

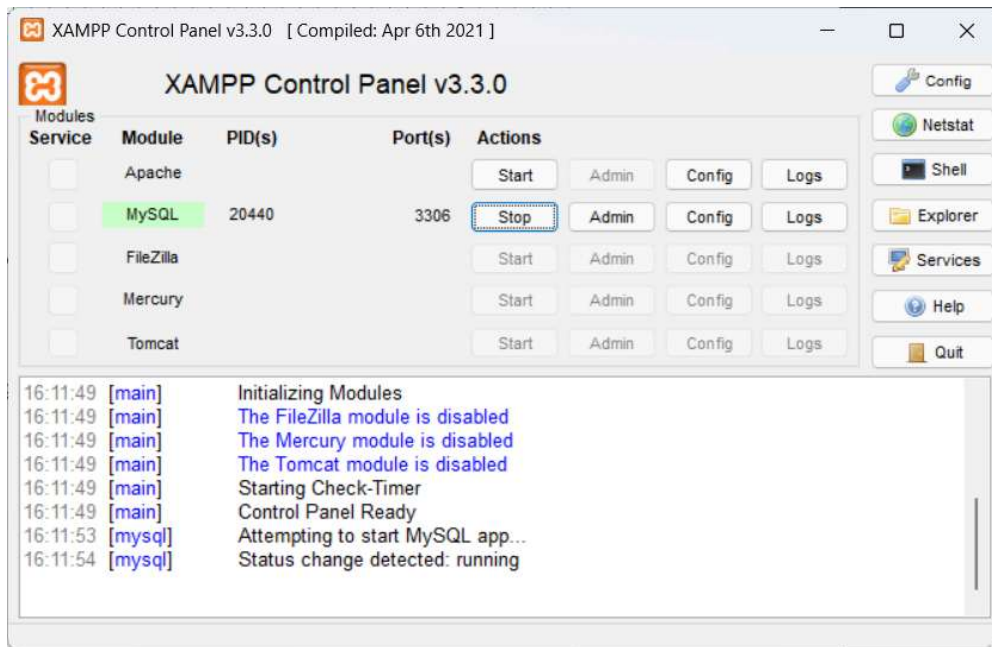
My github repository

https://github.com/tkeatkaew/data_engineer_training_psu

1) หน้าจอจาก Xampp, Compass แสดงว่าข้อมูลอยู่ใน MongoDB แล้ว 3 คะแนน

MySQL

Start MySQL Server.



Assignment

1. จงเขียนคำสั่งเพื่อสร้างฐานข้อมูลใหม่ชื่อ topic
2. จงเขียนคำสั่งเพื่อเรียกใช้ฐานข้อมูล topic แล้วสร้างตารางใหม่ชื่อ student ที่เก็บข้อมูล id ชนิด int และ name ชนิด varchar(50)
3. จงเขียนคำสั่งเพื่อเพิ่มข้อมูลจำนวนหนึ่งระเบียบลงในตาราง customer ของฐานข้อมูล lab ด้วยข้อมูล id = 9 ตามด้วย name เป็นชื่อภาษาอังกฤษของผู้ทดลอง (เช่น Kamon) และ credit เป็น 9999.9
4. จงเขียนคำสั่งเพื่อแสดงข้อมูลทั้งหมดของตาราง customer ของฐานข้อมูล lab
5. จงเขียนคำสั่งเพื่อเปลี่ยนแปลงข้อมูลหนึ่งระเบียบของตาราง customer ของฐานข้อมูล lab ที่มีค่า id = 2 โดยเปลี่ยนแปลงให้ค่าของ name เป็นชื่อภาษาอังกฤษของผู้ทดลอง (เช่น Kamon)

Connect MySQL using the command line.

Display all databases.

```
PowerShell 7.5.1
PS C:\Users\se-rumtl> d:
PS D:\> cd D:\DataEngineerTraining\day3\data_engineer_training_psu\day3
PS D:\DataEngineerTraining\day3\data_engineer_training_psu\day3> mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 8
Server version: 10.4.32-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| lab |
| mysql |
| performance_schema |
| phpmyadmin |
| test |
| topic |
+-----+
7 rows in set (0.048 sec)

MariaDB [(none)]>
```

แสดง lab DB ที่สร้างขึ้นมาและผลของการทำตาม slide: C3-MySQL.pdf

```
MariaDB [(none)]> use lab;
Database changed
MariaDB [lab]> show tables;
+-----+
| Tables_in_lab |
+-----+
| customer |
+-----+
1 row in set (0.001 sec)

MariaDB [lab]> select * from customer;
+----+-----+-----+
| id | name      | credit |
+----+-----+-----+
| 1  | First Count | 10000.15 |
| 3  | Lek      | 50000.35 |
| 9  | Thanit    | 9999.9 |
+----+-----+-----+
3 rows in set (0.053 sec)

MariaDB [lab]>
```

แสดง topic DB

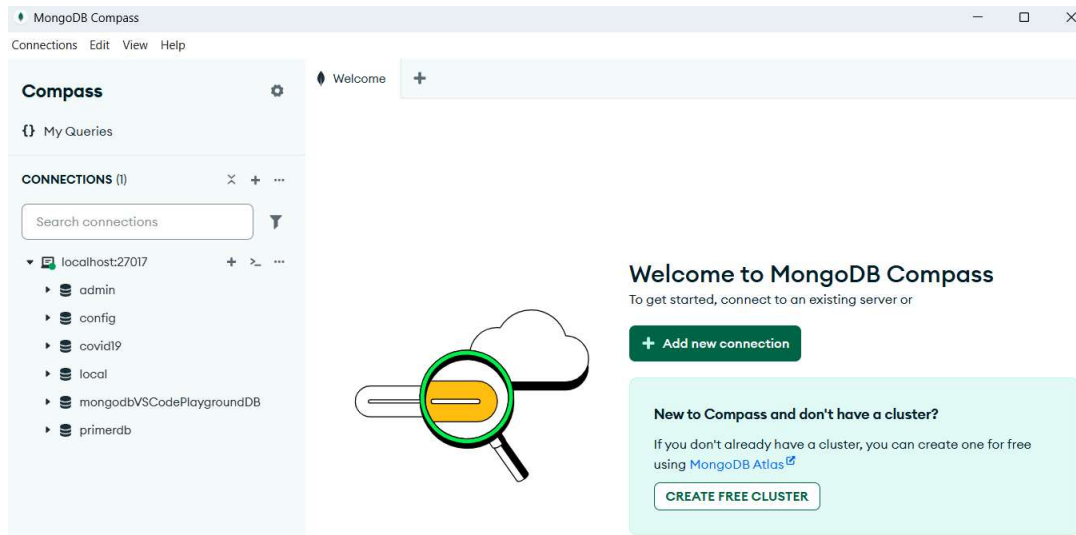
```
MariaDB [lab]> use topic;
Database changed
MariaDB [topic]> show tables;
+-----+
| Tables_in_topic |
+-----+
| student |
+-----+
1 row in set (0.001 sec)

MariaDB [topic]> select * from student;
+----+-----+-----+
| id | name      | credit |
+----+-----+-----+
| 9  | Thanit    | 9999.9 |
+----+-----+-----+
1 row in set (0.004 sec)

MariaDB [topic]>
```

MongoDB

Connect to MongoDB and show all existing databases and my import data.



Connect MongoDB using the command line.

Display all databases.

```
mongosh mongodb://127.0.0.1
primerdb> show databases
admin                40.00 KiB
config               72.00 KiB
covid19              8.00 KiB
local                72.00 KiB
mongodbVSCodePlaygroundDB 40.00 KiB
primerdb             4.07 MiB
primerdb>
```

From my github repository, these are my related command lines which used for this lab,

```
mysql_works.md U  mongodb_works.md M  mongodb.md M  test-mysql-connector.ipynb
day3 > # At powershell
5
6 # mongosh basic command
7 show databases
8 use mongodbVSCodePlaygroundDB
9 show collections
10
11 // Insert a few documents into the sales collection.
12 db.getCollection('sales').insertMany([
13   { 'item': 'abc', 'price': 10, 'quantity': 2, 'date': new Date('2025-06-18T08:00:00Z') },
14   { 'item': 'jkl', 'price': 20, 'quantity': 1, 'date': new Date('2025-06-18T09:00:00Z') },
15 ]);
16
17 db.sales.find()
18 db.sales.find({'item':'abc'})
```

```
switched to db mongodbVSCodePlaygroundDB
mongodbVSCodePlaygroundDB> db.getCollection('sales').insertMany([
...   { 'item': 'abc', 'price': 10, 'quantity': 2, 'date': new Date('2025-06-18T08:00:00Z') },
...   { 'item': 'jkl', 'price': 20, 'quantity': 1, 'date': new Date('2025-06-18T09:00:00Z') },
... ]);
...
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6852a759a0fb583cbd50eb67'),
    '1': ObjectId('6852a759a0fb583cbd50eb68')
  }
}
```

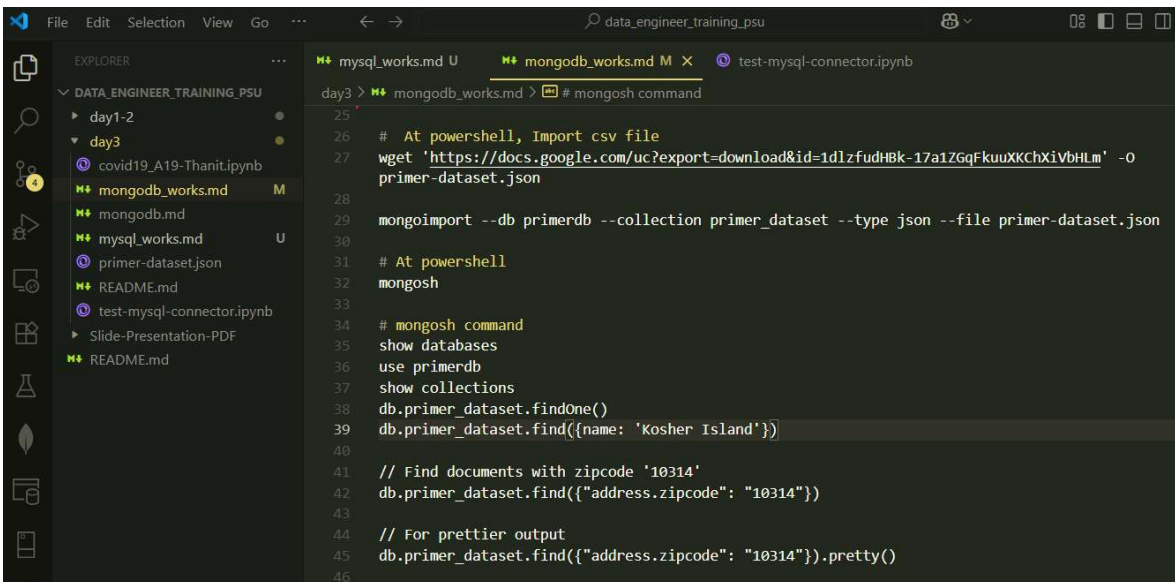
```
primerdb> use mongodbVSCodePlaygroundDB
switched to db mongodbVSCodePlaygroundDB
mongodbVSCodePlaygroundDB> db.getCollection('sales').insertMany([
...   { 'item': 'abc', 'price': 10, 'quantity': 2, 'date': new Date('2025-06-18T08:00:00Z') },
...   { 'item': 'jkl', 'price': 20, 'quantity': 1, 'date': new Date('2025-06-18T09:00:00Z') },
... ]);
...
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('6852a759a0fb583cbd50eb67'),
    '1': ObjectId('6852a759a0fb583cbd50eb68')
  }
}
```

Then, test import with csv file

```
wget
'https://docs.google.com/uc?export=download&id=1dlzfudHBk-17a1ZGqFkuuXKChXiVbHLm' -O primer-dataset.json

mongoimport --db primerdb --collection primer_dataset --type json --file
primer-dataset.json
```

From my github repository, these are my related command lines which used for this lab,

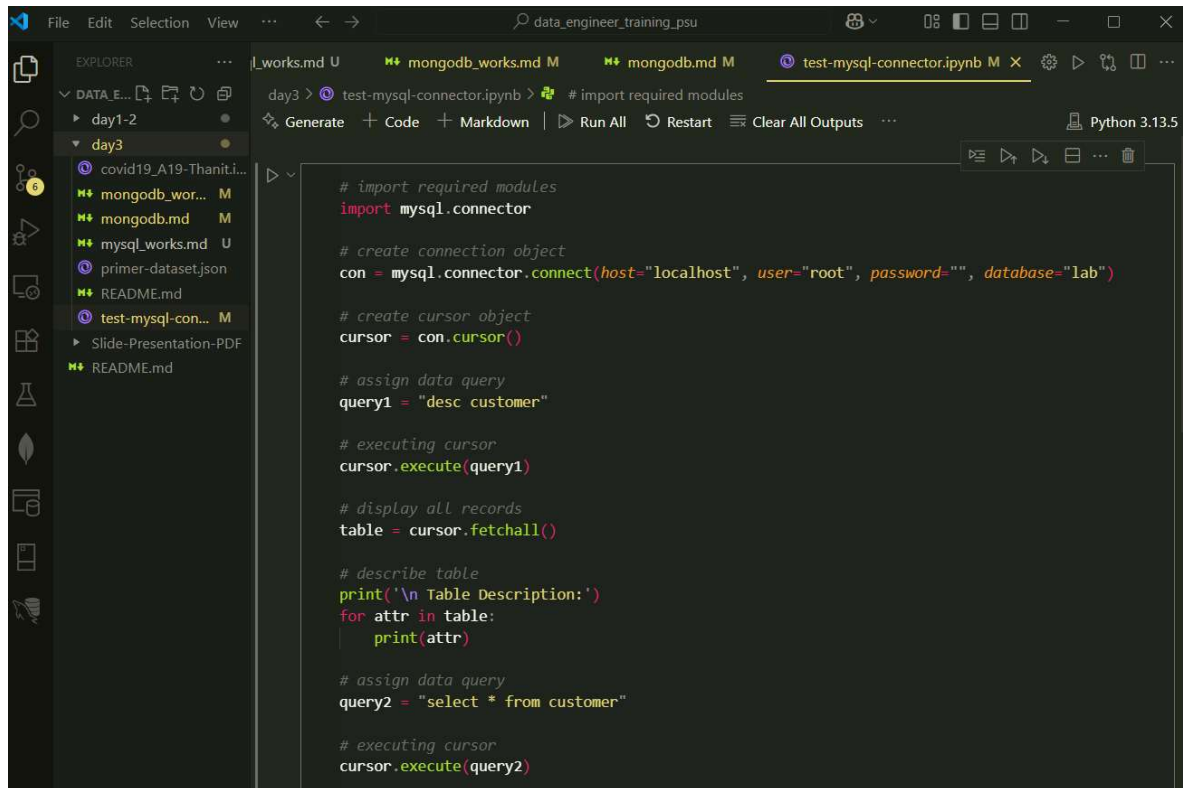


```
day3 > # mongosh command
25
26 # At powershell, Import csv file
27 wget 'https://docs.google.com/uc?export=download&id=1dlzfudHBk-17a1ZGqFkuuXKChXiVbHLm' -O
primer-dataset.json
28
29 mongoimport --db primerdb --collection primer_dataset --type json --file primer-dataset.json
30
31 # At powershell
32 mongosh
33
34 # mongosh command
35 show databases
36 use primerdb
37 show collections
38 db.primer_dataset.findOne()
39 db.primer_dataset.find({name: 'Kosher Island'})
40
41 // Find documents with zipcode '10314'
42 db.primer_dataset.find({"address.zipcode": "10314"})
43
44 // For prettier output
45 db.primer_dataset.find({"address.zipcode": "10314"}).pretty()
46
```

```
primerdb> show collections
primer_dataset
primerdb> db.primer_dataset.findOne()
{
  _id: ObjectId('685277cf01576891627ab8cb'),
  address: {
    building: '2206',
    coord: [ -74.1377286, 40.6119572 ],
    street: 'Victory Boulevard',
    zipcode: '10314'
  },
  borough: 'Staten Island',
  cuisine: 'Jewish/Kosher',
  grades: [
    { date: ISODate('2014-10-06T00:00:00.000Z'), grade: 'A', score: 9 },
    {
      date: ISODate('2014-05-20T00:00:00.000Z'),
      grade: 'A',
      score: 12
    },
    {
      date: ISODate('2013-04-04T00:00:00.000Z'),
      grade: 'A',
      score: 12
    },
    { date: ISODate('2012-01-24T00:00:00.000Z'), grade: 'A', score: 9 }
  ],
  name: 'Kosher Island',
  restaurant_id: '40356442'
}
primerdb>
```

2) หน้าจอดึงข้อมูลได้ด้วย sql-connector หรือ pymongo 4 คะแนน

Connect to MySQL using Python



```
# import required modules
import mysql.connector

# create connection object
con = mysql.connector.connect(host="localhost", user="root", password="", database="lab")

# create cursor object
cursor = con.cursor()

# assign data query
query1 = "desc customer"

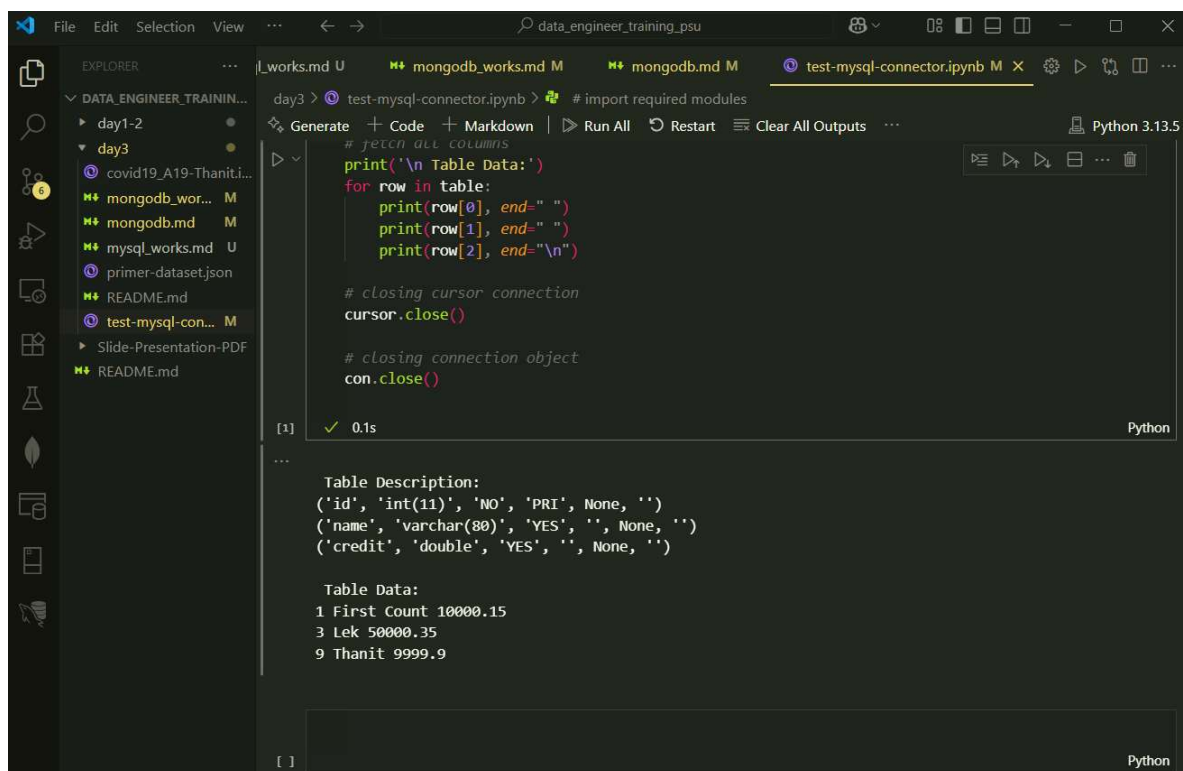
# executing cursor
cursor.execute(query1)

# display all records
table = cursor.fetchall()

# describe table
print('\n Table Description:')
for attr in table:
    print(attr)

# assign data query
query2 = "select * from customer"

# executing cursor
cursor.execute(query2)
```



```
# fetch all columns
print('\n Table Data:')
for row in table:
    print(row[0], end=" ")
    print(row[1], end=" ")
    print(row[2], end="\n")

# closing cursor connection
cursor.close()

# closing connection object
con.close()
```

[1] ✓ 0.1s Python

...

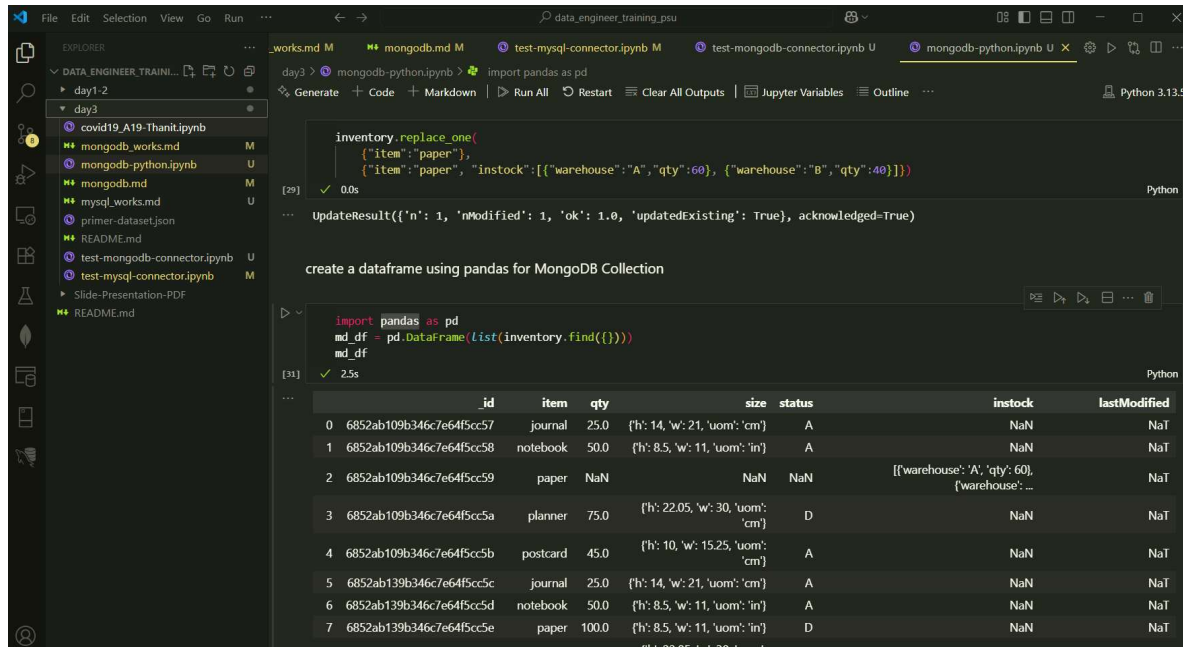
Table Description:
('id', 'int(11)', 'NO', 'PRI', None, '')
('name', 'varchar(80)', 'YES', '', None, '')
('credit', 'double', 'YES', '', None, '')

Table Data:
1 First Count 10000.15
3 Lek 50000.35
9 Thanit 9999.9

[] Python

Connect to MongoDB using Python

Final screen of original `mongodb-python.ipynb`, when connect python to MongoDB and work with pandas



The screenshot shows a Jupyter Notebook titled 'mongodb-python.ipynb' in a VS Code environment. The notebook has two cells. The first cell contains a MongoDB update operation:

```
inventory.replace_one(
    {"item": "paper"},
    {"item": "paper", "instock": [{"warehouse": "A", "qty": 60}, {"warehouse": "B", "qty": 40}]}
)
```

The output of this cell is an `UpdateResult` object: `UpdateResult({'n': 1, 'modified': 1, 'ok': 1.0, 'updatedExisting': True}, acknowledged=True)`.

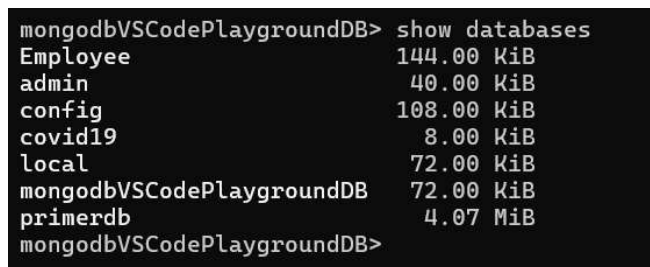
The second cell contains a prompt: 'create a dataframe using pandas for MongoDB Collection'. The code in this cell is:

```
import pandas as pd
md_df = pd.DataFrame(List(inventory.find({})))
md_df
```

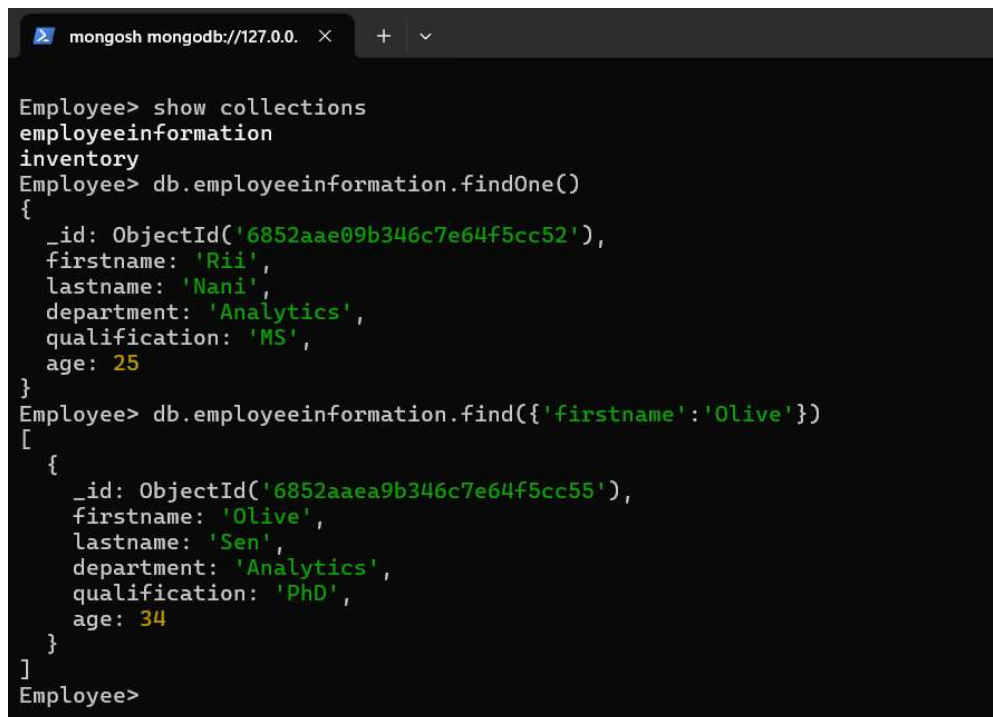
The output of this cell is a pandas DataFrame with 8 rows and 8 columns: `id`, `item`, `qty`, `size`, `status`, `instock`, and `lastModified`. The data is as follows:

	id	item	qty	size	status	instock	lastModified
0	6852ab109b346c7e64f5cc57	journal	25.0	{'h': 14, 'w': 21, 'uom': 'cm'}	A	NaN	NaT
1	6852ab109b346c7e64f5cc58	notebook	50.0	{'h': 8.5, 'w': 11, 'uom': 'in'}	A	NaN	NaT
2	6852ab109b346c7e64f5cc59	paper	NaN	NaN	NaN	[{'warehouse': 'A', 'qty': 60}, {'warehouse': 'B', 'qty': 40}]	NaT
3	6852ab109b346c7e64f5cc5a	planner	75.0	{'h': 22.05, 'w': 30, 'uom': 'cm'}	D	NaN	NaT
4	6852ab109b346c7e64f5cc5b	postcard	45.0	{'h': 10, 'w': 15.25, 'uom': 'cm'}	A	NaN	NaT
5	6852ab139b346c7e64f5cc5c	journal	25.0	{'h': 14, 'w': 21, 'uom': 'cm'}	A	NaN	NaT
6	6852ab139b346c7e64f5cc5d	notebook	50.0	{'h': 8.5, 'w': 11, 'uom': 'in'}	A	NaN	NaT
7	6852ab139b346c7e64f5cc5e	paper	100.0	{'h': 8.5, 'w': 11, 'uom': 'in'}	D	NaN	NaT

Some results of its data.



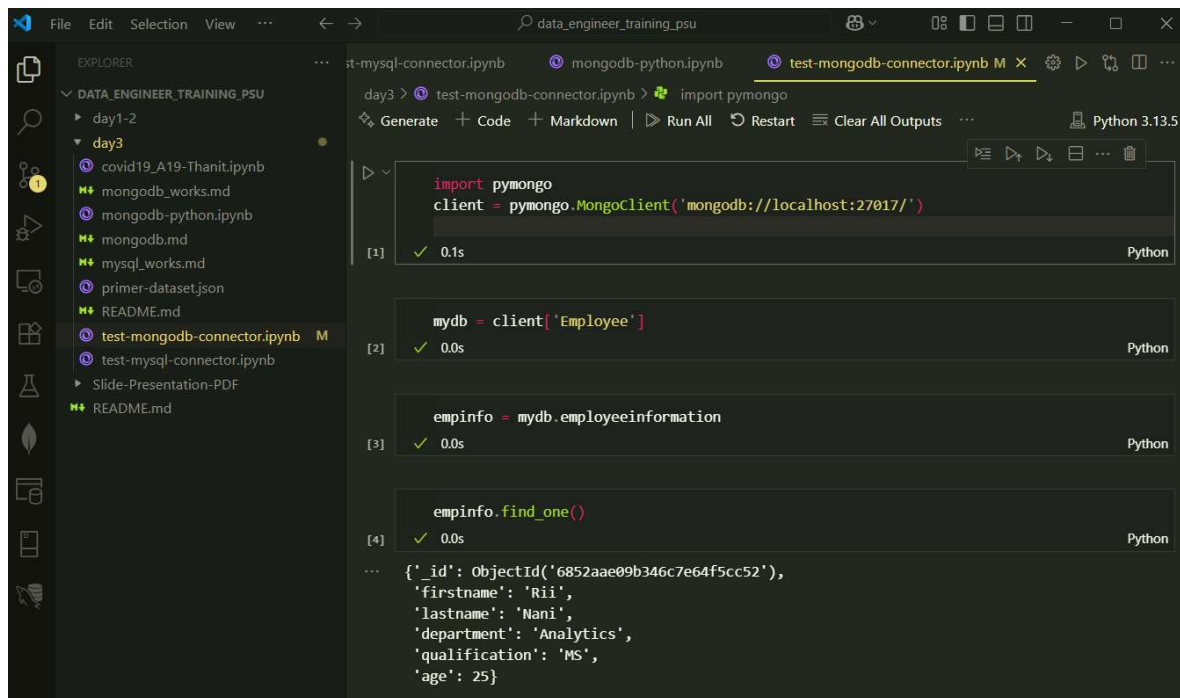
```
mongodbVSCodePlaygroundDB> show databases
Employee      144.00 KiB
admin         40.00 KiB
config        108.00 KiB
covid19        8.00 KiB
local         72.00 KiB
mongodbVSCodePlaygroundDB 72.00 KiB
primerdb       4.07 MiB
mongodbVSCodePlaygroundDB>
```



```
Employee> show collections
employeeinformation
inventory
Employee> db.employeeinformation.findOne()
{
  _id: ObjectId('6852aae09b346c7e64f5cc52'),
  firstname: 'Rii',
  lastname: 'Nani',
  department: 'Analytics',
  qualification: 'MS',
  age: 25
}
Employee> db.employeeinformation.find({'firstname': 'Olive'})
[
  {
    _id: ObjectId('6852aaea9b346c7e64f5cc55'),
    firstname: 'Olive',
    lastname: 'Sen',
    department: 'Analytics',
    qualification: 'PhD',
    age: 34
  }
]
Employee>
```

3) หน้าจอแสดงการประมวลผลด้วย Pandas เช่น แสดงข้อมูลเฉพาะจังหวัด หรือวันที่ที่ต้องการ 3 คะแนน
My Code here:

https://github.com/tkeatkaew/data_engineer_training_psu/blob/main/day3/test-mongodb-connector.ipynb



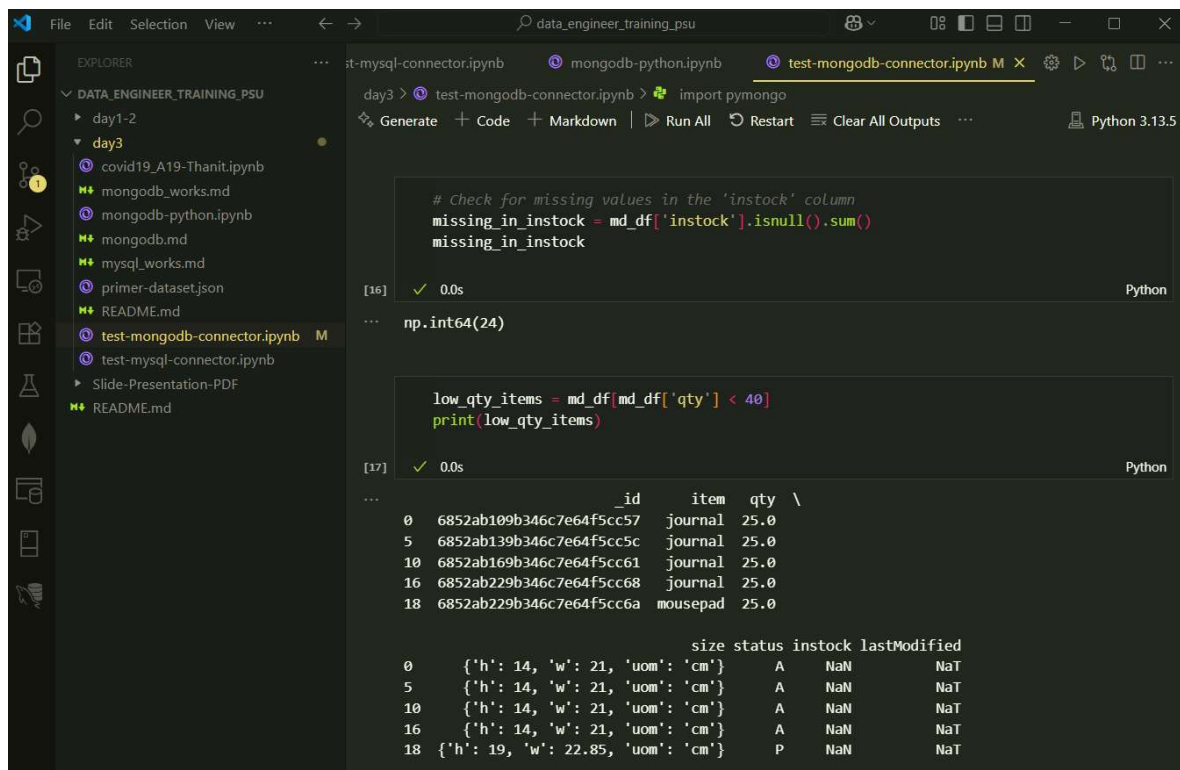
```
import pymongo
client = pymongo.MongoClient('mongodb://localhost:27017/')

mydb = client['Employee']

empinfo = mydb.employeeinformation

empinfo.find_one()

{'_id': ObjectId('6852aae09b346c7e64f5cc52'),
 'firstname': 'Rii',
 'lastname': 'Nani',
 'department': 'Analytics',
 'qualification': 'MS',
 'age': 25}
```



```
# Check for missing values in the 'instock' column
missing_in_instock = md_df['instock'].isnull().sum()
missing_in_instock

np.int64(24)

low_qty_items = md_df[md_df['qty'] < 40]
print(low_qty_items)

   _id      item  qty \
0  6852ab109b346c7e64f5cc57  journal  25.0
5  6852ab139b346c7e64f5cc5c  journal  25.0
10 6852ab169b346c7e64f5cc61  journal  25.0
16 6852ab229b346c7e64f5cc68  journal  25.0
18 6852ab229b346c7e64f5cc6a  mousepad  25.0

   size status instock lastModified
0  {'h': 14, 'w': 21, 'uom': 'cm'}    A   NaN         NaT
5  {'h': 14, 'w': 21, 'uom': 'cm'}    A   NaN         NaT
10 {'h': 14, 'w': 21, 'uom': 'cm'}    A   NaN         NaT
16 {'h': 14, 'w': 21, 'uom': 'cm'}    A   NaN         NaT
18 {'h': 19, 'w': 22.85, 'uom': 'cm'}  P   NaN         NaT
```

Thank you.