

Big Data with

Lab#1



Installation (MongoDB)

- MongoDB Current Stable Release (4.0.1)

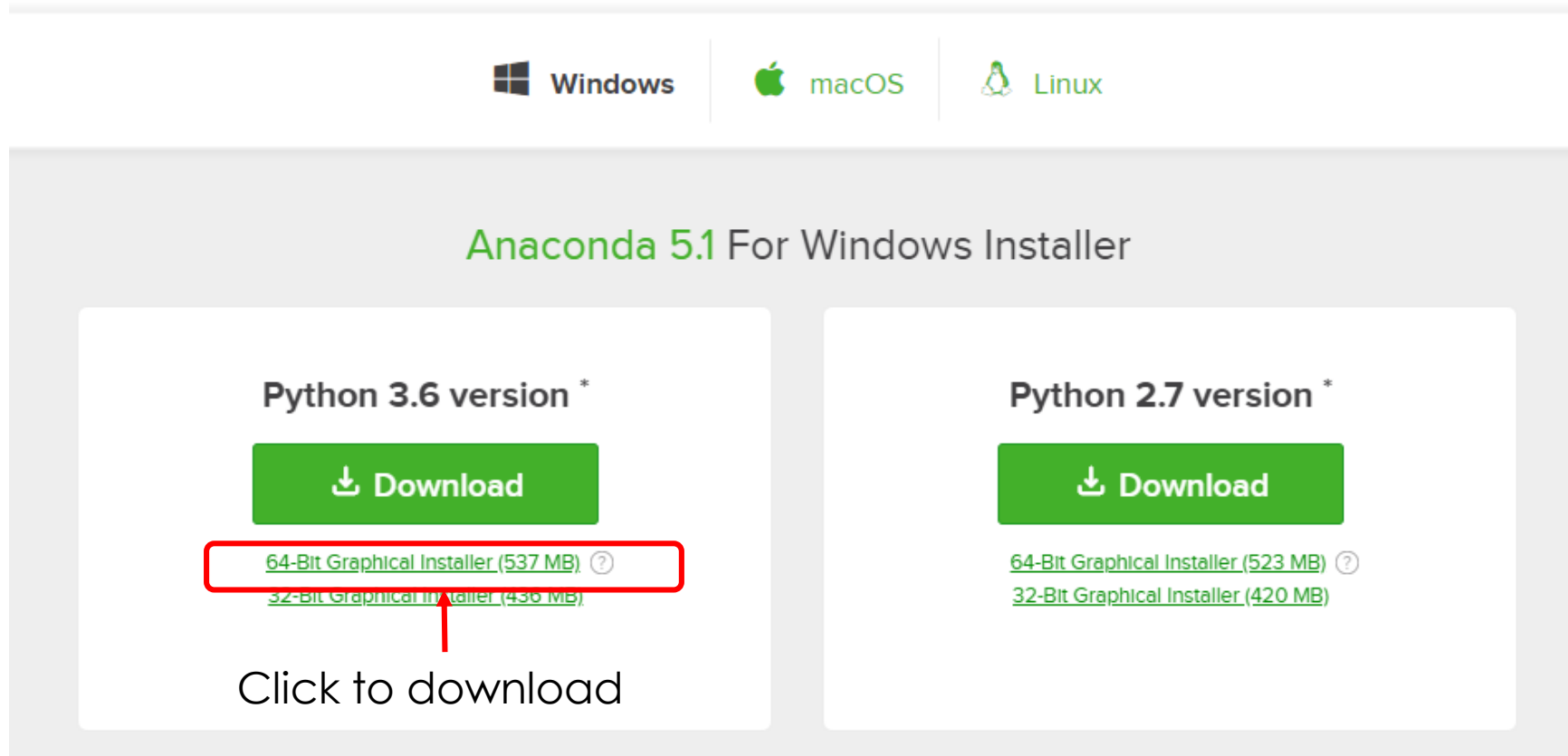
<https://www.mongodb.com/download-center#community>

The screenshot shows the MongoDB download center for the Community Server. The top navigation bar includes links for Atlas, Community Server (highlighted in green), Enterprise Server, Ops Manager, Compass, Connector for BI, and Charts. Below the navigation bar, there are tabs for Current Release, Previous Releases, and Development Releases. The main content area displays the 'Current Stable Release (4.0.1)' with a date of 08/03/2018 and links to Release Notes, Changelog, and Download Source. It also shows the operating system selection: Windows, Linux, and OSX. A dropdown menu for 'Version' is set to 'Windows 64-bit x64'. At the bottom, there is a green button labeled 'DOWNLOAD (msi)' which is circled in red with an arrow pointing to it.

| Atlas | Community Server | Enterprise Server | Ops Manager | Compass | Connector for BI | Charts | | | | | | |
|--|------------------|-------------------|-------------|---------|------------------|--------|---------|-------|-----|---|--|--|
| <p>Current Release Previous Releases Development Releases</p> <p>Current Stable Release (4.0.1) 08/03/2018: Release Notes Changelog Download Source: tgz zip</p> <table border="1"><thead><tr><th>Windows</th><th>Linux</th><th>OSX</th></tr></thead><tbody><tr><td colspan="3"><p>Version:</p><p>Windows 64-bit x64 ▼</p><p>Installation Package:</p><p>DOWNLOAD (msi)</p></td></tr></tbody></table> | | | | | | | Windows | Linux | OSX | <p>Version:</p> <p>Windows 64-bit x64 ▼</p> <p>Installation Package:</p> <p>DOWNLOAD (msi)</p> | | |
| Windows | Linux | OSX | | | | | | | | | | |
| <p>Version:</p> <p>Windows 64-bit x64 ▼</p> <p>Installation Package:</p> <p>DOWNLOAD (msi)</p> | | | | | | | | | | | | |

Installation (Anaconda)

◉ <https://www.anaconda.com/download/>



The screenshot shows the Anaconda website's download page for Windows. At the top, there are three tabs: 'Windows' (selected), 'macOS', and 'Linux'. Below the tabs, the heading 'Anaconda 5.1 For Windows Installer' is displayed. There are two main sections for download: 'Python 3.6 version *' and 'Python 2.7 version *'. Each section has a green 'Download' button. Below the buttons, there are links for '64-Bit Graphical Installer' and '32-Bit Graphical Installer'. A red box highlights the '64-Bit Graphical Installer (537 MB)' link in the Python 3.6 section, with a red arrow pointing to it from the text 'Click to download' below.

Windows macOS Linux

Anaconda 5.1 For Windows Installer

Python 3.6 version *

Download

[64-Bit Graphical Installer \(537 MB\) ?](#)

[32-Bit Graphical Installer \(436 MB\)](#)

Click to download

Python 2.7 version *

Download

[64-Bit Graphical Installer \(523 MB\) ?](#)

[32-Bit Graphical Installer \(420 MB\)](#)

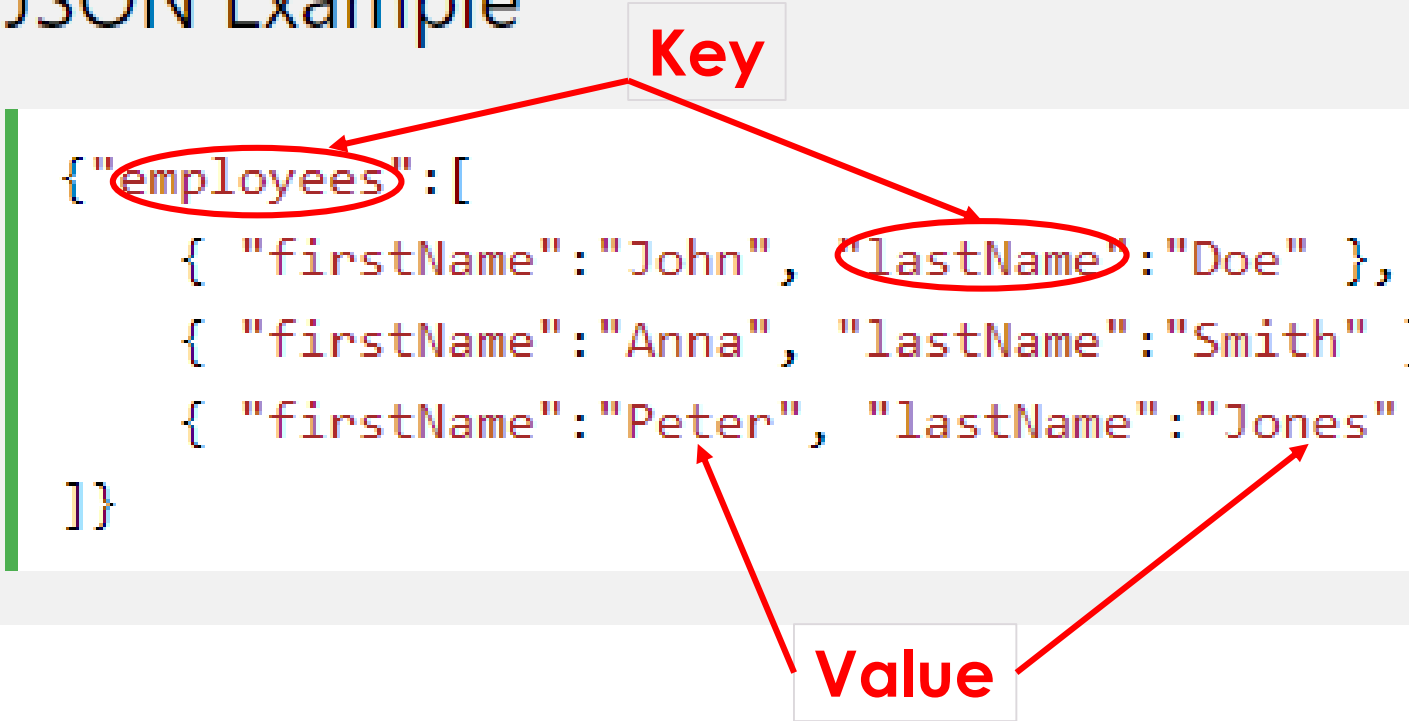
JSON: JavaScript Object Notation.

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

JSON: JavaScript Object Notation.

JSON Example

```
{"employees": [  
  { "firstName": "John", "lastName": "Doe" },  
  { "firstName": "Anna", "lastName": "Smith" },  
  { "firstName": "Peter", "lastName": "Jones" }  
]}
```



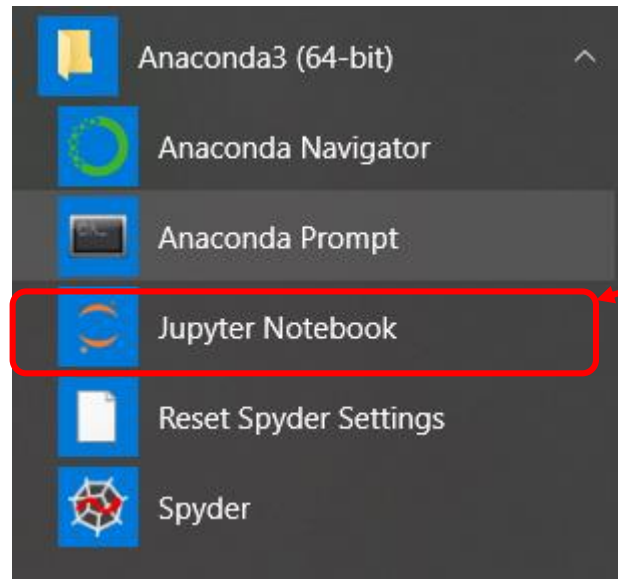
The diagram shows a JSON object with a key "employees" pointing to an array of three objects. Red circles highlight the key "employees" and the "lastName" property in the first object. Red arrows point from a "Key" label to the "employees" key and from a "Value" label to the "lastName" value "Doe".

Value

Jupyter Notebook

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

Start menu->



Click to open Jupyter Notebook

Jupyter Notebook



Logout

Files Running Clusters

Select items to perform actions on them.

Upload New ↻

Click at Documents

| <input type="checkbox"/> 0 ▾ | / | Name ▾ | Last Modified |
|------------------------------|-------------------|--------|------------------|
| <input type="checkbox"/> | 3D Objects | | 1 เดือนที่แล้ว |
| <input type="checkbox"/> | Anaconda3 | | 7 ชั่วโมงที่แล้ว |
| <input type="checkbox"/> | Contacts | | 1 เดือนที่แล้ว |
| <input type="checkbox"/> | Desktop | | 15 วันที่แล้ว |
| <input type="checkbox"/> | Documents | | 3 นาทีที่แล้ว |
| <input type="checkbox"/> | Downloads | | 12 นาทีที่แล้ว |
| <input type="checkbox"/> | eclipse-workspace | | 16 วันที่แล้ว |
| <input type="checkbox"/> | Favorites | | 1 เดือนที่แล้ว |
| <input type="checkbox"/> | Links | | 1 เดือนที่แล้ว |
| <input type="checkbox"/> | Music | | 1 เดือนที่แล้ว |
| <input type="checkbox"/> | OneDrive | | 1 เดือนที่แล้ว |
| <input type="checkbox"/> | Pictures | | 1 เดือนที่แล้ว |
| <input type="checkbox"/> | Saved Games | | 1 เดือนที่แล้ว |
| <input type="checkbox"/> | Searches | | 1 เดือนที่แล้ว |
| <input type="checkbox"/> | Videos | | 16 วันที่แล้ว |

Jupyter Notebook

Files

Running

Clusters

Click to **crate** folder

Select items to perform actions on them.

0 ▾ / Documents

- ..
- ☐ Custom Office Templates
- ☐ My Tableau Repository
- ☐ Python Scripts

Upload New ↻

Notebook:
Python 3

Other:
Text File
Folder
Terminal

Files Running Clusters

Rename Move

1 ▾ / Documents

- ..
- ☐ Custom Office Templates
- ☐ My Tableau Repository
- ☐ Python Scripts
- ☒ Untitled Folder

Click to **rename** the folder

2

3

4

Rename directory

Enter a new directory name:

MyCrawler

5

Cancel Rename

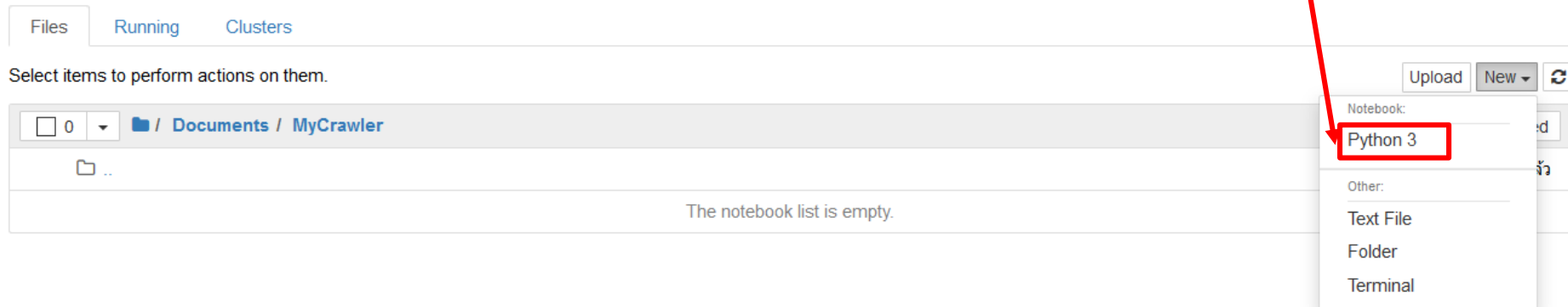
ไม่กี่วินาทีที่แล้ว

Jupyter Notebook

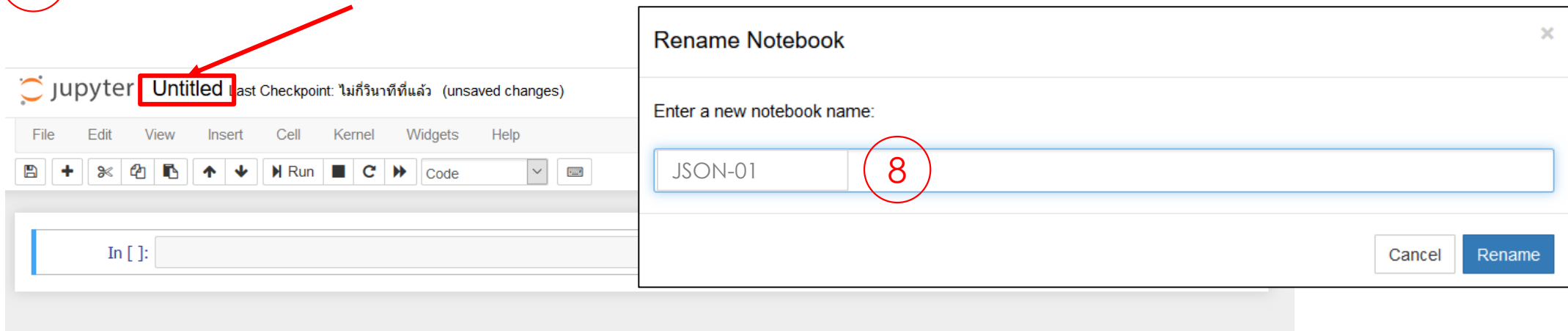
Click to **crate** Python Application

6

Click into MyCrawler Folder



7 Click to rename the Application



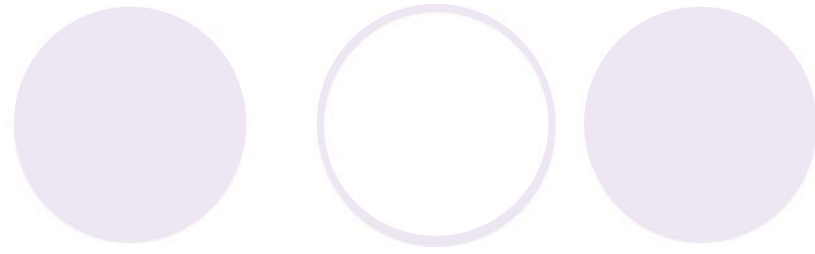
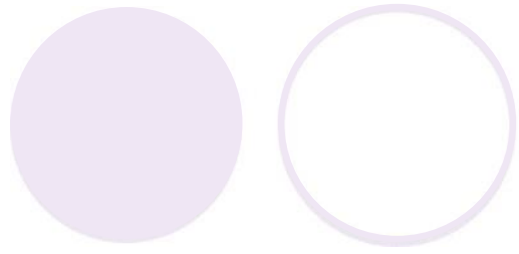
Work with JSON in Python

```
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












pythonTM

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 |
|  <u>mongod.exe</u> | 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



D:\MongoDB\bin\mongod.exe

```
2018-08-27T20:13:57.962+0700 I CONTROL [main] Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDis
abledProtocols 'none'
2018-08-27T20:13:58.457+0700 I CONTROL [initandlisten] MongoDB starting : pid=83776 port=27017 dbpath=D:\data\db\ 64-bi
t host=SPT-Halisa
2018-08-27T20:13:58.457+0700 I CONTROL [initandlisten] targetMinOS: Windows 7/Windows Server 2008 R2
2018-08-27T20:13:58.458+0700 I CONTROL [initandlisten] db version v4.0.1
2018-08-27T20:13:58.458+0700 I CONTROL [initandlisten] git version: 54f1582fc6eb01de4d4c42f26fc133e623f065fb
2018-08-27T20:13:58.458+0700 I CONTROL [initandlisten] allocator: tcmalloc
2018-08-27T20:13:58.458+0700 I CONTROL [initandlisten] modules: none
2018-08-27T20:13:58.458+0700 I CONTROL [initandlisten] build environment:
2018-08-27T20:13:58.458+0700 I CONTROL [initandlisten]     distmod: 2008plus-ssl
2018-08-27T20:13:58.458+0700 I CONTROL [initandlisten]     distarch: x86_64
2018-08-27T20:13:58.458+0700 I CONTROL [initandlisten]     target_arch: x86_64
2018-08-27T20:13:58.459+0700 I CONTROL [initandlisten] options: {}
2018-08-27T20:13:58.462+0700 I STORAGE [initandlisten] Detected data files in D:\data\db\ created by the 'wiredTiger' s
torage engine, so setting the active storage engine to 'wiredTiger'.
2018-08-27T20:13:58.463+0700 I STORAGE [initandlisten] wiredtiger_open config: create,cache_size=7619M,session_max=2000
0,eviction=(threads_min=4,threads_max=4),config_base=false,statistics=(fast),log=(enabled=true,archive=true,path=journal
```

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

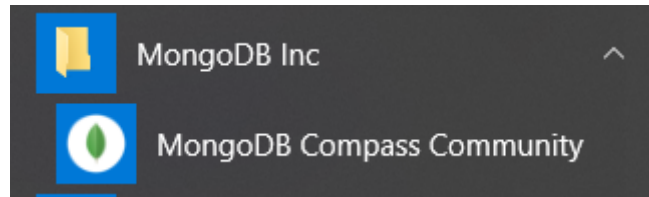
Created 5b8411f9bdc81d9851a4d0b8

Created 5b8411f9bdc81d9851a4d0b9

MongoDB Compass Community

Connect to Host

Start menu->



Hostname localhost

Port 27017

SRV Record

Authentication None

Replica Set Name

Read Preference Primary

SSL None

SSH Tunnel None

Connect to Server

Favorite Name Local

1

CREATE FAVORITE

CONNECT

MongoDB Compass Community

Click to Check inserted data

2

The screenshot shows the MongoDB Compass interface. On the left sidebar, the 'business' database is selected, and the 'reviews' collection is highlighted. A red arrow points from the number '2' to the 'business' database entry. The main panel displays the 'business.reviews' collection with 2 documents and 1 index. The documents are listed in a table view:

| DOCUMENTS | TOTAL SIZE | AVG. SIZE | INDEXES | TOTAL SIZE | AVG. SIZE |
|-----------|------------|-----------|---------|------------|-----------|
| 2 | 170B | 85B | 1 | 16.0KB | 16.0KB |

Below the table, the documents are displayed in a list view:

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

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

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)
```

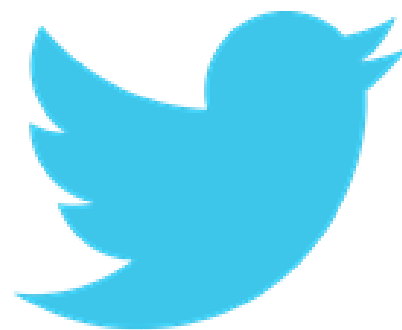
```
{'n': 1, 'updatedExisting': True, 'ok': 1.0, 'nModified': 1}
```

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

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

MongDB: more...

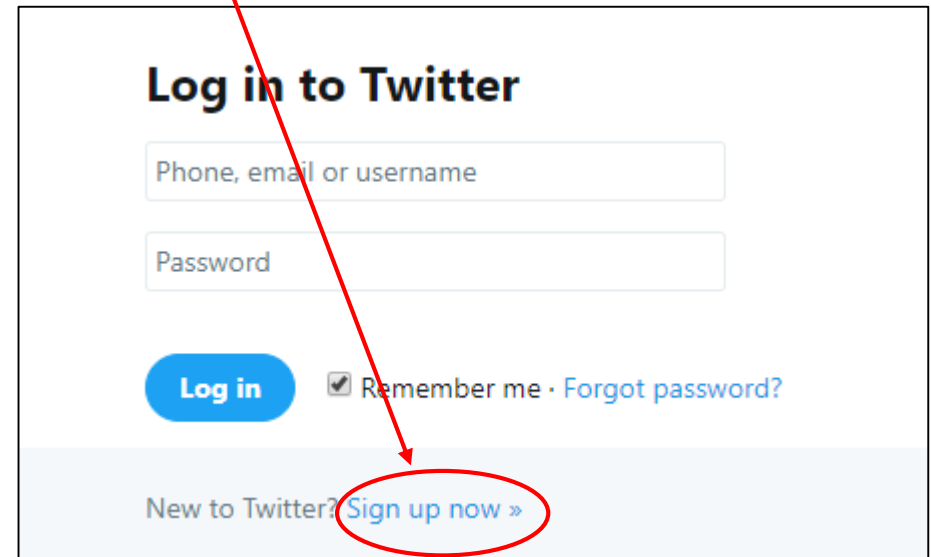
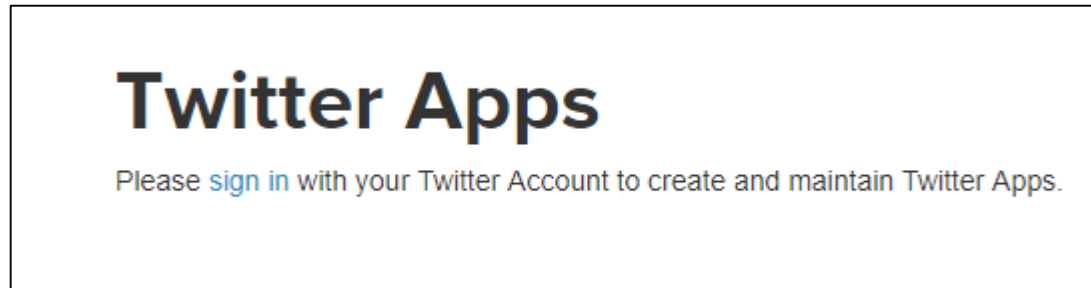
- ◉ Go to : <https://github.com/9meo/SIIT-MongoDB-Lab1>
- ◉ Try it!!!



twitter

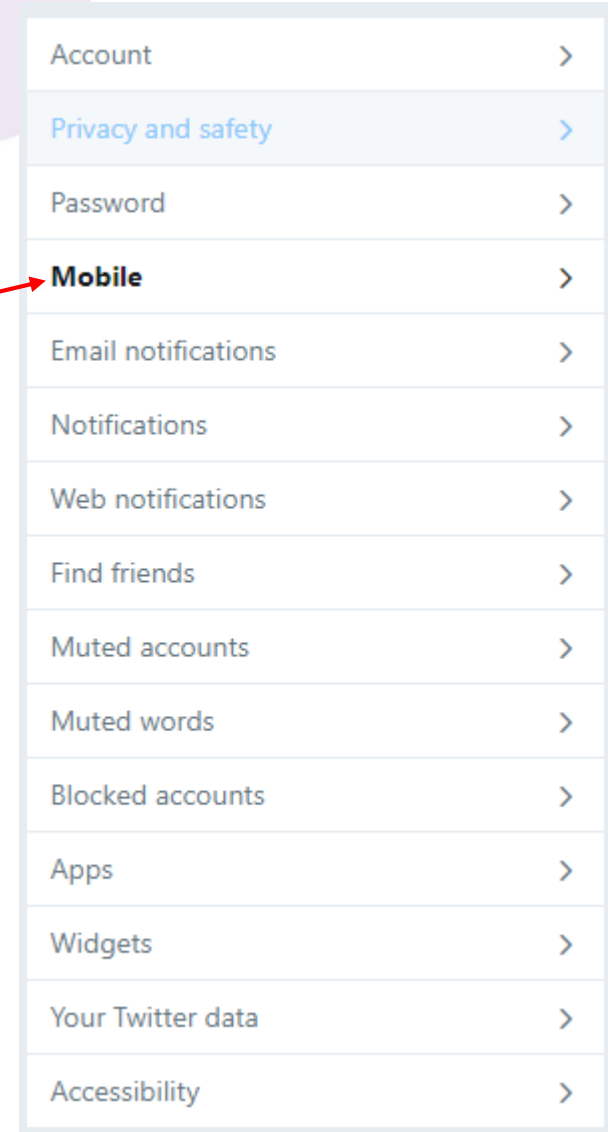
Twitter API Registration

- ◉ 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

A screenshot of the Twitter settings menu. The menu is a vertical list of items, each with a right-pointing chevron. A red arrow points from the text 'Click Mobile to identify' in the list group to the 'Mobile' item in the settings menu.

| | |
|---------------------|---|
| Account | > |
| Privacy and safety | > |
| Password | > |
| Mobile | > |
| Email notifications | > |
| Notifications | > |
| Web notifications | > |
| Find friends | > |
| Muted accounts | > |
| Muted words | > |
| Blocked accounts | > |
| Apps | > |
| Widgets | > |
| Your Twitter data | > |
| Accessibility | > |

Twitter API KEY

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

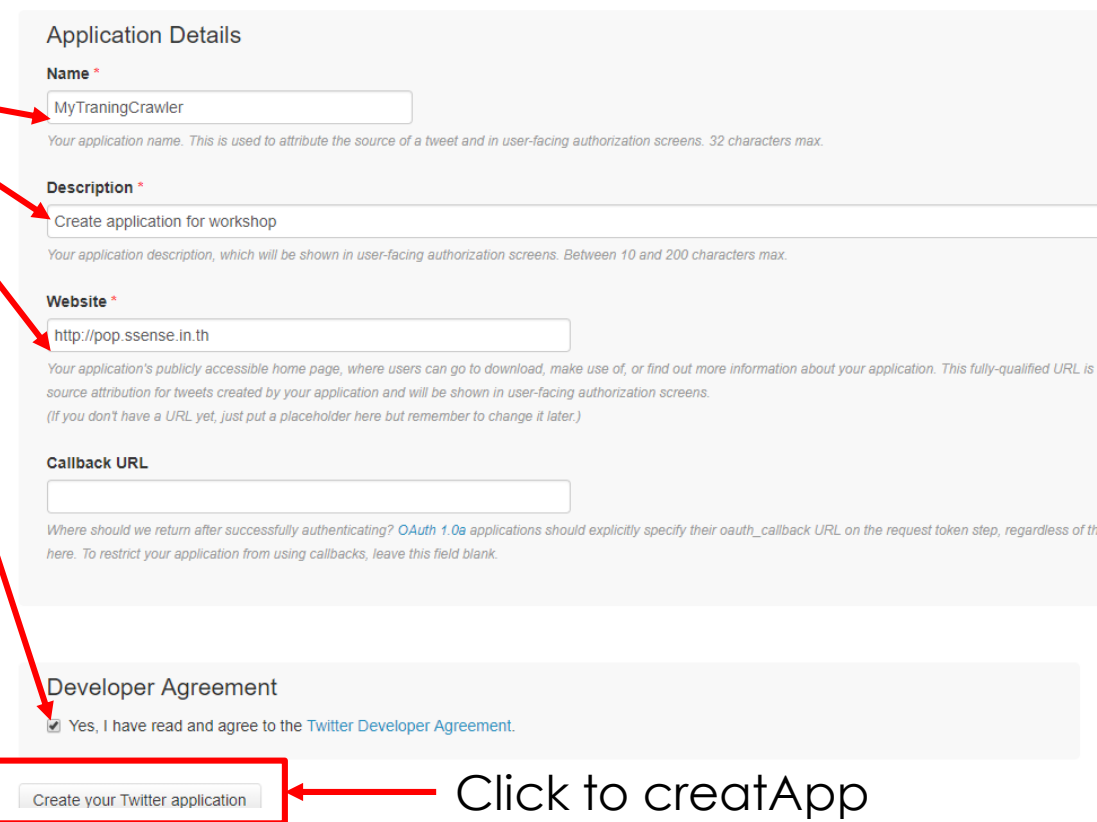
Twitter Apps

Create New App

Twitter API KEY

Create an application

Fill in the form



The screenshot shows the 'Create an application' form on the Twitter developer portal. It is divided into two main sections: 'Application Details' and 'Developer Agreement'. The 'Application Details' section contains four input fields: 'Name' (filled with 'MyTraningCrawler'), 'Description' (filled with 'Create application for workshop'), 'Website' (filled with 'http://pop.ssense.in.th'), and 'Callback URL' (empty). Each field has a small text description below it. The 'Developer Agreement' section has a checkbox labeled 'Yes, I have read and agree to the [Twitter Developer Agreement](#)' which is checked. At the bottom, there is a button labeled 'Create your Twitter application' which is highlighted with a red rectangle. Red arrows point from the text 'Fill in the form' to the 'Name', 'Description', 'Website', and 'Developer Agreement' sections. Another red arrow points from the text 'Click to creatApp' to the 'Create your Twitter application' button.

Application Details

Name *
MyTraningCrawler
Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description *
Create application for workshop
Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

Website *
http://pop.ssense.in.th
Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is source attribution for tweets created by your application and will be shown in user-facing authorization screens.
(If you don't have a URL yet, just put a placeholder here but remember to change it later.)

Callback URL

Where should we return after successfully authenticating? [OAuth 1.0a](#) applications should explicitly specify their oauth_callback URL on the request token step, regardless of th here. To restrict your application from using callbacks, leave this field blank.

Developer Agreement
☒ Yes, I have read and agree to the [Twitter Developer Agreement](#).

Create your Twitter application

Click to creatApp

Twitter API KEY

Click to create API KEY

mytweetpy995

Details Settings **Keys and Access Tokens** Permissions



Test for tweetpy

<http://www.sansarn.com>

Organization

Information about the organization or company associated with your application. This information is optional.

| | |
|----------------------|------|
| Organization | None |
| Organization website | None |

Application Settings

Your application's Consumer Key and Secret are used to [authenticate](#) requests to the Twitter Platform.

| | |
|-------------------------|---|
| Access level | Read and write (modify app permissions) |
| Consumer Key (API Key) | cxYrPjo1JvLYo9yAhoqdtiEPD (manage keys and access tokens) |
| Callback URL | None |
| Callback URL Locked | No |
| Sign in with Twitter | Yes |
| App-only authentication | https://api.twitter.com/oauth2/token |
| Request token URL | https://api.twitter.com/oauth/request_token |
| Authorize URL | https://api.twitter.com/oauth/authorize |
| Access token URL | https://api.twitter.com/oauth/access_token |

Twitter API KEY

MyTraningCrawler

[Test OAuth](#)[Details](#)[Settings](#)[Keys and Access Tokens](#)[Permissions](#)

Application Settings

Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.

Consumer Key (API Key) NzDRw1V12GnxXrMrk5sOPgZ6s

Consumer Secret (API Secret) 9aMDJXSFBu1P3k4Llq1uYMUJcfymQbmSDELIWmNQfRulsAe1HI

Access Level Read and write ([modify app permissions](#))

Owner 9meo

Owner ID 92082793

KEY for Twitter API

Application Actions

[Regenerate Consumer Key and Secret](#)[Change App Permissions](#)

Your Access Token

You haven't authorized this application for your own account yet.

By creating your access token here, you will have everything you need to make API calls right away. The access token generated will be assigned your application's current permission level.

Token Actions

[Create my access token](#)

Click to createAccess token

Twitter API KEY

Your Access Token

This access token can be used to make API requests on your own account's behalf. Do not share your access token secret with anyone.

Access Token

92082793-
o2WvZU1FK4LqQNYqgmh1ouW9yXCgPLmtY0bBM2y6S

Access Token Secret

2UmBa0omH2dYc5crqXg175SL5n3WLeC0bDJtFnDf3HiX

TOKEN KEY for Twitter
API

Twitter api with Tweepy

- ◉ Go to: <https://github.com/9meo/IIIT-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
- ◉ Submit your notebook to <https://goo.gl/6qVqLP>
 - ◉ By rename to ID_FirstName_Lastname.ipynb
 - ◉ For example: 5922040604_sarawoot_kongyoung.ipynb

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