

PJT명	파일 시스템을 활용하여 수집된 파일 데이터를 가공하고 생성형 AI를 이용하여 요약하기	
단계	[Python PJT]	
진행일자	2025.07.25	
예상 구현 시간	필수기능	5H
	추가기능	2H
	심화기능	1H

1. 목표

- Python 기초 문법에 대하여 이해하고 활용할 수 있다.
- 파일 입출력에 대하여 이해한다.
- JSON 파일로 주어진 데이터를 프로그램에 활용할 수 있다.
- 데이터 구조를 분석하고 재구성하여 프로그램에 활용할 수 있다.
- 생성형 AI를 활용하여 도서 데이터를 요약하고 간단한 인사이트를 도출해본다.

2. 준비사항

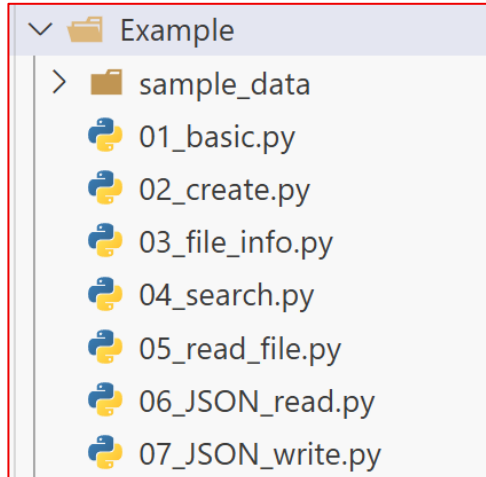
1) 프로젝트 구조

- 프로젝트는 총 두개의 폴더와 13개의 Python 코드로 구성되어 있음
 1. problem_A1.py ~ problem_C3.py: 요구사항 구현을 위한 스켈레톤 코드
 - problem_XN.py의 형태로 이름이 지어짐
 - X: 과제 그룹
 - N: 요구사항 번호
 2. Example 폴더: 아래 2) 사용 데이터에 설명된 Example 폴더
 3. data 폴더: 아래 2) 사용 데이터에 설명된 data 폴더

2) 사용 데이터

- Example 폴더: 이번 프로젝트 해결을 위해 알아야 하는, 혹은 직접적인 도움이 될 수 있는 코드

예시:



```
# 경로 다루기
```

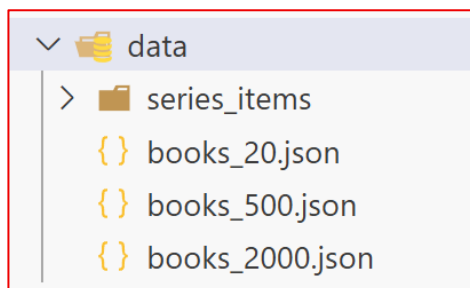
```
# 현재 작업 디렉토리
```

```
current_path = Path.cwd()
```

```
print(f'현재 작업 경로 : {current_path}')
```

- data 폴더: 실제 문제 풀이에 사용되는 데이터

예시:



```
{
  "version": "20131101",
  "logo": "http://image.aladin.co.kr/img/header",
  "title": "알라딘 검색결과 - 강아지",
  "link": "http://www.aladin.co.kr/search/wsear",
  "pubDate": "Wed, 18 Dec 2024 08:43:00 GMT",
  "totalResults": 1088,
  "startIndex": 1,
```

3) 개발언어 및 툴

- Python 3.11+ / Visual Studio Code

4) 필수 라이브러리 / 오픈소스

- Python 내장 json 모듈
- Python 내장 pathlib 모듈

3. 작업 순서

- 1) 팀원과 같이 요구사항(기본/추가/심화)을 확인하고, GitLab에 프로젝트를 생성한다.
 - 프로젝트 이름은 01-pjt로 지정한다.
 - 각 반 담당 강사님을 Maintainer로 설정한다.
- 2) 제공된 examples의 코드를 확인하고, 요구사항에 필요한 코드를 파악한다.
- 3) 팀원과 합의하여 과제 그룹을 선정한다.
 - 과제 그룹 별로 요구사항이 다르며, 난이도가 상이하다.
 - 팀원과 합의하여 적정 수준의 그룹을 선정하도록 한다.
- 4) 각 문제 코드를 확인하고, data에 주어진 파일을 활용하여 필수 요구사항을 구현한다.
- 5) 작성한 코드들을 정리하고, README를 작성한다.
- 6) README 작성이 완료되면 심화 학습을 진행한다.
- 7) 제출 기한에 맞춰 모든 산출물이 GitLab에 업로드 될 수 있도록 한다.

4. 요구사항

AI 기반 도서 분석과 창작 지원 통합 솔루션 서비스를 구축하려고 한다. 다양한 장르와 형태의 도서 데이터를 수집 및 관리하고, 이를 기반으로 AI 기반 도서 분석 기능을 설계 및 구현한다. 또한 작가들에게 아이디어 발상 지원, 플롯 구성 제안, 문장 교정 및 스타일 개선 등 창작 전반에 걸친 다양한 AI 기반 창작 지원 기능을 제공한다. 또한 도서에 대한 사용자 리뷰 및 감상평 공유 커뮤니티 기능을 제공하여, 사용자들이 활발하게 소통하고 정보를 교환할 수 있는 기능을 제공한다. 사용자는 자신이 읽은 도서를 평가하고, 다른 사용자의 리뷰를 참고하여 다음 도서를 선택하는데 도움을 받을 수 있다. 나아가, 관심 도서 목록을 맞춤형으로 구성하는 등 다양한 편의 기능을 제공한다. 팀원과 상의하여 아래 요구사항을 만족할 수 있도록 요구 사항 명세서를 작성 및 구현해보자.

본격적인 서비스 구현에 앞서서 제공되는 데이터 파일을 기반으로, 필요한 정보를 파싱한 후 프로그램 내부에서 정리할 수 있는 프로그램을 구현해보자.

과제 그룹 A, B, C 중 하나를 선택해서 진행한다.

● 요구사항 예시(참고용)

- 아래의 내용을 참고하여 추가적인 아이디어에 대해 요구사항을 추가 또는 수정하여 기능을 구현한다. 단, **필수 기능은 구현해야 하며, 수정할 수 없다.**

번호	분류	요구사항명	요구사항 상세	우선순위
기능적 요구사항				
F01	도서 데이터	데이터 정리	주어진 파일에서 필요한 정보만 추출하여 정리하는 기능	필수
F02	도서 데이터	데이터 집계	주어진 파일의 데이터를 바탕으로, 전체의 가격을 집계하는 기능	필수
F03	도서 데이터	데이터 집계	주어진 파일의 데이터를 바탕으로, 전체의 고객 리뷰를 집계하는 기능	필수
F04	도서 데이터	데이터 변환	주어진 파일의 데이터를 바탕으로, 필요한 정보를 다른 형태로 변환하는 기능	필수
F05	도서 데이터	데이터 분류	주어진 파일의 데이터를 바탕으로, 가격 범위에 따라 데이터를 분류하는 기능	필수
F06	도서 데이터	데이터 찾기	주어진 파일의 데이터를 바탕으로, 특정 분류가 아닌 도서만 추출하는 기능	필수
F07	도서 데이터	데이터 집계	주어진 파일의 데이터를 바탕으로, 출판사 별 도서 개수를 집계하는 기능	필수
F08	도서 데이터	데이터 집계	주어진 파일의 데이터를 바탕으로, 카테고리 별 도서 수와 평균 가격을 집계하는 기능	필수
F09	도서 데이터	데이터 정리	주어진 파일의 데이터를 바탕으로, 도서를 출판일 기준으로 정리하는 기능	필수
F10	도서 데이터	데이터 찾기	주어진 파일의 데이터를 바탕으로, 도서의 할인율을 계산하고, 할인율을 기준으로 정리하는 기능	필수
F11	도서 데이터	데이터 집계	생성형 AI를 활용하여, 파일로 주어진 데이터의 월별 책 정보를 모아보고 평균	필수

			가격을 계산하는 기능	
F12	도서 데이터	파일 만들기	생성형 AI를 활용하여, 파일로 주어진 데이터의 카테고리를 기준으로 새로운 파일을 만드는 기능	필수
F13	도서 데이터	데이터 집계	생성형 AI를 활용하여, 파일로 주어진 데이터 중 시리즈 정보가 있는 책들끼리 묶어주는 기능	필수
F14	도서 데이터	데이터 요약 정리 프롬프트	데이터에서 인사이트를 뽑아낼 수 있는 지표와 분석 내용을 구성하고, 생성형 AI에 요약을 요청하는 기능	심화
F15	도서 데이터	AI 기반 인사이트	데이터의 지표의 상관관계를 AI를 이용하여 인사이트를 요청하는 기능	심화
F16	도서 데이터	데이터 생성	생성형 AI를 활용하여 가상의 책 정보를 만들고 저장하는 기능	심화
...

1) 기본(필수) 기능 (그룹 A)

① 책 제목을 리스트로 정리하기

pathlib 라이브러리의 Path 클래스를 이용하여, books_20.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 제목(title)의 값을 리스트에 추가한다. 생성된 리스트를 출력한다.

- 요구사항 번호: F01
- 주어진 problem_A1.py을 수정하여 구현
- 구현해야 할 화면 (콘솔 출력)

책 제목 리스트:

['강아지 탐정 와츠 - 꼭꼭 묻힌 단서, 공공 숨긴 보물', '냥만 강아지 봉봉 7 - 나는 스타 강아지', '1000개 숨은그림찾기 : 강아지 - 찾아도 찾아도 끝판왕', '냥만 강아지 봉봉 6 - 이층집의 비밀', '냥만 강아지 봉봉 4 - 부러진 이빨 사건', '신발 신은 강아지 - 2015 미국 어린이도서관협회 선정도서', '냥만 강아지 봉봉 5 - 위험천만 숨바꼭질', '나의 똑똑한 강아지', '냥만 강아지 봉봉 1 - 똥개 아니고 번개', '귀여운 강아지 에코 수세미 100 - 코바늘 한 자루로 완성, 깨끗한 우리 집을 부탁해!', '책 읽는 강아지 몽몽 - 제3회 비룡소 문학상 대상 수상작', '내 강아지 마음 상담소 - 강아지 마음에 대한 소소한 질문들', '내 강아지에게', '소녀와 강아지', '냥만 강아지 봉봉 2 - 진짜 주인 찾기', '냥만 강아지 봉봉 3 - 거리의 비밀 요원', '강아지 복실이', '멍멍, 강아지와 동물 친구들', '냥만 강아지 봉봉 1~7권 세트/아동도서+문구세트 증정', '손끝에서 시작하는 특별한 강아지옷 만들기 - 소잉도그의 노하우를 담은 강아지 옷 만들기 A부터 Z까지 완벽 정리']

- books_20.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

파일이 존재하지 않습니다: ../data/books_20.json

② 도서 평균 가격 계산

pathlib 라이브러리의 Path 클래스를 이용하여, books_20.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 가격 데이터(priceSales)의 값을 리스트에 추가한다. 저장된 가격 데이터의 평균을 계산한다.

- 요구사항 번호: F02
- 주어진 problem_A2.py을 수정하여 구현
- 구현해야 할 화면 (콘솔 출력)

도서 평균 가격: 15792.00원

- books_20.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

③ 도서의 평균 고객 리뷰 순위 리스트

pathlib 라이브러리의 Path 클래스를 이용하여, books_20.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 고객 리뷰 데이터(customerReviewRank)의 값을 리스트에 추가한다. 저장된 고객 리뷰 순위 데이터의 평균을 계산한다.

- 요구사항 번호: F03
- 주어진 problem_A3.py을 수정하여 구현
- 구현해야 할 화면 (콘솔 출력)

도서의 평균 고객 리뷰 순위: 7.95

- books_20.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

④ 도서별 작가 리스트

pathlib 라이브러리의 Path 클래스를 이용하여, books_20.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 제목(title)을 key로, 작가 이름(author)을 value로 하는 dictionary를 출력한다.

- 요구사항 번호: F04
- 주어진 problem_A4.py을 수정하여 구현
- 구현해야 할 화면 (콘솔 출력, 결과 일부 생략)

도서 별 작가 리스트:

제목: 강아지 탐정 와츠 - 꼭꼭 묻힌 단서, 뽀뽀 숨긴 보물
작가: 메그 맥래런 (지은이), 심연희 (옮긴이)

제목: 낭만 강아지 봉봉 7 - 나는 스타 강아지
작가: 홍민정 (지은이), 김무연 (그림)

제목: 1000개 숨은그림찾기 : 강아지 - 찾아도 찾아도 끝판왕
작가: 양혜민, 홍혜련 (그림)

- books_20.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

⑤ 정가 범위 분류하기

pathlib 라이브러리의 Path 클래스를 이용하여, books_20.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 정가(priceStandard)를 확인한 다음, 특정 가격 기준으로 도서를 분류하고 제목을 저장한다.

- 요구사항 번호: F04
- 주어진 problem_A5.py을 수정하여 구현
- 구현해야 할 화면 (콘솔 출력, 결과 일부 생략)

정가 범위 분류:

1만원 이상 2만원 미만:

- 강아지 탐정 와츠 - 꼭꼭 묻힌 단서, 공공 숨긴 보물
- 낭만 강아지 봉봉 7 - 나는 스타 강아지
- 낭만 강아지 봉봉 6 - 이층집의 비밀
- 낭만 강아지 봉봉 4 - 부러진 이빨 사건
- 신발 신은 강아지 - 2015 미국 어린이도서관협회 선정도서
- 낭만 강아지 봉봉 5 - 위험천만 숨바꼭질
- 나의 똑똑한 강아지

1만원 미만:

- 1000개 숨은그림찾기 : 강아지 - 찾아도 찾아도 끝판왕
- 귀여운 강아지 에코 수세미 100 - 코바늘 한 자루로 완성, 깨끗한 우리 집을 부탁해!

2만원 이상:

- 멍멍, 강아지와 동물 친구들
- 낭만 강아지 봉봉 1~7권 세트/아동도서+문구세트 증정

- books_20.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

2) 기본(필수) 기능 (그룹 B)

① 어린이 도서가 아닌 책 찾기

pathlib 라이브러리의 Path 클래스를 이용하여, books_500.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 카테고리(categoryName)에 '어린이'가 포함되지 않은 도서의 제목(title)과 카테고리를 새로운 리스트로 저장하고 출력한다.

- 요구사항 번호: F06
- 주어진 problem_B1.py을 수정하여 구현
- 구현해야 할 화면 (콘솔 출력, 결과 일부 생략)

어린이 도서가 아닌 책 목록:
1000개 숨은그림찾기 : 강아지 - 찾아도 찾아도 끝판왕 - 국내도서>유아>놀이책>숨은그림찾기
나의 똑똑한 강아지 - 국내도서>소설/시/희곡>한국소설>2000년대 이후 한국소설
귀여운 강아지 에코 수세미 100 - 코바늘 한 자루로 완성, 깨끗한 우리 집을 부탁해! - 국내도서>건강/취미>공예>뜨개질/퀼트/십자수/바느질
내 강아지 마음 상담소 - 강아지 마음에 대한 소소한 질문들 - 국내도서>건강/취미>반려동물

- books_500.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

파일이 존재하지 않습니다: ..\data\books_500.json

② 출판사별 도서 수 집계하기

pathlib 라이브러리의 Path 클래스를 이용하여, books_500.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 출판사명(publisher)를 key로, 해당 출판사의 도서 개수를 value로 하는 dictionary를 출력한다.

- 요구사항 번호: F07
- 주어진 problem_B2.py을 수정하여 구현
- 구현해야 할 화면 (콘솔 출력, 결과 일부 생략)

```
출판사별 도서 수:  
한빛에듀: 3권  
다산어린이: 16권  
위즈덤하우스: 10권  
서삼독: 1권  
참돌: 1권  
비룡소: 10권  
헤다: 1권  
문학동네: 8권  
반출판사: 1권
```

- books_500.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

③ 카테고리별로 도서 수, 평균 가격 구하기

pathlib 라이브러리의 Path 클래스를 이용하여, books_500.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 카테고리(categoryName) 별로 도서 수와 가격(priceStandard)을 집계한 뒤, 카테고리별 평균 가격을 출력한다.

- 요구사항 번호: F08
- 주어진 problem_B3.py를 수정하여 구현
- 구현해야 할 화면 (콘솔 출력, 결과 일부 생략)

```
카테고리별 도서 통계:
국내도서>어린이>동화/명작/고전>외국창작동화: 20권, 평균 가격 10930.50원
국내도서>어린이>동화/명작/고전>국내창작동화: 63권, 평균 가격 14375.08원
국내도서>유아>놀이책>숨은그림찾기: 2권, 평균 가격 8820.00원
국내도서>어린이>초등1~2학년>그림책: 2권, 평균 가격 13500.00원
국내도서>소설/시/희곡>한국소설>2000년대 이후 한국소설: 14권, 평균 가격 10125.00원
국내도서>소설/시/희곡>한국소설>2000년대 이전 한국소설: 14권, 평균 가격 15245.00원
```

- books_500.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

④ 출판일 기준 최신 도서 10개 목록 리스트로 정리

pathlib 라이브러리의 Path 클래스를 이용하여, books_500.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 출판일(publishDate)를 기준으로 최신 도서 10개를 추출한다.

- 요구사항 번호: F09
- 주어진 problem_B4.py를 수정하여 구현
- sorted 함수의 key 매개변수를 사용하여 상위 10개 도서를 선택한다.
- 구현해야 할 화면 (콘솔 출력, 결과 일부 생략)

```
최신 도서 10개:  
전쟁과 개 고양이 대학살 - 인간의 전쟁에서 지워진 동물 학살의 역사, 재구성하다  
- 2025-01-31  
우리가 지구를 걷는 법 1 - 2024-12-17  
고양이 레이의 선물 - 2024-12-16  
지구화 - 2024-12-13  
내 강아지에게 - 2024-12-12  
검은 고양이와 마녀의 교실 8 - 2024-12-12  
쿡판다의 수상한 만두카 2 - 쿡판다 학교에 가다! - 2024-12-12  
파이썬으로 탐구하는 지구과학 데이터 분석 & 시각화 - 그래프로 그려보는 지구별  
자연법칙 - 2024-12-11  
고양이별 펠리 - 2024-12-10  
2026 수능대비 Xistory 자이스토리 지구과학 1 (2025년) - 2024-12-06
```

- books_500.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

⑤ 할인율이 가장 큰 도서 5개 찾기

pathlib 라이브러리의 Path 클래스를 이용하여, books_500.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 내부의 item 리스트를 순회하며 판매가(priceSales)와 정가(priceStandard)를 가지고 할인율을 계산, 이후 할인율이 높은 상위 5개 도서를 추출한다.

- 요구사항 번호: F10
- 주어진 problem_B6.py을 수정하여 구현
- $\text{할인율} = ((\text{정가} - \text{판매가}) / \text{정가}) * 100$
- sorted 함수의 key 매개변수를 사용하여 상위 5개 도서를 선택한다.
- 구현해야 할 화면 (콘솔 출력, 결과 일부 생략)

할인율 상위 5개 도서:

조각 조각 스티커 아트북 3종 세트 4 : 탈것, 우리집, 강아지 - 전3권 - 할인율: 30.00%

2025 판다 달력 - 할인율: 30.00%

멍멍, 강아지와 동물 친구들 - 할인율: 20.00%

조각 조각 스티커 아트북 : 강아지 - 할인율: 20.00%

자연 관찰 스티커북 : 강아지 - 할인율: 20.00%

- books_500.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

3) 기본(필수) 기능 (그룹 C)

① 생성형 AI를 활용한 월별 책 정보의 평균 가격 계산하기

pathlib 라이브러리의 Path 클래스를 이용하여, books_2000.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 생성형 AI를 활용하여 출판일 기준, 도서를 월별로 분류하는 코드를 생성한 다음, 정상 동작하는 코드인지 검증한다.

생성된 코드가 정상적으로 동작하면, 월별로 도서 가격을 합산하고, 도서 수를 구하여 각 월의 평균 가격을 계산한다.

- 요구사항 번호: F11
- 주어진 problem_C1.py을 수정하여 구현
- 구현해야 할 화면 (콘솔 출력)

월별 평균 가격 및 도서 수:
1월: 평균 가격 16202.50원 (총 132권)
2월: 평균 가격 16030.43원 (총 93권)
3월: 평균 가격 14643.14원 (총 156권)
4월: 평균 가격 16551.34원 (총 127권)
5월: 평균 가격 15271.88원 (총 117권)
6월: 평균 가격 20990.61원 (총 131권)
7월: 평균 가격 13564.70원 (총 134권)
8월: 평균 가격 16628.98원 (총 137권)
9월: 평균 가격 14838.22원 (총 180권)
10월: 평균 가격 18057.42원 (총 271권)
11월: 평균 가격 19008.96원 (총 318권)
12월: 평균 가격 16654.90원 (총 204권)

- books_2000.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

파일이 존재하지 않습니다: ..\data\books_2000.json

② 생성형 AI를 활용하여 카테고리별 JSON 파일로 만들기

pathlib 라이브러리의 Path 클래스를 이용하여, books_500.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 생성형 AI를 활용하여 category_id와 category_name을 기준으로 도서 정보를 분류하는 코드를 생성한 다음, 정상 동작하는 코드인지 검증한다.

생성된 코드가 정상적으로 동작하면, 각 카테고리별로 도서 정보를 담은 새로운 JSON 파일을 생성한다.

- 요구사항 번호: F12
- 주어진 problem_C2.py을 수정하여 구현
- 도서 제목, 저자, 출판사, 출판일, ISBN, 가격 정보를 포함하여 정리
- 구현해야 할 화면 (콘솔 출력)

JSON 파일이 생성되었습니다: category_books.json

- 완성된 파일 일부

```
{
  "48874": {
    "name": "국내도서>어린이>동화/명작/고전>국내창작동화",
    "books": [
      {
        "title": "호랑이 빵집 4 : 사랑은 어려워!",
        "author": "서지원 (지은이), 홍그림 (그림)",
        "publisher": "아르볼",
        "pubDate": "2024-10-30",
        "isbn": "9791162041772",
        "price": 12600
      },
      {
        "title": "호랑이 뱃속에서 고래 잡기 - 김용택 선생님이
```

- books_2000.json 이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

③ 생성형 AI를 활용한 시리즈 정보가 있는 책들끼리 묶기

pathlib 라이브러리의 Path 클래스를 이용하여, books_500.json이 존재하는지 확인한다. 존재할 경우 해당 파일을 열어 JSON 데이터를 읽은 뒤 Python dictionary로 변환한다.

이후 생성형 AI를 활용하여 시리즈 정보를 포함한 책들을 추출, 시리즈에 대한 정보와 해당 시리즈에 속한 책들의 정보를 리스트로 모아 저장하는 코드를 생성한 다음, 정상 동작하는 코드인지 검증한다.

생성된 코드가 정상적으로 동작하면, 병합된 시리즈 정보를 담은 새로운 JSON 파일을 생성한다.

- 요구사항 번호: F13
- 주어진 problem_C3.py을 수정하여 구현
- 구현해야 할 화면 (콘솔 출력)

모든 시리즈 데이터가 series.json 파일로 병합되었습니다.

- 완성된 파일 일부

```
{
  "1036021": {
    "seriesId": 1036021,
    "seriesName": "잠뜰TV 오리지널 추리 스토리북 ",
    "books": [
      {
        "title": "잠뜰TV 오리지널 추리 스토리북 1~3권",
        "link": "http://www.aladin.co.kr/shop/wprodu",
        "author": "한바리, 잠뜰TV, 시우시 (지은이)",
        "pubDate": "2022-12-23",
        "description": "",
        "isbn": "6000595252",
        "isbn13": "9791169231206",
```

- books_2000.json이 존재하지 않을 경우 적절한 오류 메시지를 출력한다.

4) 심화 학습

프롬프트를 진행하면서 다음과 같은 심화 학습을 시도해볼 수 있다.

① 데이터 요약 정리 프롬프트 작성하기

데이터에서 인사이트를 뽑아낼 수 있는 지표나 분석 내용을 스스로 구성한 다음, 수집한 데이터를 기반으로 생성형 AI에 요약 요청해 본다.

- 요구사항 번호: F14

② 데이터를 활용하여 ChatGPT로 인사이트 얻기

할인율과 판매량의 관계, 저자별 도서 생산성, 추가 분석 항목을 직접 도출하여 실험해 본다.

- 요구사항 번호: F15

③ 가상의 책 정보를 생성형 AI를 활용하여 만들어 저장하기

가상의 책 정보를 생성형 AI를 활용하여 만들어 저장한다. 가상의 책 정보에는 제목, 저자, 출판사, 출판일, 가격, ISBN, 카테고리 등의 필드를 포함하고, JSON 형태로 저장해본다.

- 요구사항 번호: F16

상기된 과제를 전부 완료한 경우 명시된 요구사항 외 추가 개발을 자유롭게 진행한다.

5. 참고자료

- json – JSON encoder and decoder
<https://docs.python.org/3.11/library/json.html>
- pathlib – Object-oriented filesystem paths
<https://docs.python.org/3.11/library/pathlib.html>
- 프롬프트 전략 가이드
<https://www.promptingguide.ai/kr>

6. 결과

제출 기한은 진행일 18시까지이므로 제출 기한을 지킬 수 있도록 한다. 제출은 GitLab을 통해서 이뤄진다.

- 산출물과 제출
 - 단계별로 구현 과정 중 학습한 내용, 어려웠던 부분, 새로 배운 것들 및 느낀 점을 상세히 기록한 README.md
 - 완성된 각 문제 별 소스코드 및 실행 화면 캡처본
 - 프로젝트 이름은 01-pjt로 지정, 각자의 계정에 생성할 것
 - 각 반 담당 강사님을 Maintainer로 설정

- 끝 -