For this project, I built an ETL pipeline to analyze how weather patterns have changed in Charlottesville, VA over the past decade. The pipeline extracted data from a local historical weather CSV and from the Weatherstack API, transformed the data for consistency, merged the datasets, and stored the results in a SQL database. I also implemented summary statistics and visualized trends in temperature and humidity over time.

Challenges Encountered:

One of the main challenges was understanding the structure of the local CSV data and ensuring the correct columns were retained for analysis. Another issue involved working with the Weatherstack API, where I had to handle potential API errors and incomplete data (e.g., fields missing or in unexpected formats). Integrating these two sources—one static and historical, the other real-time—required careful standardization of data types, especially for date formatting and column naming.

Aspects That Were Easier Than Expected:

Using pandas for data manipulation was more intuitive than I initially expected. Once I figured out the column names and formatting, cleaning and merging the datasets went smoothly. Additionally, exporting the merged data to different formats (CSV, JSON, and SQL) was simple using built-in pandas and SQLAlchemy functionality.

Aspects That Were More Difficult Than Expected:

One surprising difficulty was plotting the data meaningfully. While simple line charts were straightforward, dual-axis plots (e.g., comparing temperature and humidity) required careful configuration to avoid misinterpretation. Also, making sure the merged dataset had consistent data types between the API and the CSV source took more debugging than expected.

Usefulness of This Utility:

This type of ETL utility is incredibly valuable for future data science projects. It allows automated ingestion of up-to-date and historical data from various sources, supports flexible format conversions, and prepares datasets for analysis or machine learning workflows. Having a modular and reusable pipeline like this can save significant time and help ensure consistent preprocessing across multiple projects. Additionally, this setup provides a foundation for building dashboards or integrating with cloud-based analytics tools.