. i do have some backgroundf in java .. a few things i already had an idea but most of the things are new to me . but i kept reading and rereading and as it went on it becamese easier and clearer.



Saved: 6/16/2025 9:45:07 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 of the HW was getting everything to work together. i spent the majority of the time debugging going back and forth on the examples i felt like quitting but i pused through and as i kept going to each assignment it became slightly easier.



Saved: 6/16/2025 9:43:43 PM

