



#

```
=====
```

## app.py : LangGraph 기반 실행 진입점

```
=====
```

```
from graph.flow.pipeline_graph import run_graph
```

```
=====
```

## 메인 실행 블록

```
=====
```

```
if name == "main":
```

```
# (1) 입력 CSV 경로
```

```
forms_csv = "data/raw/forms_2025-10-09.csv"
```

```
# (2) 분석할 BQ2 PDF 파일 경로
```

```
neuro_pdf = "data/raw/neuro/S001_오은빈_2025-10-03.pdf"
```

```
# (3) LangGraph 기반 파이프라인 실행
```

```
run_graph(forms_csv, neuro_pdf)
```

해석해줘

이 코드의 의미는 LangGraph 기반 데이터 처리 파이프라인을 실행하는 진입점 스크립트를 정의한 것입니다.

- `from graph.flow.pipeline_graph import run_graph`  
→ `graph/flow/pipeline_graph.py` 파일 안에 있는 `run_graph` 함수를 불러옵니다. 이 함수가 실제 데이터 처리 흐름(그래프 실행)을 담당합니다.

- `if __name__ == "__main__":`  
→ 이 스크립트가 직접 실행될 때만 아래 코드를 실행하도록 합니다.
- `forms_csv = "data/raw/forms_2025-10-09.csv"`  
→ 첫 번째 입력값: 분석할 CSV 파일 경로를 지정합니다.
- `neuro_pdf = "data/raw/neuro/S001_오은빈_2025-10-03.pdf"`  
→ 두 번째 입력값: 분석 대상 BQ2 유형의 PDF 파일 경로를 지정합니다.
- `run_graph(forms_csv, neuro_pdf)`  
→ 위 두 파일 경로를 입력으로 하여 LangGraph 기반 파이프라인을 실행합니다.  
실행 시 CSV 데이터를 불러오고 PDF를 분석하는 워크플로우가 수행됩니다.

즉, 이 스크립트를 실행하면 지정된 CSV와 PDF를 LangGraph 파이프라인으로 처리·분석하는 작업이 시작됩니다.

원하시면 제가 `run_graph` 내부 처리가 어떤 구조로 동작하는지까지 분석해 드릴까요?