stock_broker writeup

WO1 Clayton E. Williams

October 2023

0.1 Project Summary

stock_broker is a command line utility tool that, through the use of command-line options, allows a broker to manage customers, their accounts, and the various functions associated with them, such as depositing and withdrawing funds, and buying and selling stocks.

0.2 Challenges

The main challenge with this project was simply making the transition from C back to Python. There were several solutions to the project that I over-engineered, or over complicated by not knowing or remembering some of the built in Python functions. Since I just finished DSA, I didn't have any difficulty with the project design or program flow, but as with implementation, forgetting much of what makes Python so easy to use was not considered in the initial design. As I worked through the project and started remembering how to effectively write Python, I saw where areas of the code could have been done better and more 'Pythonic'. This resulted in self-induced stress and Challenges as I ended up rewriting and refactoring a large part of my code base.

0.3 Successes

My biggest success on this project was my Interface base class that I was able to create and the 3 menu sub-classes from it. I used classes for my menus on ipyfamily, and while they worked, the implementation was terrible. Each menu was its own class, so I ended up with 4 or 5 different classes, with only one instance of each. Being able to make each menu an instance inheriting from the base class gave me much more functionality and makes it much easier to quickly look at the code and understand what it is doing. After the first OOP block, I am pleased with my ability to see my short-comings from my menus in ipyfamily, design a better way to implement it, and actually implement it.

0.4 Lessons Learned

My design process has gotten exponentially better since my first projects, but now being somewhat proficient in C and Python, the language being used needs more attention during the design process. As mentioned in Challenges there were some difficulties in the transition between languages. During the design process, more thought needs to be put into not just what functionality is needed, but how

it will be achieved in that specific language. I tackled this like a DSA project, likening the classes needed to C structs, but spending more time addressing how to interact with and manipulate those classes in Python would have taken additional design time, but saved refactor time.