SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

04 PJT

챕터의 포인트

- Django 에서 Data Science 활용하기
- DB를 활용한 웹 페이지 구현

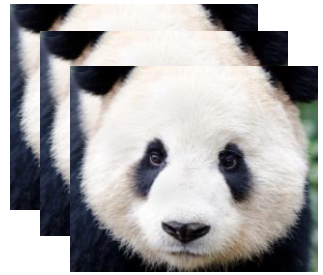
SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

목표

목표

- Django 에서 데이터 사이언스 패키지 사용하기



이전 데이터 사이언스 프로젝트와의 차이점

- 이전 프로젝트
  - Matplotlib, Pandas, Numpy 를 주피터 노트북에서 구동함

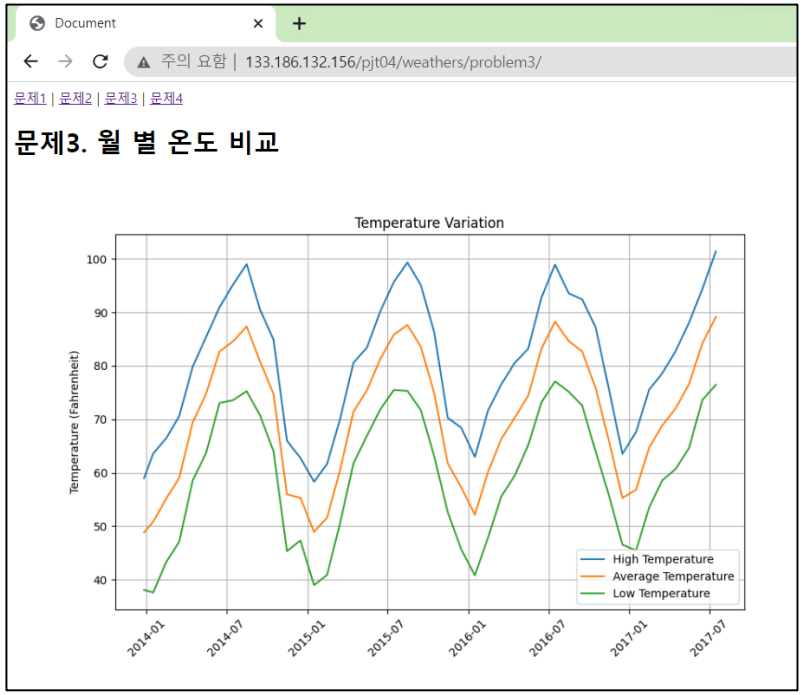
- 이번 프로젝트 실습
  - Matplotlib, Pandas, Numpy 를 Django 에서 구동함

금융상품비교

영화추천서비스

Django 에서 데이터 사이언스 패키지를 사용하는 이유

- 결과를 웹 페이지에서 보여주기 위함



이번 프로젝트 결과 화면 예시

금융상품비교

영화추천서비스

Django 에서 데이터 사이언스 패키지를 사용하기 위해 알아야 할 내용

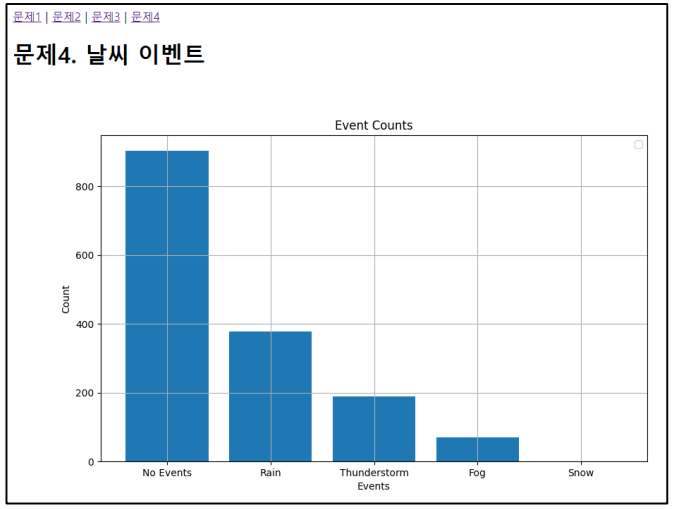
- 데이터 사이언스 3종 패키지 사용 방법
- Django 기본 사용 방법
  - 웹 페이지 구성(template)
  - 데이터 전달(View -> Template)
- 파이썬 BytesIO 패키지(곧, 학습 예정)

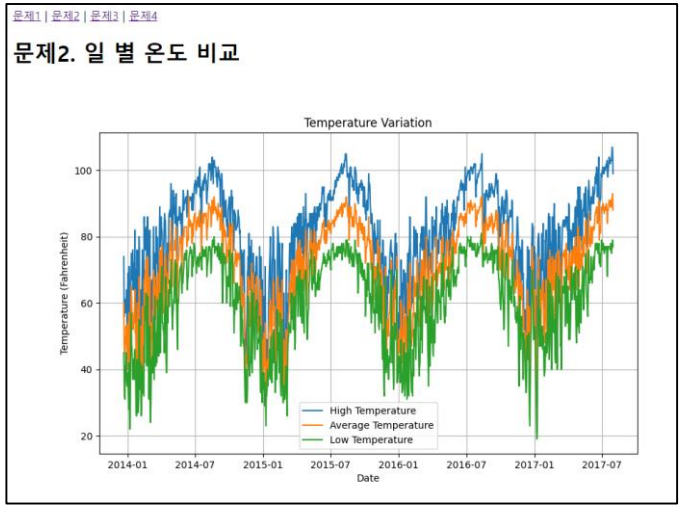
금융상품비교

영화추천서비스

프로젝트 목표 세 가지

1. Numpy, Pandas, Matplotlib 복습
2. Django Web Framework 복습
3. Django로 데이터 사이언스 패키지 활용 방법 학습





SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

준비사항

개발도구

- Visual Studio Code
- Google Chrome
- Python 3.9 +

필수 라이브러리

- 가상환경을 설정하여 아래 라이브러리 설치 후 요구사항을 구현합니다.
  - Django 4.2 +
  - Pandas
  - Numpy
  - Matplotlib

테스트 데이터

- Gitlab 을 통해 test_data.csv 를 다운로드 받습니다.

SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

기본 그래프 출력하기

기본 그래프 출력

- 아래 더미데이터를 하드코딩 합니다.

```
x = [1, 2, 3, 4, 5]
y = [1, 2, 3, 4, 5]
```

- Matplotlib 을 활용하여 그래프를 생성하고, 화면으로 출력합니다.

View 에서 Template 으로 이미지 전달하기

- View 에서 Template 으로 이미지 형식의 데이터를 직접 전달할 수 없습니다.
- 저장된 이미지의 경로를 전달하여 Template 에서 출력해야 합니다.
- matplotlib 의 그래프를 버퍼에 이미지 형식으로 저장 후 저장된 경로를 전달합니다.
  - 버퍼(buffer): 임시로 데이터를 저장하는 공간
- Python “BytesIO” 클래스
  - 파이썬의 내장 모듈인 “io” 모듈에 포함된 클래스
  - 메모리 내에 데이터를 저장 및 조작할 수 있는 기능 제공

View 에서 Template 으로 이미지 전달하기

- views.py

```python
import base64
from io import BytesIO

def test(request):
    x = [1, 2, 3, 4, 5]
    y = [1, 2, 3, 4, 5]

    # 그래프 초기화
    plt.clf()

    # plot 생성
    plt.plot(x, y)
    plt.title('Test Graph')
    plt.xlabel('x label')
    plt.ylabel('y label')
    plt.grid(True)

    # 그래프 이미지를 임시로 저장할 버퍼 생성
    buffer = BytesIO()
    # 그래프를 버퍼에 저장. 이미지 형식은 png 로 설정
    plt.savefig(buffer, format='png')
    # 버퍼의 내용을 base64 로 인코딩
    image_base64 = base64.b64encode(buffer.getvalue()).decode('utf-8').replace('\n', '')
    buffer.close()
    context = {
        # 저장된 이미지의 경로를 전달
        'chart_image': f'{image_base64}',
    }

    return render(request, 'test.html', context)
```

- test.html

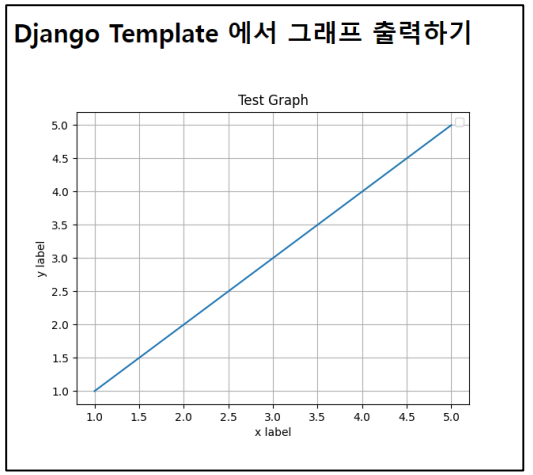
```
{% extends 'base.html' %}

{% block content %}

<h1>Django Template 에서 그래프 출력하기</h1>
<img src="{{ chart_image }}" alt="">

{% endblock content %}
```

- 출력 결과 예시



SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

CSV 파일 활용하기

CSV 파일 활용하기

- 다운로드 받은 test_data.csv 를 pandas 의 dataframe 으로 구성합니다.
- 날짜 별 데이터를 DB에 저장합니다.
- 고려사항
  - 사용자의 어떤 요청이 왔을 때 데이터를 저장할까 ?
  - 저장 요청이 두 번 온다면 중복된 데이터 처리는 ?
  - DB 테이블의 필드 구성은 어떻게 할까 ?

SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

데이터 출력하기

데이터 출력하기

- DB 에 저장한 날짜 별 데이터를 출력합니다.
- 세 가지 데이터를 하나의 페이지에 적절히 출력합니다.

SAMSUNG SW ACADEMY FOR YOUTH

함께가요 미래로!
Enabling People

도전 과제

관통 Ver1 - PJT04 도전 과제

- 프로젝트명: Django에서 Data Science 활용하기
- 목표
  - 데이터 사이언스 패키지를 Django 에서 활용하는 방법 익히기
- 특징
  - 웹 페이지에서 결과를 확인할 수 있다
  - 캐글에서 데이터 다운로드

관통 Ver2 - PJT04 도전 과제

- 프로젝트명: DB를 활용한 웹 페이지 구현
- 목표
  - 영화 데이터 생성, 조회, 수정, 삭제가 가능한 애플리케이션 완성
- 특징
  - 영화 커뮤니티 웹 서비스의 화면 및 데이터 구성 단계
  - 영화 데이터 CRUD가 가능한 애플리케이션 완성