1. \*\*프로젝트 개요\*\* \* \*\*목표\*\*: 단일 종목(예: 삼성전자)의 일별 주가 흐름, 이벤트(급등/급락), 뉴스, 투자자 수급 정보를 바탕으로 LLM을 활용해 종합적인 자연어 리포트를 자동 생성하고 Slack에 전송 \* \*\*핵심 기술\*\*: \* 🔍 \*\*Retrieval-Augmented Generation (RAG)\*\*: 실시간 데이터를 검색해 LLM 입력으로 활용 \* 🧠 \*\*LLM 기반 리포트 생성\*\*: 시계열 요약, 뉴스 해석, 투자자 수급 판단 등 \* 📊 \*\*CRAG 구조 확장 적용\*\*: 주가 이벤트 발생 전 뉴스를 기반으로 인과 분석 ✅ 2. \*\*완료된 주요 작업\*\* 📦 \*\*데이터 수집\*\* 항목방법실시간 주가 데이터`naver\_finance\_crawler.py`로 구현 (분봉 기반)투자자 수급 데이터`investor\_crawler.py`를 통해 기관/외국인 순매수량 수집뉴스 데이터`news\_api\_caller.py` 및 `search\_news\_advanced()` 함수 사용 (Naver Open API 기반) 🧠 \*\*분석 및 이벤트 감지\*\* 항목설명이벤트 탐지`detect\_price\_events\_by\_day()`로 1% 이상 상승/하락 감지뉴스 매칭`match\_news\_before\_events()`로 이벤트 이전 뉴스만 필터링 연결이벤트별 분석용 Prompt`build\_event\_prompt()` 생성 🤖 \*\*LLM 분석 생성\*\* 항목설명LLM 리포트 생성`llm\_reporter.py` → GPT 기반 리포트 생성 함수 구현이벤트별 인과분석 생성각 이벤트마다 관련 뉴스 기반 분석 Prompt 생성 및 결과 생성 📨 \*\*Slack 연동\*\* \* Slack Webhook을 통해 결과 자동 전송 (`slack\_sender.py`) \* 이벤트별 분석 결과, 종합 리포트 전송 기능 완료 🧩 3. \*\*적용된 핵심 개념 (LLM/Tool/RAG) 요약\*\* 개념구현 내용RAG (검색 증강 생성)실시간 뉴스 + 주가 데이터를 LLM 입력으로 제공CRAG (Causal RAG)주가 이벤트 전 뉴스만 LLM에 제공하여 원인 추론Tool 사용Pandas 기반 시계열 분석, 뉴스 필터링, API 연결Prompt Engineering주가 이벤트, 뉴스 타이밍, 투자자 수급을 결합한 동적 프롬프트 설계 🔜 4. \*\*남은 작업\*\* ✅ [1] 뉴스 검색 보완 \* 현재 필터링된 뉴스가 거의 없거나 `"관련 뉴스 없음"`으로 출력 \* \*\*대응 필요\*\*: \* `news\_api\_caller.py` 내 `filter\_news\_by\_date()` 정밀 개선 \* `pubDate`의 파싱 실패 케이스 확인 \* `title` 중복 제거 및 다중 쿼리 조합 보완 필요 ✅ [2] 이벤트별 뉴스 인과 분석 자동화 \* 이미 `build\_event\_prompt()`와 `analyze\_events\_with\_llm()` 기반은 있음 \* \*\*보완 필요\*\*: \* 동일 시간대 중복 이벤트 병합 (예: 1분 간격 상승 다수 감지 시) \* 관련 뉴스가 너무 적은 경우 백업 설명 전략 (예: “시장 분위기” 등) ✅ [3] 정량적 평가/품질 검증 \* 향후 추가 실험으로 다음 작업을 고려 가능: \* \*\*LLM-as-a-Judge 평가\*\* 도입 (GPT-4로 리포트 비교 평가) \* \*\*다양한 종목에 적용\*\* → 대형주 vs 테마주 성능 비교 ✅ [4] 개선 아이디어 \* 뉴스 감성 분석 추가 (`sentiment` → 긍정/부정 판단) \* 이벤트 필터링 기준을 사용자가 조정 가능하게 변경 \* 종합 보고서 PDF/HTML 자동 저장 기능