

Unit 12: HandsOn Lab 1

Jupyter PyMongo_InClassHandsOnLab1 Last Checkpoint: 34 minutes ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted | Python 3 C

```

Install pymongo library

In [1]: !pip install pymongo

Requirement already satisfied: pymongo in /anaconda3/lib/python3.6/site-packages (3.7.1)

In [2]: ## This notebook contains sample code to connect to a MongoDB collection , insert data and retrieve data

import pymongo
import pandas as pd
from pymongo import MongoClient
import json

In [ ]: ##### Connect to the local mongodb and collection sample

In [3]: client = MongoClient('mongodb://127.0.0.1:27017')
##Connect to the nlpExampleDB database
db = client.nlpExampleDB

##Connect to the sampleReview collection
collection = db.sampleReviews

In [4]: ## Read the sample data file and load it into a json objec
with open("SampleData.json","rb") as fp:
    jsonObj = json.load(fp)

In [5]: for doc in jsonObj["data"]:
    collection.insert_one(doc)

In [6]: ## Read from Database

list(collection.find({},{"_id":0,"ReviewText":1}).limit(1))

Out[6]: [{'ReviewText': 'Excellent Central Location in Boston I stayed here for 6 nights while attending a conference in Bos
ton. This is an excellent hotel. It is a very typical Fairmont experience with excellent service, great food and con
fortable rooms. The rooms are well equipped including high speed internet and hdtv. It is very clean. There is a lot
of history throughout the hotel, which was very interesting.The location of the hotel is the biggest highlight for m
e. It is very well situated, just two blocks away from Newbury Street, where there is some of the best shopping in N
orth America. It is walking distance to a lot of great bars, pubs and restaurnats. Above all, it is right next to tw
o different T subway stops, so it's just a hop and a skip to get anywhere in the Boston area, including Cambridge, F
reedom Trail, etc etc. The only drawback is that the hotel is perhaps slightly overpriced. However, if you find a gr
eat deal, this is a fantastic place and it does earn a highly recommended from me."}]

In [7]: ## count the # of reviews with adjective "good"
((collection.find({"good":1},{"_id":0,"ReviewText":1}).count()))

/Users/shanqinggu/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:2: DeprecationWarning: count is deprec
ated. Use Collection.count_documents instead.

Out[7]: 30

In [8]: ## find and list the ReviewText of the reviews with adjective "good"

list(collection.find({"good":1},{"_id":0,"ReviewText":1}))

Out[8]: [{'ReviewText': 'Awesome Hotel, fantastic service in a great location Just returned from our 2nd stay at the Lenno
x and it did not disapoint!The room was clean spacious, beds were very comfortable rooms now have added bonus of
a fridge and in room safe Staff couldn't do enough for you very polite and helpful, the Lennox is situated in a gr
eat central location in Boston easy to get around on foot or by using the T (stations are very close by)good resta
urants and shopping easily accessible, After a very busy sightseeing day too tired to go out so used the hotels in

```

Unit 12: HandsOn Lab 2

Content Management Database using MongoDB

InClass Assignment :

Add a new service to search for documents by the tags and return to the customer

This exercise builds the backend services for a sample content management . The code builds the following services

The Rest Services can be started by running

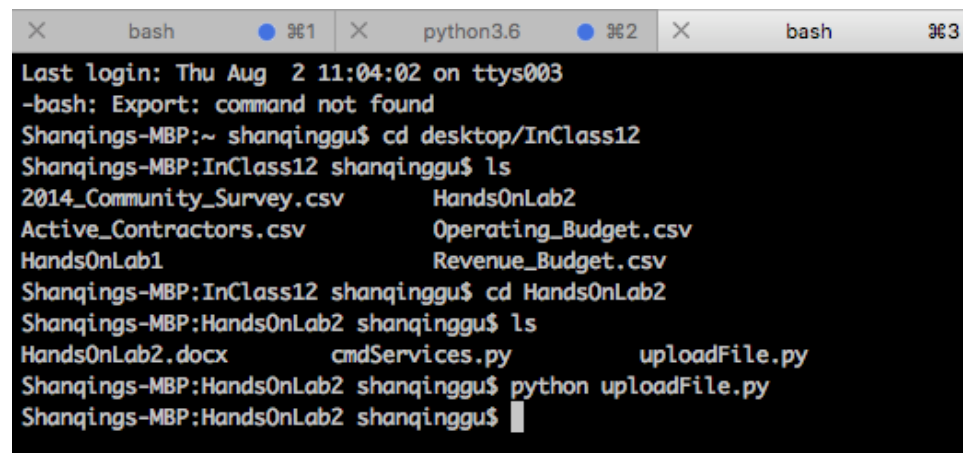
`python cmdServices.py`

- uploadDocument

This service takes any file as its input and stores it in the database. The service also accepts additional meta tags about the files which can then be used for searching. A sample client program `uploadFile.py` is provided to “post” files to this service . The `uploadFile.py` can be edited to include the file to be upload from your local machine and the relevant tags

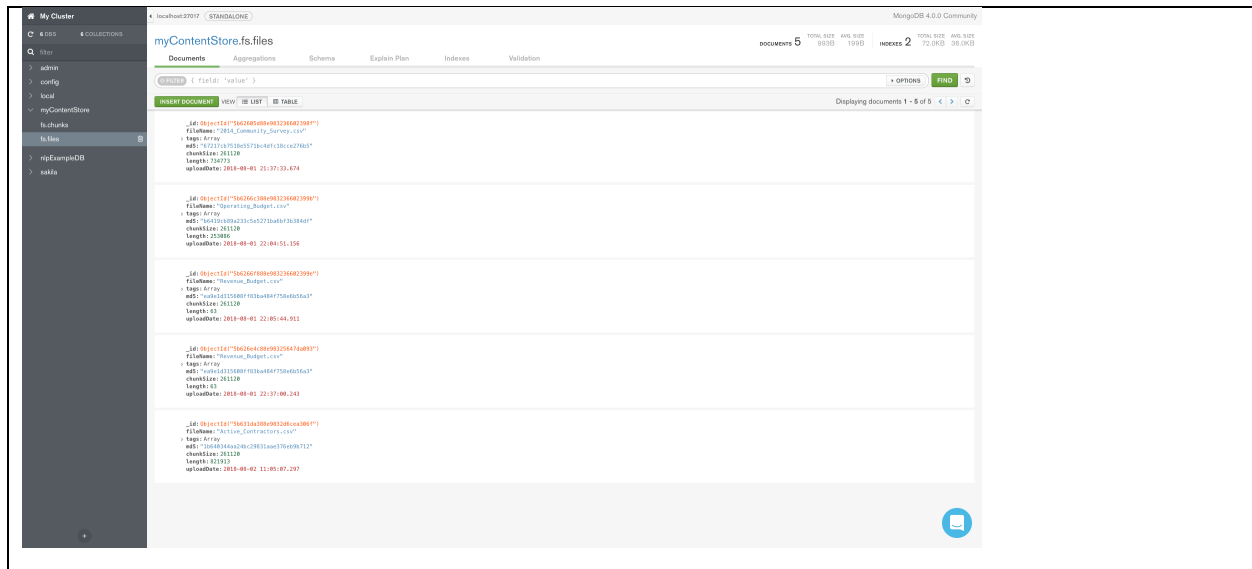
E.g. usage
`python uploadFile.py`

Upload file `Active_Contractors.csv`

A terminal window with three tabs: 'bash', 'python3.6', and 'bash'. The 'python3.6' tab is active. The terminal shows the following commands and output:

```
Last login: Thu Aug  2 11:04:02 on ttys003
-bash: Export: command not found
Shanqings-MBP:~ shanqinggu$ cd desktop/InClass12
Shanqings-MBP:InClass12 shanqinggu$ ls
2014_Community_Survey.csv      HandsOnLab2
Active_Contractors.csv        Operating_Budget.csv
HandsOnLab1                    Revenue_Budget.csv
Shanqings-MBP:InClass12 shanqinggu$ cd HandsOnLab2
Shanqings-MBP:HandsOnLab2 shanqinggu$ ls
HandsOnLab2.docx      cmdServices.py      uploadFile.py
Shanqings-MBP:HandsOnLab2 shanqinggu$ python uploadFile.py
Shanqings-MBP:HandsOnLab2 shanqinggu$
```

Check uploaded file `Active_Contractors.csv` in MongoDB Compass



- listDocument

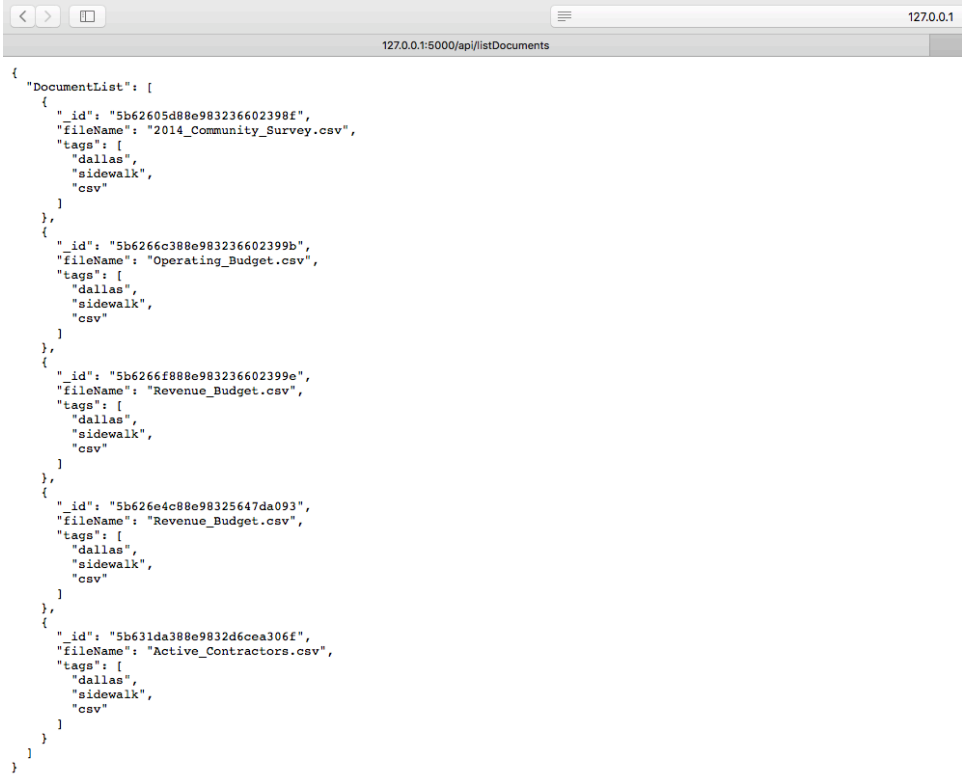
This service lists all the documents that are uploaded in the database. Launch the following url in the browser <http://localhost:5000/api/listDocuments>

List uploaded documents

```

X      bash      361  X      bash      362  X      python3.6      363
Last login: Thu Aug  2 11:02:48 on ttys002
-bash: Export: command not found
Shanqings-MBP:~ shanqinggu$ cd desktop/InClass12
Shanqings-MBP:InClass12 shanqinggu$ ls
2014_Community_Survey.csv      HandsOnLab2
Active_Contractors.csv        Operating_Budget.csv
HandsOnLab1                    Revenue_Budget.csv
Shanqings-MBP:InClass12 shanqinggu$ cd HandsOnLab2
Shanqings-MBP:HandsOnLab2 shanqinggu$ ls
HandsOnLab2.docx      cmdServices.py      uploadFile.py
Shanqings-MBP:HandsOnLab2 shanqinggu$ python cmdServices.py
* Serving Flask app "cmdServices" (lazy loading)
* Environment: production
  WARNING: Do not use the development server in a production environment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 623-067-097
127.0.0.1 - - [02/Aug/2018 11:05:07] "POST /api/uploadDocument HTTP/1.1" 200 -
127.0.0.1 - - [02/Aug/2018 11:06:23] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [02/Aug/2018 11:06:23] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [02/Aug/2018 11:06:28] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [02/Aug/2018 11:06:40] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [02/Aug/2018 11:06:52] "GET /api/listDocuments HTTP/1.1" 200 -
127.0.0.1 - - [02/Aug/2018 11:07:41] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [02/Aug/2018 11:07:50] "GET /api/listDocuments HTTP/1.1" 200 -
127.0.0.1 - - [02/Aug/2018 11:08:45] "GET /api/getDocument?fileId=5b631da388e9832d6cea306f HTTP/1.1" 200 -

```



- getDocument

This service takes the id value of the document and returns the file from the database
E.g. <http://localhost:5000/api/getDocument?fileId=5b631da388e9832d6cea306f>

Get file Active_Contractors.csv from <http://localhost:5000/api/getDocument?fileId=5b631da388e9832d6cea306f>

