MARKET DATA ANALYTICS PIPELINE

PHASE 1:

1. Project setup and environment configuration
   1. Create environment using – python -m venv <env name>
   2. Activate environment using - <env name>\Scripts\activate
   3. Install yfinance package – pip install yfinance
2. Use a json file to refer to the tickers for which data needs to be extracted- sample format of data in the json –

{

"tickers": [

{"symbol": "AAPL", "name": "Apple Inc."},

{"symbol": "MSFT", "name": "Microsoft Corporation"},

…

}

1. Create python script – extract.py which uses yfinance api to fetch and store data
   1. It takes three arguments-
      1. --path\_tickers - Path for the ticker list
      2. --path\_extract - Path for extracted raw data
      3. --business\_date - Business date for data extract
   2. Extracted data is stored at - f"{out\_path}/raw/{business\_date}"
   3. One file is created per ticker – AAPL, MSFT, ...

PHASE 2:

1. Google cloud setup
   1. Log in to GCS console and setup a new project.
   2. Enable BigQuery API for the project
   3. Use the BigQuery console to run the DDL statements-
      1. CREATE SCHEMA `<project id>.raw\_data`;
      2. CREATE SCHEMA `<project id>.analytics`;
   4. For testing out, use BQ console to create table and manually load json data
   5. E.g. we create aapl\_raw table in the raw schema for AAPL
2. dbt setup

Step 1: Install dbt-bigquery

$ pip install dbt-bigquery

$ dbt –version

Step 2: Initialize dbt project

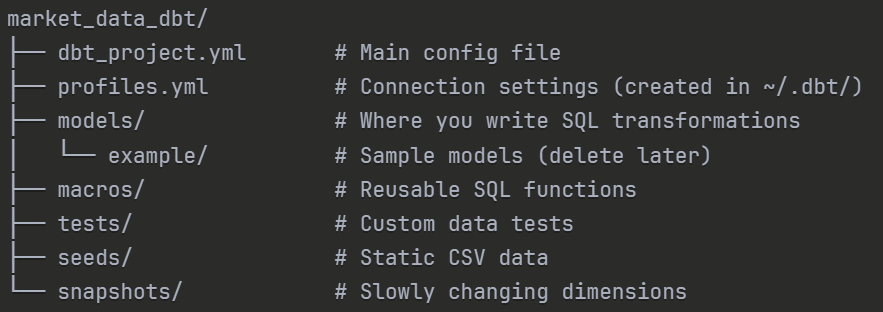
$ cd <*Your project root>*

$ dbt init market\_data\_dbt

It will prompt you for:

* **Which database?** → bigquery
* **Authentication method?** → oauth
* **Project ID?** → <gcp project id>
* **Dataset (schema)?** → analytics
* **Threads?** → 1 (default is fine)
* **Location?** → US

This creates a market\_data\_dbt/ folder with dbt project structure.



Also need to install googlecloud sdk

Download the installer: https://cloud.google.com/sdk/docs/install#windows

Run the installer (GoogleCloudSDKInstaller.exe)

Follow the prompts (default settings are fine)

Important: Check the box "Run gcloud init" at the end

After Installation:

$ gcloud auth application-default login

$ gcloud config set project market-data-pipeline-alpha

$ cd market\_data\_dbt

$dbt debug

Phase 3:

1. Dbt model setup:
   1. Step 1: Clean up example models and create folder structure
      1. *# Inside market\_data\_dbt/* $ rm -rf models/example
      2. $ mkdir models/staging
      3. $ mkdir models/marts
   2. Step 2: Define your source (raw BigQuery table)
      1. Create models/staging/sources.yml:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

version: 2

sources:

- name: raw\_data

database: market-data-pipeline-alpha

schema: raw\_data

tables:

- name: aapl\_raw

description: "Raw daily market data from yfinance for AAPL"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. Step 4: Build your first staging model
     1. Create models/staging/stg\_daily\_prices.sql:

SELECT

CAST(Date AS DATE) as date,

'AAPL' as ticker,

CAST(Open AS FLOAT64) as open\_price,

CAST(High AS FLOAT64) as high\_price,

CAST(Low AS FLOAT64) as low\_price,

CAST(Close AS FLOAT64) as close\_price,

CAST(Volume AS INT64) as volume,

CAST(Dividends AS FLOAT64) as dividends,

CAST(`Stock Splits` AS FLOAT64) as stock\_splits

FROM {{ source('raw\_data', 'aapl\_raw') }}

* + 1. dbt run
  1. Step 6: Verify in BigQuery

SELECT \* FROM `market-data-pipeline-alpha.analytics.stg\_daily\_prices` LIMIT 5;

Edit dbt\_project.yml and remove these lines:

yaml

models:

market\_data\_dbt:

example: *# ← Delete*

+materialized: view

* 1. Step 7: Create dim\_tickers - Create models/marts/dim\_tickers.sql:

SELECT DISTINCT

ticker,

-- Add metadata columns as needed later

CURRENT\_TIMESTAMP() as created\_at

FROM {{ ref('stg\_daily\_prices') }}

* 1. Step 9: Create fct\_market\_daily

SELECT

CONCAT(ticker, '\_', CAST(date AS STRING)) as market\_daily\_key,

date,

ticker,

open\_price,

high\_price,

low\_price,

close\_price,

volume,

dividends,

stock\_splits

FROM {{ ref('stg\_daily\_prices') }}

* 1. dbt run
  2. Verify all 3 tables in BigQuery:

SELECT \* FROM `market-data-pipeline-alpha.analytics.stg\_daily\_prices` LIMIT 5;

SELECT \* FROM `market-data-pipeline-alpha.analytics.dim\_tickers` LIMIT 5;

SELECT \* FROM `market-data-pipeline-alpha.analytics.fct\_market\_daily` LIMIT 5;

1. Add data quality tests
   1. Create models/marts/schema.yml

version: 2

models:

- name: dim\_tickers

description: "Dimension table containing unique ticker symbols"

columns:

- name: ticker

description: "Stock ticker symbol"

tests:

- unique

- not\_null

- name: fct\_market\_daily

description: "Fact table with daily OHLCV market data"

columns:

- name: market\_daily\_key

description: "Primary key: ticker\_date"

tests:

- unique

- not\_null

- name: close\_price

description: "Closing price"

tests:

- not\_null

- name: volume

description: "Trading volume"

tests:

- not\_null