**market.py** Python的「功能規格說明」(Functional Specification)。

**📄 功能規格說明：market.py**

**1. 系統目的**

本程式主要負責從 **Polygon API** 取得股市資料，包括：

* 判斷市場是否開盤
* 查詢所有股票**收盤價** (End-of-Day, EOD)
* 查詢單一股票即時或收盤價格
* 將市場資料快取並寫入資料庫

若無法成功從 Polygon 取得資料，則回傳隨機價格作為 fallback 機制。

**2. 系統變數**

* polygon\_api\_key
  + 從環境變數 POLYGON\_API\_KEY 讀取
  + 用於存取 Polygon API
* polygon\_plan
  + 從環境變數 POLYGON\_PLAN 讀取
  + 可能值："paid"、"realtime" 或空值
* is\_paid\_polygon (bool)
  + 是否為付費帳號
* is\_realtime\_polygon (bool)
  + 是否為即時帳號

**3. 功能規格**

**3.1 判斷市場是否開盤**

def is\_market\_open() -> bool

* 功能：
  + 呼叫 Polygon API get\_market\_status()
  + 回傳市場狀態是否為 "open"

**3.2 取得所有股票的 EOD 價格**

def get\_all\_share\_prices\_polygon\_eod() -> dict[str, float]

* 功能：
  + 查詢 SPY 的前一日收盤資料，確定日期 (處理時區問題)
  + 呼叫 Polygon API get\_grouped\_daily\_aggs()，取得該日所有股票的收盤價
  + 回傳字典 { 股票代號: 收盤價 }

**3.3 取得指定日期的市場資料 (快取版)**

@lru\_cache(maxsize=2)

def get\_market\_for\_prior\_date(today)

* 功能：
  + 從資料庫讀取指定日期的市場資料
  + 若資料不存在：
    - 呼叫 get\_all\_share\_prices\_polygon\_eod() 取得最新 EOD 資料
    - 將結果寫入資料庫
  + 回傳該日期的市場資料 (字典)

**3.4 取得單一股票的 EOD 價格**

def get\_share\_price\_polygon\_eod(symbol) -> float

* 功能：
  + 取得當日日期字串
  + 從快取市場資料中找出指定股票代號的收盤價
  + 若查無資料，回傳 0.0

**3.5 取得單一股票的即時最小價**

def get\_share\_price\_polygon\_min(symbol) -> float

* 功能：
  + 呼叫 Polygon API get\_snapshot\_ticker()，取得股票快照
  + 回傳：
    - 最小成交價 (min.close)
    - 或前一日收盤價 (prev\_day.close)

**3.6 根據帳號類型決定查價方式**

def get\_share\_price\_polygon(symbol) -> float

* 功能：
  + 若為付費帳號：
    - 呼叫 get\_share\_price\_polygon\_min()，取得即時價
  + 否則：
    - 呼叫 get\_share\_price\_polygon\_eod()，取得 EOD 價格

**3.7 對外查詢單一股票價格**

def get\_share\_price(symbol) -> float

* 功能：
  + 若設定了 Polygon API Key：
    - 嘗試呼叫 get\_share\_price\_polygon() 查詢
    - 若 API 失敗，顯示錯誤並 fallback
  + 若無 API Key：
    - 回傳隨機價格 (1 ~ 100)

**4. 資料庫依賴**

* write\_market(today, market\_data)
  + 將當天市場資料寫入資料庫
* read\_market(today)
  + 從資料庫讀取當天市場資料

**5. 錯誤處理**

* 若 Polygon API 呼叫失敗：
  + 顯示錯誤訊息
  + 回傳隨機股價作為替代方案

**6. 快取機制**

* get\_market\_for\_prior\_date() 使用 lru\_cache 快取
  + 減少重複呼叫 Polygon API
  + 預設快取最多 2 天的資料

**✅ 適用場景**

* 建立**股市監控機器人**
* 撰寫**投資策略回測**
* **實時更新股票價格系統**
* **模擬交易環境**