PRACTICAL LESSON

MONGODB

slides on : https://github.com/JoGreen/mongo_workshop

install mongodb, and show /usr/local/etc/mongodb.conf

create a database and a collection

crud operations insert/update/delete/find

bulk operations

transaction is on document level

will support multi-doc transactions

MySQL vs MongoDB

MySQL and MongoDB

indexes, aggregations, text search, cursor

INSTALLATION...

- Windows
  - download at https://www.mongodb.com/download-center?jmp=nav#community
- Linux Ubuntu-based
  - sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv 2930ADAE8CAF5059EE73BB4B58712A2291FA4AD5
  - 14.04 -> echo "deb [arch=amd64] https://repo.mongodb.org/apt/ubuntu trusty/mongodb-org/3.6 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.6.list
  - 16.04/17.10 -> echo "deb [arch=amd64,arm64] https://repo.mongodb.org/apt/ubuntu xenial/mongodb-org/3.6 multiverse" | sudo tee /etc/apt/sources.list.d/mongodb-org-3.6.list
  - sudo apt-get update
  - $ sudo apt-get install -y mongodb-org
- OSX
  - brew update
  - brew install mongodb

LAUNCH IT

- Windows
  - MongoDB's default data directory path is the absolute path \data\db => command line -> md \data\db
  - run C:\Program Files\MongoDB\Server\3.6\bin\mongod.exe [server]
  - run C:\Program Files\MongoDB\Server\3.6\bin\mongo.exe [client]
- Linux Ubuntu-based
  - sudo systemctl start/status/stop mongod [server]
  - mongo [client]
- OSX
  - mongod [server]
  - mongo [client]

SOME EASY CONFIG OPTIONS

- to specify a specific port or storage directory launch mongod with –port/–dbpath option
    - es: mongod –port 27018 –dbpath /data
    - or you can edit appropriately “mongod.conf” file and launch mongod without any options

Db and Collections

- from mongo shell :
  - create a new db and a new collection
    - insert / remove / find / updateOne /replaceOne
      - use mydbname
      - db.mycollectionname.insert({“lab”:”basi di dati”})
      - db.mycollectionname.find({})

LET' S CODE NOW

- repo clone link :
  - https://github.com/JoGreen/mongo_workshop.git

- repo link:
  - https://github.com/JoGreen/mongo_workshop

LET' S CODE NOW

- we will use python 2.7 and a mongoldb local installation.
- launch from the shell the setup.py file with the follow command:
  - python setup.py install
- you need to be able to connect to a mysql server. if you miss something reading the errors you will find all you need searching on google(you will use a remote mysql server with all the data you need for this exercises).
- if you want to use a python ide i suggest Pycharm community edition.

LET' S CODE NOW

- Exercise 1
  - simple crud operations. Follow the instructions in the comments
  - complete mongo_crud.py
  - to launch the script from the shell:
    - go to the root of the project (mongo_workshop directory)
    - prompt: python -m exercise1.mongo_crud
  - to launch the script from Pycharm right click on mongo_crud.py and click run mongo_crud.py
  - mongo cursor: they are a way to iterate on the docs in the query result without loading all those docs in ram
- Exercise 2.0
  - run migration.py just to create a collection with data from mysql tables
  - have a look on how we copy the same data with a different structure.
  - we don t need any join to retrieve any info we want
  - the data schema is more similar to your mind schema of the all data . Do you agree ?
  - you don' t need to declare any schema. you could declare a schema but it' s not mandatory and in many use cases it is useless.

Exercise 2.1

- Follow the instructions in the comments in query_comparison.py
- launch the script as in the previous exercises
- some log using the same host for both dbs
  - 7.80879998207s first query (sql)
  - 0.0024471282959s first query (mongo)
  - 7.92992305756s first query (sql)
  - 2.28897500038s second query (mongo) loading all the results
- your logs will be using a remote mysql instance and a local mongodb instance
- what do you think about these time logs ?

- Exercise 3
  - use both the dbs together
  - use all we have already seen and complete the assignment.
  - if there is time try to see time statistic differences doing both operations in sql and in mongo

TEXT SEARCH

- MongoDB supports query operations that perform a text search of string content. To perform text search, MongoDB uses a text index and the $text operator.
- create index: db.papers_workshop.createIndex({ name: "text", title: "text" })
- db.papers_workshop.find({ $text: { $search: “web acid" } })
- unsorted order by default, however, text search queries will compute a relevance score for each document that specifies how well a document matches the query.
- db.papers_workshop.find({ $text: { $search: "destribuited" } }, { score: { $meta: "textScore" } }).sort({ score: { $meta: "textScore" } })

EXTRACT THE ARCHIVE IN FOLDER DATA

- restore the collection on mongo (mongorestore) from the dump in the archive
  - uncompress short_tweet.tar.gz
  - mongorestore –db twitter –collection tweets tweets_short_dump /twitter/tweets/tweets.bson
- create one or more text search on different string fields as we have seen before
- use text search to find tweet with high score on a couple of linked keywords (many tweets are on F1)