

Homework Assignment 03

DS 4300 - Spring 2025

- **EC Due Date:** Feb 16, 2025 @ 11:59pm
- **Regular Due Date:** Feb 18, 2025 @ 11:59pm
- Upload to GradeScope (no question/solutions to Match)

```
In [ ]: # Set up your connection to Mongo DB here.
import pymongo
from bson.json_util import dumps

# --> Update the URI with your username and password <--

uri = "mongodb://kenneth:123@localhost:27018"
client = pymongo.MongoClient(uri)
mflixdb = client.mflix
```

Directions:

- Use the mflix sample database to prepare a pymongo query each of the following prompts.
- Be sure to print the results of your query using the `dumps` function.

Question 1:

Give the street, city, and zipcode of all theaters in Massachusetts.

```
In [ ]: data = mflixdb.theaters.find({"location.address.state": "MA"}, {"_id": 0, "location": 1})
print(dumps(data, indent=2))
```

Question 2:

How many theaters are there in each state? Order the output in alphabetical order by 2-character state code.

```
In [ ]: data = mflixdb.theaters.aggregate([{"$group": {"_id": {"state": "$location.address.state"}, "count": {"$sum": 1}}])
print(dumps(data, indent=2))
```

Question 3:

How many movies are in the Comedy genre?

```
In [ ]: numMovies = mflixdb.movies.count_documents({"genres": {"$in": ["Comedy"]}})
print(f'Number of movies in Comedy: {numMovies}')
```

Question 4:

What movie has the longest run time? Give the movie's title and genre(s).

```
In [ ]: data = mflixdb.movies.find({}, {"_id": 0, "title":1, "genres":1}).sort("runtime", -
print(dumps(data, indent=2))
```

Question 5:

Which movies released after 2010 have a Rotten Tomatoes viewer rating of 3 or higher? Give the title of the movies along with their Rotten Tomatoes viewer rating score. The viewer rating score should become a top-level attribute of the returned documents. Return the matching movies in descending order by viewer rating.

```
In [ ]: data = mflixdb.movies.aggregate([{"$match": {"year": {"$gt":2010}, "tomatoes.viewer
{"$project": {"_id": 0, "title":1, "rating":"$toma
print(dumps(data, indent=2))
```

Question 6:

How many movies released each year have a plot that contains some type of police activity (i.e., plot contains the word "police")? The returned data should be in ascending order by year.

```
In [ ]: data = mflixdb.movies.aggregate([{"$match": {"plot": {"$regex": "police", "$options
{"$group": {"_id": {"year": "$year"}, "count": {"$
print(dumps(data, indent=2))
```

Question 7:

What is the average number of imdb votes per year for movies released between 1970 and 2000 (inclusive)? Make sure the results are order by year.

```
In [ ]: data = mflixdb.movies.aggregate([{"$match": {"year": {"$gte": 1970, "$lte": 2000}}}
{"$group": {"_id": "$year", "avg_votes": {"$avg":
print(dumps(data, indent=2))
```

Question 8:

What distinct movie languages are represented in the database? You only need to provide the list of languages.

```
In [ ]: data = mflixdb.movies.aggregate([{"$unwind": "$languages"}, {"$group": {"_id": "$la  
print(dumps(data, indent=2))
```