import pandas as pd

from sklearn.metrics.pairwise import cosine\_similarity

from sklearn.feature\_extraction.text import CountVectorizer

data = {

'movie\_id': [1, 2, 3, 4, 5],

'title': ['The Matrix', 'Inception', 'Interstellar', 'The Dark Knight', 'The Prestige'],

'genres': ['Action Sci-Fi', 'Action Sci-Fi Thriller', 'Adventure Drama Sci-Fi', 'Action Crime Drama', 'Drama Mystery Sci-Fi']

}

movies = pd.DataFrame(data)

vectorizer = CountVectorizer()

genre\_matrix = vectorizer.fit\_transform(movies['genres'])

similarity = cosine\_similarity(genre\_matrix)

def recommend\_movie(movie\_title, num\_recommendations=3):

if movie\_title not in movies['title'].values:

return f"Movie '{movie\_title}' not found in database."

idx = movies[movies['title'] == movie\_title].index[0]

scores = list(enumerate(similarity[idx]))

sorted\_scores = sorted(scores, key=lambda x: x[1], reverse=True)

top\_movies = sorted\_scores[1:num\_recommendations+1]

recommendations = [movies.iloc[i[0]]['title'] for i in top\_movies]

return recommendations

user\_input = "Inception"

recommended = recommend\_movie(user\_input)

print(f"Because you liked '{user\_input}', you might also like: {recommended}")