

Project 1 Report

Garrett Nix

1-23-2026

1 Introduction

As a generality, this project has been a useful experience for getting familiarized with the Java programming language. The language itself does not seem entirely complex syntactically, however, there are some points where finding an equivalency between familiar concepts between it and C++ caused many hours scouring the documentation pages of Oracle, as well as W3Schools. The main point of learning in this project seemed to be getting used to modularization, and a more Object-Oriented Programming (OOP) approach. This report aims to provide a personal reflection of the project and things that I had learned along the way and potential things that could be done better for the rest of the semester.

2 Learned Material

As stated, some material and concepts of OOP were already familiar to me as I have taken some courses where these concepts were taught in a brief sense. I have also spent the better part of a year working on a personal TUI project, and have thus reinforced some of the concepts I learned in my own time with this project as well. Syntactically, I learned some interesting “tricks” from reading up on the documentation of Java syntax, as well as some open resources online. The main use of these resources were to familiarize myself with the Java syntax and think of how to use them in my own projects, opting to learn concepts rather than be given the answer directly from forums or LLMs that provide a lacking sense of knowledgeable gain from being handed a solution. Some examples of nice points of learning I had during the project was learning about checked and unchecked exceptions when I was struggling to figure out why I was receiving an error with my custom exceptions I was trying to create. I had also learned some interesting tricks with syntax throughout this project that were fun to discover. A couple examples of this are the for each syntax as well as calling a method directly in a construction of a new class object rather than separating the operations on newlines. This was useful as I only needed to use one method for a single object, so it was an interesting way to avoid 2 lines and simply have a single line.

3 Some Things I Could Do Better in The Future

There were some things early on that I realized may be beneficial to notice early in the course so that way I can more effectively avoid potential pitfalls and time wasting habits when working on the coming projects. One of these of course is the scope of the program. I became very invested in attempting to create some very overly complex functions that are outside of my knowledge base that I wasted an unfortunate amount of time attempting to figure out. For instance, one function would take in the header of a CSV, parse through each column name, and then sort based on the example csv. The intention was to make a robust portion of the program with this, but I soon realized I needed to have not only an implementation of a stack, but an implementation of some type of sorting algorithm that could do this. I soon figured out that this was well beyond what I could think of within the time frame of the assignment, and so I had to reel in my ambitions and come back to it another time. I suppose that’s the main aspect that I need to realize for

the semester is that I can still have robust programs, but there may need to be times where an exception is the best choice given a restraint, and refactoring can come later.

4 Concluding Remarks

All in all, I believe this project to be a good introduction into the course as it was well-rounding in its approach. Many things were learned and experienced, good or bad. I hope to use this knowledge gained and begin to create better, more sophisticated code in the coming projects, and throughout the rest of the semester.