Data Loading

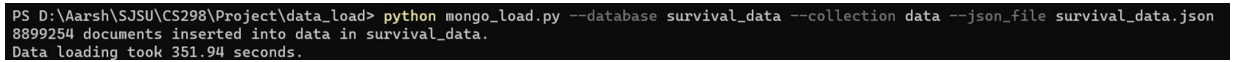
After data generation,next step is data loading.I am using Python and database drivers to make connection to the database and load data in it with different parameters the user can specify for each database which executing the script

MongoDB

Driver: pymongo

Example code :

python mongo_load.py --database survival_data --collection data --json_file survival_data.json



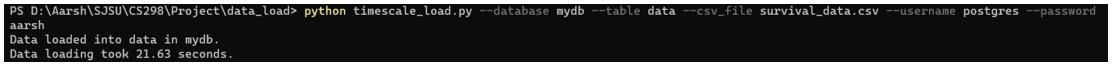
```
survival_data> db.data.find();
[
  {
    _id: ObjectId("65162b4a00ca2595b1b55f16"),
    age_start_observed: 1,
    age_end: 50,
    date_start_observed: '2009-03-13',
    date_end_observed: '2020-12-31',
    is_truncated: false,
    is_censored: false,
    is_dead: true
  },
  {
    _id: ObjectId("65162b4a00ca2595b1b55f17"),
    age_start_observed: 1,
    age_end: 42,
    date_start_observed: '2020-12-31',
    date_end_observed: '2020-12-31',
    is_truncated: false,
    is_censored: true,
    is_dead: false
  },
  {
    _id: ObjectId("65162b4a00ca2595b1b55f18"),
    age_start_observed: 49,
    age_end: 70,
    date_start_observed: '1950-01-01',
    date_end_observed: '1970-12-09',
    is_truncated: true,
    is_censored: false,
    is_dead: true
  },
```

TimeScaleDB

Driver: psycopg2

Example code :

python timescale_load.py --database mydb --table data --csv_file survival_data.csv --username postgres --password aarsh



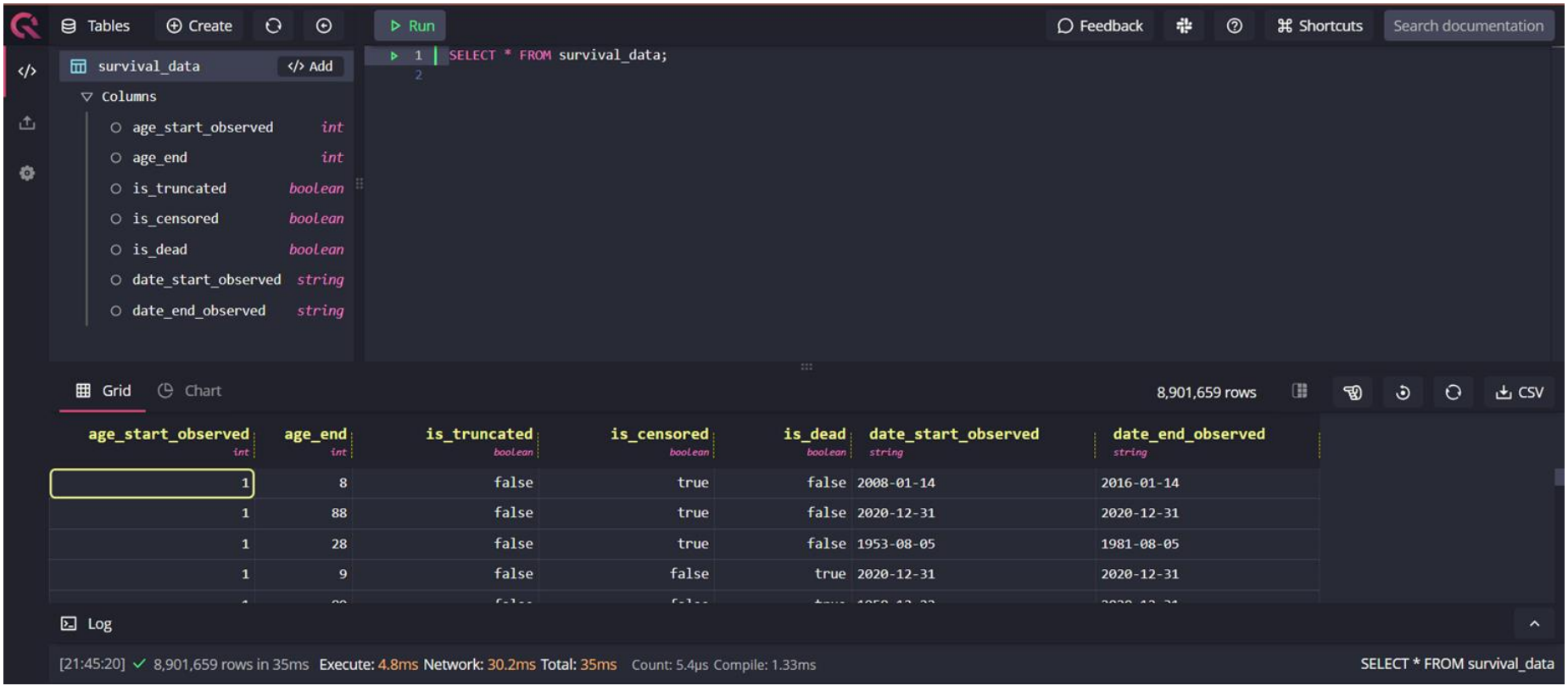
```
mydb=# SELECT * FROM mytable;
 age_start_observed | age_end | is_truncated | is_censored | is_dead | date_start_observed | date_end_observed
--------------------+---------+--------------+-------------+---------+---------------------+-------------------
 1                  | 8       | False        | True        | False   | 2008-01-14          | 2016-01-14
 1                  | 88      | False        | True        | False   | 2020-12-31          | 2020-12-31
 1                  | 28      | False        | True        | False   | 1953-08-05          | 1981-08-05
 1                  | 9       | False        | False       | True    | 2020-12-31          | 2020-12-31
 1                  | 89      | False        | False       | True    | 1958-12-22          | 2020-12-31
 1                  | 51      | False        | False       | True    | 2020-12-31          | 2020-12-31
 1                  | 85      | False        | False       | True    | 2017-07-22          | 2020-12-31
 1                  | 79      | False        | False       | True    | 1981-06-14          | 2020-12-31
 1                  | 8       | False        | True        | False   | 1961-08-14          | 1969-08-14
 1                  | 60      | False        | False       | True    | 1957-12-05          | 2017-12-05
 1                  | 51      | False        | False       | True    | 2020-12-31          | 2020-12-31
 34                 | 81      | True         | False       | True    | 1950-01-01          | 1996-01-16
 1                  | 18      | False        | False       | True    | 2007-05-31          | 2020-12-31
 1                  | 3       | False        | True        | False   | 2011-01-08          | 2014-01-08
 1                  | 18      | False        | True        | False   | 2020-12-31          | 2020-12-31
 1                  | 59      | False        | False       | True    | 1994-07-27          | 2020-12-31
 1                  | 78      | False        | False       | True    | 1995-08-13          | 2020-12-31
 1                  | 63      | False        | False       | True    | 1996-07-22          | 2020-12-31
 1                  | 15      | False        | True        | False   | 1976-12-18          | 1991-12-19
 1                  | 34      | False        | True        | False   | 1982-09-24          | 2016-09-24
 1                  | 41      | False        | True        | False   | 2020-12-31          | 2020-12-31
 1                  | 45      | False        | True        | False   | 2003-09-28          | 2020-12-31
 1                  | 17      | False        | True        | False   | 1974-04-19          | 1991-04-20
 47                 | 93      | True         | False       | True    | 1950-01-01          | 1995-06-13
 1                  | 47      | False        | False       | True    | 2015-06-10          | 2020-12-31
 1                  | 67      | False        | False       | True    | 2020-12-31          | 2020-12-31
 1                  | 73      | False        | True        | False   | 2020-12-31          | 2020-12-31
```

QuestDB

Example code :

python questdb_load.py http://localhost:9000 survival_data survival_data.csv

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python questdb_load.py http://localhost:9000 data survival_data.csv
Data loaded into data
Data loading took 14.64 seconds.
```


Tables
Create
Run
Feedback
Shortcuts
Search documentation
survival_data
Add
Columns
age_start_observed int
age_end int
is_truncated boolean
is_censored boolean
is_dead boolean
date_start_observed string
date_end_observed string
1 SELECT * FROM survival_data;
2
Grid
Chart
8,901,659 rows
CSV
age_start_observed int
age_end int
is_truncated boolean
is_censored boolean
is_dead boolean
date_start_observed string
date_end_observed string
1 8 false true false 2008-01-14 2016-01-14
1 88 false true false 2020-12-31 2020-12-31
1 28 false true false 1953-08-05 1981-08-05
1 9 false false true 2020-12-31 2020-12-31
Log
[21:45:20] ✓ 8,901,659 rows in 35ms Execute: 4.8ms Network: 30.2ms Total: 35ms Count: 5.4µs Compile: 1.33ms
SELECT * FROM survival_data

Cassandra

Driver: Cluster

Example code :

python cassandra_load.py --keyspace survival_data --table data --csv_file survival_data.csv

```
PS D:\Aarsh\SJSU\CS298\Project\data_load> python cassandra_load.py --keyspace survival_data --table data --csv_file survival_data.csv
Using 7 child processes

Starting copy of survival_data.data with columns [age_start_observed, age_end, date_end_observed, date_start_observed, is_censored, is_dead, is_truncated].
Processed: 8902255 rows; Rate:   17762 rows/s; Avg. rate:   25371 rows/s
8902255 rows imported from 1 files in 0 day, 0 hour, 5 minutes, and 50.879 seconds (0 skipped).
Data loading took 353.21 seconds.
```

```
cqlsh> select * from survival_data.data;
```

age_start_observed	age_end	date_end_observed	date_start_observed	is_censored	is_dead	is_truncated
23	94	1950-01-01	2020-07-04	True	False	True
33	97	1950-01-01	2013-05-14	True	True	True
5	92	1950-01-01	2020-12-31	True	True	True
28	63	1950-01-01	1984-02-17	True	True	True
42	68	1950-01-01	1975-11-07	True	False	True
49	85	1950-01-01	1985-04-11	True	True	True
10	97	1950-01-01	2020-12-31	True	True	True
16	84	1950-01-01	2017-06-08	True	True	True
13	71	1950-01-01	2007-08-11	True	True	True
30	52	1950-01-01	1971-07-02	True	False	True
11	43	1950-01-01	1981-09-27	True	False	True
1	80	2011-07-02	2020-12-31	True	True	True
19	71	1950-01-01	2001-03-11	True	False	True
46	50	1950-01-01	1953-06-07	True	False	True
43	67	1950-01-01	1973-04-08	True	True	False
8	91	1950-01-01	2020-12-31	True	False	True
2	66	1950-01-01	2013-01-12	True	True	False
45	80	1950-01-01	1984-05-28	True	True	True
4	49	1950-01-01	1994-03-22	True	True	False
18	66	1950-01-01	1997-12-30	True	True	True
47	83	1950-01-01	1985-05-15	True	False	True
44	81	1950-01-01	1986-09-16	True	True	False
15	73	1950-01-01	2007-10-14	True	False	True
22	64	1950-01-01	1991-07-06	True	True	True
27	82	1950-01-01	2004-09-11	True	True	True
20	83	1950-01-01	2012-01-16	True	False	True
7	99	1950-01-01	2020-12-31	True	True	True
36	75	1950-01-01	1988-09-12	True	True	True
40	80	1950-01-01	1989-10-04	True	True	True
38	64	1950-01-01	1975-05-15	True	False	True
39	60	1950-01-01	1970-09-22	True	False	True
6	27	1950-01-01	1970-05-15	True	False	True
29	80	1950-01-01	2000-07-25	True	True	True
37	93	1950-01-01	2005-02-15	True	True	True
9	72	1950-01-01	2012-03-20	True	True	True
14	87	1950-01-01	2020-12-31	True	True	True
26	64	1950-01-01	1987-09-16	True	True	True
21	84	1950-01-01	2012-09-14	True	False	True
17	69	1950-01-01	2001-12-29	True	False	True
35	88	1950-01-01	2002-07-30	True	True	True

Data Loading Results

Database	Time Taken
MongoDB	351.94 sec
TimeScaleDB	21.63 sec
QuestDB	14.64 sec
Cassandra	353.21 sec