Big Data with

Lab#1

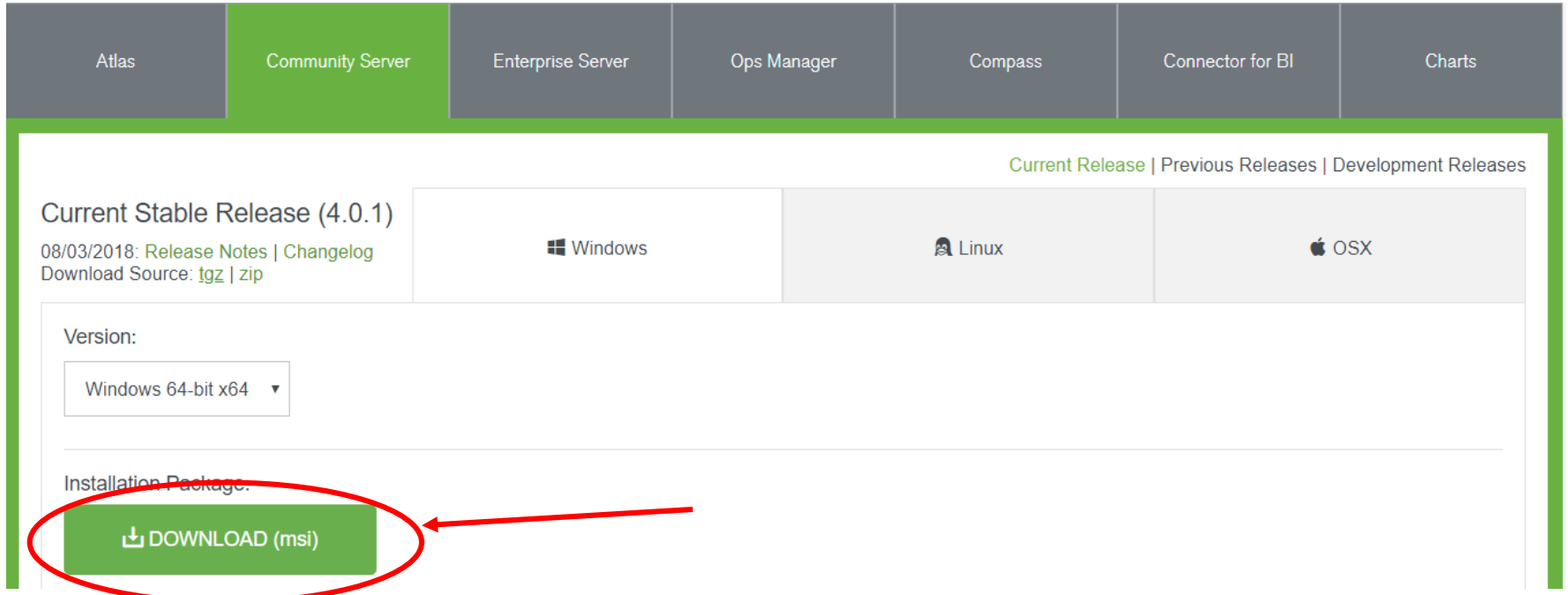
mongoDB

SIIT 25th ANNIVERSARY

Installation (MongoDB)

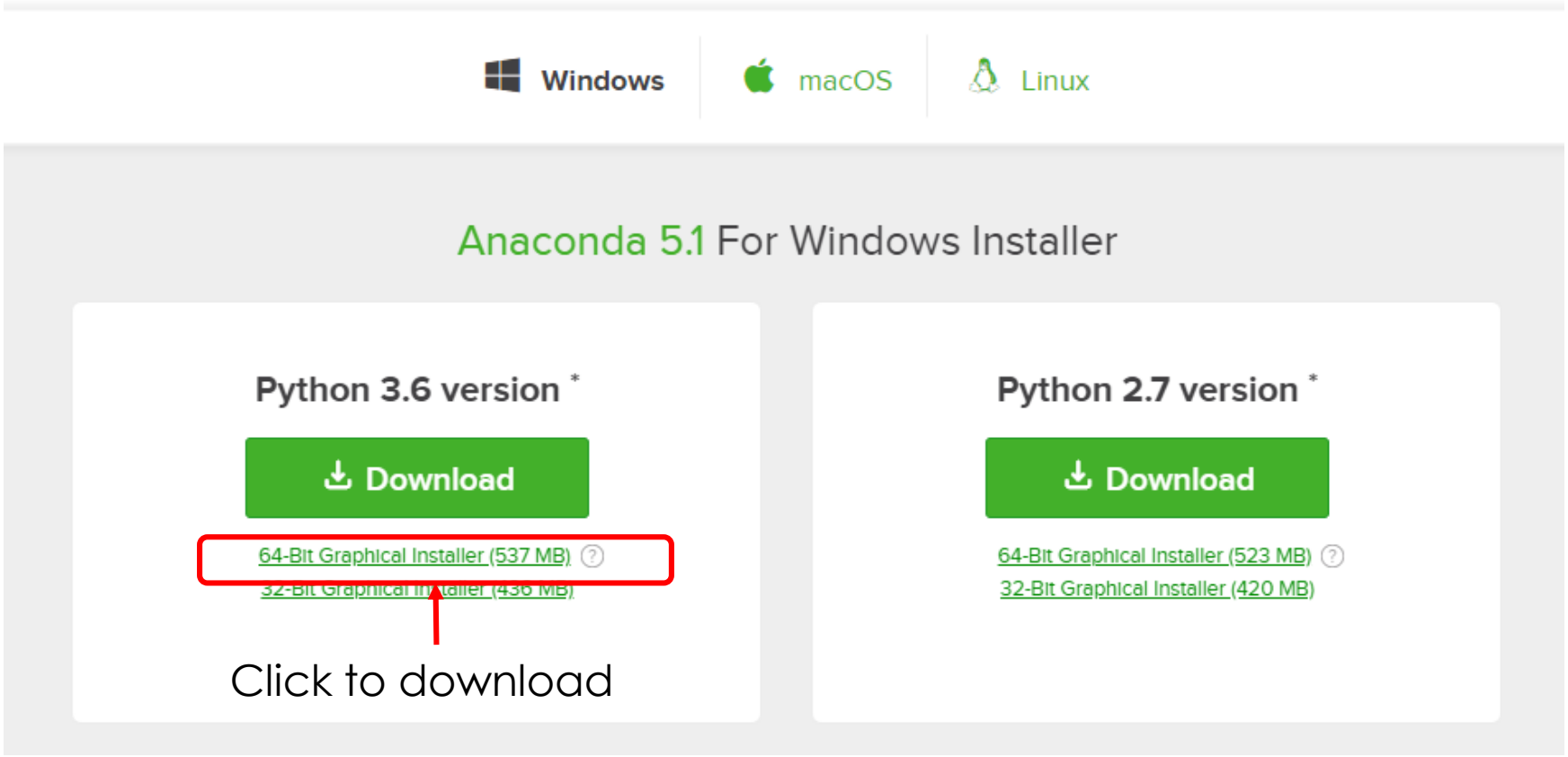
- MongoDB Current Stable Release (4.0.1)

https://www.mongodb.com/download-center#community



Installation (Anaconda)

- https://www.anaconda.com/download/



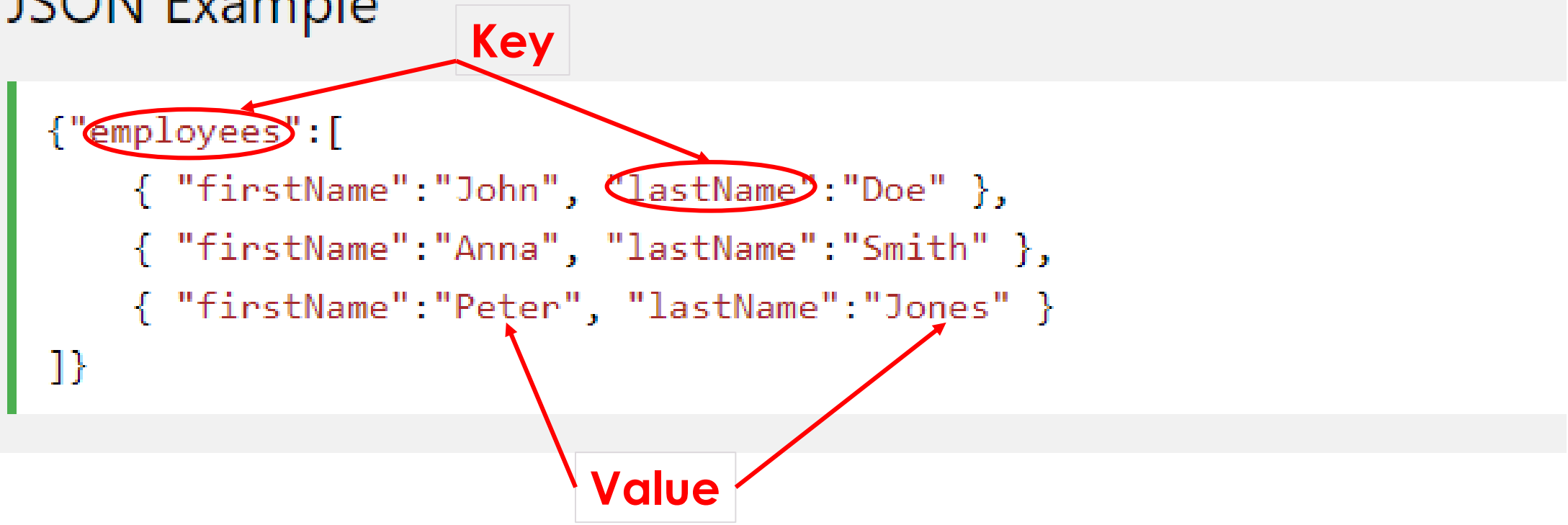
JSON: JavaScript Object Notation.

- JSON is a syntax for storing and exchanging data.
- JSON is text, written with JavaScript object notation.

https://www.w3schools.com/js/js_json_intro.asp

JSON: JavaScript Object Notation.

JSON Example

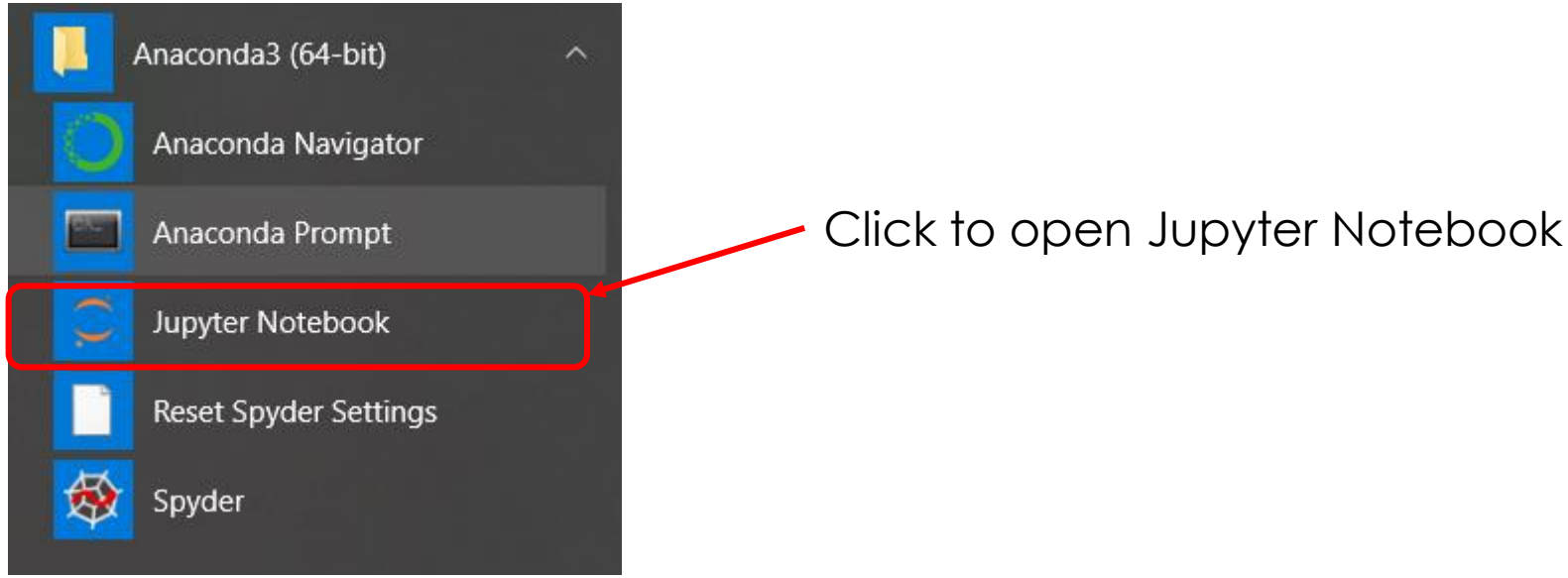


https://www.w3schools.com/js/js_json_xml.asp

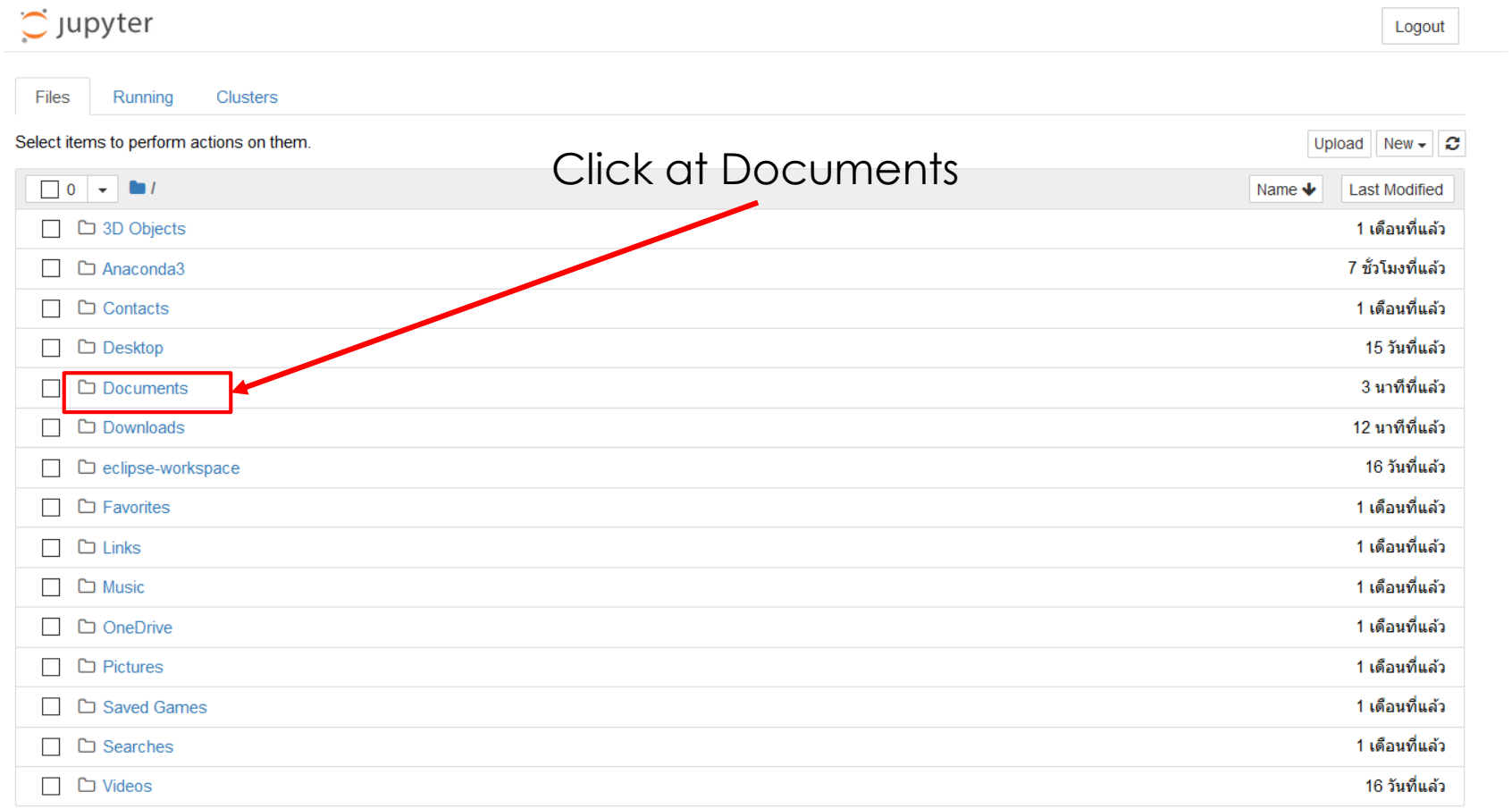
Jupyter Notebook

- Python Editor
- An open-source web application that allows you to create and share documents that contain live code

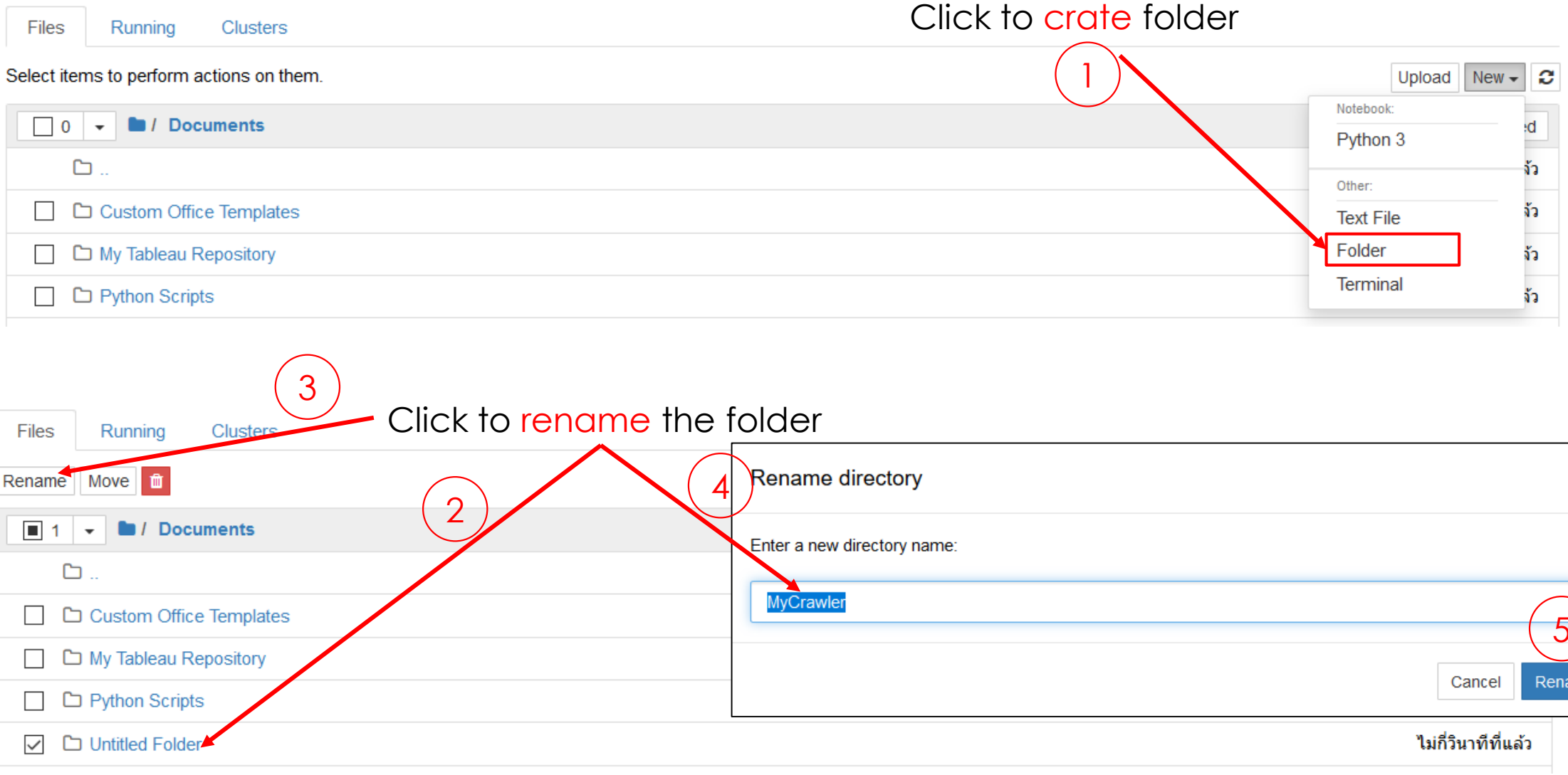
Start menu->



Jupyter Notebook



Jupyter Notebook

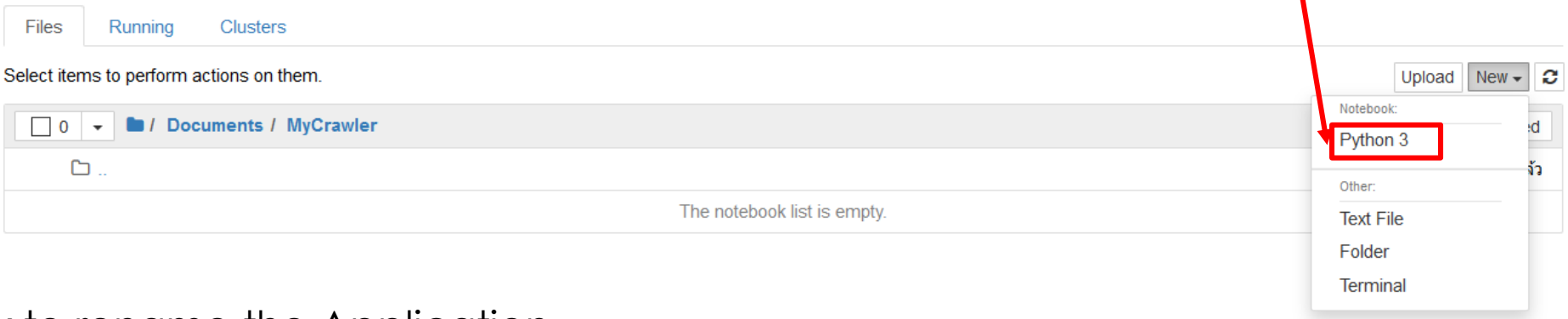


Jupyter Notebook

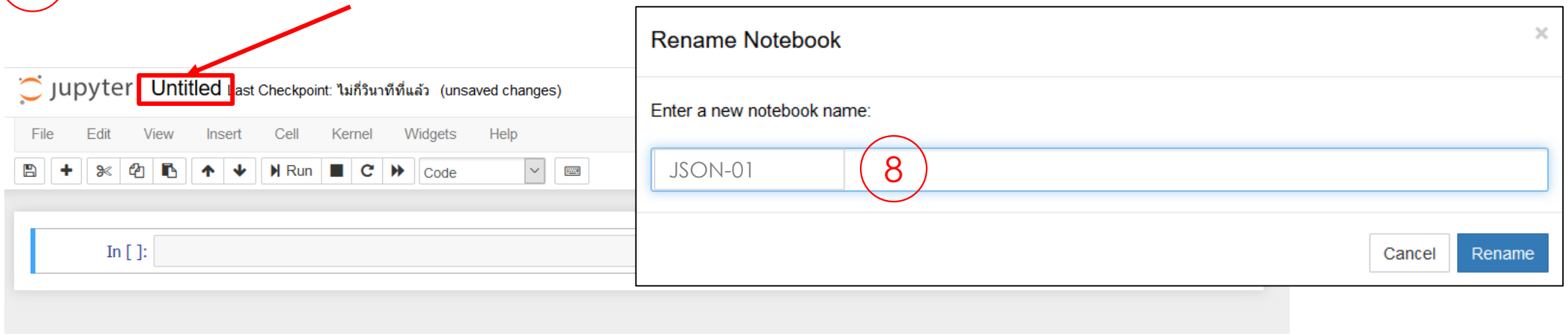
- Click into MyCrawler Folder

Click to crate Python Application

6



7 Click to rename the Application



Work with JSON in Python

In [ ]:
```
data = []
com1 = {'name': 'State Salty Inc', 'cuisine': 'American', 'rating': 2}
com2 = {'name': 'Animal Animal Company', 'rating': 5, 'cuisine': 'Bar Food'}
data.append(com1)
data.append(com2)
```

In [ ]:
```
data[0]['name']
```

In [ ]:
```
for i in range(len(data)):
    print(data[i])
```

In [ ]:
```

```

mongoDB

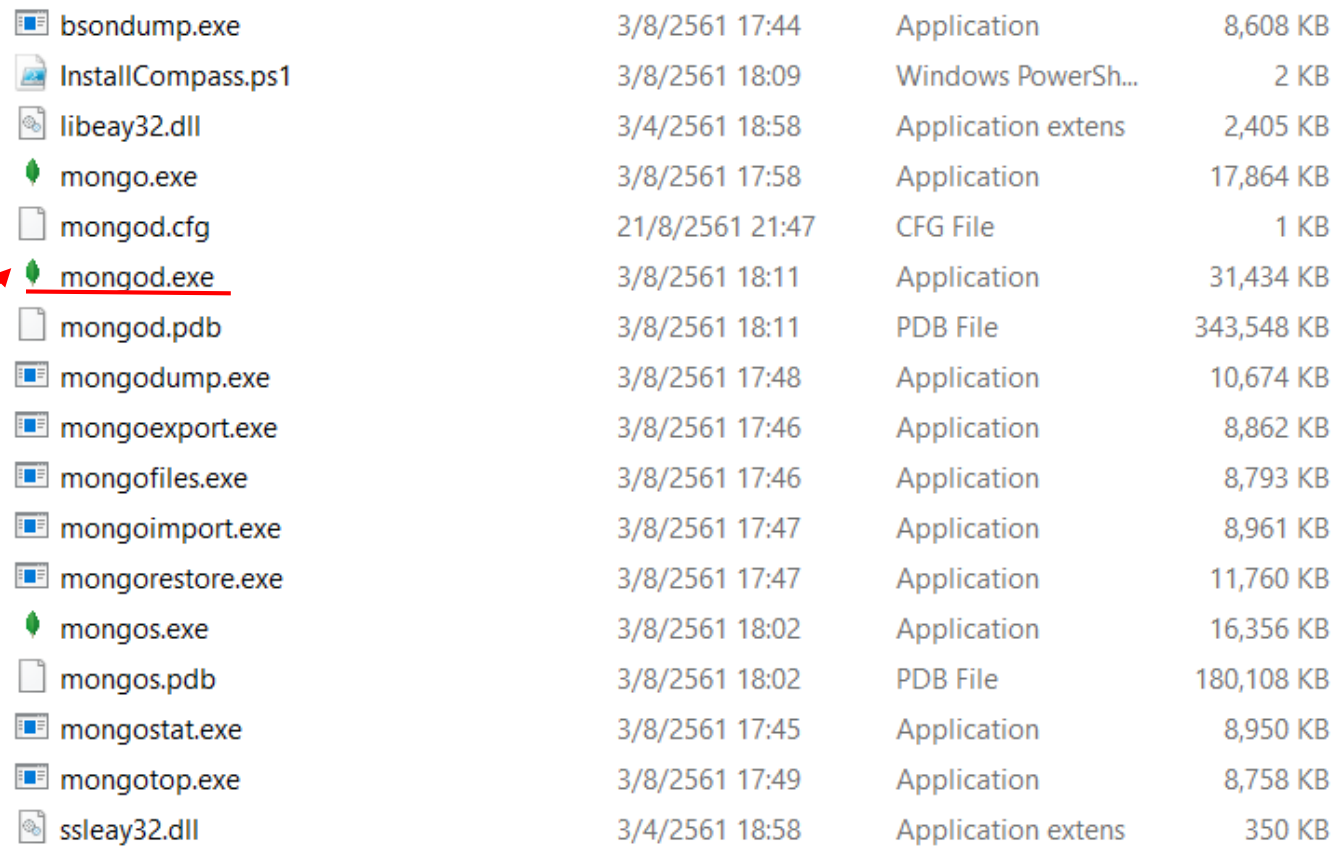
python™

Start MongoDB Server

MongoDB\bin

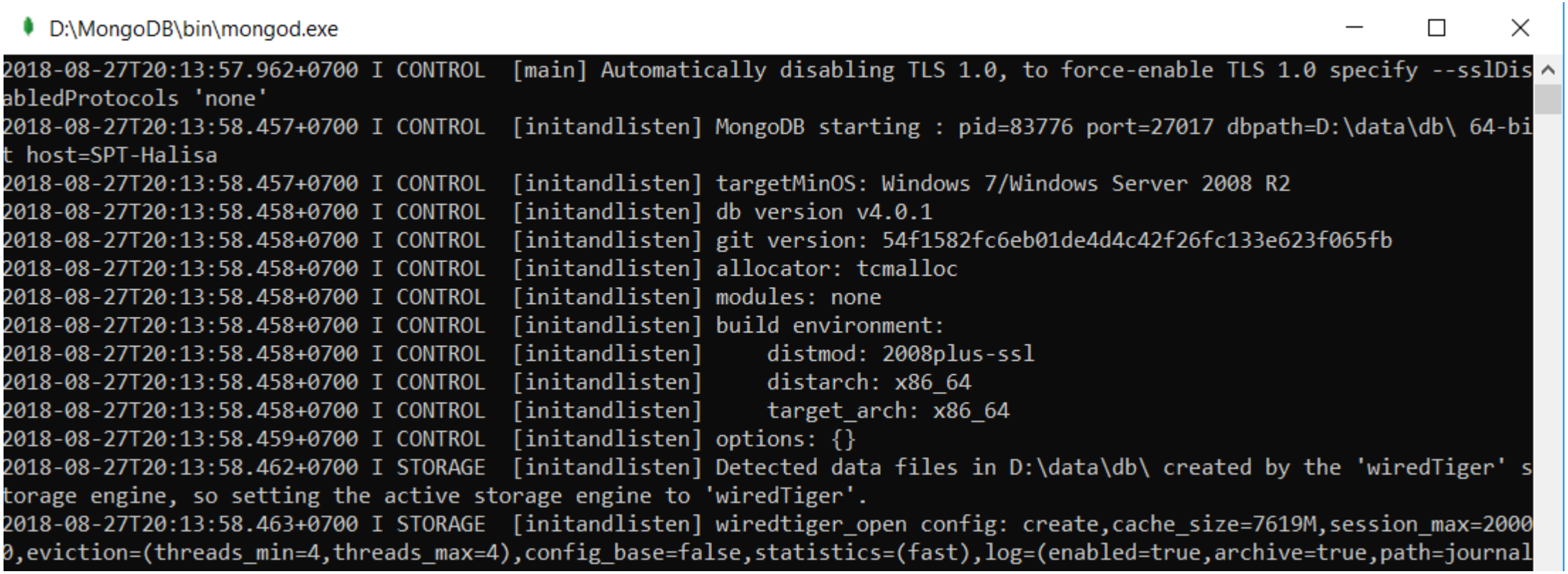
Double Click to start server

1



bsondump.exe	3/8/2561 17:44	Application	8,608 KB
InstallCompass.ps1	3/8/2561 18:09	Windows PowerSh...	2 KB
libeay32.dll	3/4/2561 18:58	Application extens	2,405 KB
mongo.exe	3/8/2561 17:58	Application	17,864 KB
mongod.cfg	21/8/2561 21:47	CFG File	1 KB
mongod.exe	3/8/2561 18:11	Application	31,434 KB
mongod.pdb	3/8/2561 18:11	PDB File	343,548 KB
mongodump.exe	3/8/2561 17:48	Application	10,674 KB
mongoexport.exe	3/8/2561 17:46	Application	8,862 KB
mongofiles.exe	3/8/2561 17:46	Application	8,793 KB
mongoimport.exe	3/8/2561 17:47	Application	8,961 KB
mongorestore.exe	3/8/2561 17:47	Application	11,760 KB
mongos.exe	3/8/2561 18:02	Application	16,356 KB
mongos.pdb	3/8/2561 18:02	PDB File	180,108 KB
mongostat.exe	3/8/2561 17:45	Application	8,950 KB
mongotop.exe	3/8/2561 17:49	Application	8,758 KB
ssleay32.dll	3/4/2561 18:58	Application extens	350 KB

Start MongoDB Server



pyMongo, the driver for MongoDB

Press Shift+Enter to Run each program

Install driver

1

```
!pip install pymongo
```

```
Collecting pymongo
  Downloading https://files.pythonhosted.org/
o-3.7.1-cp35-cp35m-win_amd64.whl (311kB)
Installing collected packages: pymongo
Successfully installed pymongo-3.7.1
```

MongoDB: Create database

Press Shift+Enter to Run each program

```
from pymongo import MongoClient
```

Connect MongoDB server port 27017 @ localhost

```
client = MongoClient(port=27017)
```

Create Database named "business"

```
db=client.business
```

MongoDB: Insert

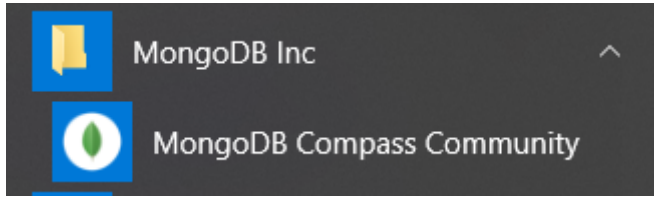
```
com1 = {'name': 'State Salty Inc', 'cuisine': 'American', 'rating': 2}
com2 = {'name': 'Animal Animal Company', 'rating': 5, 'cuisine': 'Bar Food'}
```

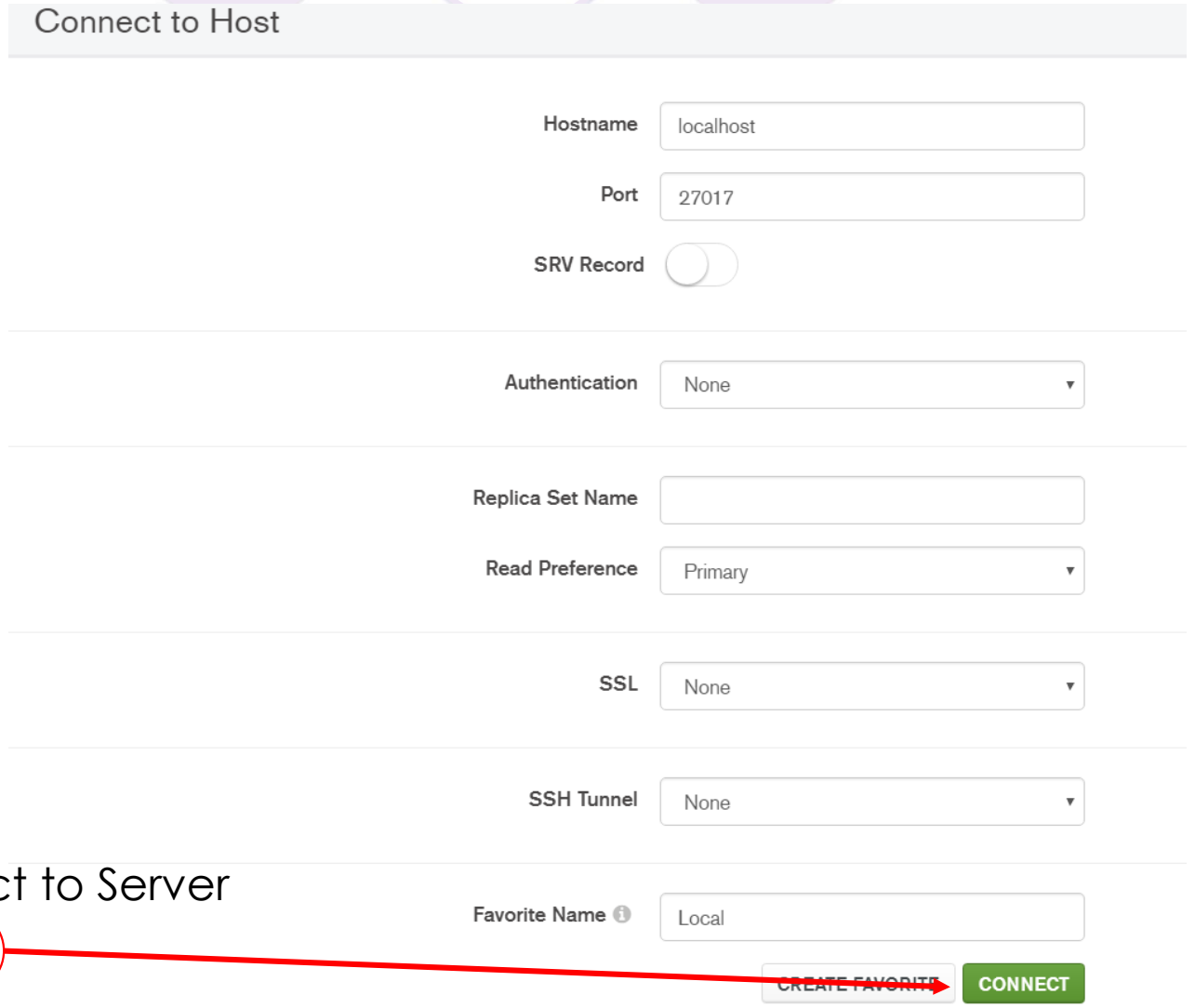
```
result=db.reviews.insert_one(com1)
print('Created {0}'.format(result.inserted_id))
result=db.reviews.insert_one(com2)
print('Created {0}'.format(result.inserted_id))
```

```
Created 5b8411f9bdc81d9851a4d0b8
Created 5b8411f9bdc81d9851a4d0b9
```

MongoDB Compass Community

- Start menu->



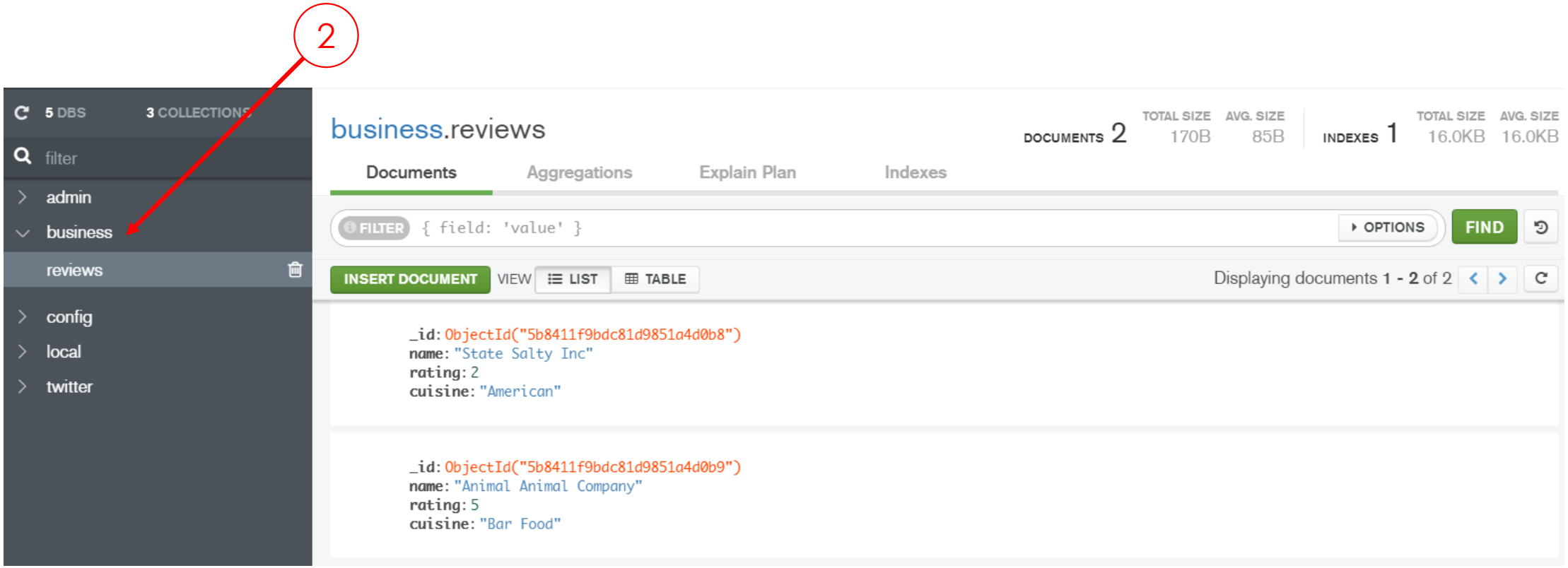


Connect to Server

1

MongoDB Compass Community

Click to Check inserted data



MongoDB: Query

Query Data

```
data = db.reviews.find()
for d in data:
    print(d)
```

```
{'name': 'State Salty Inc', 'rating': 2, '_id': ObjectId('5b8411f9bdc81d9851a4d0b8'), 'cuisine': 'American'}
{'name': 'Animal Animal Company', 'rating': 5, '_id': ObjectId('5b8411f9bdc81d9851a4d0b9'), 'cuisine': 'Bar Food'}
```

Query 1 record

```
one = db.reviews.find_one()
print(one)
```

```
{'name': 'State Salty Inc', 'rating': 2, '_id': ObjectId('5b8411f9bdc81d9851a4d0b8'), 'cuisine': 'American'}
```

Query Data with condition {'rating'=5}

```
fivestar = db.reviews.find_one({'rating': 5})
print(fivestar)
```

```
{'name': 'Animal Animal Company', 'rating': 5, '_id': ObjectId('5b8411f9bdc81d9851a4d0b9'), 'cuisine': 'Bar Food'}
```

MongoDB: Delete

Count Document in database

```
db.reviews.estimated_document_count()
```

```
2
```

Delete data in Database

```
res=db.reviews.delete_one({'rating': 5})
print(res.deleted_count)
```

```
1
```

```
data = db.reviews.find()
for d in data:
    print(d)
```

```
{'name': 'State Salty Inc', 'rating': 2, '_id': ObjectId('5b8411f9bdc81d9851a4d0b8'), 'cuisine': 'American'}
```

MongoDB: Update

Update data in Database

```
res=db.reviews.update_one({'rating': 2}, { '$set': {'rating': 5} })
print(res.raw_result)
```

```
{'n': 1, 'updatedExisting': True, 'ok': 1.0, 'nModified': 1}
```

```
data = db.reviews.find()
for d in data:
    print(d)
```

```
{'name': 'State Salty Inc', 'rating': 5, '_id': ObjectId('5b8411f9bdc81d9851a4d0b8'), 'cuisine': 'American'}
```

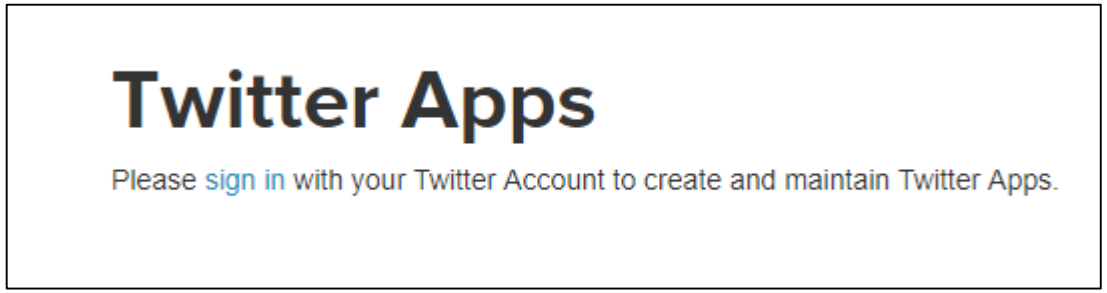
MongDB: more…

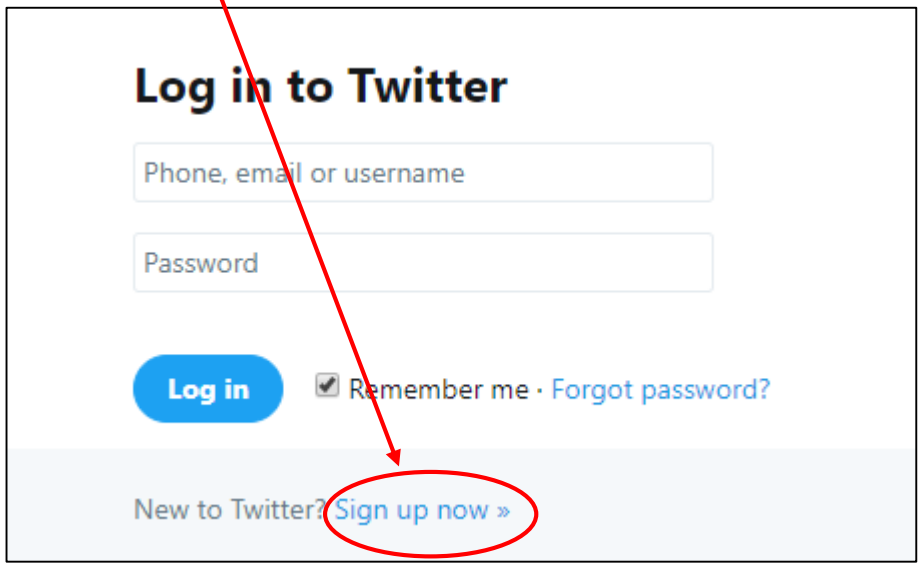
- Go to : https://github.com/9meo/SIIT-MongoDB-Lab1
- Try it!!!


twitter

Twitter API Registration

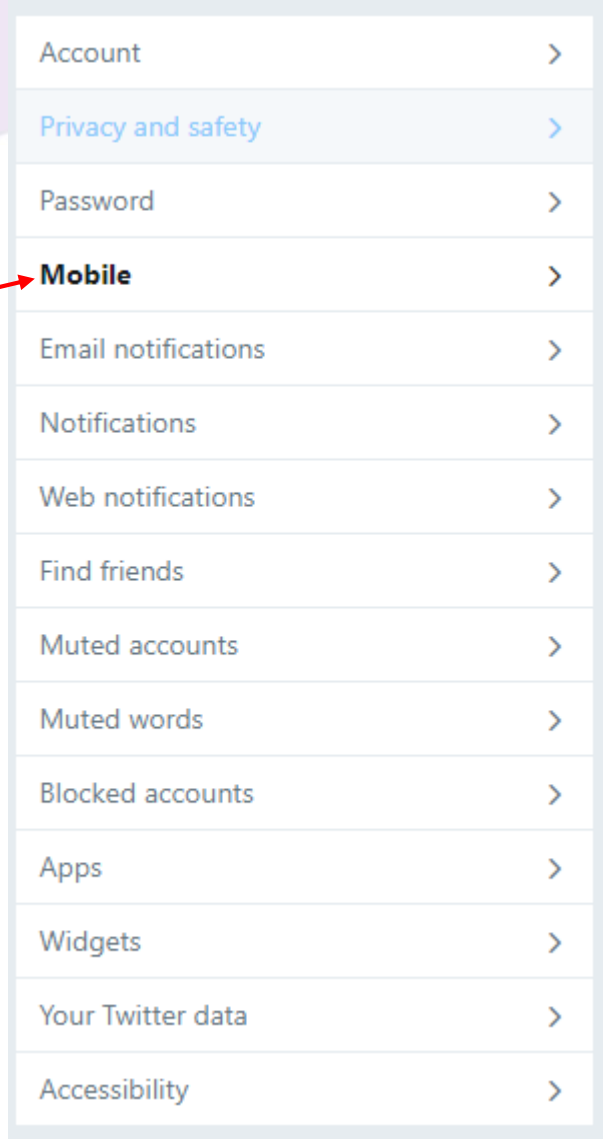
- Go to: https://apps.twitter.com/
- Login with twitter account or create new





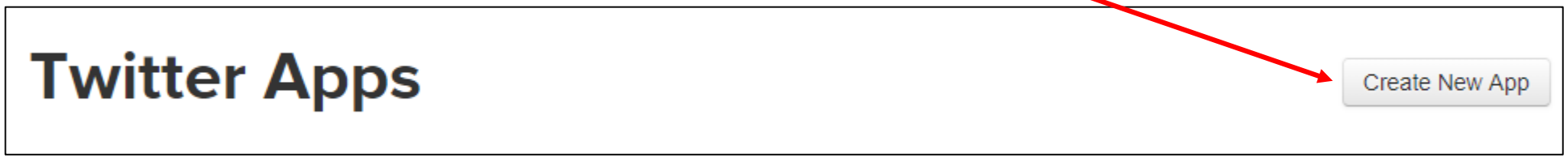
Twitter API Registration

- Go to: https://twitter.com/settings/account
- Click Mobile to identify



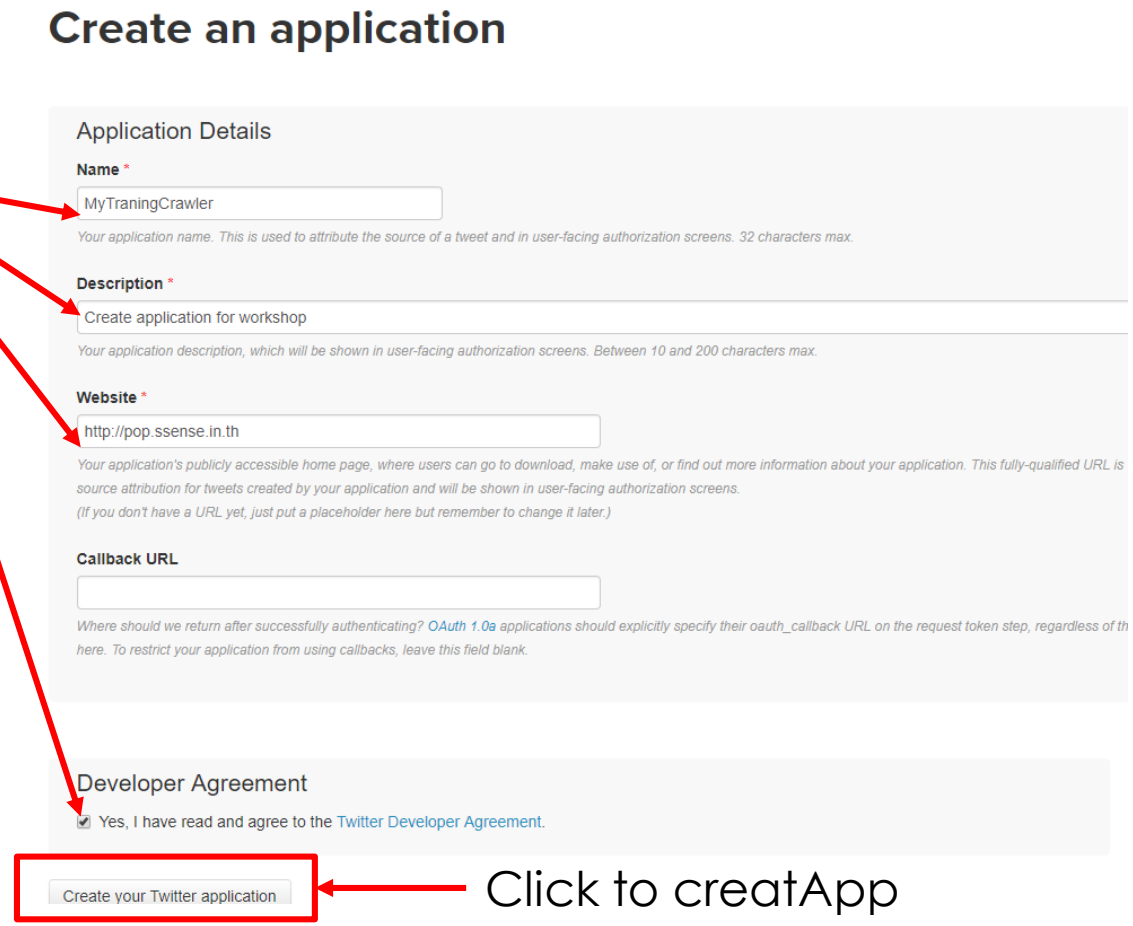
Twitter API KEY

- After login and registered go to: https://apps.twitter.com/
- Click to Create New App for creating the API KEY



Twitter API KEY

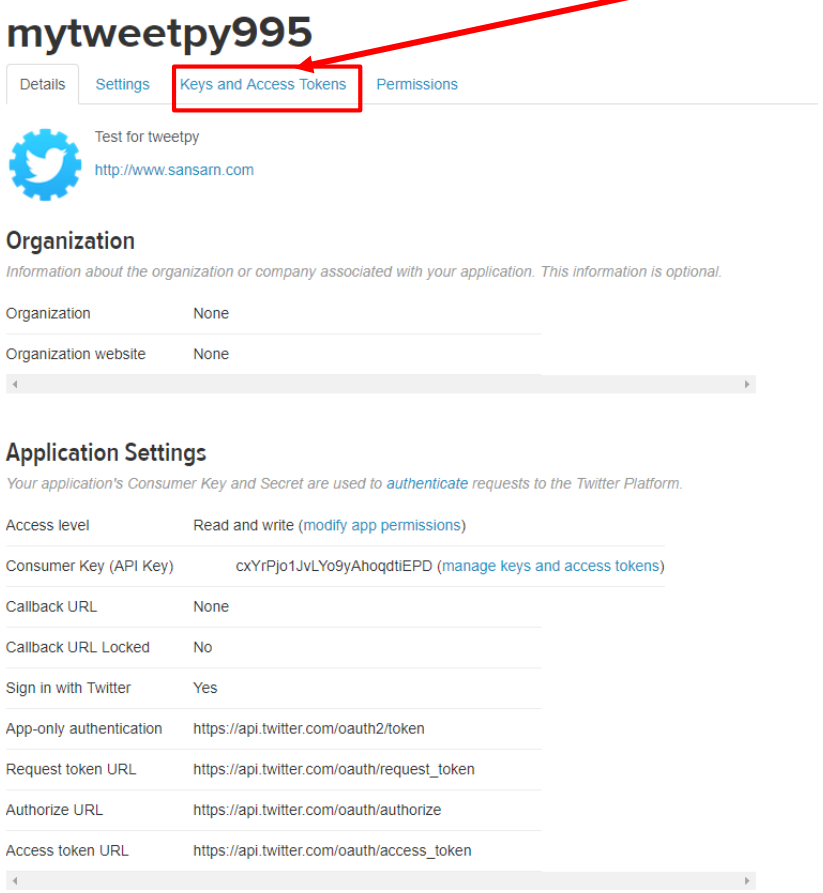
Fill in the form



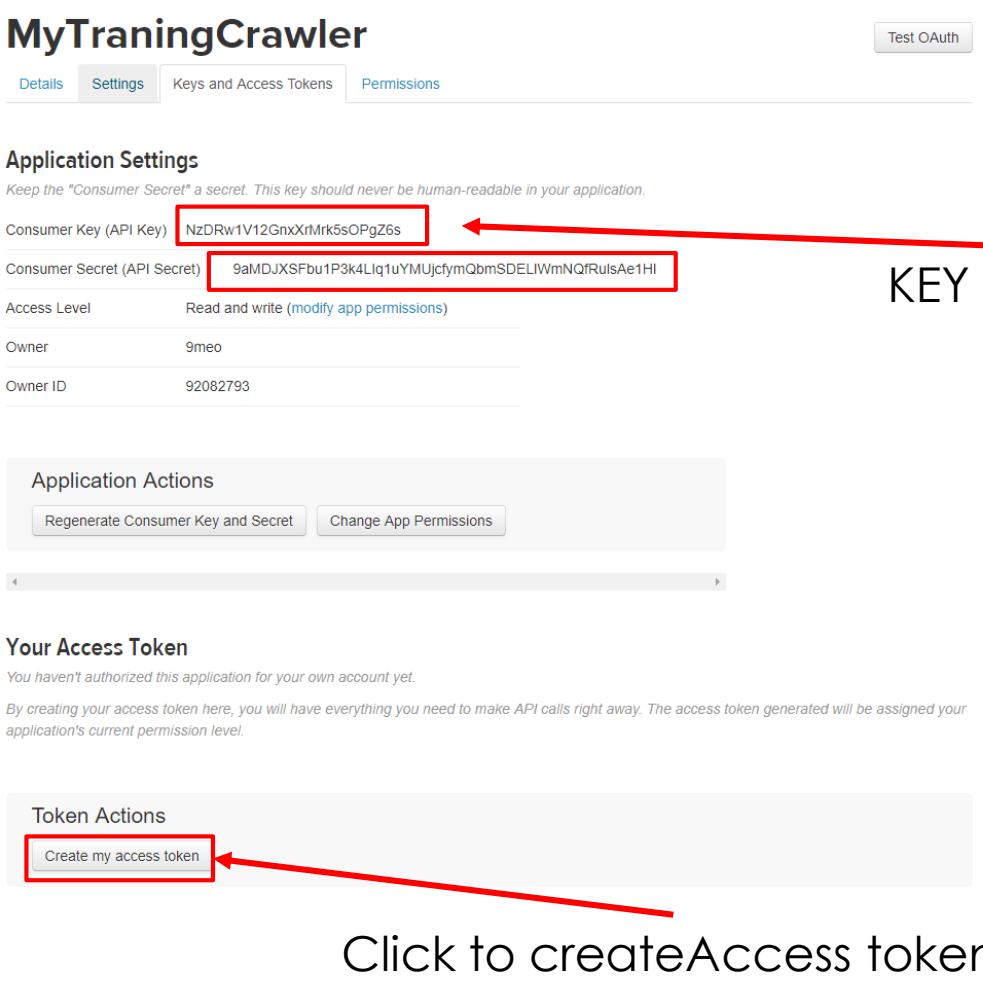
Click to creatApp

Twitter API KEY

Click to create API KEY



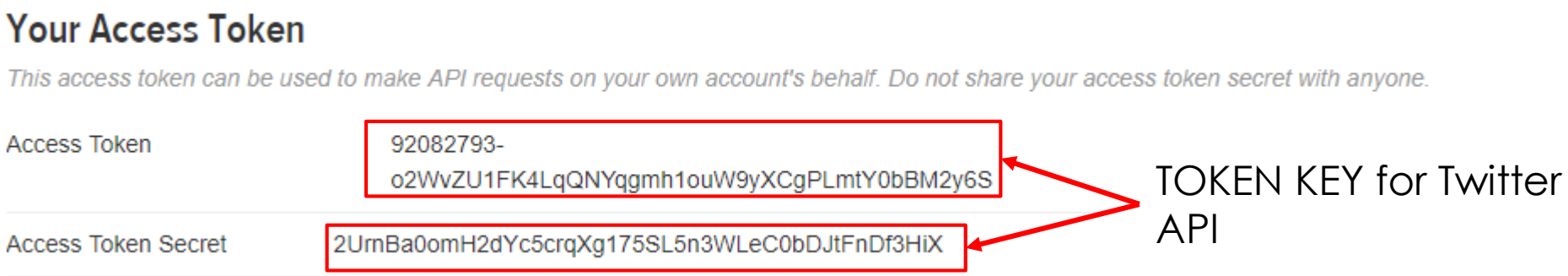
Twitter API KEY



KEY for Twitter API

Click to createAccess token

Twitter API KEY



Twitter api with Tweepy

- Go to: https://github.com/9meo/SIIT-MongoDB-Lab1
- Download file: TwitterMongoDB.ipynb

Create program to call Twitter API for collecting the data

Press Shift+Enter to Run each program

In [1]:
```
import tweepy
import re
```
#call Library

In [2]:
```
consumer_key = ''
consumer_secret = ''
access_token = ''
access_token_secret = ''
```
#insert API KEY

In [3]:
```
# OAuth process, using the keys and tokens
auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)

# Creation of the actual interface, using authentication
api = tweepy.API(auth)
```
#login process with api key

Assignment (Deadline: 4-Sept-2018 23.00)

- Collect twitter using Tweepy stream api by following hashtag
  - ['#bigdata', '#AI', '#datascience', '#machinelearning', '#thailand', '#TheGiftedSeries']
- Find:
  - Document count
  - Top 5 hashtag in your database

```
{'_id': 'IoT', 'sum': 806}
{'_id': 'AI', 'sum': 630}
{'_id': 'Futurist', 'sum': 406}
{'_id': 'BlockChain', 'sum': 376}
{'_id': 'Agile', 'sum': 373}
```

- Submit your notebook to https://goo.gl/6qVqLP
  - By rename to ID_FirstName_Lastname.ipynb
  - For example: 5922040604_sarawoot_kongyoung.ipynb