

MONGODB ASSIGNMENT

LAB DOCUMENT

COMMANDS FOR CONTAINER SETUP:

1)Pulling Mongodb image using docker-compose.yml file: Instructions is docker-compose.yml file will be used to create container when mongo image is run in detach mode.

```
C:\Namra\IBA-Masters\BigDataAnalytics>docker-compose -f docker-compose.yml up -d
Pulling mongodb (mongo:...)
latest: Pulling from library/mongo
7c3b88808835: Already exists
48a403577c28: Pull complete
76bbb7dc9013: Pull complete
e81b1e5a386b: Pull complete
7bdc1db49ec9: Pull complete
7b2f602f894c: Pull complete
627a84bea5f4: Pull complete
fc06fa2a9f52: Pull complete
0133c89ee92b: Pull complete
4e990167b745: Pull complete
Digest: sha256:af367448dfeb8761d60aecfd34af5820bb675f90053146eaabf97c8b3e01c3979
Status: Downloaded newer image for mongo:latest
Creating mongodb ... done
```

2) Checking all containers whether running or exited to obtain container id of mongo container;

```
C:\Namra\IBA-Masters\BigDataAnalytics>docker ps -a
CONTAINER ID        IMAGE       COMMAND      CREATED     STATUS          PORTS          NAMES
082e182b4249        mongo       "docker-entrypoint.s..."   13 seconds ago   Up 11 seconds   0.0.0.0:27017->27017/tcp   mongodb
47b196e3b3f7        ubuntu      "bash"        4 days ago    Exited (127)   3 days ago
725e7bb4f678        ubuntu      "bash"        4 days ago    Exited (1)     4 days ago
clever_brattain
```

3) docker cp command will copy the dataset from pc path to container path that is “home” in data/db: The dataset from path is mapped from pc to container path here. Container id is also mentioned.

```
C:\Namra\IBA-Masters\BigDataAnalytics\mongodb>docker cp C:\Namra\IBA-Masters\BigDataAnalytics\mongodb\database_082e182b4249:home
```

4)Bashing into mongo container using interactive terminal mode: Inside linux mongo container, we are changing directory to home which is containers' path where we copied our dataset. Then checking where our database directory is successfully copied in home;

```
C:\Namra\IBA-Masters\BigDataAnalytics\mongodb>docker exec -it mongodb bash  
root@082e182b4249:/# cd home  
root@082e182b4249:/home# ls  
database
```

5) Changing directory to dataabse and 'ls' to see our json dataset files (collections):

```
root@082e182b4249:/home# cd database
root@082e182b4249:/home/database# ls
WiredTiger          collection-0--4070724485692793250.wt index-1--4070724485692793250.wt mongod.lock trips.json
WiredTiger.lock      collection-2--4070724485692793250.wt index-3--4070724485692793250.wt posts.json tweets.json
WiredTiger.turtle    collection-4--4070724485692793250.wt index-5--4070724485692793250.wt routes.json zips.json
WiredTiger.wt        companies.json                  index-6--4070724485692793250.wt sizeStorer.wt
WiredTigerHS.wt     diagnostic.data               inspections.json storage.bson
_mdb_catalog.wt     grades.json                   journal stories.json
```

6)Here we import our dataset using the shown command. For e.g. companies.json is json collection which is mapped to 'companies' file in container. And training is the name I have given to dataset.

```
root@082e182b4249:/home/database# mongoimport companies.json -d training -c companies --drop
2022-03-19T17:04:58.176+0000    connected to: mongodb://localhost/
2022-03-19T17:04:58.176+0000    dropping: training.companies
2022-03-19T17:05:00.778+0000    9500 document(s) imported successfully. 0 document(s) failed to import.
root@082e182b4249:/home/database# mongoimport grades.json -d training -c grades --drop
2022-03-19T17:05:54.744+0000    connected to: mongodb://localhost/
2022-03-19T17:05:54.745+0000    dropping: training.grades
[#####] training.grades      26.0MB/36.1MB (72.0%)
2022-03-19T17:05:57.745+0000    [#####] training.grades      36.1MB/36.1MB (100.0%)
2022-03-19T17:05:58.839+0000    10000 document(s) imported successfully. 0 document(s) failed to import.
root@082e182b4249:/home/database# mongoimport inspections.json -d training -c inspections --drop
2022-03-19T17:06:52.100+0000    connected to: mongodb://localhost/
2022-03-19T17:06:52.410+0000    dropping: training.inspections
2022-03-19T17:06:55.317+0000    90047 document(s) imported successfully. 0 document(s) failed to import.
root@082e182b4249:/home/database# mongoimport posts.json -d training -c posts --drop
2022-03-19T17:08:04.758+0000    connected to: mongodb://localhost/
2022-03-19T17:08:04.758+0000    dropping: training.posts
2022-03-19T17:08:05.268+0000    500 document(s) imported successfully. 0 document(s) failed to import.
root@082e182b4249:/home/database# mongoimport routes.json -d training -c routes --drop
2022-03-19T17:08:33.189+0000    connected to: mongodb://localhost/
2022-03-19T17:08:33.190+0000    dropping: training.routes
2022-03-19T17:08:35.325+0000    66985 document(s) imported successfully. 0 document(s) failed to import.
root@082e182b4249:/home/database# mongoimport stories.json -d training -c stories --drop
2022-03-19T17:09:11.472+0000    connected to: mongodb://localhost/
2022-03-19T17:09:11.472+0000    dropping: training.stories
2022-03-19T17:09:33.517+0000    9842 document(s) imported successfully. 0 document(s) failed to import.
root@082e182b4249:/home/database# mongoimport trips.json -d training -c trips --drop
2022-03-19T17:09:33.167+0000    connected to: mongodb://localhost/
2022-03-19T17:09:33.168+0000    dropping: training.trips
2022-03-19T17:09:33.832+0000    10000 document(s) imported successfully. 0 document(s) failed to import.
root@082e182b4249:/home/database# mongoimport tweets.json -d training -c tweets --drop
2022-03-19T17:10:03.658+0000    connected to: mongodb://localhost/
2022-03-19T17:10:03.659+0000    dropping: training.tweets
[#####] training.tweets      40.3MB/47.6MB (84.7%)
2022-03-19T17:10:07.202+0000    [#####] training.tweets      47.6MB/47.6MB (100.0%)
2022-03-19T17:10:07.202+0000    24832 document(s) imported successfully. 0 document(s) failed to import.
root@082e182b4249:/home/database# mongoimport zips.json -d training -c zips --drop
2022-03-19T17:10:31.183+0000    connected to: mongodb://localhost/
2022-03-19T17:10:31.184+0000    dropping: training.zips
2022-03-19T17:10:32.086+0000    29470 document(s) imported successfully. 0 document(s) failed to import.
```

7)Now, we run mongo in order to connect to server using mongo client:

```
root@082e182b4249:/home/database# mongo
MongoDB shell version v5.0.6
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("e18c7607-cb85-e47a-a7fa-f927a11d2900") }
MongoDB server version: 5.0.6
=====
Warning: the "mongo" shell has been superseded by "mongosh",
which delivers improved usability and compatibility.The "mongo" shell has been deprecated and will be removed in
an upcoming release.
For installation instructions, see
https://docs.mongodb.com/mongodb-shell/install/
=====
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
https://docs.mongodb.com/
Questions? Try the MongoDB Developer Community Forums
https://community.mongodb.com
...
The server generated these startup warnings when booting:
2022-03-19T16:58:23.169+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted
2022-03-19T16:58:23.169+00:00: /sys/kernel/mm/transparent_hugepage/enabled is 'always'. We suggest setting it to 'never'
...
...
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).
The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.
To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
```

8)To see databases in mongo, we us show dbs command. Our training file is present.

```
> show dbs
admin      0.000GB
config     0.000GB
local      0.000GB
training   0.060GB
...
```

9)We switch to training in order to start querying on collections:

```
> use training
switched to db training
```

10)show collections to see collection names:

```
> show collections
companies
grades
inspections
posts
routes
stories
trips
tweets
zips
```

11)To see some stats of training, we use db.stats():

```
> db.stats()
{
    "db" : "training",
    "collections" : 9,
    "views" : 0,
    "objects" : 331176,
    "avgObjSize" : 513.3017368408339,
    "dataSize" : 169993216,
    "storageSize" : 61198336,
    "indexes" : 9,
    "indexSize" : 3547136,
    "totalSize" : 64745472,
    "scaleFactor" : 1,
    "fsUsedSize" : 116897951744,
    "fsTotalSize" : 255412363264,
    "ok" : 1
}
```

12)use this command to see dataset name and collection names.

```
> db.getName()
training
> db.getCollectionNames()
[
    "companies",
    "grades",
    "inspections",
    "posts",
    "routes",
    "stories",
    "trips",
    "tweets",
    "zips"
]
```

13)To see first document of companies collection:

14) To see documents with student_id=0 and in order to not fetch other fields, we write _id=0:

15)inserting one document in routes.json

```
> db.routes.insertOne({"airline": {"id": {"$numberInt": "13910"}, "name": "Second Fi", "alias": "K9", "iata": "AXF"}, "src_airport": "GHU", "dst_airport": "YFD", "codeshare": "", "stops": {"$numberInt": "0"}, "airplane": "M04"});
{
    "acknowledged" : true,
    "insertedId" : ObjectId("62372193e1713092d609f313")
}
```

16) In order to insert more than 1 document, we use insertMany:

```
> db.zips.insertMany([
... {"city":"RIOS","zip":"83495","loc":{"y":{"$numberDouble":"83.456335"}, "x":{"$numberDouble":"45.453783"}}, "pop":{"$numberInt":8734}, "state":"AL"},
... {"city":"GHYI","zip":"84350","loc":{"y":{"$numberDouble":"98.364653"}, "x":{"$numberDouble":"85.384960"}}, "pop":{"$numberInt":4598}, "state":"AL"}]);
{
  "acknowledged" : true,
  "insertedIds" : [
    ObjectId("62372480e1713092d609f314"),
    ObjectId("62372480e1713092d609f315")
]
```

17) Inserted data see ISODate() in inspections.json:

```
> db.inspections.insert({"date": ISODate()});
WriteResult({ "nInserted" : 1 })
```

18) To see all documents from trips.json:

```
> db.trips.find();
{ "_id" : ObjectId("572bb8222b288919b68abf5a"), "tripduration" : 379, "start station id" : 476, "start station name" : "E 31 St & 3 Ave", "end station id" : 498, "end station name" : "Broadway & W 32 St", "bikeid" : 17827, "usertype" : "Subscriber", "birth year" : 1969, "gender" : 1, "start station location" : { "type" : "Point", "coordinates" : [ -73.97966069, 40.74394314 ] }, "end station location" : { "type" : "Point", "coordinates" : [ -73.98808416, 40.74854862 ] }, "start time" : ISODate("2016-01-01T00:07:04Z") },
{ "_id" : ObjectId("572bb8222b288919b68abf5d"), "tripduration" : 923, "start station id" : 268, "start station name" : "Howard St & Centre St", "end station id" : 3002, "end station name" : "South End Ave & Liberty St", "bikeid" : 22285, "usertype" : "Subscriber", "birth year" : 1958, "gender" : 1, "start station location" : { "type" : "Point", "coordinates" : [ -73.99973333, 40.711512 ] }, "end station location" : { "type" : "Point", "coordinates" : [ -74.015756, 40.711512 ] }, "start time" : ISODate("2016-01-01T00:41Z"), "stop time" : ISODate("2016-01-01T00:16:04Z") },
{ "_id" : ObjectId("572bb8222b288919b68abf5e"), "tripduration" : 332, "start station id" : 3104, "start station name" : "Broadway & N 7 St", "end station id" : 389, "end station name" : "Broadway & Berry St", "bikeid" : 15788, "usertype" : "Subscriber", "birth year" : 1984, "gender" : 1, "start station location" : { "type" : "Point", "coordinates" : [ -73.96150225, 40.72057 ] }
```

19) To see result in a nice format, use pretty() with find():

```
> db.trips.find().pretty();
{
  "_id" : ObjectId("572bb8222b288919b68abf5a"),
  "tripduration" : 379,
  "start station id" : 476,
  "start station name" : "E 31 St & 3 Ave",
  "end station id" : 498,
  "end station name" : "Broadway & W 32 St",
  "bikeid" : 17827,
  "usertype" : "Subscriber",
  "birth year" : 1969,
  "gender" : 1,
  "start station location" : {
    "type" : "Point",
    "coordinates" : [
      -73.97966069,
      40.74394314
    ]
  },
  "end station location" : {
    "type" : "Point",
    "coordinates" : [
      -73.98808416,
      40.74854862
    ]
  },
  "start time" : ISODate("2016-01-01T00:00:45Z"),
  "stop time" : ISODate("2016-01-01T00:07:04Z")
}
{
  "_id" : ObjectId("572bb8222b288919b68abf5d"),
  "tripduration" : 923,
  "start station id" : 268,
  "start station name" : "Howard St & Centre St",
  "end station id" : 3002,
  "end station name" : "South End Ave & Liberty St",
```

20)Fetching all distinct names from companies.json:

```
> db.companies.distinct("name");
"Adjust",
"Adjustables",
"Adlib Software",
"Admeld",
"Adneuron",
"AdoTube",
"Adooga",
"Adoos",
"Adoroi",
"Adreel",
"Ads by Athena",
"Adscape",
"Adteractive",
"Advaliant",
"Advameg",
"Advanced Control Components",
"Advanced Millennium Technologies",
"Advanced Power Projects",
"Advanced Virtual Surveillance",
"Advanta Bank",
"Advantia VoIP Systems",
"Advantiv",
"Advasense",
"Advent Software",
"Adventure Areas",
"Adventure Engine",
"AdventureDrop",
"Adverfix",
"Advertising Solutions",
"Advisor",
"Advisor Marketing ServicesGroup",
"Adwest Systems"
```

21)To see src_airport field where dst_airport is "UIO". We write _id:0 to not fetch whole document.

```
> db.routes.find({"dst_airport": "UIO"}, {"src_airport:1, _id:0});
{"src_airport" : "GYE"}
{"src_airport" : "BOG"}
{"src_airport" : "OCC"}
{"src_airport" : "GYE"}
{"src_airport" : "MIA"}
{"src_airport" : "AMS"}
{"src_airport" : "MEX"}
{"src_airport" : "BOG"}
{"src_airport" : "LIM"}
{"src_airport" : "SAL"}
{"src_airport" : "BOG"}
{"src_airport" : "PTY"}
{"src_airport" : "ATL"}
{"src_airport" : "BOG"}
{"src_airport" : "CUE"}
{"src_airport" : "ESM"}
{"src_airport" : "ETR"}
{"src_airport" : "GRU"}
{"src_airport" : "GYE"}
{"src_airport" : "HAV"}
Type "it" for more
```

22)Counting documents in trips.json:

```
> db.trips.estimatedDocumentCount();
10000
```

23)Counting documents where filed usertype has value “subscriber”:

```
> db.trips.count({"usertype":"Subscriber"});
8011
```

Comparison:

24)Fetching “sector” and “result” value where date is 25th feb:

```
> db.inspections.find({"date":"Feb 25 2015"},{sector:1, result:1, _id:0});
{
  "result" : "No Violation Issued", "sector" : "Misc Non-Food Retail - 817"
}
{
  "result" : "No Violation Issued", "sector" : "Salons And Barbershop - 841"
}
{
  "result" : "No Violation Issued", "sector" : "Cigarette Retail Dealer - 127"
}
{
  "result" : "No Violation Issued", "sector" : "Mobile Food Vendor - 881"
}
{
  "result" : "Pass", "sector" : "Scale Dealer/Repairer - 107"
}
{
  "result" : "Violation Issued", "sector" : "Tax Preparers - 891"
}
{
  "result" : "Pass", "sector" : "Pedicab Business - 130"
}
{
  "result" : "Pass", "sector" : "Mobile Food Vendor - 881"
}
{
  "result" : "No Violation Issued", "sector" : "Home Improvement Contractor - 100"
}
{
  "result" : "Violation Issued", "sector" : "Fuel Oil Dealer - 814"
}
{
  "result" : "Fail", "sector" : "Fuel Oil Dealer - 814"
}
{
  "result" : "Fail", "sector" : "Fuel Oil Dealer - 814"
}
{
  "result" : "No Violation Issued", "sector" : "Cigarette Retail Dealer - 127"
}
{
  "result" : "No Violation Issued", "sector" : "Salons And Barbershop - 841"
}
{
  "result" : "Pass", "sector" : "Gas Station-Retail - 815"
}
{
  "result" : "Out of Business", "sector" : "Salons And Barbershop - 841"
}
{
  "result" : "No Evidence of Activity", "sector" : "Secondhand Dealer [General] - 006"
}
{
  "result" : "Violation Issued", "sector" : "Cigarette Retail Dealer - 127"
}
{
  "result" : "Violation Issued", "sector" : "Drug Store Retail - 810"
}
{
  "result" : "No Violation Issued", "sector" : "Cigarette Retail Dealer - 127"
}
Type "it" for more
>
```

25)Fetching “tripduration” where “birth year” is less than 1963:

```
> db.trips.find({"birth year":{$lt:"1963"}}, {"tripduration":1, _id:0});
{
  "tripduration" : 858
}
{
  "tripduration" : 839
}
{
  "tripduration" : 110
}
{
  "tripduration" : 2023
}
{
  "tripduration" : 1236
}
{
  "tripduration" : 1684
}
{
  "tripduration" : 1167
}
{
  "tripduration" : 1169
}
{
  "tripduration" : 1485
}
{
  "tripduration" : 1488
}
{
  "tripduration" : 1364
}
{
  "tripduration" : 1358
}
{
  "tripduration" : 1325
}
{
  "tripduration" : 1315
}
{
  "tripduration" : 1475
}
{
  "tripduration" : 1474
}
{
  "tripduration" : 2426
}
{
  "tripduration" : 1437
}
{
  "tripduration" : 950
}
{
  "tripduration" : 941
}
Type "it" for more
>
```

26)Fetching name where permalink is facebook:

```
> db.companies.find({"permalink":"facebook"}, {"name:1, _id:0});
{ "name" : "Facebook" }
```

27)Fetching “tripduration” where birth year is not equal to 2004:

```
> db.trips.find({"birth_year":{$ne:"2004"}}, {"tripduration":1, _id:0});
{ "tripduration" : 379 }
{ "tripduration" : 923 }
{ "tripduration" : 332 }
{ "tripduration" : 2321 }
{ "tripduration" : 889 }
{ "tripduration" : 694 }
{ "tripduration" : 1376 }
{ "tripduration" : 1770 }
{ "tripduration" : 589 }
{ "tripduration" : 1480 }
{ "tripduration" : 525 }
{ "tripduration" : 240 }
{ "tripduration" : 667 }
{ "tripduration" : 539 }
{ "tripduration" : 1245 }
{ "tripduration" : 615 }
{ "tripduration" : 858 }
{ "tripduration" : 479 }
{ "tripduration" : 888 }
{ "tripduration" : 839 }
Type "it" for more
```

28) Fetching gender where birth year is not in the range of 1970 and 1988 inclusive:

```
> db.trips.find({"birth_year":{$nin:["1970","1988"]}}, {"gender":1, _id:0});
{ "gender" : 1 }
{ "gender" : 2 }
{ "gender" : 1 }
{ "gender" : 1 }
{ "gender" : 1 }
{ "gender" : 2 }
{ "gender" : 1 }
{ "gender" : 0 }
{ "gender" : 1 }
{ "gender" : 1 }
{ "gender" : 0 }
{ "gender" : 1 }
{ "gender" : 1 }
{ "gender" : 0 }
Type "it" for more
>
```

30) Fetching gender where birth year is not in this given range:

```
> db.trips.find({"birth_year":{$nin:["1970","2004"]}}, {"gender":1, _id:0});
{ "gender" : 1 }
{ "gender" : 2 }
{ "gender" : 1 }
{ "gender" : 2 }
{ "gender" : 1 }
{ "gender" : 0 }
{ "gender" : 1 }
{ "gender" : 1 }
{ "gender" : 0 }
Type "it" for more
```

31) Fetching documents where name does not equal to “CBS”:

Very Large Output:

```
> db.companies.find({name:{'$not': { '$eq': "CBS"}}});  
[{"_id": ObjectId("52cdefc4babbbd675297d8a"), "name": "Wetpaint", "permalink": "abc2", "crunchbase_url": "http://www.crunchbase.com/company/wetpaint", "home_page_url": "http://wetpaint-inc.com", "blog_url": "http://digitalquarters.net/", "blog_feed_url": "http://digitalquarters.net/feed/", "twitter_username": "BachelorWetpaint", "category_code": "web", "number_of_employees": 47, "founded_year": 2005, "founded_month": 10, "founded_day": 17, "deadpooled_year": 1, "tag_list": ["wiki", "seattle", "elowitz", "media-industry", "media-platform", "social-distribution-system"], "alias_list": "", "email_address": "info@wetpaint.com", "phone_number": "+206.859.6300", "overview": "<p>Wetpaint is a technology platform company that uses its proprietary state-of-the-art technology and expertise in social media to build and monetize audiences for digital publishers. Wetpaint's own online property, Wetpaint Entertainment, an entertainment news site that attracts more than 12 million unique visitors monthly and has over 2 million Facebook fans, is a proof point to the company's success in building and engaging audiences. Media companies can license Wetpaint's platform which includes a dynamic playback tailored to their individual needs and comprehensive training. Founded by Internet pioneer Ben Elowitz, and with offices in New York and Seattle, Wetpaint is backed by Accel Partners, the investors behind Facebook.</p>", "image": {"available_sizes": [ [ [ 150, 75 ], "assets/images/resized/0000/3604/3604v14-max-150x150.jpg" ], [ [ 250, 125 ], "assets/images/resized/0000/3604/3604v14-max-150x150.jpg" ] ], "products": [ { "name": "Wikison Wetpaint", "permalink": "wetpaint-wiki" }, { "name": "Wetpaint Social Distribution System", "permalink": "wetpaint-social-distribution-system" } ], "relationships": [ { "is": false, "title": "Co-Founder and VP, Social and Audience Development", "person": { "first_name": "Michael", "last_name": "Howell", "permalink": "michael-howell" } }, { "is": false, "title": "Co-Founder/CEO/Board of Directors", "person": { "first_name": "Ben", "last_name": "Elowitz", "permalink": "ben-elowitz" } }, { "is": false, "title": "COO/Board of Directors", "person": { "first_name": "Rob", "last_name": "Grady", "permalink": "rob-grady" } }, { "is": false, "title": "SVP, Strategy and Business Development", "person": { "first_name": "Chris", "last_name": "Kollas", "permalink": "chris-kollas" } }, { "is": false, "title": "Board", "person": { "first_name": "Theresa", "last_name": "Ranzetta", "permalink": "theresa-ranzetta" } }, { "is": false, "title": "Board Member", "person": { "first_name": "Gus", "last_name": "Tal", "permalink": "gus-tal" } }, { "is": false, "title": "Board", "person": { "first_name": "Len", "last_name": "Jordan", "permalink": "len-jordan" } }, { "is": false, "title": "Board", "person": { "first_name": "Jordan", "last_name": "Ranzetta", "permalink": "jordan-ranzetta" } } ] } ]
```

32)Fetching permalink where name is not equal to Yahoo and is equal to Facebook:

```
> db.companies.find({$or: [{"name":{$eq: "Yahoo!"}}, {"name":{$eq: "Facebook"}]}]}, {permalink:1, _id:0});  
{ "permalink" : "facebook" }  
{ "permalink" : "yahoo" }  
>
```

33) Fetching crunbase url where name is not equal to “CBS” and permalink is twitter;

```
> db.companies.find({$and: [{"name":{$not: {$eq: "CBS"} }}, {"permalink":"twitter"}]}, {crunchbase_url:1, _id:0});  
{ "crunchbase_url" : "http://www.crunchbase.com/company/twitter" }
```

34)Fetching src_airport where neither airline is 470 nor name is Helvetic Airways:

```
> db.routes.find({$nor: [{"airline": "470"}, {"name": "Helvetic Airways"}]}, {src_airport:1, _id:0});  
[{"src_airport": "DME"}, {"src_airport": "CEK"}, {"src_airport": "ASF"}, {"src_airport": "CEK"}, {"src_airport": "DME"}, {"src_airport": "ASF"}, {"src_airport": "DME"}, {"src_airport": "AER"}, {"src_airport": "DME"}, {"src_airport": "GYD"}, {"src_airport": "KGD"}, {"src_airport": "EGO"}, {"src_airport": "KZN"}, {"src_airport": "KZN"}, {"src_airport": "KZN"}, {"src_airport": "KZN"}, {"src_airport": "KZN"}, {"src_airport": "LED"}, {"src_airport": "LED"}  
Type "it" for more  
>
```

35)Fetching business_name where we use 2 arrays to be included but within them is a choice:

```
> db.inspections.find({$and: [{$or: [{"result": "No Violation Issued"}, {"result": "Fail"}]}, {$or: [{"date": "Dec 23 2015"}, {"business_name": "BUILDING TO SERVE INC."}]}]});  
[{"_id": ObjectId("56d61033a378eccde8a83560"), "id": "1044-2016-ENFO", "certificate_number": 9318079, "business_name": "AZMY KIROLES", "date": "Dec 23 2015", "result": "No Violation Issued", "sector": "Mobile Food Vendor - 881", "address": {"city": "JERSEY CITY", "zip": 7306, "street": "VROOM ST", "number": 160}}, {"_id": ObjectId("56d61033a378eccde8a83565"), "id": "1048-2016-ENFO", "certificate_number": 9319054, "business_name": "QUIRINO MORALES", "date": "Dec 3 2015", "result": "No Violation Issued", "sector": "Mobile Food Vendor - 881", "address": {"city": "EAST ELMHURST", "zip": 11370, "street": "80TH ST", "number": 3216}}, {"_id": ObjectId("56d61033a378eccde8a83578"), "id": "1112-2016-ENFO", "certificate_number": 9319053, "business_name": "SAMER EKAD", "date": "Dec 23 2015", "result": "No Violation Issued", "sector": "Mobile Food Vendor - 881", "address": {"city": "BROOKLYN", "zip": 11209, "street": "77TH ST", "number": 571}}, {"_id": ObjectId("56d61033a378eccde8a83572"), "id": "1085-2016-ENFO", "certificate_number": 9319055, "business_name": "QADIR AHMAD", "date": "Dec 23 2015", "result": "No Violation Issued", "sector": "Mobile Food Vendor - 881", "address": {"city": "BROOKLYN", "zip": 11218, "street": "DORCHESTER RD", "number": 1107}}, {"_id": ObjectId("56d61033a378eccde8a83625"), "id": "1642-2016-ENFO", "certificate_number": 9318938, "business_name": "MOHAMED MOHAMED", "date": "Dec 23 2015", "result": "No Violation Issued", "sector": "Mobile Food Vendor - 881", "address": {"city": "BROOKLYN", "zip": 11209, "street": "95TH ST", "number": 450}}, {"_id": ObjectId("56d61033a378eccde8a83db2"), "id": "648-2016-ENFO", "certificate_number": 9318937, "business_name": "ABDELKADER BOUTALHA", "date": "Dec 23 2015", "result": "No Violation Issued", "sector": "Mobile Food Vendor - 881", "address": {"city": "BRONX", "zip": 10472, "street": "E 172ND ST", "number": 1509}}, {"_id": ObjectId("56d61033a378eccde8a84033"), "id": "753-2016-ENFO", "certificate_number": 9310979, "business_name": "THE QUEENS WU CORP.", "date": "Dec 23 2015", "result": "No Violation Issued", "sector": "Laundry - 064", "address": {"city": "LITTLE NECK", "zip": 11362, "street": "NORTHERN BLVD", "number": 24825}}, {"_id": ObjectId("56d61033a378eccde8a841dc"), "id": "1287-2016-ENFO", "certificate_number": 9323992, "business_name": "C R CHAMBERLAIN INC", "date": "Dec 23 2015", "result": "No Violation Issued", "sector": "Laundry Jobber - 066", "address": {"city": "NEW YORK", "zip": 10019, "street": "7TH AVE", "number": 912}}, {"_id": ObjectId("56d61033a378eccde8a843ea"), "id": "821-2016-ENFO", "certificate_number": 9311320, "business_name": "218 BAYSIDE OPERATING, LLC", "date": "Dec 23 2015", "result": "No Violation Issued", "sector": "Cigarette Retail Dealer - 127", "address": {"city": "BAYSIDE", "zip": 11361, "street": "10TH AVENUE", "number": 218}}  
Type "it" for more  
>
```

36)same as above

```
> db.inspections.find({$and: [{$or: [{"result": "No Violation Issued"}, {"result": "Fail"}]}, {$or: [{"date": "Dec 23 2015"}, {"business_name": "BUILDING TO SERVE INC."}]}]}, {"certificate_number":1, _id:0});  
[{"certificate_number": 9318079}, {"certificate_number": 9319054}, {"certificate_number": 9319053}, {"certificate_number": 9319055}, {"certificate_number": 9318938}, {"certificate_number": 9318937}, {"certificate_number": 9310979}, {"certificate_number": 9323992}, {"certificate_number": 9311320}, {"certificate_number": 9319038}, {"certificate_number": 9318143}, {"certificate_number": 9311755}, {"certificate_number": 9317469}, {"certificate_number": 9311751}, {"certificate_number": 50065617}, {"certificate_number": 9322023}, {"certificate_number": 9308442}, {"certificate_number": 50065618}, {"certificate_number": 9311747}, {"certificate_number": 9318144}  
Type "it" for more  
>
```

37)Fetching city where zip is of type 2

```
> db.zips.find({"zip":{$type:2}}, {city:1, _id:0});  
{ "city" : "BAILEYTON" }  
{ "city" : "HUEYTOWN" }  
{ "city" : "ALPINE" }  
{ "city" : "BESSEMER" }  
{ "city" : "BRIERFIELD" }  
{ "city" : "BREMEN" }  
{ "city" : "BLOUNTSVILLE" }  
{ "city" : "ADGER" }  
{ "city" : "CALERA" }  
{ "city" : "ACMAR" }  
{ "city" : "CENTREVILLE" }  
{ "city" : "ARAB" }  
{ "city" : "CHELSEA" }  
{ "city" : "KEYSTONE" }  
{ "city" : "NEW SITE" }  
{ "city" : "BRENT" }  
{ "city" : "ADAMSVILLE" }  
{ "city" : "CRANE HILL" }  
{ "city" : "CULLMAN" }  
{ "city" : "CLEVELAND" }  
Type "it" for more  
>
```

38)Fetching city where zip is in given data type

```
> db.zips.find({"zip":{$type:"string"}}, {city:1, _id:0});  
{ "city" : "BAILEYTON" }  
{ "city" : "HUEYTOWN" }  
{ "city" : "ALPINE" }  
{ "city" : "BESSEMER" }  
{ "city" : "BRIERFIELD" }  
{ "city" : "BREMEN" }  
{ "city" : "BLOUNTSVILLE" }  
{ "city" : "ADGER" }  
{ "city" : "CALERA" }  
{ "city" : "ACMAR" }  
{ "city" : "CENTREVILLE" }  
{ "city" : "ARAB" }  
{ "city" : "CHELSEA" }  
{ "city" : "KEYSTONE" }  
{ "city" : "NEW SITE" }  
{ "city" : "BRENT" }  
{ "city" : "ADAMSVILLE" }  
{ "city" : "CRANE HILL" }  
{ "city" : "CULLMAN" }  
{ "city" : "CLEVELAND" }  
Type "it" for more  
>
```

39)Only fetch 1st 2 records

```
> db.grades.find().limit(2);
{ "_id" : ObjectId("56d5f7eb604eb380b0d8d8cf"), "student_id" : 0, "scores" : [ { "type" : "exam", "score" : 91.97520018439039 }, { "type" : "quiz", "score" : 95.80410375967175 }, { "type" : "homework", "score" : 89.62485475572984 }, { "type" : "homework", "score" : 51.621532832724846 } ], "class_id" : 350 }
{ "_id" : ObjectId("56d5f7eb604eb380b0d8d8d2"), "student_id" : 0, "scores" : [ { "type" : "exam", "score" : 41.25131199553351 }, { "type" : "quiz", "score" : 91.7351500084582 }, { "type" : "homework", "score" : 24.198828271948415 }, { "type" : "homework", "score" : 79.77471812670814 } ], "class_id" : 391 }
```

40)using aggregate function. Group by user to get the total sum of followers_count:

```
> db.tweets.aggregate([ {$group: {_id:"$user", followers_count:{$sum:1}}}] );
{ "_id" : { "friends_count" : 214, "profile_sidebar_fill_color" : "7AC3E", "location" : "Los Angeles", "verified" : false, "follow_request_sent" : null, "favourites_count" : 1, "profile_sidebar_border_color" : "65B0DA", "profile_image_url" : "http://a0.twimg.com/profile_images/1114154584/buttanormal.jpg", "geo_enabled" : false, "created_at" : "Mon Sep 21 00:24:26 +0000 2009", "description" : "Every1 knows me as Shea/Coach, I'm a High School girls Bball & VBall coach @ The very lovely Windward School in LA, CA I love my life & all my kids!", "time_zone" : "Pacific Time (US & Canada)", "url" : "http://courtcred.com/", "screen_name" : "SheaButtaCoco", "notifications" : null, "profile_background_color" : "642DB", "listed_count" : 6, "lang" : "en", "profile_background_image_url" : "http://a3.twimg.com/profile_background_images/128565937/courtcred.jpg", "statuses_count" : 5830, "following" : null, "profile_text_color" : "3D1957", "protected" : false, "show_all_inline_media" : false, "profile_background_tile" : true, "name" : "Le'Sheala Dawson ", "contributors_enabled" : false, "profile_link_color" : "F0000", "followers_count" : 380, "id" : 75913689, "profile_use_background_image" : true, "utc_offset" : -28800, "followers_count" : 2 }
{ "id" : { "friends_count" : 79, "profile_sidebar_fill_color" : "efefef", "location" : "Em baixo da sua cama.", "verified" : false, "follow_request_sent" : null, "favourites_count" : 0, "profile_sidebar_border_color" : "eeeeee", "profile_image_url" : "http://a0.twimg.com/profile_images/1113545632/P1010063xcxz_normal.JPG", "geo_enabled" : false, "created_at" : "Thu Jan 14 22:05:14 +0000 2010", "description" : "Compacto pra que vê. Gigante pra quem anda.", "time_zone" : "Greenland", "url" : null, "screen_name" : "Iggy_online", "notifications" : null, "profile_background_color" : "131516", "listed_count" : 14, "lang" : "en", "profile_background_image_url" : "http://a1.twimg.com/profile_background_images/141029182/img-thing.jpg", "statuses_count" : 7443, "following" : null, "profile_text_color" : "333333", "protected" : false, "show_all_inline_media" : false, "profile_background_tile" : true, "name" : "Iggy ", "contributors_enabled" : false, "profile_link_color" : "009999", "followers_count" : 443, "id" : 104947670, "profile_use_background_image" : true, "utc_offset" : -10800, "followers_count" : 1 }
{ "id" : { "friends_count" : 199, "profile_sidebar_fill_color" : "f6fffd", "location" : "Provo, Utah", "verified" : false, "follow_request_sent" : null, "favourites_count" : 4, "profile_sidebar_border_color" : "ffffad", "profile_image_url" : "http://a2.twimg.com/profile_images/1103300826/asciid60_normal.PNG", "geo_enabled" : true, "created_at" : "Wed Oct 15 19:37:17 +0000 2008", "description" : "lists: /politics /local /news /sports /geek /tech /videogames /music /celebs /humor /wtf", "time_zone" : "Mountain Time (US & Canada)", "url" : null, "screen_name" : "BillBirrell", "notifications" : null, "profile_background_color" : "FFF04D", "listed_count" : 0, "lang" : "en", "profile_background_image_url" : "http://s.twimg.com/a/12833518/images/themes/theme19/bg.gif", "statuses_count" : 171, "following" : null, "profile_text_color" : "333333", "protected" : false, "show_all_in
```

41)same explanation:

```
> db.tweets.aggregate([ {$group: {_id:"$user", retweeted:{$sum:1}}}] );
{ "_id" : { "friends_count" : 90, "profile_sidebar_fill_color" : "DDEEF6", "location" : "Minas Gerais :)", "verified" : false, "follow_request_sent" : null, "favourites_count" : 0, "profile_sidebar_border_color" : "C0DEED", "profile_image_url" : "http://a3.twimg.com/profile_images/1113925879/100_3425-crop_normal.jpg", "geo_enabled" : true, "created_at" : "Sun Jan 24 18:38:42 +0000 2010", "description" : "", "time_zone" : "Greenland", "url" : "http://www.orkut.com.br/Main#Profile?uid=5624452063611616555", "screen_name" : "naniliikaah", "notifications" : null, "profile_background_color" : "C0DEED", "listed_count" : 0, "lang" : "es", "profile_background_image_url" : "http://a1.twimg.com/profile_background_images/69767650/Naanika.Cunha.jpg", "statuses_count" : 509, "following" : null, "profile_text_color" : "333333", "protected" : false, "show_all_inline_media" : false, "profile_background_tile" : true, "name" : "Caroline Izabela", "contributors_enabled" : false, "profile_link_color" : "0084B4", "followers_count" : 33, "id" : 108066489, "profile_use_background_image" : true, "utc_offset" : -10800, "retweeted" : 1 }
{ "id" : { "friends_count" : 168, "profile_sidebar_fill_color" : "0fff92", "location" : "", "verified" : false, "follow_request_sent" : null, "favourites_count" : 1, "profile_sidebar_border_color" : "87bc44", "profile_image_url" : "http://a1.twimg.com/profile_images/1088994741/topsteward_normal.jpg", "geo_enabled" : false, "created_at" : "Sun Oct 25 19:48:45 +0000 2009", "description" : "Luton town fan dad of two lagen man ", "time_zone" : null, "url" : "http://lutontown", "screen_name" : "topsteward", "notifications" : null, "profile_background_color" : "9ae4e8", "listed_count" : 7, "lang" : "en", "profile_background_image_url" : "http://s.twimg.com/a/1282351897/images/themes/theme1/bg.png", "statuses_count" : 2515, "following" : null, "profile_text_color" : "000000", "protected" : false, "show_all_inline_media" : false, "profile_background_tile" : false, "name" : "Rob", "contributors_enabled" : false, "profile_link_color" : "0000ff", "followers_count" : 109, "id" : 85150796, "profile_use_background_image" : true, "utc_offset" : null }, "retweeted" : 1 }
{ "id" : { "friends_count" : 136, "profile_sidebar_fill_color" : "252429", "location" : "ÜT: 34.138563,-118.268855", "verified" : false, "follow_request_sent" : null, "favourites_count" : 1, "profile_sidebar_border_color" : "181A1E", "profile_image_url" : "http://a3.twimg.com/profile_images/306103595/IMG00008-20090604-0118_normal.jpg", "geo_enabled" : true, "created_at" : "Fri Jul 03 15:15:05 +0000 2009", "description" : "in there like swim wear ", "time_zone" : "Alaska", "url" : "http://theeproactive.blogspot.com", "screen_name" : "pacmanjoey", "notifications" : null, "profile_background_color" : "1A1B1F", "listed_count" : 3, "lang" : "en", "profile_background_image_url" : "http://a1.twimg.com/profile_background_images/135071260/product_thumb.jpg", "statuses_count" : 391, "following" : null, "profile_text_color" : "666666", "protected" : false, "show_all_inline_media" : false, "profile_background_tile" : true, "name" : "joseph madison ", "contributors_enabled" : false, "profile_link_color" : "2FC2EF", "followers_count" : 43, "id" : 53419218, "profile_use_background_image" : true, "utc_offset" : -32400, "retweeted" : 1 }
{ "id" : { "friends_count" : 99, "profile_sidebar_fill_color" : "e0ff92", "location" : "Bucks County", "verified" : false, "follow_request_sent" : null, "favourites_count" : 1, "profile_sidebar_border_color" : "87bc44", "profile_image_url" : "http://a2.twimg.com/profile_images/1113990538/ijustdontgiveafuckfortwitter_normal.jpg", "geo_enabled" : false, "created_at" : "Fri Sep 18 13:41:54 +0000 2009", "description" : "Not everybody trusts paintings but people believe photographs. Ansel Adams ", "time_zone" : null, "url" : "http://www.adrienneparisphotographs.com", "screen_name" : "ApaisPhoto", "notifications" : null, "profile_background_color" : "9ae4e8", "listed_count" : 0, "lang" : "en", "profile_background_image_url" : "http://
```

42)Grouping by id. Summing all ids:

```
> db.tweets.aggregate([ {$group: {_id:"$oid", followers_count:{$sum:1}}}]);
{ "_id" : null, "followers_count" : 24832 }
```

43)Summing all population for all cities:

```
> db.zips.aggregate([ {$group: {_id:"city", pop:{$sum:1}}}]);
{ "_id" : "city", "pop" : 29476 }
```

44)Query an Array: fetching using an array

```
> db.companies.find({"products": {"name": "Slacker Web Player", "permalink": "slacker-web-player"}, {"name": "Slacker Desktop Radio", "permalink": "slacker-desktop-radio"}, {"name": "Slacker Portable Player", "permalink": "slacker-portable-player"}});
{
  "_id": ObjectId("52cdef7c4bab8bd675297d98"),
  "name": "Slacker",
  "permalink": "slacker",
  "crunchbase_url": "http://www.crunchbase.com/company/slacker",
  "homepage_url": "http://www.slacker.com",
  "blog_url": "",
  "blog_feed_url": "",
  "twitter_username": "SlackerRadio",
  "category_code": "music",
  "number_of_employees": null,
  "founded_year": 2006,
  "founded_month": null,
  "founded_day": null,
  "deadpooled_year": null,
  "deadpooled_month": null,
  "deadpooled_day": null,
  "deadpooled_url": null,
  "tag_list": "",
  "alias_list": "",
  "email_address": "support@slacker.com",
  "phone_number": "",
  "description": "Music Personalization Suite",
  "created_at": "Mon Jun 04 08:45:22 UTC 2007",
  "updated_at": "Sun Mar 17 08:05:45 UTC 2013",
  "overview": "KpxSlacker is the most complete music service on Earth. The company delivers free and subscription-based access to millions of songs and hundreds of expert-programmed stations, plus news, sports and talk, on the web, mobile devices, in-car infotainment systems and consumer electronics platforms. By blending cutting-edge analytics with a team of expert music curators and content programmers, Slacker delivers a seamless music discovery experience and uniquely personalized entertainment. Slacker was founded in 2006 and is headquartered in San Diego with offices in Palo Alto, CA and New York City. For more information about Slacker, or to listen for free, visit www.slacker.com/p>",
  "image": {
    "available_sizes": [
      [ [ 150, 61 ], "assets/images/resized/0001/7589/17589v4-max-150x150.jpg" ],
      [ [ 250, 101 ], "assets/images/resized/0001/7589/17589v4-max-250x250.jpg" ],
      [ [ 450, 183 ], "assets/images/resized/0001/7589/17589v4-max-450x450.jpg" ]
    ],
    "attribution": null
  },
  "products": [
    {
      "name": "Slacker Web Player",
      "permalink": "slacker-web-player"
    },
    {
      "name": "Slacker Desktop Radio",
      "permalink": "slacker-desktop-radio"
    },
    {
      "name": "Slacker Portable Player",
      "permalink": "slacker-portable-player"
    }
  ],
  "relationships": [
    {
      "is_past": false,
      "title": "President & CEO",
      "person": {
        "first_name": "Carter",
        "last_name": "Brokaw",
        "permalink": "carter-brokaw"
      }
    },
    {
      "is_past": false,
      "title": "CMO",
      "person": {
        "first_name": "Craig",
        "last_name": "Rechennacher",
        "permalink": "craig-rechennacher"
      }
    },
    {
      "is_past": false,
      "title": "VP of Engineering",
      "person": {
        "first_name": "Brad",
        "last_name": "Kindig",
        "permalink": "brad-kindig"
      }
    }
  ],
  "is_past": false,
  "title": "SVP Strategic Development",
  "person": {
    "first_name": "Jack",
    "last_name": "Rist",
    "permalink": "jack-rist"
  }
}
```

45)Fetching certificate_number and business_name of given city and sorting as per certificate_number then business:

```
> db.inspections.find({address:{city:"BROOKLYN"}}, {certificate_number:1, business_name:1, _id:0}).sort({certificate_number:1,business_name:-1});
>
```

```
> db.trips.aggregate([{$group:{_id:"start station id", avg_trip_duration:{$avg:"$tripduration"}}}]);
{
  "_id": "start station id",
  "avg_trip_duration": 1129.9943
}
>
> db.trips.aggregate([{$group:{_id:"start station id", avg_trip_duration:{$avg:"$tripduration"}},{$project:{avg_trip_duration:1, _id:0}}}], {
  "avg_trip_duration": 1129.9943
})
>
> db.zips.find({}).sort({"city": 1, "state": -1}).skip(10).limit(3);
{
  "_id": ObjectId("5c8ecc1caa187d17ca75693"),
  "city": "23157",
  "zip": "23157",
  "loc": {
    "y": 37.340897,
    "x": 76.297135
  },
  "pop": 313,
  "state": "VA"
},
{
  "_id": ObjectId("5c8ecc1caa187d17ca75767"),
  "city": "23973",
  "zip": "23973",
  "loc": {
    "y": 36.766267,
    "x": 78.219555
  },
  "pop": 796,
  "state": "VA"
},
{
  "_id": ObjectId("5c8ecc1caa187d17ca757f3"),
  "city": "24329",
  "zip": "24329",
  "loc": {
    "y": 36.889764,
    "x": 80.827174
  },
  "pop": 159,
  "state": "VA"
}
>
```

Very Large Output:

```
> db.companies.find().readConcern("majority");
{
  "_id": ObjectId("52cdef7c4bab8bd675297d8a"),
  "name": "Wetpaint",
  "permalink": "abc2",
  "crunchbase_url": "http://www.crunchbase.com/compaint",
  "homepage_url": "http://wetpaint-inc.com",
  "blog_url": "http://digitalquarters.net/",
  "blog_feed_url": "http://digitalquarters.net",
  "twitter_username": "BachelderWetpaint",
  "category_code": "web",
  "number_of_employees": 47,
  "founded_year": 2005,
  "founded_month": 10,
  "founded_day": 17,
  "deadpooled_year": 1,
  "tag_list": "wiki, seattle, elowitz, media-industry, media-platform, social-distribution-system",
  "alias": "",
  "email_address": "info@wetpaint.com",
  "phone_number": "206.859.6300",
  "description": "Technology Platform Company",
  "created_at": "2007-05-25T06:51:27Z",
  "updated_at": "Sun Dec 08 07:15:44 UTC 2013",
  "overview": "KpxWetpaint is a technology platform company that uses proprietary state-of-the-art technology and expertise in social media to build and monetize audiences for digital publishers. Wetpaint's own online fans, is a proof point to the company's success in building and engaging audiences. Media companies can license Wetpaint's platform which is dynamic playbook tailored to their individual needs and comprehensive training. Founded by Internet pioneer Ben Elowitz, and with offices in New York and Seattle, Wetpaint is backed by Accel Partners, the investors behind Facebook.</p>",
  "image": {
    "available_sizes": [
      [ [ 150, 75 ], "assets/images/resized/0000/3604/v14-max-150x150.jpg" ],
      [ [ 250, 125 ], "assets/images/resized/0000/3604/v14-max-250x250.jpg" ],
      [ [ 450, 225 ], "assets/images/resized/0000/3604/v14-max-450x450.jpg" ]
    ],
    "attribution": null
  },
  "products": [
    {
      "name": "Wikison Wetpaint",
      "permalink": "wetpaint-wiki"
    }
  ],
  "relationships": [
    {
      "is_past": false,
      "title": "Co-Founder and VP, Social and Audience Development",
      "person": {
        "first_name": "Michael",
        "last_name": "Howell",
        "permalink": "michael-howell"
      }
    },
    {
      "is_past": false,
      "title": "Co-Founder/CEO/Board of Directors",
      "person": {
        "first_name": "Ben",
        "last_name": "Elowitz",
        "permalink": "ben-elowitz"
      }
    },
    {
      "is_past": false,
      "title": "COO/Board of Directors",
      "person": {
        "first_name": "Rob",
        "last_name": "Grady",
        "permalink": "rob-grady"
      }
    },
    {
      "is_past": false,
      "title": "SVP, Strategy and Business Development",
      "person": {
        "first_name": "Chris",
        "last_name": "Kallas",
        "permalink": "chris-kallas"
      }
    },
    {
      "is_past": false,
      "title": "Board",
      "person": {
        "first_name": "Theresa",
        "last_name": "Ranzetta",
        "permalink": "theresia-ranzetta"
      }
    },
    {
      "is_past": false,
      "title": "Board Member",
      "person": {
        "first_name": "Gus",
        "last_name": "Tai",
        "permalink": "gus-tai"
      }
    },
    {
      "is_past": false,
      "title": "Board",
      "person": {
        "first_name": "Len",
        "last_name": "Jordan",
        "permalink": "len-jordan"
      }
    }
  ],
  "is_past": false,
  "title": "Board Member",
  "person": {
    "first_name": "Tina",
    "last_name": "Fitzgerald",
    "permalink": "tina-fitzgerald"
  }
}
```

```
> db.stories.update({_id: 1}, {$pop: {"container": -1}});
WriteResult({ "nMatched": 0, "nUpserted": 0, "nModified": 0 })
```

Updated trips.json with usertype:local where birth year:1969: Here it has updated 125 document fields(usertype) with “local”:

```
> db.trips.updateMany({"birth year": 1969}, {$set: {"usertype": "local"}});
{ "acknowledged" : true, "matchedCount" : 125, "modifiedCount" : 125 }
>
```

Checking result: Here we can see that usertype is changed to “local” and there are 125 records in total that matched birth year as 1969.

```
> db.trips.find({"birth year": 1969},{usertype:1, _id:0});
{ "usertype" : "local" }
Type "it" for more
```

Updates trips.json with bikeid:14566 where birth year:1952:

```
> db.trips.update({"birth year": 1952}, {$set: {"bikeid": "14566"}});
WriteResult({ "nMatched" : 1, "nUpserted" : 0, "nModified" : 1 })
>
```

Checking the updated change where we can see that first entry is bikeid:14566 as we updated earlier:

```
> db.trips.find({"birth year": 1952}, {bikeid:1, _id:0});
{ "bikeid" : "14566" }
{ "bikeid" : 23643 }
{ "bikeid" : 24072 }
{ "bikeid" : 23961 }
{ "bikeid" : 21037 }
{ "bikeid" : 20379 }
{ "bikeid" : 18662 }
{ "bikeid" : 18331 }
{ "bikeid" : 23469 }
{ "bikeid" : 15828 }
{ "bikeid" : 16649 }
{ "bikeid" : 20541 }
{ "bikeid" : 19153 }
{ "bikeid" : 22964 }
{ "bikeid" : 21481 }
{ "bikeid" : 21481 }
{ "bikeid" : 22979 }
{ "bikeid" : 16743 }
{ "bikeid" : 23590 }
{ "bikeid" : 23063 }
Type "it" for more
```

```
> db.routes.replaceOne({airline: 470}, {name: "Air Sai" , alias: "21"});
{ "acknowledged" : true, "matchedCount" : 0, "modifiedCount" : 0 }
> db.routes.update({_id: 1}, {$set: {airline: "1338"}, $setOnInsert: {alias: 120}}, {upsert: true});
WriteResult({ "nMatched" : 0, "nUpserted" : 1, "nModified" : 0, "_id" : 1 })
>
> db.routes.find({_id:1}).pretty();
{ "_id" : 1, "airline" : "1338", "alias" : 120 }
```

Updating and saving this certificate numbers' business_name as shown:

```
> db.inspections.save({"certificate_number": "3022423", "business_name": "SIXTEENSEVEN"});  
WriteResult({ "nInserted" : 1 })  
>
```

Checking whether our saved entry actually persists: We can see the result which tells us that business_name is saved and updating successfully.

```
> db.inspections.find({"certificate_number": "3022423"},{business_name:1, _id:0});  
{ "business_name" : "SIXTEENSEVEN" }  
>
```