from pyspark.sql import SparkSession

from pyspark.sql.functions import col

from pyspark.sql import functions as F

spark = SparkSession.builder.appName("Movies").getOrCreate()

movies\_file\_path = "/content/sample\_data/movies.csv"

# 1. \*\*Load the Dataset\*\*:

#    - Read the CSV file into a PySpark DataFrame.

df\_movies = spark.read.format("csv").option("header","true").option("inferSchema","true").load(movies\_file\_path)

df\_movies.show()

# 2. \*\*Filter Movies by Genre\*\*:

#    - Find all movies in the "Sci-Fi" genre.

scifi\_df = df\_movies.filter(col("genre") == "Sci-Fi")

scifi\_df.show()

# 3. \*\*Top-Rated Movies\*\*:

#    - Find the top 3 highest-rated movies

top\_rated\_df = df\_movies.orderBy(col("rating").desc()).limit(3)

top\_rated\_df.show()

# 4. \*\*Movies Released After 2010\*\*:

#    - Filter out all movies released after the year 2010.

movies\_after\_2010\_df = df\_movies.filter(F.year(F.to\_date(col("date"),'yyyy-MM-dd')) > 2010)

movies\_after\_2010\_df.show()

# 5. \*\*Calculate Average Box Office Collection by Genre\*\*:

#    - Group the movies by `genre` and calculate the average box office collection for each genre.

box\_office\_genre\_df = df\_movies.groupBy("genre").agg(F.avg("box\_office").alias("AverageCollection"))

box\_office\_genre\_df.show()

# 6. \*\*Add a New Column for Box Office in Billions\*\*:

#    - Add a new column that shows the box office collection in billions.

df\_movies\_billions = df\_movies.withColumn("InBillions", col("box\_office")/ 1000000000)

df\_movies\_billions.show()

# 7. \*\*Sort Movies by Box Office Collection\*\*:

#    - Sort the movies in descending order based on their box office collection.

df\_movies.orderBy(col("box\_office").desc()).show()

# 8. \*\*Count the Number of Movies per Genre\*\*:

#    - Count the number of movies in each genre.

movies\_genre\_df = df\_movies.groupBy("genre").agg(F.count("title").alias("NumberOfMovies"))

movies\_genre\_df.show()