

# 사용자 입력을 바탕으로 한 기업 공시 데이터 가공과 시각화

글로벌 아카데미

팀명 : 2팀

팀원 : 김주섭, 이서영, 성시건

멘토 : --

---

00.

## 프로젝트 목차

- |     |                 |   |
|-----|-----------------|---|
| 01. | 프로젝트 개요         | ↘ |
| 02. | 프로젝트 팀 구성 및 역할  | ↘ |
| 03. | 프로젝트 수행 절차 및 방법 | ↘ |
| 04. | 프로젝트 수행 경과      | ↘ |
| 05. | 자체 평가 의견        | ↘ |

# 01.

## 프로젝트 개요

### #1 프로젝트 주제 및 선정 배경과 기획 의도

#### 프로젝트 주제

다트(Dart) 공시 데이터를 바탕으로  
사용자에게 변수를 입력받고,  
그에 맞게 데이터를 가공하고 시각화합니다.

#### 선정 배경

사용자의 다양한 변수를 입력받고 그에 맞게 데이터 처리 및 가공과 시각화를 하고 싶었습니다.

프로젝트 시작 전에 개인적으로 다트(Dart) 데이터를 임의로 가공과 분석을 하며 데이터에 대한 이해도를 높였습니다.

해당 데이터를 어떤 유형의 사용자가 주로 사용할지 고민했고, 결과적으로 데이터 사용자를 투자자로 선정했습니다.

#### 기획 의도

과거의 지표만 갖고 기업 투자를 결정하기에는 위험하지만 참고 자료로 사용될 수는 있습니다.

해당 프로젝트는 머신러닝과 같은 기술을 사용하지 않기에 미래에 대한 예측은 불가능합니다.

하지만 사용자가 분석을 원하는 회사의 과거 데이터를 가공 및 시각화 함으로써 투자의 참고 자료로 기능합니다.

# 01.

## 프로젝트 개요

### #2 프로젝트 내용

1

다트(Dart) 재무정보 일괄다운로드 페이지에서 2015년부터 최근 년도까지 재무상태표, 손익계산서, 현금흐름표의 모든 사업보고서를 다운로드 받습니다.

2

다운로드 받은 파일을 임의로 지정한 폴더에 옮기고 압축을 해제한 후에 파이썬 웹 어플리케이션을 실행합니다.

3

웹 어플리케이션이 실행되면 압축을 해제했을 때 생긴 데이터를 하나의 파일로 취합합니다.

4

취합한 파일을 바탕으로 사용자가 분석을 원하는 주제를 변수로 입력받습니다. 변수에 알맞게 데이터를 가공 및 시각화하고, 결과를 브라우저에 보여줌과 동시에 분석 데이터를 CSV 파일로 저장하고 시각화 그래프는 이미지 파일로 저장합니다.

5

미리 준비된 엑셀 양식 파일에 저장한 CSV 파일과 시각화 그래프를 기입하여 보고서를 작성합니다.

# 01.

## 프로젝트 개요

### #3

#### 활용 도구 및 라이브러리



- Python
- Flask(웹 프레임워크)
- Pandas(데이터분석 라이브러리)
- Pycharm(IDE)
- Anaconda(Interpreter)
- BrityRPA
- Excel



# 01.

## 프로젝트 개요

### #4

#### 프로젝트 구조 및 역할



#### Python

1

웹 어플리케이션을 실행하여  
사용자로부터 변수를 입력받습니다.

2

입력받은 사용자의 변수에 따라 판다스 라이브러리를  
활용해 데이터 가공과 시각화 및 저장합니다.

3

웹 어플리케이션을 exe 실행 파일로 변환하여  
파이썬이 없는 컴퓨터에서도 실행할 수 있도록 합니다.

#### BrityRPA

파이썬 웹 어플리케이션이 기능하기 위해 폴더를 생성해  
기본 환경을 구성하고 웹 앱을 실행시킵니다.

사용자에게 변수를 입력받은 후 처리된 결과를 바탕으로  
엑셀 작업을 진행하여 보고서를 작성합니다.

01.

## 프로젝트 개요

### #5 활용 방안 및 기대 효과

#### 활용 방안

사용자가 절차에 따라 변수를 입력했다면  
어떠한 항목이던 간에 시각화가 가능합니다.  
그렇기 때문에 당기순이익이나 매출액과 같은  
자주 사용하는 항목 뿐만 아니라 판관비나  
특정 기업에 특화된 정보 등 자주 사용되지 않는  
항목도 시각화가 가능하여 사용자의 다양한  
요구에 대응할 수 있습니다.

#### 기대 효과

사용자는 요구에 따라 분석하고 시각화된  
다양한 자료들을 투자에 참고합니다.

## 02.

# 프로젝트 팀 구성 및 역할

훈련생	역할	담당 업무
김주섭	팀장	데이터 분석, BrityRPA 작성, 웹 앱 개발
이서영	팀원	데이터 분석, ppt 작성
성시건	팀원	데이터 분석
--	팀원	--
--	멘토	--



### 03.

## 프로젝트 수행 절차 및 방법

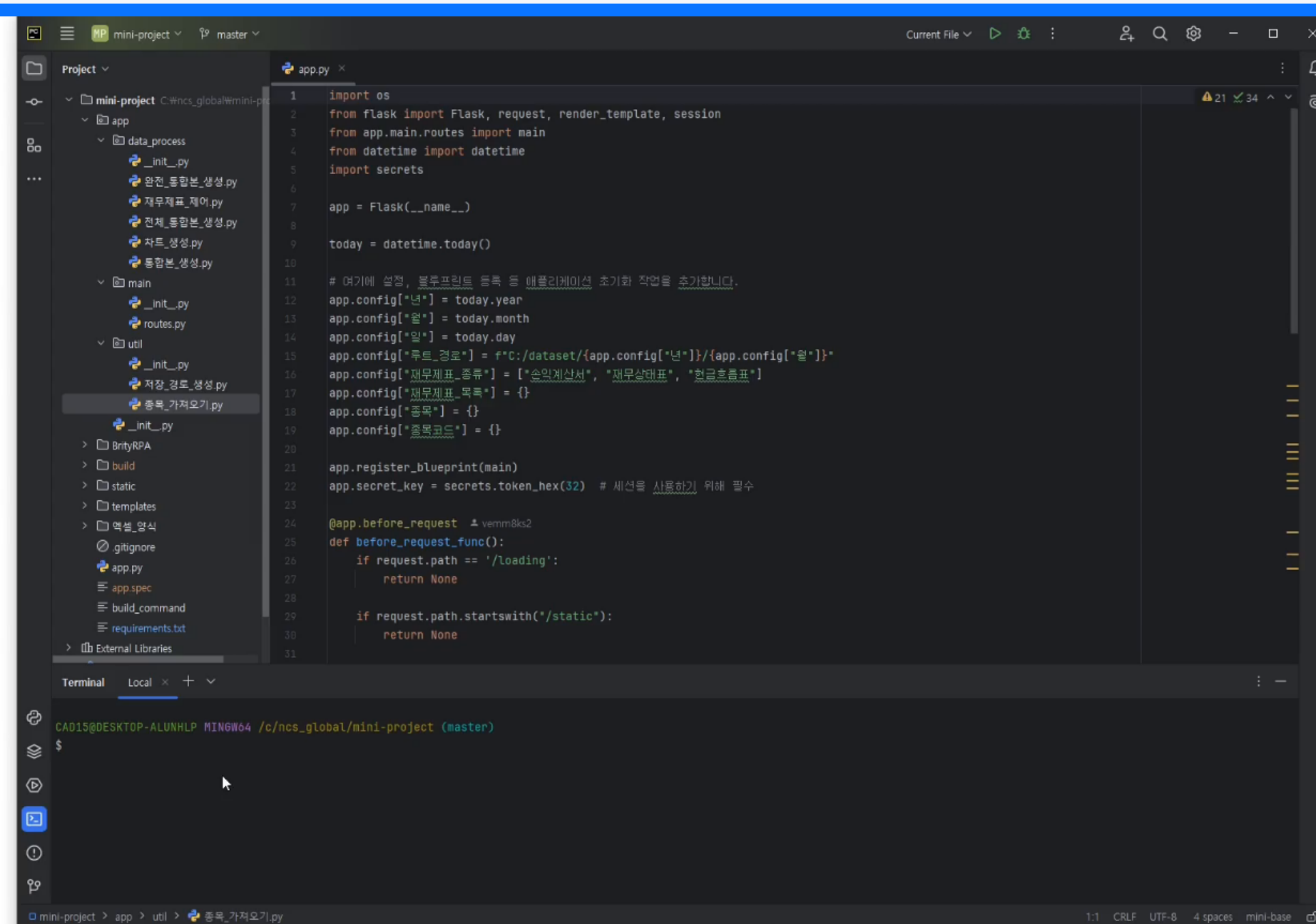


## 04.

# 프로젝트 수행 경과

## #1 파이썬 웹 어플리케이션 역할

영상 자료



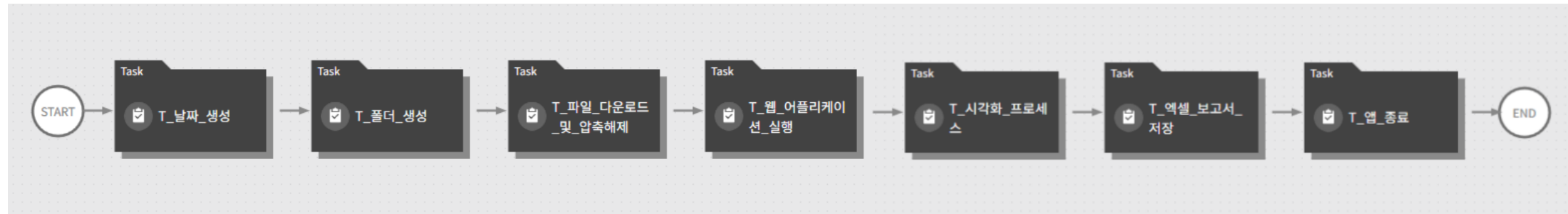
```
1 import os
2 from flask import Flask, request, render_template, session
3 from app.main.routes import main
4 from datetime import datetime
5 import secrets
6
7 app = Flask(__name__)
8
9 today = datetime.today()
10
11 # 여기에 설정, 블루프린트 등록 등 애플리케이션 초기화 작업을 추가합니다.
12 app.config["년"] = today.year
13 app.config["월"] = today.month
14 app.config["일"] = today.day
15 app.config["루트_경로"] = f"C:/dataset/{app.config["년"]}/{app.config["월"]}"
16 app.config["재무제표_종류"] = ["손익계산서", "재무상태표", "현금흐름표"]
17 app.config["재무제표_목록"] = {}
18 app.config["종목"] = {}
19 app.config["종목코드"] = {}
20
21 app.register_blueprint(main)
22 app.secret_key = secrets.token_hex(32) # 세션을 사용하기 위해 필수
23
24 @app.before_request
25 def before_request_func():
26     if request.path == '/loading':
27         return None
28
29     if request.path.startswith('/static'):
30         return None
31
```

영상 링크 : <https://youtu.be/dylEoqrYcE8>

## 04. 프로젝트 수행 경과

#2 BrityRPA 실행 순서  
#3 날짜 생성 및 폴더 생성

### BrityRPA 실행 순서



### 날짜 생성

프로젝트 전역적으로 사용할 날짜를 변수로 생성합니다.

### 폴더 생성

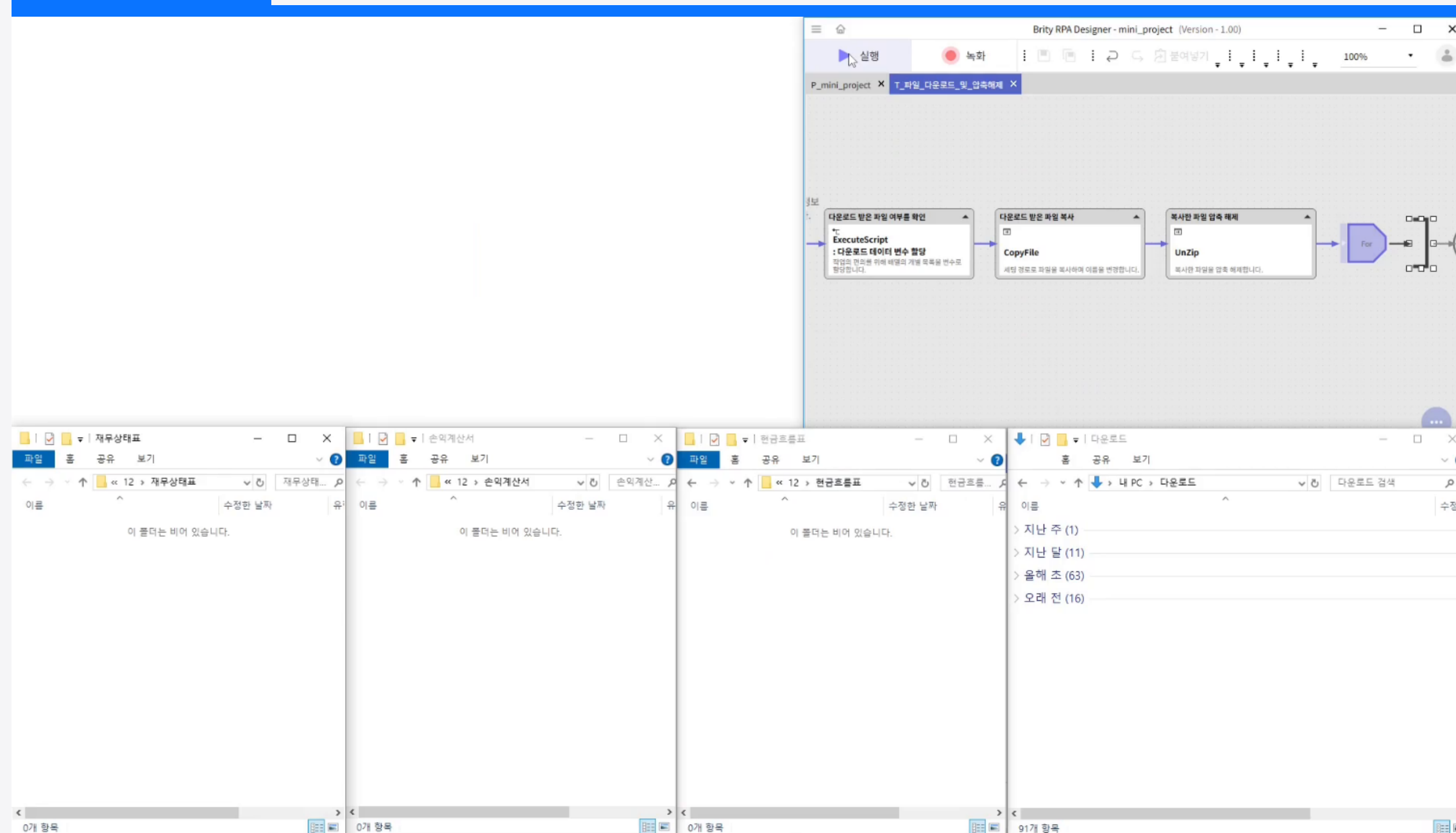
프로세스 진행에 필요한 폴더를 생성하여 기본 환경을 세팅합니다.

## 04.

# 프로젝트 수행 경과

## #4 파일 다운로드 및 압축 해제

영상 자료



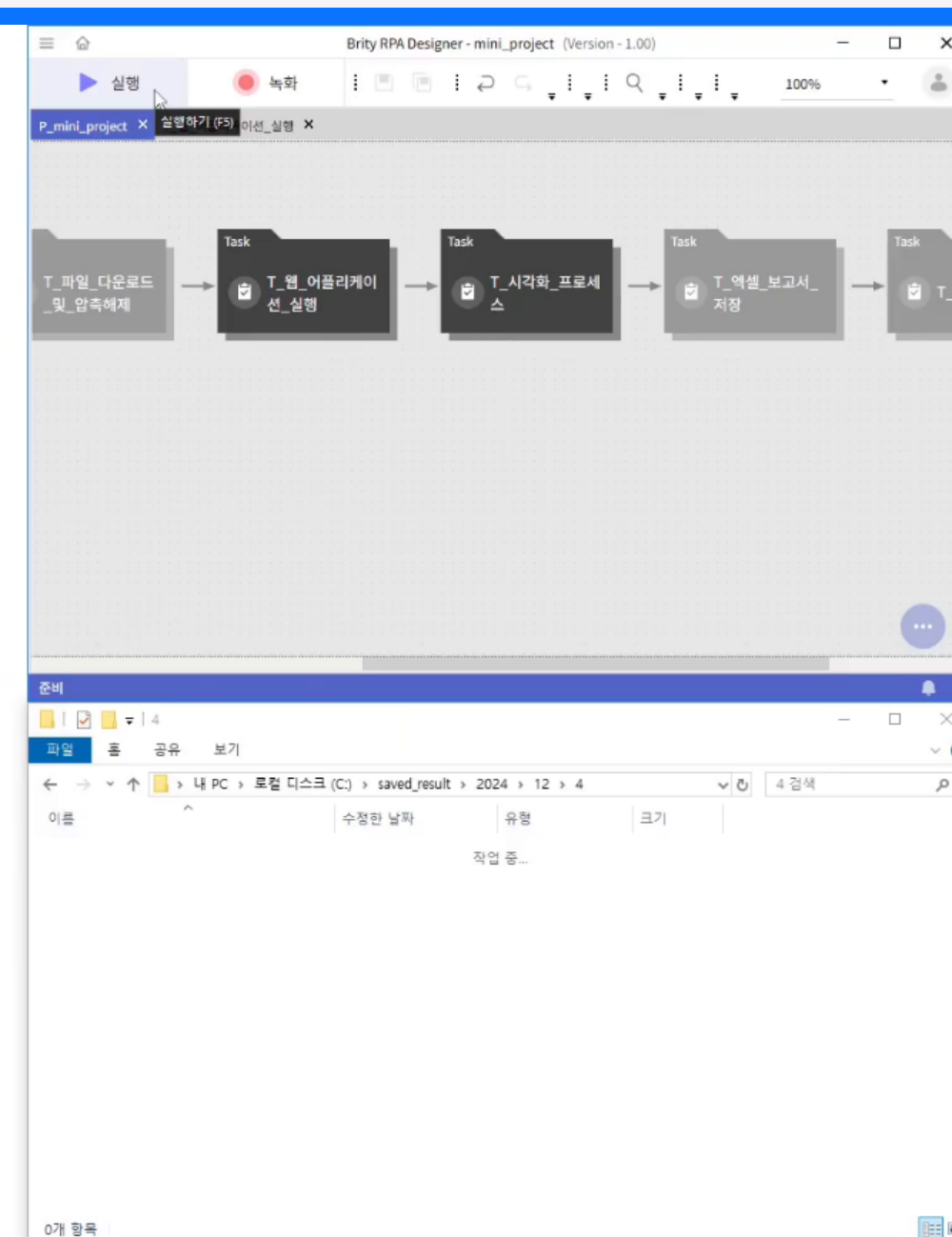
영상 링크 : <https://youtu.be/BfqLeNYOwDM>

04.

## 프로젝트 수행 경과

#5 웹 어플리케이션 실행, 시각화 프로세스

영상 자료

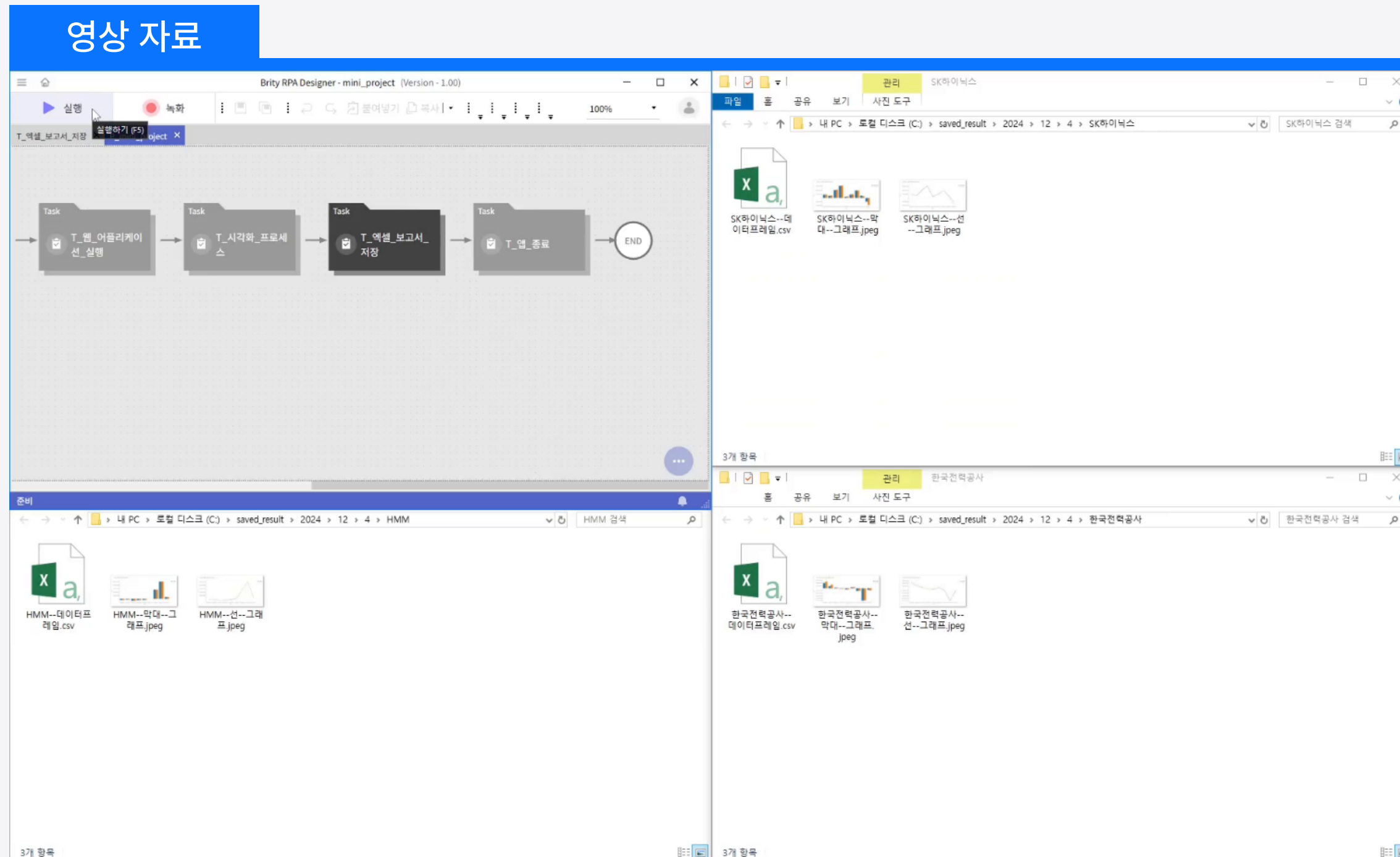


영상 링크 : [https://youtu.be/251ki\\_20304](https://youtu.be/251ki_20304)

04.

## 프로젝트 수행 경과

#6 엑셀 보고서 저장



영상 링크 : <https://youtu.be/hyJuCpCYWQY>

04.

## 프로젝트 수행 경과

#7 BrityRPA 전체 프로세스 영상

영상 자료



영상 링크 : <https://youtu.be/XPdeSVnReWA>

05.

## 자체 평가 의견

3팀

### #1 프로젝트 결과물에 대한 완성도 평가

프로젝트 기간

총 개발 기간

약 10일  
2024.11.20 ~ 12.05

기획 기간

2024.11.20 ~ 11.22

개발 기간

2024.11.25 ~ 12.05

80점

시간 부족으로 인해 더 많은 기능을 구현하지 못한 점과  
에러에 대한 핸들링이 없다는 점은 아쉬웠습니다.  
하지만 처음에 계획했던 대로 구현은 완료했기 때문에  
불만은 없습니다.



05.

## 자체 평가 의견

### 잘한 점

.....

개발을 진행하기 전에 데이터를 먼저  
분석하여 데이터 가공과 시각화에  
큰 어려움이 없었습니다.

시나리오를 과하지 않게 설정하여  
목표를 달성할 수 있었습니다.



#2

### 잘한 점과 아쉬운 점

### 아쉬운 점

.....

웹 어플리케이션 개발에 집중하다보니  
자동화 관점의 생각이 부족했습니다.

멤버 간 수행할 역할을 골고루  
분배하지 못했습니다.



05.

## 자체 평가 의견

01

웹 앱을 실행한 후 처음으로 데이터를 통합하는 과정에서 상당한 시간이 소요됩니다. 이를 개선할 방안을 찾고 시간을 단축하게 된다면 사용자 경험이 크게 향상될 것입니다.

02

사용자에게 받은 변수가 적절하지 못하여 빈 데이터프레임을 반환할 경우 이에 대한 에러 핸들링이 없습니다. 마찬가지로 이를 개선한다면 사용자 경험이 향상될 것입니다.

03

BrityRPA가 구동중일 때 사용자에게 직접 입력을 받는데 이를 특정 양식의 엑셀 파일로 대체하고 이를 읽어와 사용한다면 더 높은 수준의 자동화가 가능할 것입니다.

04

파이썬 웹 앱 동작의 일부가 BrityRPA 동작에 종속되어 있기에 이를 분리하면 가용성이 더 늘어날 것입니다.

#3 개선 방안

05.

## 자체 평가 의견

### #4 느낀점 및 경험 성과

01

파이썬 Flask로 웹 개발을 해본 적은 없지만 기존의 지식을 바탕으로 큰 어려움 없이 진행되었고 파이썬 개발에 대해 익숙해지는 시간을 가졌습니다.

02

웹 지식을 바탕으로 HTML 요소와 속성을 제어하기 위해 BrityRPA의 다양한 카드를 사용하였고, 결과적으로 빠른 속도로 크롬 브라우저를 제어하게 되었으며 BrityRPA 사용 방법에 대한 이해도가 올라갔습니다.

03

데이터를 효과적으로 가공하고 시각화하기 위해서는 데이터셋에 대한 이해가 요구되고, 결측치나 이상치를 분석할 필요가 있으며 이에 대한 전처리가 매우 중요하다는 점을 깨닫게 되었습니다.

---

## 06. 마무리

프로젝트의 파이썬 코드 및 BrityRPA 시나리오 링크

<https://github.com/vemm8ks2/mini-project>

실행 영상 재생목록 링크

<https://www.youtube.com/playlist?list=PLjQ0nq-EU7dxPiTmMS4nT3XpZl9P76SPt>

링크를 클릭하면 해당 페이지로 이동이 가능합니다.  
감사합니다.