

Annand_module08_lab01

March 4, 2025

0.1 Import Pymongo

```
[1]: #import pymongo
import pymongo

#create your connection string
connect_string = "mongodb+srv://annandj:kitkatbaka@cluster0.mmpel.mongodb.net/?
↳retryWrites=true&w=majority&appName=Cluster0"
#create a connection to your Atlas cluster
client = pymongo.MongoClient(connect_string)

#connect to the sample_mflix.movies db
collection = client['sample_mflix']['movies']
```

0.2 Exercise 1

Create a mongo_db connection with pymongo to your database
<https://pymongo.readthedocs.io/en/stable/examples/authentication.html>

For the homework we will be using the sample_mflix.movies collection.

What is the title of the movie with the highest IMDB rating?

Note match, sort, limit, project. collection.aggregate(query) is the syntax for aggregation pipelines in Python.

<https://pymongo.readthedocs.io/en/stable/examples/aggregation.html>

```
[2]: result1 = collection.aggregate([
    {
        '$match': {
            'imdb.rating': {
                '$type': 'number'
            }
        }
    }, {
        '$sort': {
            'imdb.rating': -1
        }
    }, {
```

```

        '$limit': 1
    }, {
        '$project': {
            'title': 1,
            'imdb_rating': '$imdb.rating'
        }
    }
])

result1 = list(result1)

for doc in result1:
    print(doc.items())

```

```
dict_items([(' _id', ObjectId('573a139ff29313caabd003c4')), ('title', 'Band of
Brothers'), ('imdb_rating', 9.6)])
```

0.3 Exercise 2

Which year had the most titles released? **Note** group, sort, limit

```

[3]: result2 = collection.aggregate([
    {
        '$group': {
            '_id': '$year',
            'count': {
                # setting $sum equal to one adds one for each document in the
                ↪group
                '$sum': 1
            }
        }
    }, {
        '$sort': {
            'count': -1
        }
    }, {
        '$limit': 1
    }
])

result2 = list(result2)

for doc in result2:
    print(doc.items())

```

```
dict_items([(' _id', 2013), ('count', 1105)])
```

0.4 Exercise 4

What are the four directors with the most titles accredited to them? **Note** project, unwind, group, sort, limit

```
[4]: result3 = collection.aggregate([
    {
        '$project': {
            'title': 1,
            'directors': 1
        }
    }, {
        '$unwind': {
            'path': '$directors'
        }
    }, {
        '$group': {
            '_id': '$directors',
            'count': {
                '$sum': 1
            }
        }
    }, {
        '$sort': {
            'count': -1
        }
    }, {
        '$limit': 4
    }
])

result3 = list(result3)

for doc in result3:
    print(doc.items())
```

```
dict_items([('id', 'Woody Allen'), ('count', 40)])
dict_items([('id', 'Martin Scorsese'), ('count', 32)])
dict_items([('id', 'Takashi Miike'), ('count', 31)])
dict_items([('id', 'John Ford'), ('count', 29)])
```

0.5 Exercise 4

Show the title and number of languages the movie was produced in for the following: Year:2013, genre:'Action'

```
[5]: result4 = collection.aggregate([
    {
        '$unwind': {
```

```

        'path': '$genres'
    }
}, {
    '$match': {
        '$and': [
            {
                'genres': 'Action'
            }, {
                'year': 2013
            }
        ]
    }
}, {
    '$project': {
        'title': 1,
        'num_of_lang': {
            # use $ifNull to replace null values with empty array
            # worked fine in Atlas without but Pymongo raised error
            '$size': { '$ifNull': ['$languages', []] }
        }
    }
}
])

result4 = list(result4)

# print contents of first five documents
for i in range(5):
    print(result4[i])

```

```

{'_id': ObjectId('573a13b7f29313caabd4a861'), 'title': 'Man of Steel',
 'num_of_lang': 1}
{'_id': ObjectId('573a13b9f29313caabd4de39'), 'title': 'World War Z',
 'num_of_lang': 4}
{'_id': ObjectId('573a13bbf29313caabd52703'), 'title': 'Snitch', 'num_of_lang':
 2}
{'_id': ObjectId('573a13bef29313caabd5c368'), 'title': 'Game of Assassins',
 'num_of_lang': 1}
{'_id': ObjectId('573a13bef29313caabd5d9c2'), 'title': 'Lone Survivor',
 'num_of_lang': 2}

```

```
[6]: client.close()
```