ITAS185 Name: Rajgursher Singh

Assignment Self-Assessment

*Assignment Number:* 01

1. What did you learn in completing this assignment?  
     
   “Measure a thousand times and cut once.” - Code slow, think thoroughly before touching the keys. Impatience is counterintuitive and leads to frustration.  
     
   Think about the bigger picture and then piece by piece, put it all together.  
     
   Git commit small changes. Git committing might seem like a tedious task but doing it frequently is a must to maintain sanity.   
     
   Clean as you go. Professionals do not “refactor” code, they write what avoids “refactoring” later.  
     
   Finishing is important! - In my assignment, this is the principle I used when I added pagination for displaying movies. There were only 5 movies in the sample dictionary but that doesn’t mean it is never going to grow into thousands. Hence, adding pagination now is a good measure to avoid fixing the code later.
2. How did you go about completing the assignment and solving problems you had along the way?

I will be honest, even though I had written the IPO diagram and pseudocode, I underestimated the complexities and just jumped into writing code. I was soon humbled by the complexities that arose out of writing chunky and repetitive functions. Once I got to the writing error handling part, I lost all track of what function does what and how – regardless of naming them with highly descriptive names.  
  
I soon realized the importance of writing modular code, and to always think things through before acting. I started breaking functions to use helper functions to avoid repetition and adding comments where needed.

1. What did you have difficulty with?

In terms on conceptualization, I had difficulty accepting that quality of the work or code must not be sacrificed due to time constraints. And learning that learning is about taking time to understand.  
  
In terms of technicalities, I had difficulty making a proper flow of the program. The IPO diagram helped. But I think there is a need of a separate hierarchal diagram that breaks down the process to have helper functions. Initially, I thought that I’d only need to build CRUD but in reality, I had to build several other helper functions. Some for formatting lists, others for simply validating choices and so on.

1. What did you do well?

I believe I was able to build a foundational program. The functions are small and reusable. There is some good error handling. And the program is easy to follow even by a novice computer user.  
  
Also, I kept in mind that this is a basic program so it is developed in a way that it can later be turned into an API so that users can build their own libraries, or build their websites with a GUI which using Python script for backend. Better yet, this program can be changed faster with a framework such as Django or Flask.

1. How many hours did you spend in completing this assignment?

21:5 hours or 47 pomodoros

1. What took you the most time?

Short answer: Refactoring code when adding error handling took too much time and I was still confused. I was also not using git when I started writing this assingments so the only way for me to go back was to start with a clean slate. This wasted a lot of time initially but I was able to do it right the next time.