

1- Install MongoDB on your EC2 instance

Follow the instruction on

https://docs.aws.amazon.com/zh_cn/dms/latest/sbs/CHAP_MongoDB2DocumentDB.02.html

<https://docs.mongodb.com/manual/tutorial/install-mongodb-on-amazon/>

Verify Linux Distribution:

```
(base) [ec2-user@ip-172-31-94-236 ~]$ grep ^NAME /etc/*release  
/etc/os-release:NAME="Amazon Linux AMI"
```

Configure the package management system (yum):

```
(base) [ec2-user@ip-172-31-94-236 ~]$ sudo vi /etc/yum.repos.d/mongodb-org-4.4.repo
```

Input:

```
[mongodb-org-4.4]  
name=MongoDB Repository  
baseurl=https://repo.mongodb.org/yum/amazon/2/mongodb-org/4.4/x86_64/  
gpgcheck=1  
enabled=1  
gpgkey=https://www.mongodb.org/static/pgp/server-4.4.asc
```

copy

Install the MongoDB packages:

Update **glibc**: if you glibc is old version <= 2.17

```
(base) [ec2-user@ip-172-31-94-236 ~]$ yum check dependencies  
Loaded plugins: priorities, update-motd, upgrade-helper  
check ['dependencies']  
(base) [ec2-user@ip-172-31-94-236 ~]$ sudo yum clean all  
Loaded plugins: priorities, update-motd, upgrade-helper  
Cleaning repos: amzn-main amzn-updates mongodb-org-4.4  
Cleaning up everything  
(base) [ec2-user@ip-172-31-94-236 ~]$ sudo yum update  
Loaded plugins: priorities, update-motd, upgrade-helper
```

```
(base) [ec2-user@ip-172-31-94-236 ~]$ yum whatprovides */libc.so.6  
Loaded plugins: priorities, update-motd, upgrade-helper  
amzn-main/latest/filelists.db
```

```
sudo yum install -y mongodb-org
```

copy

Follow the instructions in the remaining part: **Run MongoDB Community Edition**

And remember to **ulimit** the **Resource Utilization**

```
(base) [ec2-user@ip-172-31-94-236 ~]$ ulimit -a
```

Configure host name to public DNS:

```
sudo vi /etc/mongod.conf
```

2. By default, the MongoDB server (mongod) only allows loopback connections from IP address 127.0.0.1 (localhost). To allow connections from elsewhere in your Amazon VPC, do the following:

a. Edit the `/etc/mongod.conf` file and look for the following lines.

```
# network interfaces
net:
  port: 27017
  bindIp: 127.0.0.1 # Enter 0.0.0.0,:: to bind to all IPv4 and IPv6 addresses or, alternatively, use the net.bindIpAll
```

b. Modify the `bindIp` line so that it looks like the following.

```
bindIp: public-dns-name
```

c. Replace `public-dns-name` with the actual public DNS name for your instance, for example `ec2-11-22-33-44.us-west-2.compute.amazonaws.com`.

d. Save the `/etc/mongod.conf` file, and then restart mongod.

```
sudo service mongod restart
```

Run MongoDB:

```
mongo --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
```

```
(base) [ec2-user@ip-172-31-94-236 ~]$ mongo --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
MongoDB shell version v4.4.1
connecting to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("c378419e-5198-48c7-a07f-cfee0d2fe3f1") }
MongoDB server version: 4.4.1

The server generated these startup warnings when booting:
  2020-11-11T09:26:20.663+00:00: ***** SERVER RESTARTED *****
  2020-11-11T09:26:20.677+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine. See http://dochub.mongodb.org/core/prodnotes-filesystem
  2020-11-11T09:26:21.725+00:00: Access control is not enabled for the database. Read and write access to data and configuration is unrestricted

---
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()

>
```

Next step please see the next page

2- Download GSE13355.zip (attached)

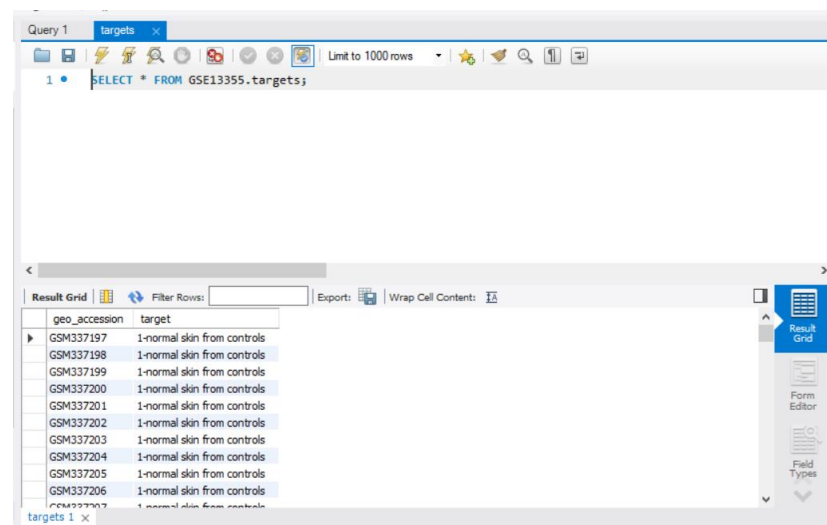
3- Create GSE13355 database on MySQL and MongoDB

4- Upload GSE13355_targets.csv to MySQL and MongoDB

MySQL:

```
CREATE SCHEMA `GSE13355` ;
```

Use MySQL Workbench to create table and import data. Result:



MongoDB:

```
use GSE13355
```

```
db.targets.insert({})
```

```
> use GSE13355
switched to db GSE13355
> db.targets.insert({})
WriteResult({ "nInserted" : 1 })
> show dbs
GSE13355  0.000GB
admin     0.000GB
config   0.000GB
local    0.000GB
test     0.000GB
>
```

Import data by shell:

```
mongoimport --db GSE13355 --collection targets --type csv --
headerline --file /home/ec2-user/GSE13355_targets.csv --host ec2-18-
211-73-183.compute-1.amazonaws.com:27017
```

Reference: https://blog.csdn.net/qg_32447321/article/details/79223332

Valid import data using UI interface: **MongoDB Compass**

Firstly config the MongoDB server:

```
sudo vi /etc/mongod.conf
```

```
# network interfaces
net:
  port: 27017
  bindIp: ec2-18-211-73-183.compute-1.amazonaws.com # Enter 0.0.0.0,:: to bind to all IPv4 and IPv6 addresses or, alternatively, use the net.bindIpAll setting.
```

Then create connection in MongoDB Compass:

New Connection ☆ FAVORITE [Paste connection string](#)

Hostname **More Options**

Hostname

Port

SRV Record ☐

Authentication

CONNECT

New Connection ☆ FAVORITE [Paste connection string](#)

Hostname **More Options**

Read Preference

SSL

SSH Tunnel

SSH Hostname

SSH Tunnel Port

SSH Username

SSH Identity File

SSH Passphrase

CONNECT

MongoDB Compass - 18.211.73.189:27017/GSE13355.targets

Local

5 DBS 3 COLLECTIONS

SSH CONNECTION VIA ec2-18-21-...aws.com:27017

HOST 127.0.0.1:29554

CLUSTER Standalone

EDITION MongoDB 4.4.1 Community

Filter your data

GSE13355

targets

admin

config

local

test

GSE13355.targets Documents

DOCUMENTS 180 TOTAL SIZE 16.0KB AVG. SIZE 91B INDEXES 1 TOTAL SIZE 36.0KB AVG. SIZE 36.0KB

Documents Aggregations Schema Explain Plan Indexes Validation

0 FILTER

ADD DATA

VIEW

Displaying documents 1 - 20 of 180

REFRESH

```
{ "_id": { "id": "sfabce8348f753334d88566" }, "geo_accession": "G0937197", "target": "1-normal skin from controls" }
{ "_id": { "id": "sfabce8348f753334d88567" }, "geo_accession": "G0937198", "target": "1-normal skin from controls" }
{ "_id": { "id": "sfabce8348f753334d88568" }, "geo_accession": "G0937199", "target": "1-normal skin from controls" }
{ "_id": { "id": "sfabce8348f753334d88569" }, "geo_accession": "G0937200", "target": "1-normal skin from controls" }
{ "_id": { "id": "sfabce8348f753334d8856a" }, "geo_accession": "G0937201", "target": "1-normal skin from controls" }
```

> _MongoSH Beta

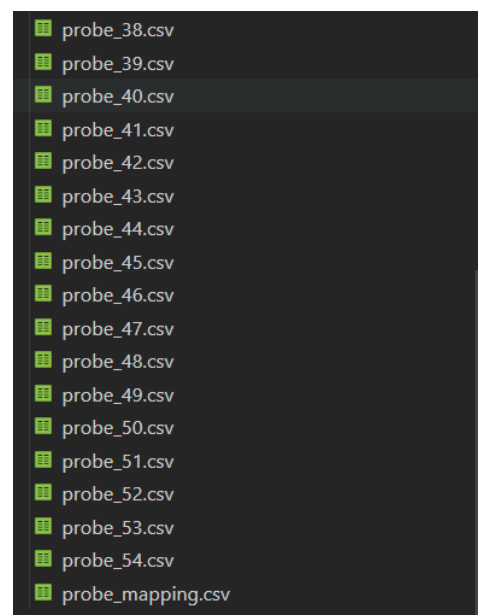
5- Transpose GSE13355_expr.csv and upload the transposed file to MySQL and MongoDB

MySQL:

Use **transpose.py** to transpose the csv file and get raw data.

```
transpose.py
1  import pandas as pd
2
3  file=open('C:\\Users\\95236\\Desktop\\MongoDB_assignment_Group5\\GSE13355_expr_transpose.csv','w')
4
5  df = pd.read_csv('C:\\Users\\95236\\Desktop\\MongoDB_assignment_Group5\\GSE13355_expr.csv',header= None)
6  data = df.values
7  # data = df.as_matrix()
8  data = list(map(list,zip(*data)))
9  data = pd.DataFrame(data)
10 data.to_csv(file, header=0, index=0)
```

Split the csv into multiple tables by **split_table.py** each has 1001 columns, since the max number of columns in MySQL is 4096, that one is the id related to the **geo_info** like GSM337197 and others are **probe_name** and the mapping between the **probe_name** and **table_name** is stored in table **probe_mapping**. And write the **geo_info** into the table **expr** with **geo_id** and **geo_name**



then use the **load** method to import the data from ec2 to rds:

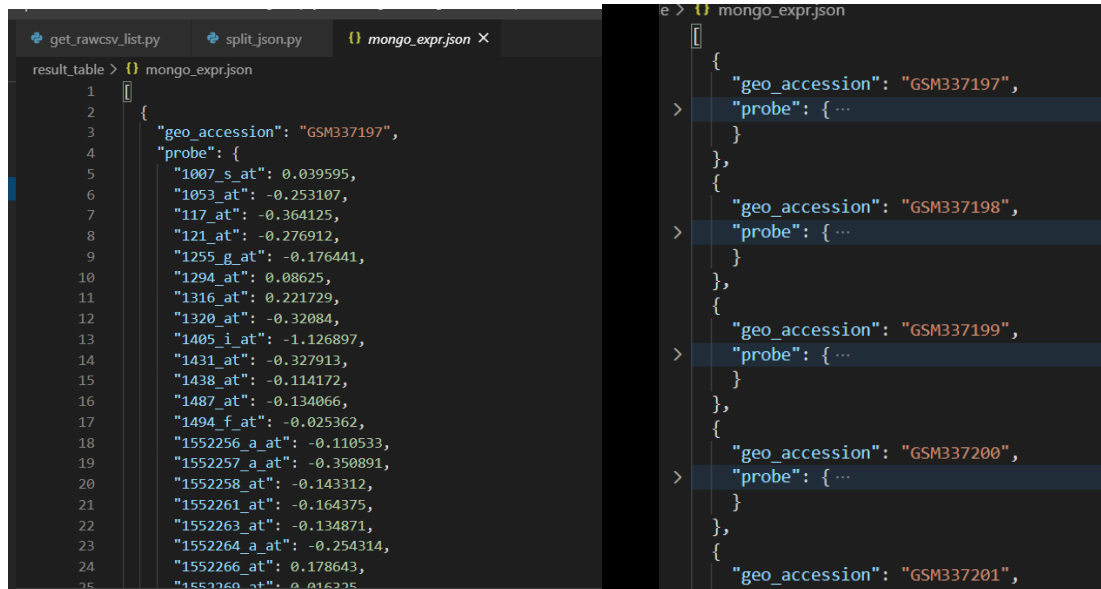
```
mysql> CREATE TABLE IF NOT EXISTS 'GSE13355'.'probe_mapping'(
-> 'probe_name' VARCHAR(500) NOT NULL COMMENT 'column name',
-> 'table_name' VARCHAR(200) NOT NULL,
-> PRIMARY KEY ('probe_name'),
-> UNIQUE INDEX ('probe_name')
-> )ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
Query OK, 0 rows affected (0.85 sec)

mysql>
mysql> LOAD DATA LOCAL INFILE '/home/ec2-user/result_table/probe_mapping.csv' INTO TABLE 'GSE13355'.'probe_mapping' FIELDS TERMINATED BY ',' ENCLOSED BY '|';
Query OK, 54675 rows affected (1.33 sec)
Records: 54675 Deleted: 0 Skipped: 0 Warnings: 0

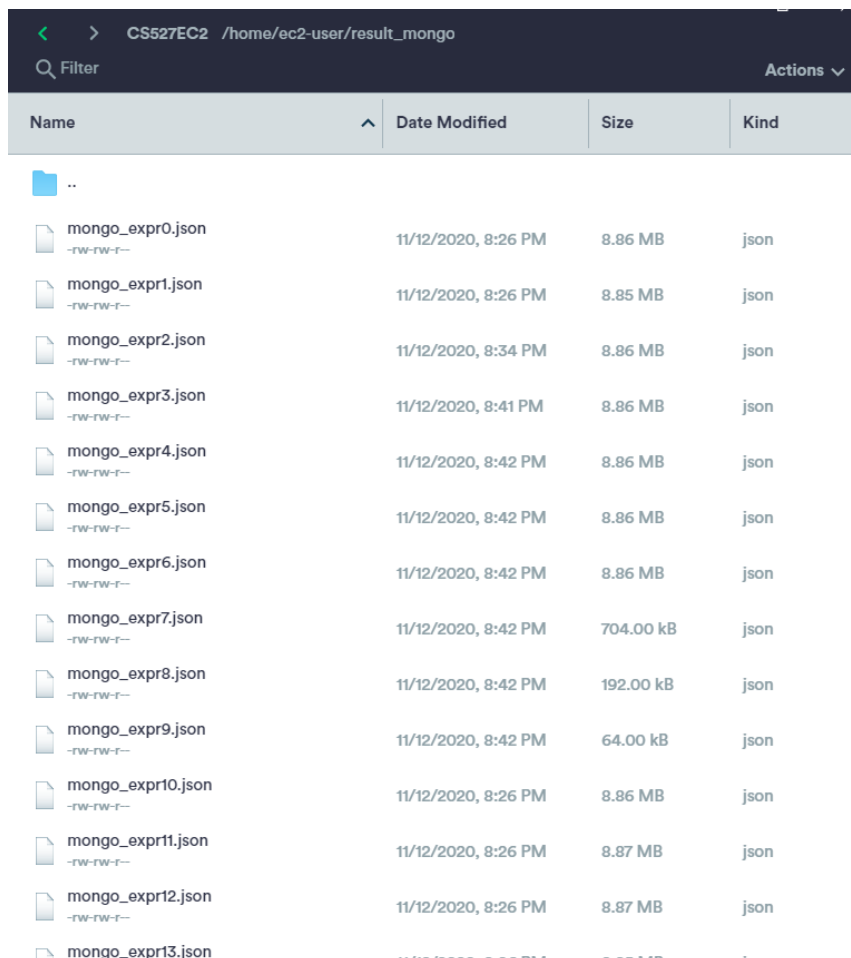
mysql>
```

MongoDB:

We use **split_json.py** to split the data to the json file as the following format, because the json result is too large and in order to import more convenient, we split the json file to 36 json files and each of it is less than 10 MB so that we can use **mongoimport --jsonArray** to import the data:



```
get_rawcsv_list.py  split_json.py  mongo_expr.json X
result_table > mongo_expr.json
1  {
2    "geo_accession": "GSM337197",
3    "probe": {
4      "1007_s_at": 0.039595,
5      "1053_at": -0.253107,
6      "117_at": -0.364125,
7      "121_at": -0.276912,
8      "1255_g_at": -0.176441,
9      "1294_at": 0.08625,
10     "1316_at": 0.221729,
11     "1320_at": -0.32084,
12     "1405_i_at": -1.126897,
13     "1431_at": -0.327913,
14     "1438_at": -0.114172,
15     "1487_at": -0.134066,
16     "1494_f_at": -0.025362,
17     "1552256_a_at": -0.110533,
18     "1552257_a_at": -0.350891,
19     "1552258_at": -0.143312,
20     "1552261_at": -0.164375,
21     "1552263_at": -0.134871,
22     "1552264_a_at": -0.254314,
23     "1552266_at": 0.178643,
24     "1552268_at": 0.016225
25   }
26 }
27 {
28   "geo_accession": "GSM337198",
29   "probe": {
30     "1007_s_at": 0.039595,
31     "1053_at": -0.253107,
32     "117_at": -0.364125,
33     "121_at": -0.276912,
34     "1255_g_at": -0.176441,
35     "1294_at": 0.08625,
36     "1316_at": 0.221729,
37     "1320_at": -0.32084,
38     "1405_i_at": -1.126897,
39     "1431_at": -0.327913,
40     "1438_at": -0.114172,
41     "1487_at": -0.134066,
42     "1494_f_at": -0.025362,
43     "1552256_a_at": -0.110533,
44     "1552257_a_at": -0.350891,
45     "1552258_at": -0.143312,
46     "1552261_at": -0.164375,
47     "1552263_at": -0.134871,
48     "1552264_a_at": -0.254314,
49     "1552266_at": 0.178643,
50     "1552268_at": 0.016225
51   }
52 }
53 {
54   "geo_accession": "GSM337199",
55   "probe": {
56     "1007_s_at": 0.039595,
57     "1053_at": -0.253107,
58     "117_at": -0.364125,
59     "121_at": -0.276912,
60     "1255_g_at": -0.176441,
61     "1294_at": 0.08625,
62     "1316_at": 0.221729,
63     "1320_at": -0.32084,
64     "1405_i_at": -1.126897,
65     "1431_at": -0.327913,
66     "1438_at": -0.114172,
67     "1487_at": -0.134066,
68     "1494_f_at": -0.025362,
69     "1552256_a_at": -0.110533,
70     "1552257_a_at": -0.350891,
71     "1552258_at": -0.143312,
72     "1552261_at": -0.164375,
73     "1552263_at": -0.134871,
74     "1552264_a_at": -0.254314,
75     "1552266_at": 0.178643,
76     "1552268_at": 0.016225
77   }
78 }
79 {
80   "geo_accession": "GSM337200",
81   "probe": {
82     "1007_s_at": 0.039595,
83     "1053_at": -0.253107,
84     "117_at": -0.364125,
85     "121_at": -0.276912,
86     "1255_g_at": -0.176441,
87     "1294_at": 0.08625,
88     "1316_at": 0.221729,
89     "1320_at": -0.32084,
90     "1405_i_at": -1.126897,
91     "1431_at": -0.327913,
92     "1438_at": -0.114172,
93     "1487_at": -0.134066,
94     "1494_f_at": -0.025362,
95     "1552256_a_at": -0.110533,
96     "1552257_a_at": -0.350891,
97     "1552258_at": -0.143312,
98     "1552261_at": -0.164375,
99     "1552263_at": -0.134871,
100    "1552264_a_at": -0.254314,
101    "1552266_at": 0.178643,
102    "1552268_at": 0.016225
103  }
104 }
105 {
106   "geo_accession": "GSM337201",
107   "probe": {
108     "1007_s_at": 0.039595,
109     "1053_at": -0.253107,
110     "117_at": -0.364125,
111     "121_at": -0.276912,
112     "1255_g_at": -0.176441,
113     "1294_at": 0.08625,
114     "1316_at": 0.221729,
115     "1320_at": -0.32084,
116     "1405_i_at": -1.126897,
117     "1431_at": -0.327913,
118     "1438_at": -0.114172,
119     "1487_at": -0.134066,
120     "1494_f_at": -0.025362,
121     "1552256_a_at": -0.110533,
122     "1552257_a_at": -0.350891,
123     "1552258_at": -0.143312,
124     "1552261_at": -0.164375,
125     "1552263_at": -0.134871,
126     "1552264_a_at": -0.254314,
127     "1552266_at": 0.178643,
128     "1552268_at": 0.016225
129   }
130 }
```



Name	Date Modified	Size	Kind
..			
mongo_expr0.json -rw-rw-r--	11/12/2020, 8:26 PM	8.86 MB	json
mongo_expr1.json -rw-rw-r--	11/12/2020, 8:26 PM	8.85 MB	json
mongo_expr2.json -rw-rw-r--	11/12/2020, 8:34 PM	8.86 MB	json
mongo_expr3.json -rw-rw-r--	11/12/2020, 8:41 PM	8.86 MB	json
mongo_expr4.json -rw-rw-r--	11/12/2020, 8:42 PM	8.86 MB	json
mongo_expr5.json -rw-rw-r--	11/12/2020, 8:42 PM	8.86 MB	json
mongo_expr6.json -rw-rw-r--	11/12/2020, 8:42 PM	8.86 MB	json
mongo_expr7.json -rw-rw-r--	11/12/2020, 8:42 PM	704.00 kB	json
mongo_expr8.json -rw-rw-r--	11/12/2020, 8:42 PM	192.00 kB	json
mongo_expr9.json -rw-rw-r--	11/12/2020, 8:42 PM	64.00 kB	json
mongo_expr10.json -rw-rw-r--	11/12/2020, 8:26 PM	8.86 MB	json
mongo_expr11.json -rw-rw-r--	11/12/2020, 8:26 PM	8.87 MB	json
mongo_expr12.json -rw-rw-r--	11/12/2020, 8:26 PM	8.87 MB	json
mongo_expr13.json -rw-rw-r--	11/12/2020, 8:26 PM	8.85 MB	json

And then as the above method, use the **mongoimport** to import the json data into MongoDB:

```
mongo --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
use GSE13355
db.targets.insert({})
```

```
(base) [ec2-user@ip-172-31-94-236 ~]$ mongo --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
MongoDB shell version v4.4.1
connecting to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("299ee530-7097-4efb-9427-2ba55652bd9f") }
MongoDB server version: 4.4.1
---
The server generated these startup warnings when booting:
  2020-11-11T13:02:49.008+00:00: ***** SERVER RESTARTED *****
  2020-11-11T13:02:49.025+00:00: Using the XFS filesystem is strongly recommended with the WiredTiger
  2020-11-11T13:02:50.082+00:00: Access control is not enabled for the database. Read and write access
---
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()
---
> use GSE13355
switched to db GSE13355
> db.expr.insert({})
WriteResult({ "nInserted" : 1 })
> show dbs
GSE13355  0.000GB
admin     0.000GB
config    0.000GB
local     0.000GB
test      0.000GB
> █
```

Example:

```
mongoimport --jsonArray --db GSE13355 --collection expr --type json --
file /home/ec2-user/result_mongo/mongo_expr1.json --host ec2-18-211-73-
183.compute-1.amazonaws.com:27017
```

```
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr0.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:35:53.077+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:35:53.190+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr1.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:22.855+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:23.094+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr18.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:23.117+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:23.357+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr11.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:24.251+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:25.494+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr12.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:25.588+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:26.365+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr13.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:26.389+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:27.202+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr14.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:27.326+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:28.109+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr15.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:28.191+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:29.074+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr16.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:29.098+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:29.898+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr17.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:29.972+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:30.864+0000 5 document(s) imported successfully, 0 document(s) failed to import.
(base) [ec2-user@ip-172-31-94-236 ~]$ mongoimport --jsonArray --db GSE13355 --collection expr --type json --file /home/ec2-user/result_mongo/mongo_expr18.json --host ec2-18-211-73-183.compute-1.amazonaws.com:27017
2020-11-11T12:36:30.888+0000 connected to: mongodb://ec2-18-211-73-183.compute-1.amazonaws.com:27017/
2020-11-11T12:36:31.738+0000 5 document(s) imported successfully, 0 document(s) failed to import.
```

6- Up to 5 teams will be selected to present their assignment