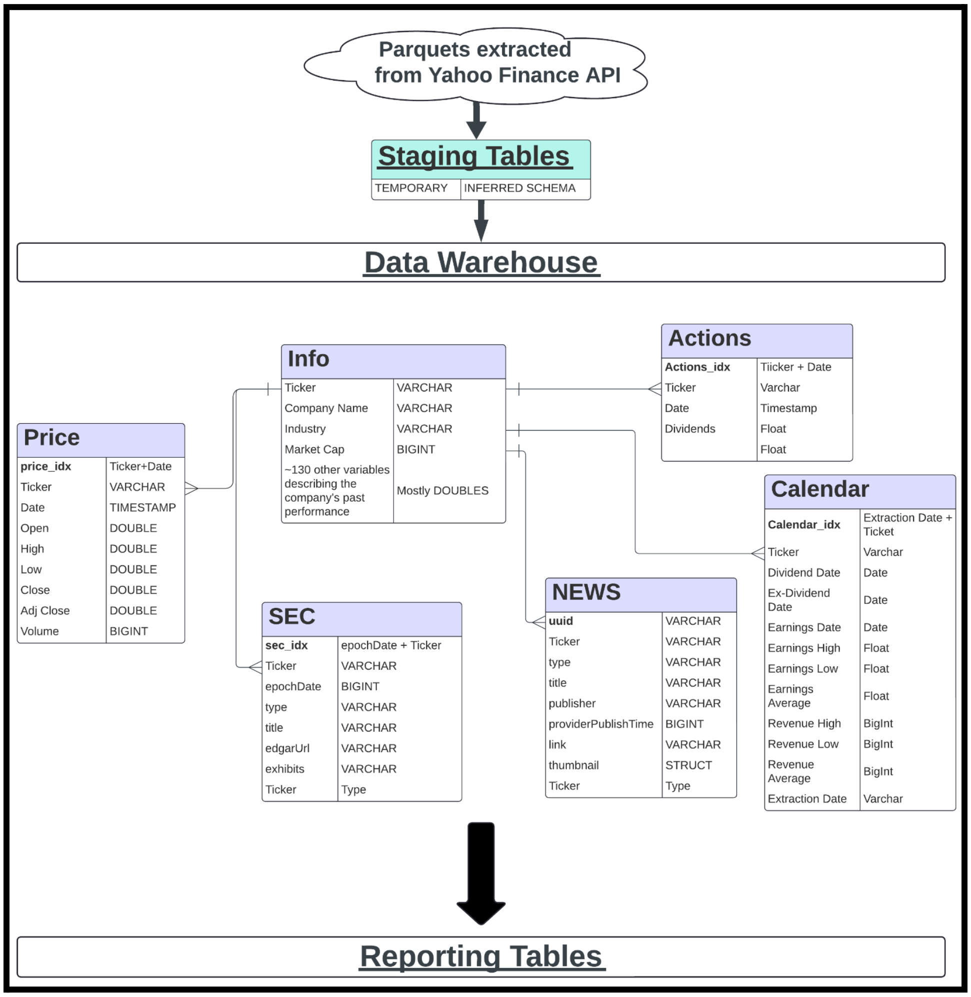
Team Project Phase 1 - Team 7

Objective: To create a unified dashboard that **aggregates** data on **stocks** the user wants to keep track of, **automatically updating** every day after market close and using **GenAI** to help users summarize and analyze the markets, specifically the latest news and SEC filings.

Data Model:



Info	
Ticker	VARCHAR
Company Name	VARCHAR
Industry	VARCHAR
Market Cap	BIGINT
~130 other variables describing the company's past performance	Mostly DOUBLES

Actions	
Actions_idx	Tiicker + Date
Ticker	Varchar
Date	Timestamp
Dividends	Float
	Float

Price	
price_idx	Ticker+Date
Ticker	VARCHAR
Date	TIMESTAMP
Open	DOUBLE
High	DOUBLE
Low	DOUBLE
Close	DOUBLE
Adj Close	DOUBLE
Volume	BIGINT

Calendar	
Calendar_idx	Extraction Date + Ticket
Ticker	Varchar
Dividend Date	Date
Ex-Dividend Date	Date
Earnings Date	Date
Earnings High	Float
Earnings Low	Float
Earnings Average	Float
Revenue High	BigInt
Revenue Low	BigInt
Revenue Average	BigInt
Extraction Date	Varchar

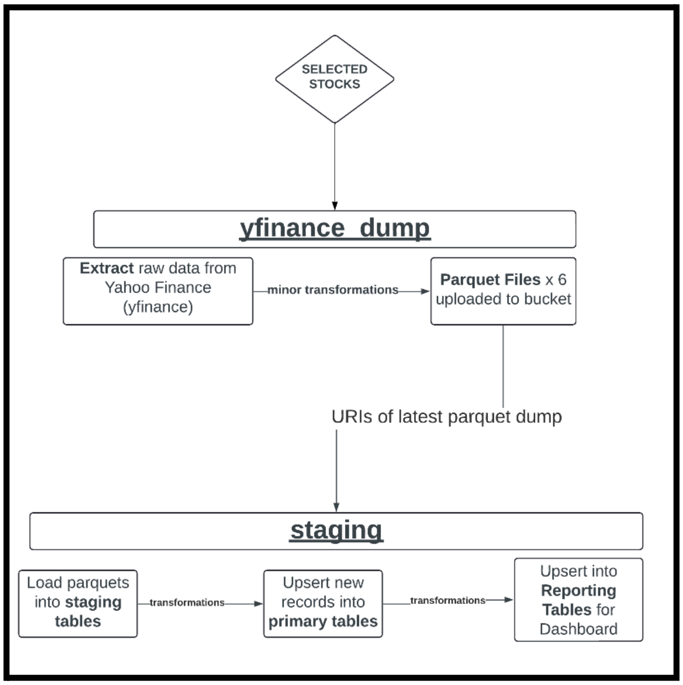
NEWS	
uuid	VARCHAR
Ticker	VARCHAR
type	VARCHAR
title	VARCHAR
publisher	VARCHAR
providerPublishTime	BIGINT
link	VARCHAR
thumbnail	STRUCT
Ticker	Type

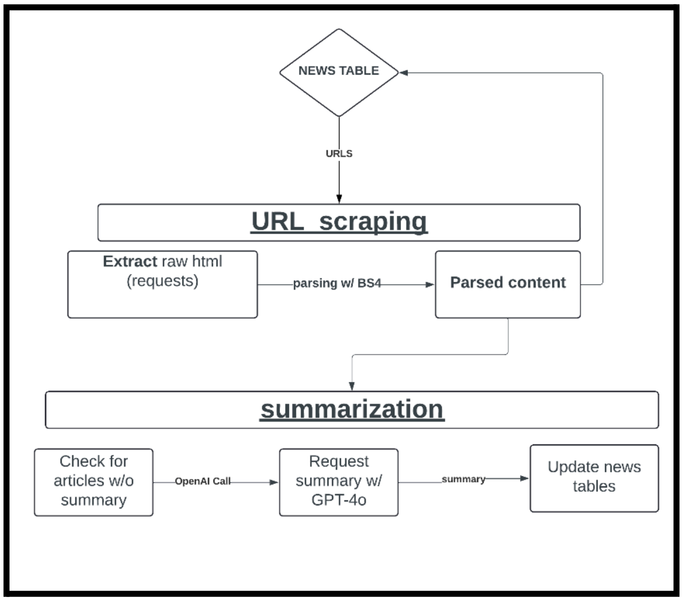
SEC	
sec_idx	epochDate + Ticker
Ticker	VARCHAR
epochDate	BIGINT
type	VARCHAR
title	VARCHAR
edgarUrl	VARCHAR
exhibits	VARCHAR
Ticker	Type

1. **Raw Data** The raw data consists of parquet files retrieved from the Yahoo Finance API. They contain structured, but unprocessed price, fundamental, and text data for the top 10 US stocks .
2. **Staging Tables** Staging tables are created and populated based on the data from the parquet files. The schema is inferred from the parquet file set during the extraction stage.
3. **Data Warehouse**: Our primary tables in our data warehouse are then updated with the latest records from the staging tables (upsert). This approach allows for efficiently ingest parquet files, perform necessary transformation and load into a more query-friendly format in the data warehouse.
4. **Reporting Tables** The reporting tables were created in MotherDuck using SQL and are designed for use in Superset for data visualization and analysis. The reporting tables mainly contain measurable, quantitative data, and dimension tables providing descriptive attributes to analyze the facts across various dimensions.

Team Project Phase 1 - Team 7

Pipeline Design:


SELECTED STOCKS
yfinance_dump
Extract raw data from Yahoo Finance (yfinance)
minor transformations
Parquet Files x 6 uploaded to bucket
URIs of latest parquet dump
staging
Load parquets into staging tables
transformations
Upsert new records into primary tables
transformations
Upsert into Reporting Tables for Dashboard



The pipeline begins with setting up a robust schema capable of efficiently handling a diverse range of data extracted from Yahoo Finance, including stock market data, news, and SEC filings. Given the broad scope of the dataset, the schema is meticulously designed to accommodate these various data types and ensure smooth integration. Cloud Run functions are then employed to manage different stages of the data workflow. These functions are responsible for creating the schema, loading, transforming, and extracting data, and ultimately transferring it into staging tables hosted on MotherDuck. This structured approach ensures the data is well-organized and prepared for subsequent analysis. **Pipeline steps**:

- **yfinance_dump** - Uses the yfinance python package to fetch information on select stocks, unpack pandas dataframes into a SQL-compatible format, and upload the information as parquet files (passing on the URIs to the next step)
    - **Files Created** - Price, Company Info. SEC, News, Actions and Calendar
- **staging** - Loads parquet files generated by yfinance_dump into temporary staging tables, performs transformations (generating unique IDs, converting epochs to datetimes…) and upserts new records into data warehouse's primary tables, which are then used for our dashboard

The pipeline is integrated with Prefect, the orchestration tool to automate and streamline the entire data flow. A worker pool is set up to manage and automate task execution, with the pipeline scheduled to run automatically at the 5pm (close of the stock market each day). This automation ensures that all the data—from stock performance to news updates and regulatory filings—is updated promptly, providing a seamless and scalable solution for ongoing data management and analysis.

Team Project Phase 1 - Team 7

Analytics: We tested an analytics pipeline that can extract, summarize and analyze news articles

- **URL Scraping** - We extract text from URLs present in the News tables using python, requests and BeautifulSoup. We plan to expand this to the SEC table as well, though this requires a more complex process (downloading PDFs → parsing text, images and tables)
- **Summarization** - We used OpenAI's GPT-4o to summarize articles scraped in the previous step, and analyze potential impact on the stock price to provide users with an easy to digest snapshot of each stock's news feeds.

These functions are still being tested, and have not been integrated into our Prefect flow. However, the preliminary results are promising, allowing us to display concise **summaries** and **analysis** of articles natively within our dashboard, eliminating the need to access multiple sites and reducing the quantity of text data needed to be manually parsed by users for market research.

Reporting: We created a dashboard combining information from all our tables using Superset connected to our data warehouse on MotherDuck. We manually created tables for charts using MotherDuck & Superset's SQL capabilities. We plan to automate these steps and integrate in Prefect next phase. We opted for Superset owing to its ease of use and robust visualization and reporting capabilities, allowing us to expand our suite of charts and reports as our pipeline expands in breadth and complexity.

Development Practices: Throughout our project, we used GitHub for version control and project management. We established a repository management structure to track tasks, assign responsibilities, and monitor progress, where one team member serves as the primary repository administrator, responsible for reviewing and managing merge requests to avoid overlapping changes.
Link: https://github.com/tangyum/BA-882-Pipeline-Project

Next Steps: For our next two phases, we will focus on enhancing our pipeline and cloud data warehouse to better perform comprehensive stock market analysis and establish a strong technical base for future development. In terms of data, we plan to broaden our dataset to include more stocks and sector-level information, as well as extract detailed content from SEC filings. This will provide a more detailed and granular understanding of market trends and individual company's performance. Our Prefect flow acts as a stable foundation upon which we can build out and scale up our data collection activities.

Our analysis will be improved by refining the summarization process using advanced LLMs, offering daily recaps for selected stocks, and potentially providing data-driven investment recommendations for investors, giving them more concise, valuable information for their investment decisions at a glance. Since we are using Cloud Run Functions, we can effectively isolate and test each addition to our pipeline before integrating it with the broader flow.

Regarding reporting, we will make improvements on our Superset dashboard and explore more ways to integrate interactive features such as a chatbot. If Superset cannot handle the interactive features we have planned, we will explore lower-level options such as Streamlit and Chainlit.