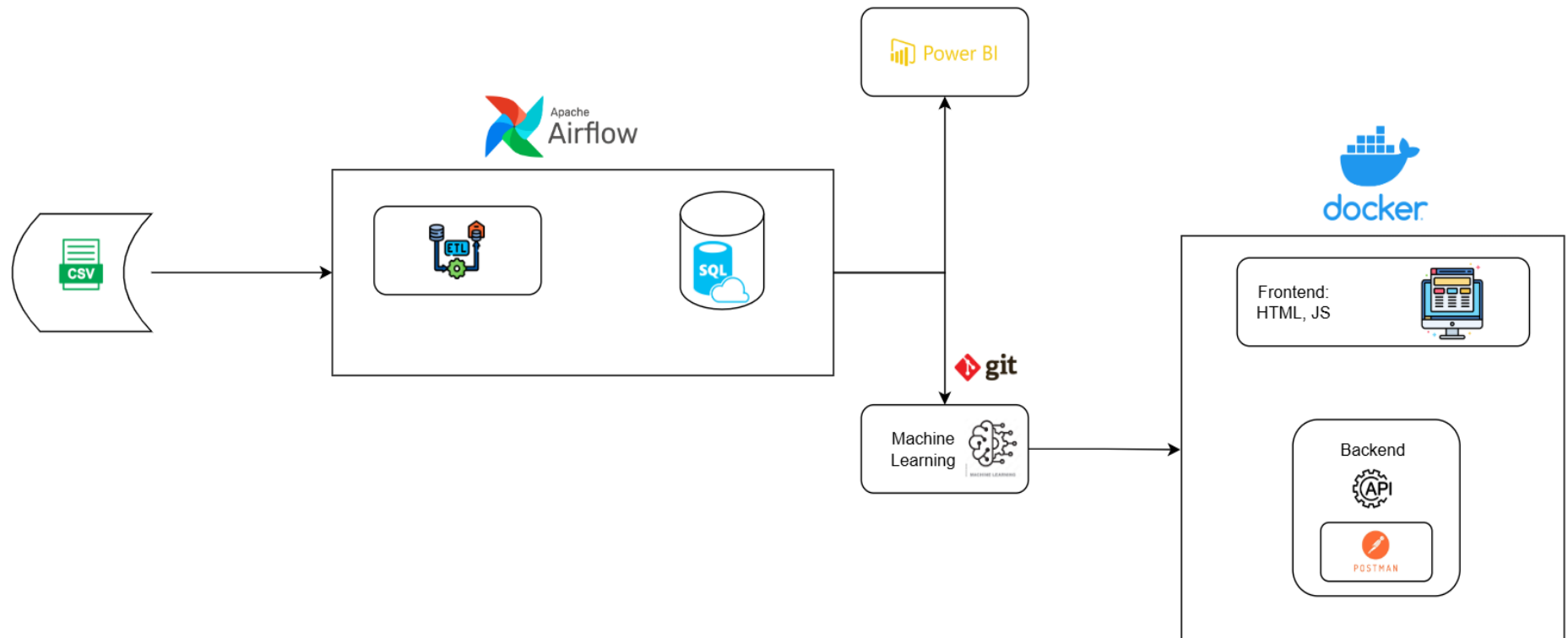
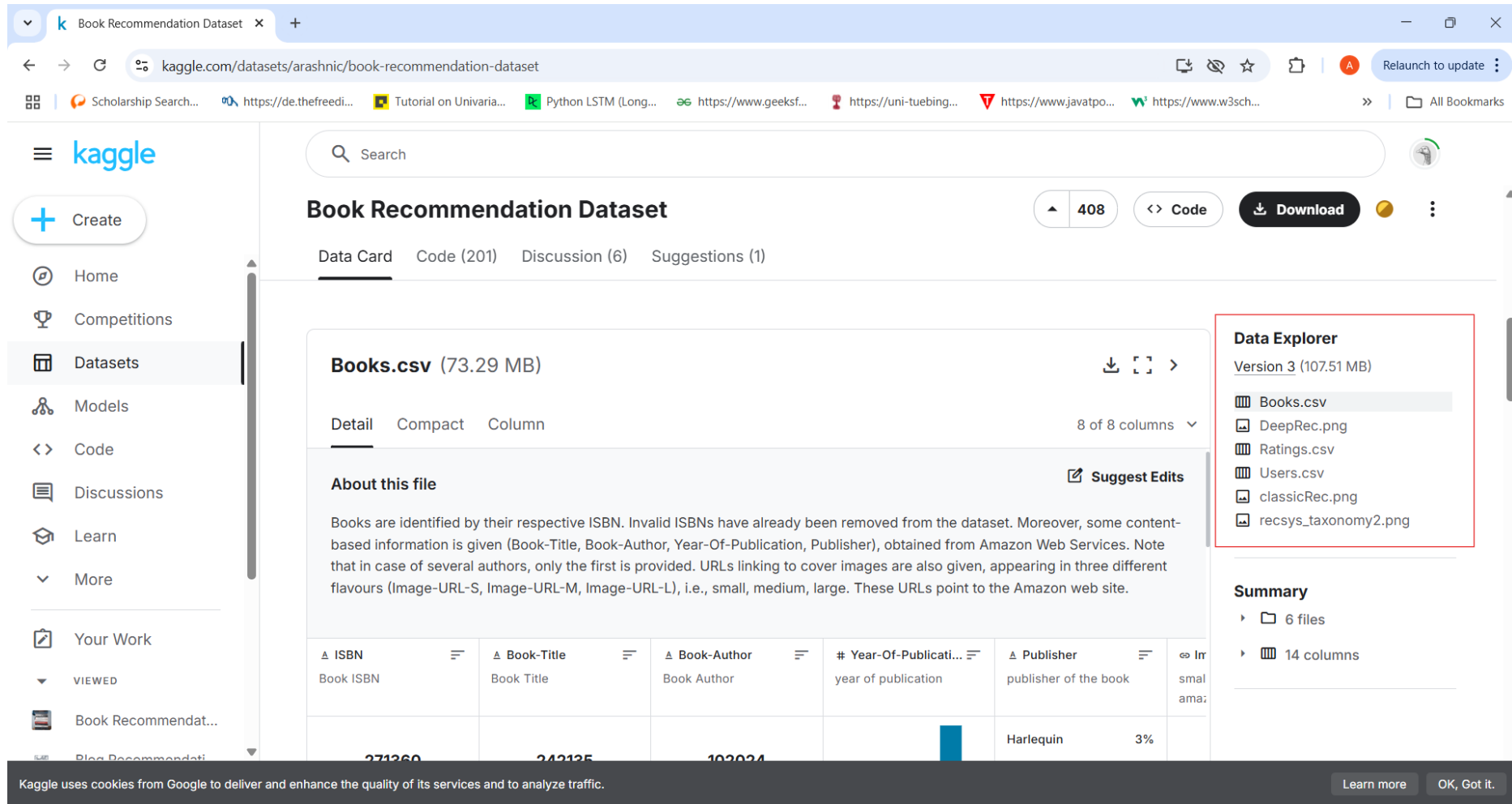


Data Engineering and Data Science Full End-to-End Project Structure



1) Getting data source



The screenshot shows the Kaggle interface for the 'Book Recommendation Dataset'. The main content area displays the 'Books.csv' file (73.29 MB) in the 'Data Card' view. The 'About this file' section provides details about the dataset, including ISBNs, book titles, authors, and publication years. The 'Data Explorer' sidebar on the right lists the files in the dataset: Books.csv, DeepRec.png, Ratings.csv, Users.csv, classicRec.png, and recsys_taxonomy2.png. The 'Summary' section shows 6 files and 14 columns.

Books.csv (73.29 MB)

Detail Compact Column 8 of 8 columns

About this file [Suggest Edits](#)

Books are identified by their respective ISBN. Invalid ISBNs have already been removed from the dataset. Moreover, some content-based information is given (Book-Title, Book-Author, Year-Of-Publication, Publisher), obtained from Amazon Web Services. Note that in case of several authors, only the first is provided. URLs linking to cover images are also given, appearing in three different flavours (Image-URL-S, Image-URL-M, Image-URL-L), i.e., small, medium, large. These URLs point to the Amazon web site.

ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher	Image-URL-S	Image-URL-M	Image-URL-L
0312360	242125	102024		Harlequin	3%		

Data Explorer
Version 3 (107.51 MB)

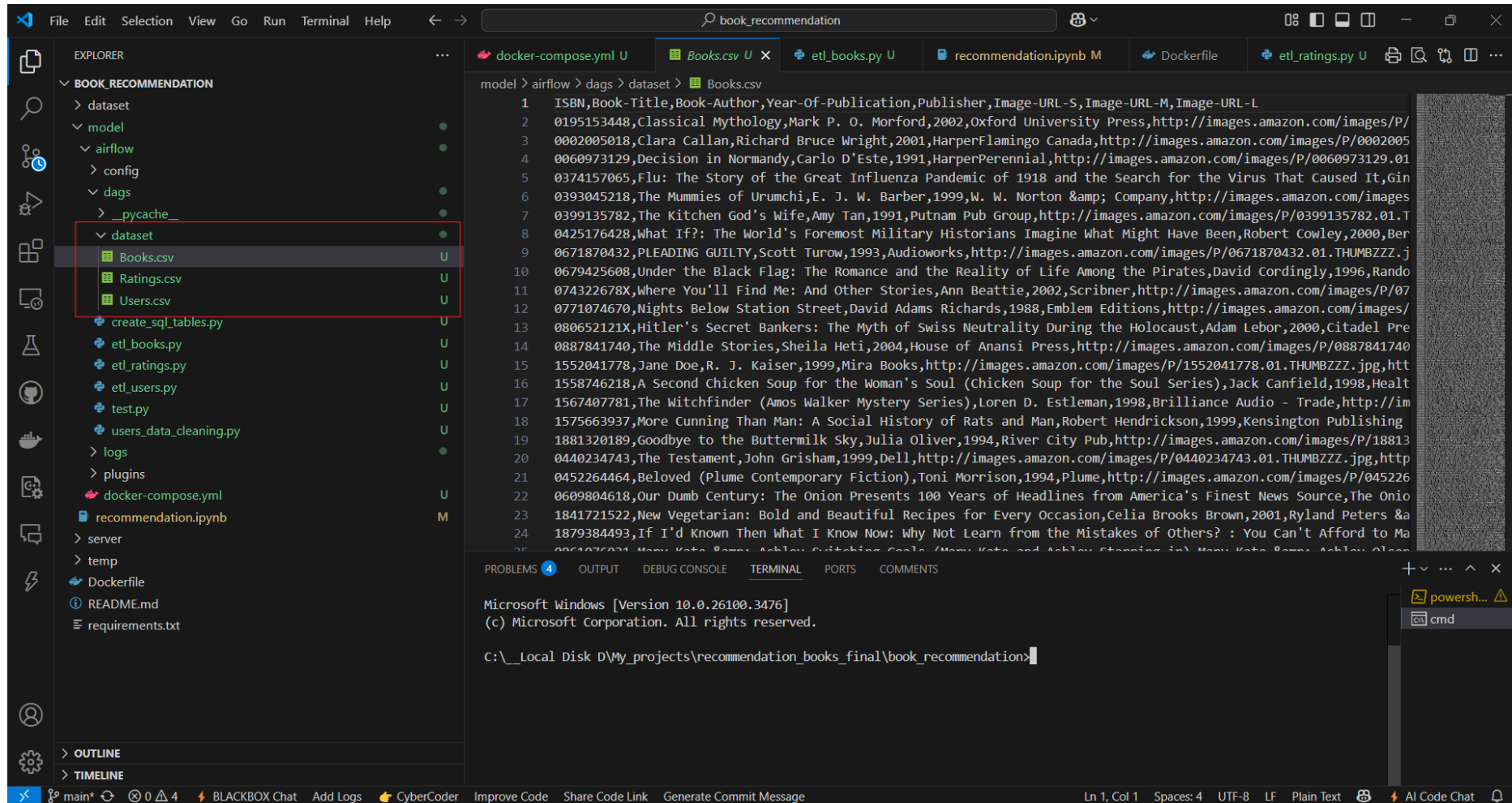
- Books.csv
- DeepRec.png
- Ratings.csv
- Users.csv
- classicRec.png
- recsys_taxonomy2.png

Summary

- 6 files
- 14 columns

Kaggle uses cookies from Google to deliver and enhance the quality of its services and to analyze traffic. [Learn more](#) [OK, Got it.](#)

2) Dataset location in our project



3) Initiating Apache Airflow locally

The screenshot displays the Visual Studio Code interface for a project named 'book_recommendation'. The Explorer sidebar on the left shows a directory structure with folders for 'dataset', 'model', 'server', and 'temp', and various files including CSVs, Python scripts, and configuration files. The 'docker-compose.yml' file is selected and open in the main editor. The code defines an 'x-airflow-common' environment with specific volumes for dags, logs, config, and plugins, and a user. It also defines a 'postgres' service with a specific image, environment variables, and a volume. The 'services' section lists 'postgres' and 'airflow' (partially visible). The Terminal at the bottom shows the command 'docker compose up' being executed, with a warning about the 'AIRFLOW_UID' variable and a list of containers being created.

```
model > airflow > docker-compose.yml
47 x-airflow-common:
54   environment:
75   volumes:
76     - ${AIRFLOW_PROJ_DIR:-.}/dags:/opt/airflow/dags
77     - ${AIRFLOW_PROJ_DIR:-.}/logs:/opt/airflow/logs
78     - ${AIRFLOW_PROJ_DIR:-.}/config:/opt/airflow/config
79     - ${AIRFLOW_PROJ_DIR:-.}/plugins:/opt/airflow/plugins
80   user: "${AIRFLOW_UID:-50000}:0"
81   depends_on:
82     &airflow-common-depends-on
83     redis:
84       condition: service_healthy
85     postgres:
86       condition: service_healthy
87
88 services:|
89   postgres:
90     image: postgres:13
91     environment:
92       POSTGRES_USER: airflow
93       POSTGRES_PASSWORD: airflow
94       POSTGRES_DB: airflow
95     volumes:
96       - postgres-db-volume:/var/lib/postgresql/data
97   ports:
```

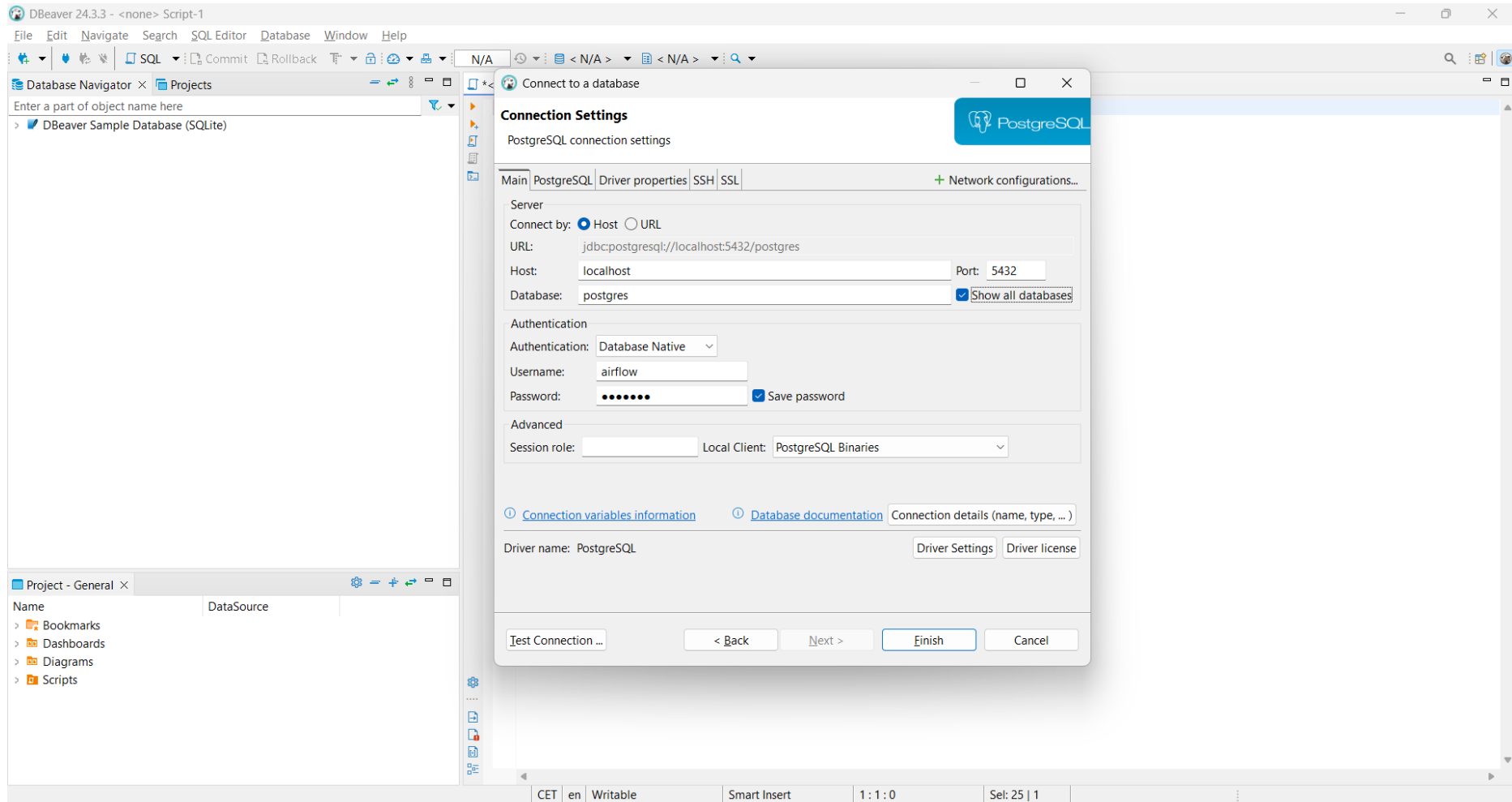
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

C:_Local Disk D\My_projects\recommendation_books_final\book_recommendation\model\airflow>docker compose up
time="2025-03-31T01:24:26+02:00" level=warning msg="The \"AIRFLOW_UID\" variable is not set. Defaulting to a blank string."
time="2025-03-31T01:24:26+02:00" level=warning msg="The \"AIRFLOW_UID\" variable is not set. Defaulting to a blank string."
[+] Running 7/0
✓ Container airflow-redis-1 Created 0.0s
✓ Container airflow-postgres-1 Created 0.0s
✓ Container airflow-airflow-init-1 Created 0.0s
✓ Container airflow-airflow-scheduler-1 Created 0.0s
✓ Container airflow-airflow-triggerer-1 Created 0.0s
✓ Container airflow-airflow-worker-1 Created 0.0s

4) Apache Airflow Interface

Version: v2.10.5
Git Version: .release:b93c3db6b1641b0840bd15ac7d05bc58ff2cccbf

5) Setting up SQL Server



6) Connecting Airflow to SQL Server

Book Recommendation Dataset x Add Connection - Airflow x +

localhost:8080/connection/add

Scholarship Search... https://de.thefreedi... Tutorial on Univaria... Python LSTM (Long... https://www.geeksf... https://uni-tuebing... https://www.javatpo... https://www.w3sch...

Airflow DAGs Cluster Activity Datasets Security Browse Admin Docs

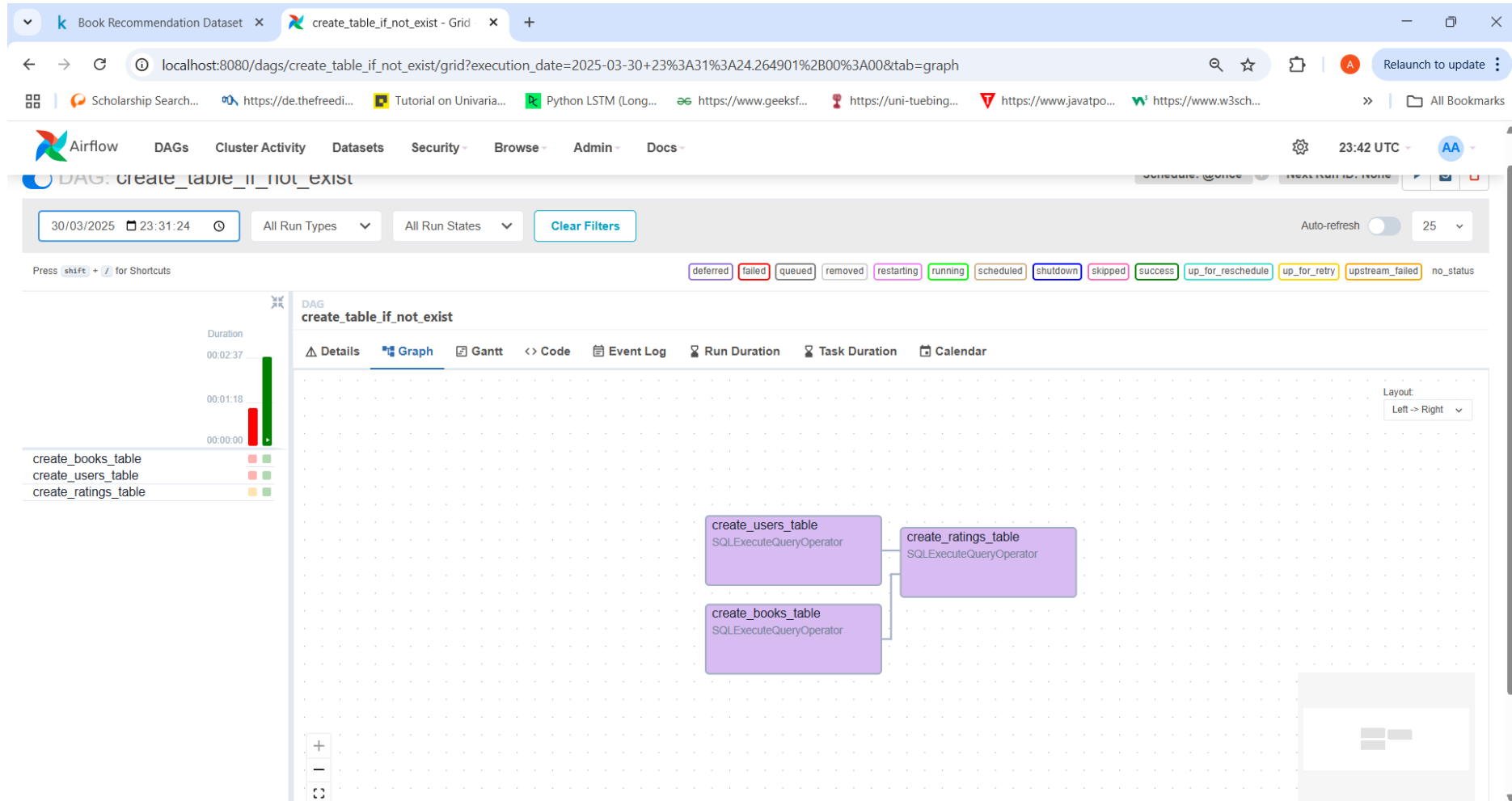
23:33 UTC AA

Warning: Fields that are currently populated can be modified but cannot be deleted. To delete data from a field, delete the Connection object and create a new one.

Add Connection

Connection Id *	postgres_localhost
Connection Type *	Postgres <small>Connection Type missing? Make sure you've installed the corresponding Airflow Provider Package.</small>
Description	
Host	host.docker.internal
Database	postgres
Login	airflow
Password	*****
Port	5432

7) Creating Tables on SQL using Apache Airflow DAGs



8) ETL Process

Book Recommendation Dataset x DAGs - Airflow x +

localhost:8080/home

Scholarship Search... https://de.thefreedi... Tutorial on Univaria... Python LSTM (Long... https://www.geeksf... https://uni-tuebing... https://www.javatpo... https://www.w3sch...

Airflow DAGs Cluster Activity Datasets Security Browse Admin Docs

23:42 UTC AA

DAGs

All 4 Active 4 Paused 0 Running 0 Failed 1 Filter DAGs by tag Search DAGs Auto-refresh

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Actions	Links
create_table_if_not_exist	azizjon	1 1	@once	2025-03-30, 23:31:24		3		...
import_books_to_database	azizjon	1	0 0 * * *	2025-03-30, 23:34:30	2025-03-31, 00:00:00	1		...
import_ratings_to_database	azizjon	1	0 0 * * *	2025-03-30, 23:41:22	2025-03-31, 00:00:00	1		...
import_users_to_database	azizjon	1	0 0 * * *	2025-03-30, 23:35:10	2025-03-31, 00:00:00	1		...

« < 1 > »

Showing 1-4 of 4 DAGs

Version: v2.10.5
Git Version: .release:b93c3db6b1641b0840bd15ac7d05bc58ff2cccbf

9) ETL Python codes behind Apache Airflow

The screenshot displays the Apache Airflow web interface. At the top, there's a navigation bar with links for DAGs, Cluster Activity, Datasets, Security, Browse, Admin, and Docs. The current time is 19:05 UTC. Below the navigation bar, a filter bar shows the date 10/04/2025 and time 19:04:57, with dropdowns for 'All Run Types' and 'All Run States', and a 'Clear Filters' button. A status bar at the top right lists various task states: deferred, failed, queued, removed, restarting, running, scheduled, shutdown, skipped, success, up_for_reschedule, up_for_retry, upstream_failed, and no_status. The main content area shows the DAG 'import_users_to_database' with a 'Code' tab selected. The code is a Python script that imports necessary modules, defines a default_args dictionary, and implements a function 'load_users_to_postgres' that reads a CSV file, processes it, and loads it into a PostgreSQL database. The code is as follows:

```
1 from airflow import DAG
2 from airflow.operators.python import PythonOperator
3 # import os
4 from airflow.providers.postgres.hooks.postgres import PostgresHook
5 from datetime import datetime, timedelta
6
7 from users_data_cleaning import get_correct_location, get_correct_age
8 from io import StringIO
9 import csv
10 import pandas as pd
11
12 default_args = {
13     "owner": "azizjon",
14     "retries": 0,
15     "retry_delay": timedelta(minutes=5)
16 }
17
18
19 def load_users_to_postgres(**context):
20     hook = PostgresHook(postgres_conn_id='postgres_localhost')
21     conn = hook.get_conn()
22     cursor = conn.cursor()
23
24     users_df = pd.read_csv('/opt/airflow/dags/dataset/users.csv')
25     users_df = get_correct_location(users_df)
26     users_df = get_correct_age(users_df)
27
28     clean_data = StringIO()
29     users_df.to_csv(clean_data, index=False, header=True, na_rep='NULL')
30
31     clean_data.seek(0)
```

At the bottom of the interface, the version information is displayed: Version: v2.10.5 and Git Version: .release:b93c3db6b1641b0840bd15ac7d05bc58ff2cccbf.

10) Running SQL commands to check for

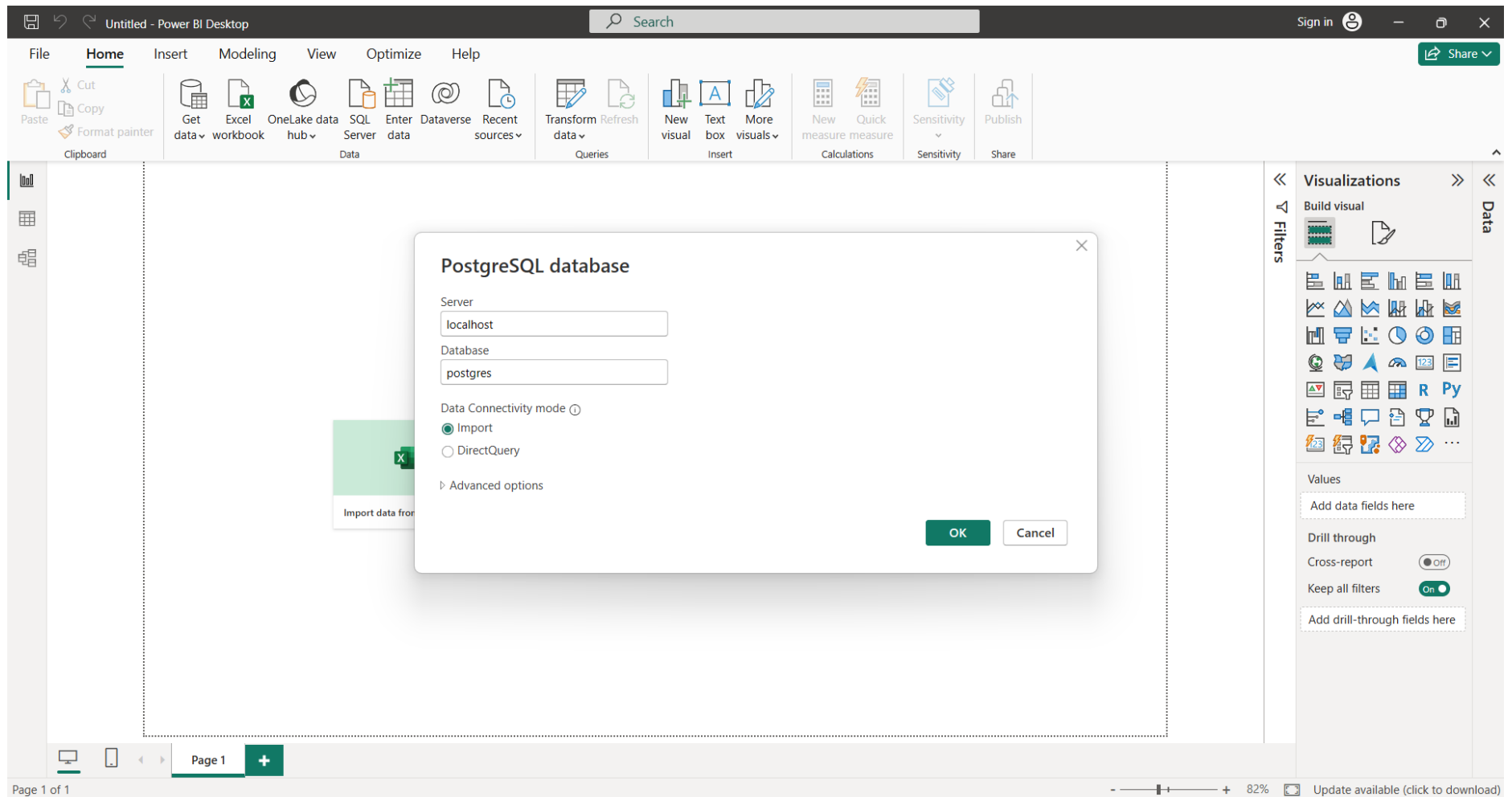
The screenshot displays the DBeaver 24.3.3 interface with a PostgreSQL database connection. The left sidebar shows the database structure, including the 'public' schema and the 'books' table. The main editor window contains a SQL query that selects book titles, authors, age groups, and average ratings, grouped by book title and author, and ordered by age group and average rating.

```
SELECT
    b.book_title,
    b.book_author,
    CASE
        WHEN u.age < 20 THEN 'Teens'
        WHEN u.age < 30 THEN '20s'
        WHEN u.age < 40 THEN '30s'
        WHEN u.age < 50 THEN '40s'
        ELSE '50+'
    END AS age_group,
    COUNT(*) AS rating_count,
    ROUND(AVG(r.book_rating), 2) AS avg_rating
FROM ratings r
JOIN books b ON r.ISBN = b.ISBN
JOIN users u ON r.user_id = u.user_id
GROUP BY b.book_title, b.book_author, age_group
HAVING COUNT(*) >= 50
ORDER BY age_group, avg_rating DESC;
```

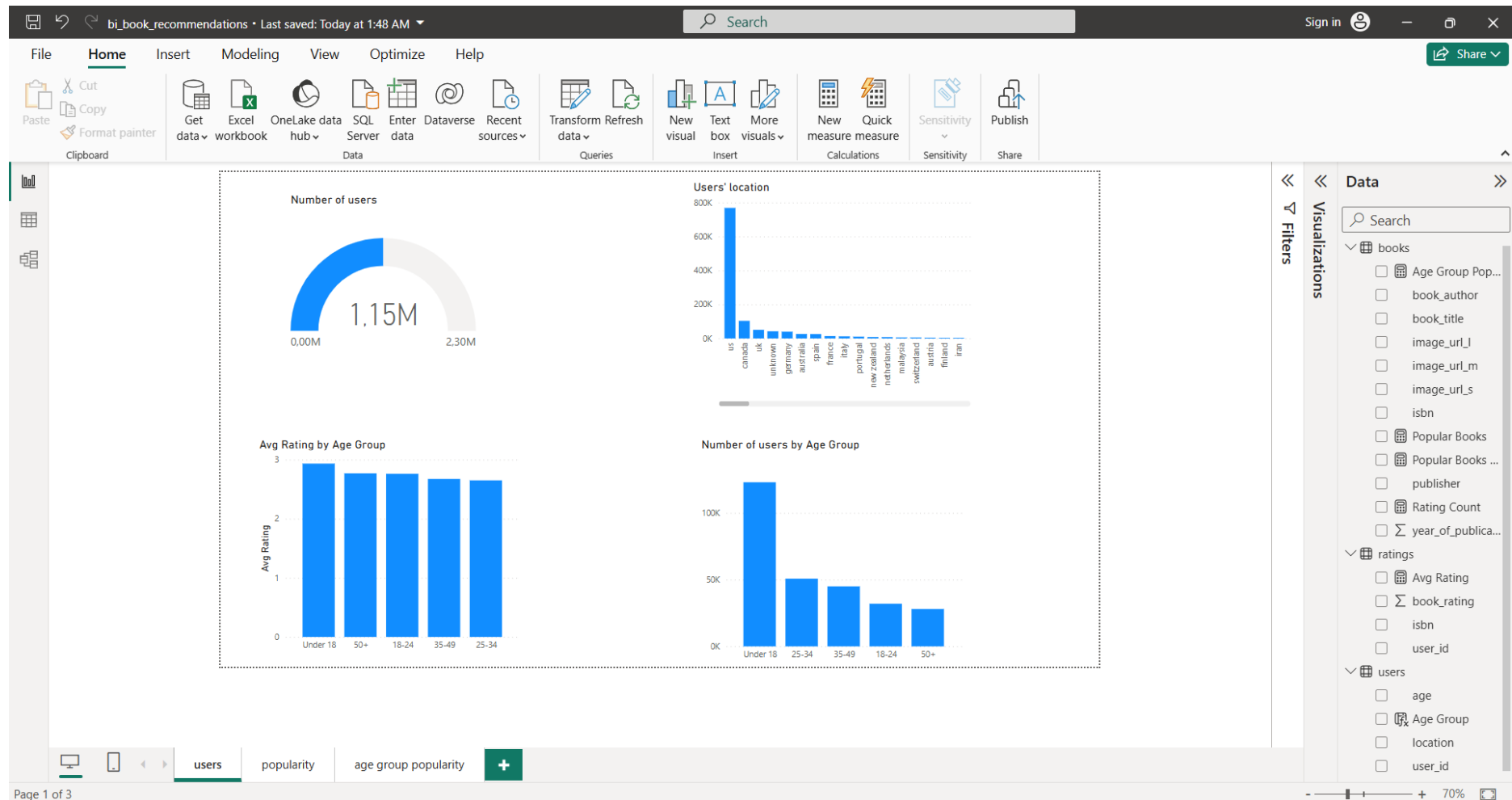
The results are displayed in a table with 13 rows and 6 columns. The first row is highlighted, showing the book 'Harry Potter and the Sorcerer's Stone' by J. K. Rowling, categorized as '20s', with a rating count of 123 and an average rating of 7.06.

	book_title	book_author	age_group	rating_count	avg_rating
1	Harry Potter and the Sorcerer's Stone	J. K. Rowling	20s	123	7.06
2	Ender's Game (Ender Wiggins Saga (Pa	Orson Scott Card	20s	54	6.69
3	Harry Potter and the Goblet of Fire (Bo	J. K. Rowling	20s	100	6.44
4	Harry Potter and the Prisoner of Azkab	J. K. Rowling	20s	106	6.33
5	Harry Potter and the Order of the Phoe	J. K. Rowling	20s	90	6.31
6	The Two Towers (The Lord of the Rings	J.R.R. TOLKIEN	20s	55	5.89
7	The Five People You Meet in Heaven	Mitch Albom	20s	71	5.46
8	Harry Potter and the Chamber of Secre	J. K. Rowling	20s	136	5.35
9	The Fellowship of the Ring (The Lord o	J.R.R. TOLKIEN	20s	73	5.33
10	Harry Potter and the Sorcerer's Stone (J. K. Rowling	20s	143	5.23
11	The Catcher in the Rye	J.D. Salinger	20s	125	5.14
12	The Gunslinger (The Dark Tower, Book	Stephen King	20s	62	5.13
13	To Kill a Mockingbird	Harper Lee	20s	111	4.94

11) Connecting PowerBI to SQL Server



12) SQL Database usage on PowerBI



13) ML Model Backend – Frontend: recommendation.ipynb file must be executed to get server artifacts

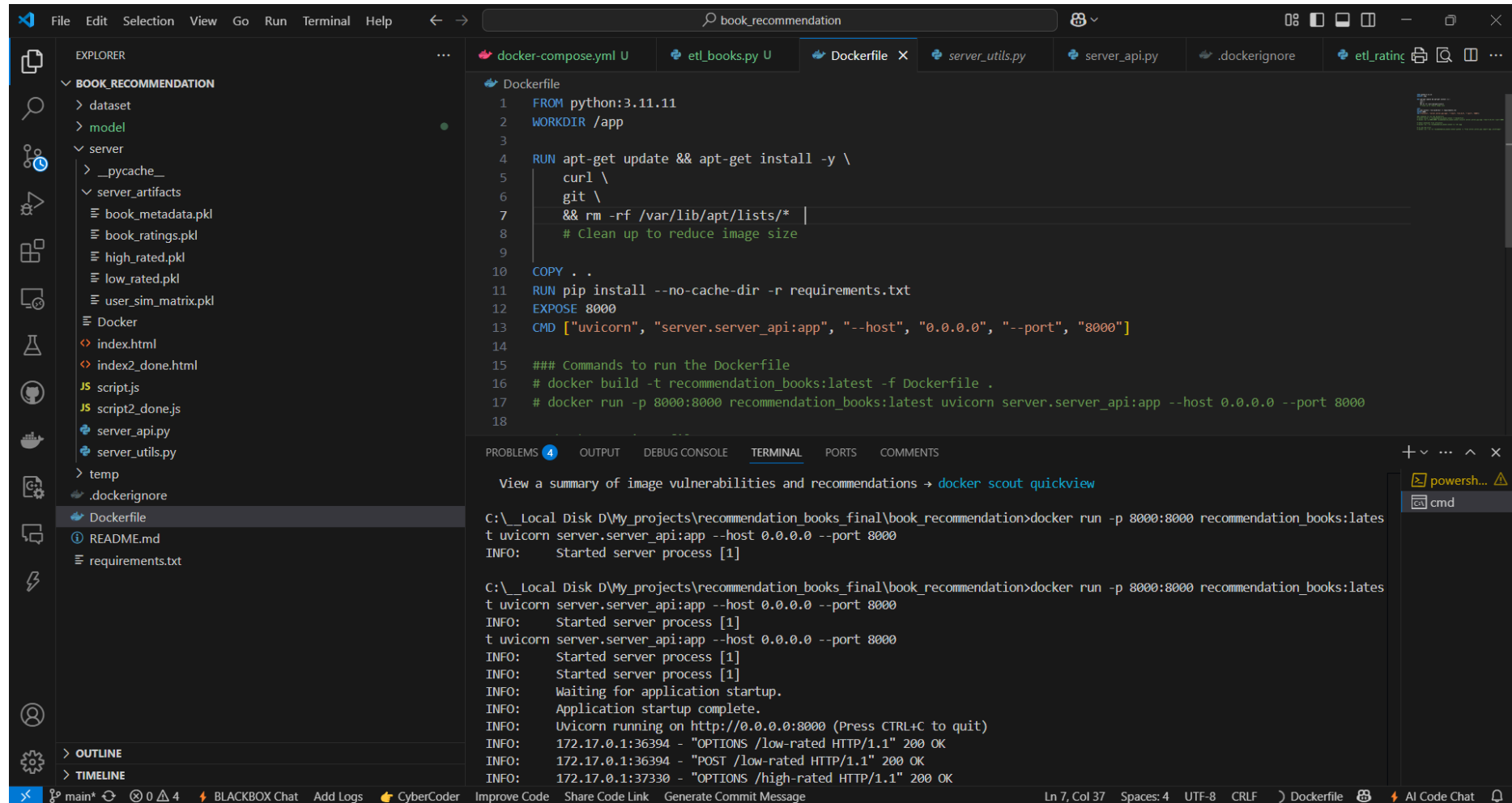
The screenshot displays the Visual Studio Code interface for a project named 'book_recommendation'. The Explorer sidebar on the left shows the project structure, with the 'model' folder expanded and 'recommendation.ipynb' selected. The main editor area shows the Jupyter Notebook 'Recommendation system' with a table of book data. The table has columns: ISBN, Book-Title, Book-Author, Year-Of-Publication, Publisher, and Image-URL-S. The data shows three entries for the book 'Clara Callan' by 'Richard Bruce Wright' published in 2001 by 'HarperFlamingo Canada'. The Terminal panel at the bottom shows the output of stopping Docker containers, indicating that the containers are stopped successfully.

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher	Image-URL-S
1	0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada	http://images.amazon.com/images/P/0002005018.0... http://images.amazon.com/images:
2	0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada	http://images.amazon.com/images/P/0002005018.0... http://images.amazon.com/images:
3	0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada	http://images.amazon.com/images/P/0002005018.0... http://images.amazon.com/images:

Terminal Output:

```
[+] Stopping 7/7
✓ Container airflow-airflow-scheduler-1 Stopped 17.0s
✓ Container airflow-airflow-webserver-1 Stopped 17.9s
✓ Container airflow-airflow-triggerer-1 Stopped 17.2s
✓ Container airflow-airflow-worker-1 Stopped 17.4s
✓ Container airflow-airflow-init-1 Stopped 0.0s
✓ Container airflow-redis-1 Stopped 1.7s
✓ Container airflow-postgres-1 Stopped 2.4s
canceled
```

14) Deploying ML Model using Docker



The screenshot shows the Visual Studio Code interface with a project named 'book_recommendation'. The Explorer panel on the left shows the project structure, including a 'server' directory with a 'Dockerfile' and 'requirements.txt'. The Dockerfile is open in the editor, showing the following content:

```
1 FROM python:3.11.11
2 WORKDIR /app
3
4 RUN apt-get update && apt-get install -y \
5     curl \
6     git \
7     && rm -rf /var/lib/apt/lists/* |
8     # Clean up to reduce image size
9
10 COPY . .
11 RUN pip install --no-cache-dir -r requirements.txt
12 EXPOSE 8000
13 CMD ["uvicorn", "server.server_api:app", "--host", "0.0.0.0", "--port", "8000"]
14
15 ## Commands to run the Dockerfile
16 # docker build -t recommendation_books:latest -f Dockerfile .
17 # docker run -p 8000:8000 recommendation_books:latest uvicorn server.server_api:app --host 0.0.0.0 --port 8000
18
```

The Terminal panel at the bottom shows the output of the Docker commands:

```
C:\_Local Disk D\My_projects\recommendation_books_final\book_recommendation>docker run -p 8000:8000 recommendation_books:lates
t uvicorn server.server_api:app --host 0.0.0.0 --port 8000
INFO: Started server process [1]

C:\_Local Disk D\My_projects\recommendation_books_final\book_recommendation>docker run -p 8000:8000 recommendation_books:lates
t uvicorn server.server_api:app --host 0.0.0.0 --port 8000
INFO: Started server process [1]
INFO: Started server process [1]
INFO: Waiting for application startup.
INFO: Application startup complete.
INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
INFO: 172.17.0.1:36394 - "OPTIONS /low-rated HTTP/1.1" 200 OK
INFO: 172.17.0.1:36394 - "POST /low-rated HTTP/1.1" 200 OK
INFO: 172.17.0.1:37330 - "OPTIONS /high-rated HTTP/1.1" 200 OK
```

The status bar at the bottom indicates the current file is 'Dockerfile' and the encoding is 'UTF-8'.

15) Postman to check for API backend

The screenshot displays the Postman API client interface. At the top, there's a navigation bar with 'Home', 'Workspaces', and 'Explore' options, along with a search bar and buttons for 'Sign In' and 'Create Account'. A notification banner states: 'You are using the Lightweight API Client, sign in or create an account to work with collections, environments and unlock all free features in Postman.'

The left sidebar shows the 'History' tab with a list of recent requests. The main workspace is configured for a POST request to 'http://localhost:8000/recommend'. The 'Body' tab is selected, showing a JSON payload:

```
1 {
2   "num_recommendations": 5,
3   "user_id": 8
4 }
5
```

Below the request editor, the 'Body' tab of the response is shown, displaying a JSON array of book recommendations:

```
1 {
2   "Book-Title": "The Progress of Love (King Penguin)",
3   "Book-Author": "Alice Munro",
4   "Year-Of-Publication": 1987,
5   "Publisher": "Penguin Books",
6   "Image-URL-L": "http://images.amazon.com/images/P/0140098798.01.LZZZZZZZ.jpg"
7 },
8 {
9   "Book-Title": "To Kill a Mockingbird",
10  "Book-Author": "Harper Lee",
11  "Year-Of-Publication": 1988,
12  "Publisher": "Little Brown & Company",
13  ...
14 }
```

At the bottom left, there's a prompt to 'Create collections in Postman' with a 'Create a Collection' button. The bottom status bar shows 'Status: 200 OK', 'Time: 25 ms', and 'Size: 1.13 KB'.

16) Simple Web UI

Book Recommendation System x FastAPI - Swagger UI x +

127.0.0.1:8000/docs/#

Scholarship Search... https://de.thefreedi... Tutorial on Univaria... Python LSTM (Long... https://www.geeksf... https://uni-tuebing... https://www.javatpo... https://www.w3sch...

Alle Lesezeichen

FastAPI 0.1.0 OAS 3.1

/openapi.json

default

- GET** / Home
- POST** /high-rated High Rated
- POST** /low-rated Low Rated
- GET** /existing-users Existing Users
- POST** /recommend Recommend Books

Schemas

17) Webpage using HTML and JS

Book Recommendation System

←

→

🔄

📄

Date

c%3A/Users/AZTURAEV/main_folder/selbst/recommendation_books/server/index.html

☆

📁

📄

🔴

⋮

Scholarship Search...https://de.thefreedi...Tutorial on Univaria...Python LSTM (Long...https://www.geeksf...https://uni-tuebing...https://www.javatpo...https://www.w3sch...

»

📁

Alle Lesezeichen

Book Recommendation System

Low-Rated Books

High-Rated Books

Existing Users

Get Recommendations

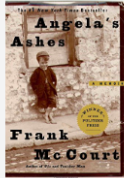
Number of Books:

57

User ID:

99

Submit

Book Title	Author	Year	Publisher	Image
ANGELA'S ASHES	Frank McCourt	1996	Scribner	
The Fall of Hyperion	Dan Simmons	1991	Bantam	