Working on this data project was extremely challenging but rewarding in the end. My project worked with year-long batting data from the players in the MLB World Series in 2022.

One of the challenges I faced came immediately in understanding the requirements of the project and what the finished product would look like. Until this point, the lessons that we have learned have been in isolation, and I struggled to understand how to apply them in a greater context. I worked through this challenge by reading questions on Discord and talking with peers to understand what the project expectations were.

Once I understood the project outline, I found it easier than expected to lay out the initial code in Python. Pulling in the data sources was not as big of a hurdle as I initially thought, and I was able to define functions to complete each of the tasks in the instructions with relatively little difficulty.

Despite the ease of the initial coding process, working through errors in the code was a challenge. I also struggled with ensuring my code's flexibility in handling the different input and output formats in the instructions. Each data type came with unique challenges, and the transformations required to convert from one format to another were difficult in their own way. For instance, converting CSV files to JSON was relatively straightforward, but handling the edge cases of inconsistent data types or missing values proved difficult.

Looking forward, this ETL utility has the potential for use in a wide range of data projects I might encounter in the future. In any data analysis role, automating the cleaning, transforming, and loading of datasets is a critical task. The ETL pipeline could serve as the foundation for automating data pipelines, which would save time on repetitive tasks.. It also provides a useful way to handle moving data from multiple systems with varying formats into one.