Data Project 1: Challenges, Easy and Hard Aspects, and Future Utility

One of the main challenges we encountered was making sure our program could handle various data formats while providing clear error messages. It was essential to build a strong error-handling system because user inputs can be unpredictable. For instance, if someone tried to upload an unsupported file type, we needed to catch that issue and let them know instead of letting the program crash. Another challenge was finding the right datasets for our project. We wanted to use an API for the JSON data to gain experience with loading files and making API calls, but it was tough to find a suitable one that offered accessible data. Many APIs had restrictions or complicated authentication processes, which made things even more challenging. On top of that, we had to manage different data structures that came with each format. Each dataset had its unique structure, so we spent time understanding these formats to make sure our program could transform the data properly. This meant we had to make our code flexible enough to adapt to various input types without breaking.

We found the pandas library to be quite user-friendly for data manipulation. Once we got the hang of it, reading files and transforming DataFrames became straightforward. The syntax made it easier to add or remove columns, which helped streamline our workflow. We were also lucky to find an API that didn't require an API key since it was hosted on a public repository. This made our job simpler, as we didn't have to deal with complex authentication steps. With direct access to the data, we could focus more on processing it rather than navigating through API limitations.

One of the trickier parts of the project was understanding all the deliverables for the assignment. At first, the various requirements felt overwhelming. We had to juggle multiple tasks, like fetching data from different sources, converting formats, and implementing effective error handling, while also keeping track of the summaries for both input and output data. This complexity really highlighted the need to step back and clarify each component before moving forward.

In the future, we think this tool will be beneficial for various data projects, especially when working with data from different sources. By automating the extraction, transformation, and loading processes, we can save time and reduce manual work. Its flexible design makes it easy to adjust for different datasets, which is a practical advantage for data analysts and engineers. Summarizing the input and output data not only clarifies the process but also helps maintain data integrity. This way, we can be confident about the data we're using and trust the results we get.