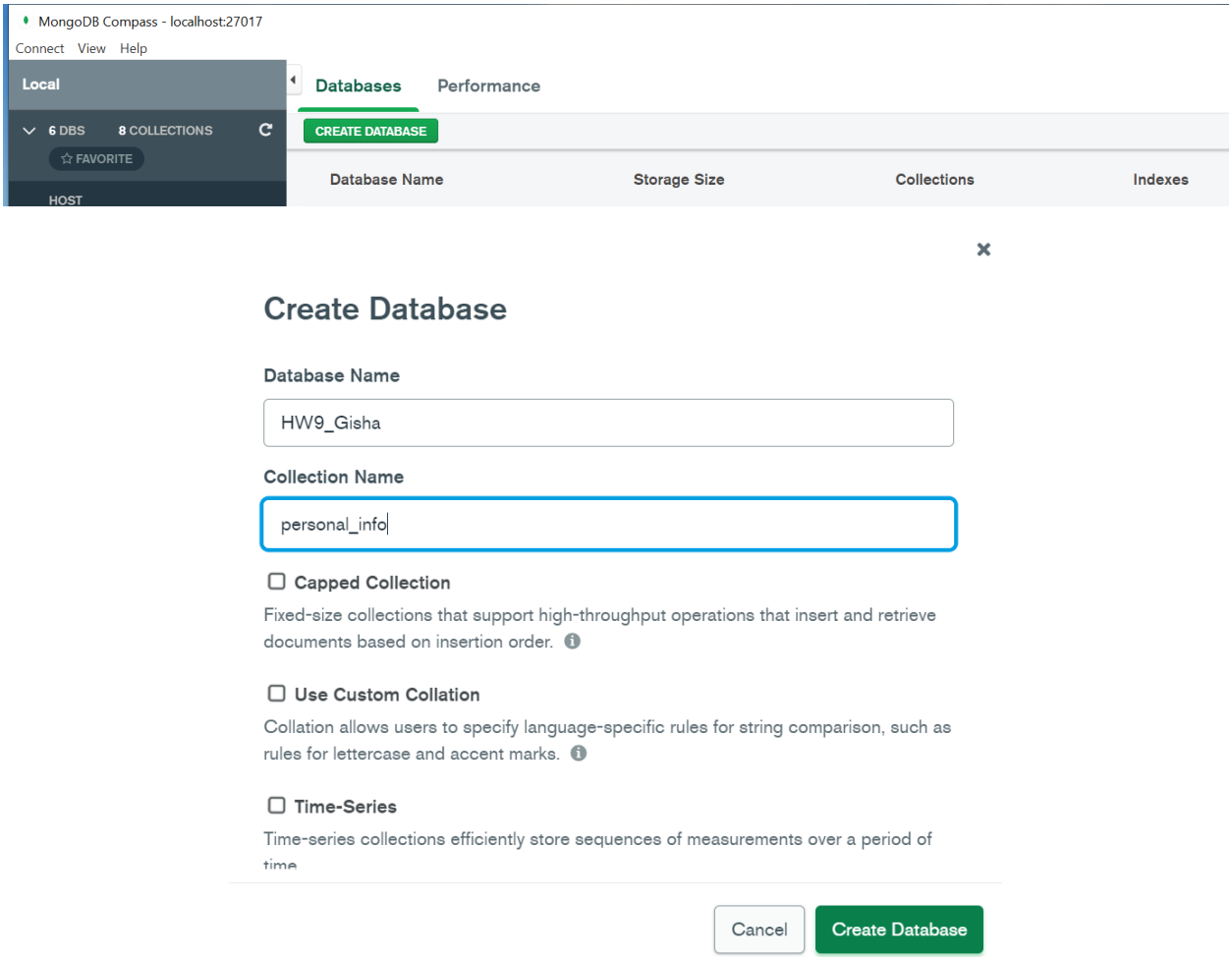
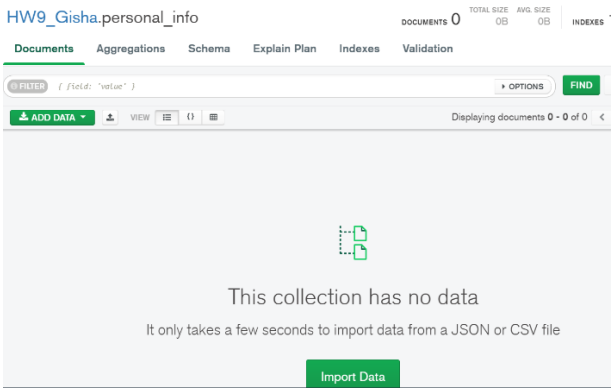
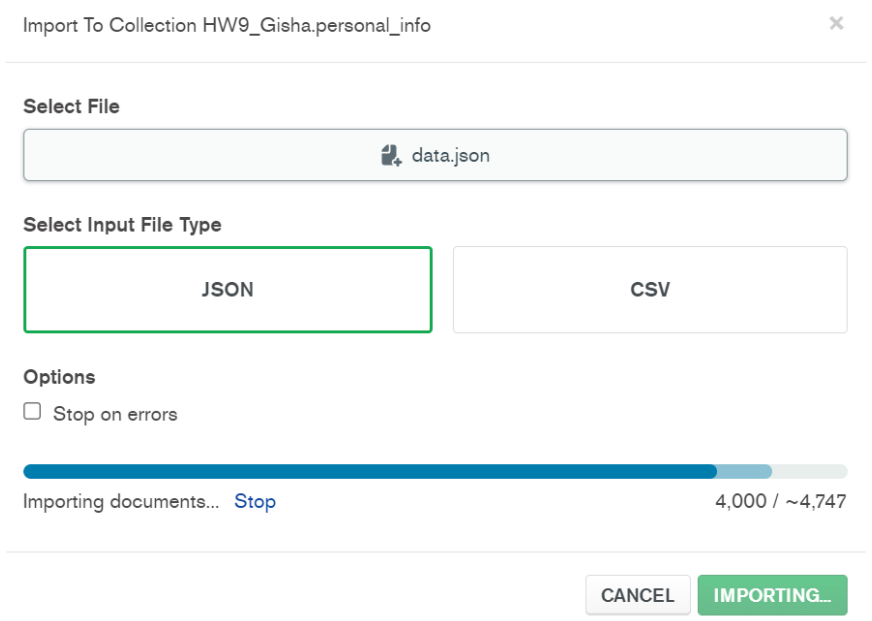
تمرین hw9 قسمت A :

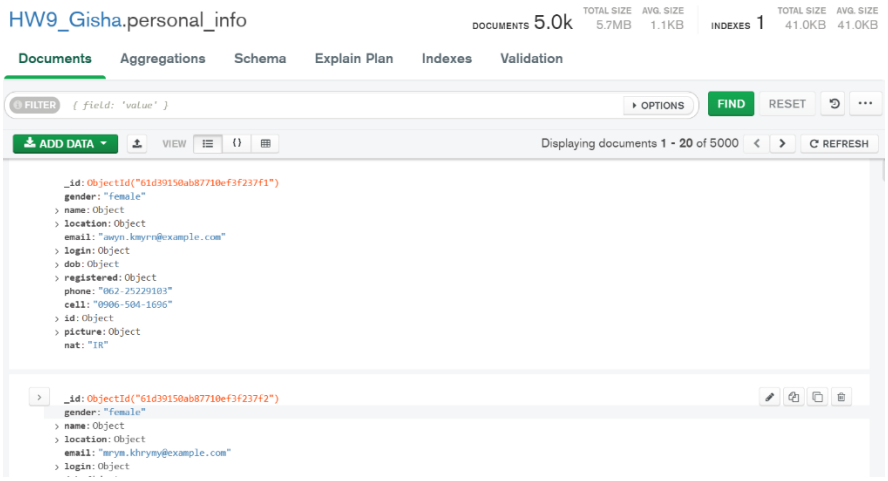
ایجاد دیتابیس و کالکشن



اضافه کردن دیتا از فایل جیسون







قسمت C :

ایجاد دیتابیس و کانت شدن به دیتابیس و کالکشن در پایتون

بعد از ایجاد کالکشن برای جلوگیری از تکرار ایجاد کد کامنت شده است .

```
from pymongo import MongoClient
import json
import datetime

client = MongoClient(host="localhost", port=27017)
hw9_db = client["HW9"]  # creat and connect to database
# new_collection = hw9_db.create_collection("personal_info") #creat collection comment it to avoid recreating
my_collection = hw9_db["personal_info"]  # after creating collection we should call the collection
```

اضافه کردن دیتا و استفاده از countدر پایتون :

برای جلوگیری از اضافه شدن مجدد کد کامنت شده است .

```python
# def insert_data():
#     with open("data.json", "r", encoding="utf8") as my_file:
#         read = json.load(my_file)
#         for i in read:
#             yield i

# read data from the json file

# my_collection.insert_many(insert_data()) #insert data to the collection comment it to avoid repeating

# using count
counting = my_collection.count_documents({})
print(f"Number of documents : {counting}")
```

نتیجه استفاده از count :

```
C:\Users\gisha\AppData\Local\Programs\Python\Python39\python.exe "D:/maktab sharif/maktab 65/HW/hw9/hw9.py"
Number of documents : 5000
```

قسمت B :

کوئری شهر ها با بیشترین و کمترین یوزر :

```python
# Tell  cities with the most and least users >>>
result = my_collection.aggregate(
    [
        {
            "$group": {
                "_id": "$location.city",
                "count_users": {
                    "$count": {}
                }
            }
        },
        {
            "$sort": {
                "count_users": 1
            }}
    ])
result = list(result)  # because the type of result is CommandCursor we change it to list of dicts
most_number, most_city = result[-1]["count_users"], result[-1]["_id"]
least_number, least_city = result[0]["count_users"], result[0]["_id"]
print(f"Most :\n{most_number} users are from {most_city}\n"
      f"least :\n{least_number} users are from {least_city}")
```

اگر در فضای شل کوئری بزنیم باید از لیمیت استفاده کنیم. اول به صورت نزولی sort میکنیم و بعد با لیمیت اولین عضو نتیجه را پرینت میگیریم به عنوان بیشترین یوزر .

برای کمترین یوزر به صورت صعودی sort میکنیم و لیمیت اولین عضو را نشان میدهد .

نتیجه کد پایتون :

```
Most :
118 users are from بیرجند
least :
72 users are from سنندج
```

کوئری مقایسه تعداد یوزر ها بر اساس استان یا state :

```
# count users base on state >>>

result = my_collection.aggregate(
    [
        {
            "$group": {
                "_id": "$location.state",
                "users": {
                    "$count": {}
                }
            }
        }
    ]
)
print("\n<<<This is result of counting users base on their state>>>\n")
for i in result:
    print(i)
```

بخشی از نتیجه :

```
<<<This is result of counting users base on their state>>>

{'_id': 'گیلان', 'users': 137}
{'_id': 'آذربایجان شرقی', 'users': 160}
{'_id': 'سمنان', 'users': 179}
{'_id': 'تهران', 'users': 166}
{'_id': 'کرمان', 'users': 161}
{'_id': 'مرکزی', 'users': 173}
{'_id': 'خوزستان', 'users': 164}
{'_id': 'سیستان و بلوچستان', 'users': 154}
{'_id': 'گلستان', 'users': 168}
{'_id': 'کردستان', 'users': 175}
{'_id': 'کرمانشاه', 'users': 174}
{'_id': 'اردبیل', 'users': 153}
{'_id': 'آذربایجان غربی', 'users': 160}
{'_id': 'همدان', 'users': 183}
{'_id': 'ایلام', 'users': 150}
{'_id': 'اصفهان', 'users': 156}
{'_id': 'هرمزگان', 'users': 163}
{'_id': 'مازندران', 'users': 164}
```

کوئری دسته بندی کردن بر اساس سن :

```
# This is categorizing  >>>
result = my_collection.aggregate([
    {
        "$facet": {
            "categorized_old": [
                {"$match": {"dob.age": {"$gt": 40}}},
                {"$group": {"_id": "$dob.age"}},
                {"$project": {"categorize": "old"}}
            ],
            "categorize_middle_age": [
                {"$match": {"dob.age": {"$gt": 16, "$lt": 40}}},
                {"$group": {"_id": "$dob.age"}},
                {"$project": {"categorize": "middle_age"}}
            ],

            "categorize_youth": [
                {"$match": {"dob.age": {"$lt": 25}}},
                {"$group": {"_id": "$dob.age"}},
                {"$project": {"categorize": "youth"}}
            ]
```

نتیجه این کوئری یک دیکشنری با value های لیست خواهد بود . برای دسترسی به کلید ها و ایتم ها ابتده ان را لیست میکنیم که یک عضو دارد . (نوع اصلی result cursor)

توجه : گروه youth افراد زیر 25 سال درنظر گرفته شد

```
result = list(result)

print("\nold categorize\n")
old = result[0]["categorized_old"]
for i in old:
    print(i)

print("\nyouth categorize\n")
youth = result[0]["categorize_youth"]
for i in youth:
    print(i)

print("\ncategorize_middle_age\n")
middle_age = result[0]["categorize_middle_age"]
for i in middle_age:
    print(i)
```

بخشی از نتیجه :

```
youth categorize

{'_id': 24, 'categorize': 'youth'}
{'_id': 23, 'categorize': 'youth'}

categorize_middle_age

{'_id': 39, 'categorize': 'middle_age'}
{'_id': 37, 'categorize': 'middle_age'}
{'_id': 33, 'categorize': 'middle_age'}
{'_id': 35, 'categorize': 'middle_age'}
{'_id': 31, 'categorize': 'middle_age'}
{'_id': 24, 'categorize': 'middle_age'}
{'_id': 29, 'categorize': 'middle_age'}
{'_id': 38, 'categorize': 'middle_age'}
{'_id': 34, 'categorize': 'middle_age'}
{'_id': 23, 'categorize': 'middle_age'}
{'_id': 30, 'categorize': 'middle_age'}
```

کوئری مقایسه میانگین تهران و بقیه شهر ها :

```
# compare avg age of users' from tehran and other cities >>>

result_1 = my_collection.aggregate(
    [
        {"$group": {
            "_id": "$location.city",
            "avg_age": {"$avg": {"$sum": "$dob.age"}},
            "counter": {"$count": {}}
        }},
        {"$match": {"_id": "تهران"}}
    ]
)

result_2 = my_collection.aggregate(
    [
        {"$match": {"location.city": {"$not": {'$regex': "تهران"}}}},
        {"$group": {
            "_id": "other_cities",
            "avg_age": {"$avg": {"$sum": "$dob.age"}},
            "counter": {"$count": {}}
        }}
    ]
```

نتیجه :

```
<<<This is result of comparing Tehran and other cities >>>

{'_id': 'تهران', 'avg_age': 46.36842105263158, 'counter': 95}
{'_id': 'other_cities', 'avg_age': 50.18246687054027, 'counter': 4905}

Process finished with exit code 0
```