Spencer Burns, Yanki Saplan

Professor Rigas

DS220

December 14, 2020

Final Project

Introduction

The dataset that we chose was the firearm permits and background checks database. The data analyzes the United States, state by state and we thought it would be a good choice for this project as we have a dataset that has many options to work with.

Firearm permits and background checks Dataset

Dataset: https://github.com/BuzzFeedNews/nics-firearm-background-checks (Links to an external site.)

The data in this repository comes from the FBI's National Instant Criminal Background Check System

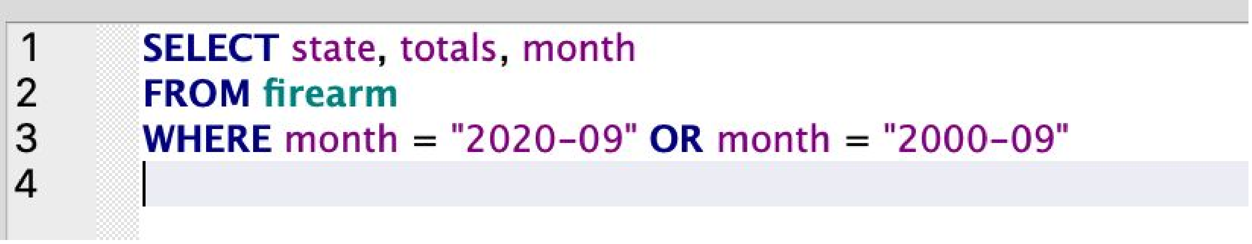
The NICS is used by the Federal Firearms Licensees to instantly determine whether a prospective buyer is eligible to buy firearms or explosives. Before ringing up the sale, cashiers call in a check to the FBI or to other designated agencies to ensure that each customer does not have a criminal record or isn't otherwise ineligible to make a purchase. There have been more than 100 million checks over the past decade, which has led to more than 700,000 denials over that timespan. The code in this GitHub repository downloads that PDF, parses it, and produces a spreadsheet/CSV of the data. The data collects data from about the last 20 years, ranging from November 1998 to April 2019.

The dataset we are working with contains these attributes:

- month
- state
- permit
- permit_recheck
- handgun
- long_gun
- other
- multiple
- admin
- prepawn_handgun
- prepawn_long_gun
- prepawn_other
- redemption_handgun
- redemption_long_gun
- redeption_other
- returned_handgun
- returned_long_gun
- returned_other
- rentals_handgun
- rentals_long_gun
- private_sale_handgun
- private_sale_long_gun
- private_sale_other
- return_to_seller_handgun
- return_to_seller_handgun
- return_to_seller_long_gun
- return_to_seller_other_totals

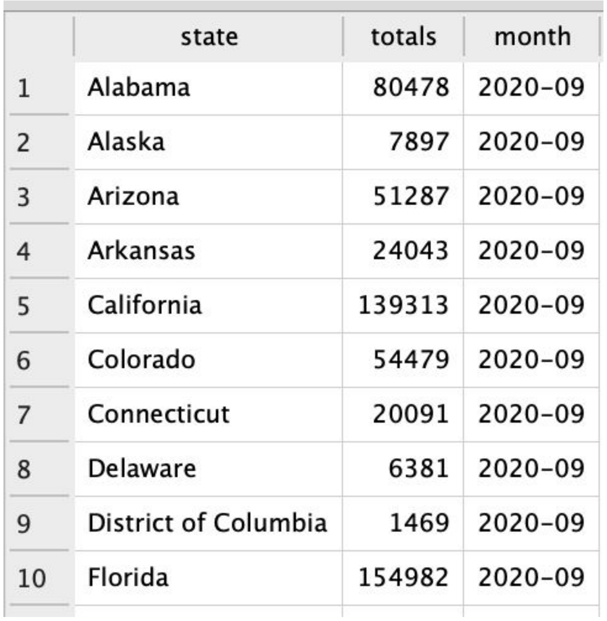
Using DB Browser for SQLite

Here, we go into the data using “firearm,” and are able to SELECT state, totals, and month from the data. This lets us identify totals by each state, month by month, but WHERE is used to only pull data from September of 2020 and September of 2009. This query allows comparison in totals between these two selected months.

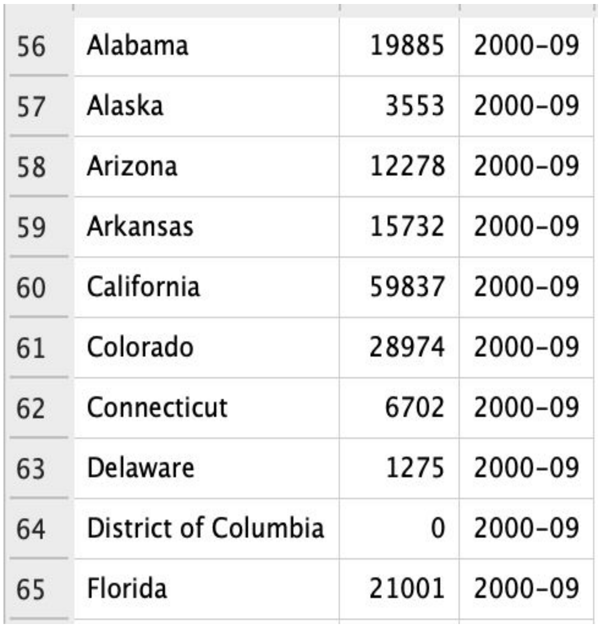


```
SELECT state, totals, month
FROM firearm
WHERE month = "2020-09" OR month = "2000-09"

```

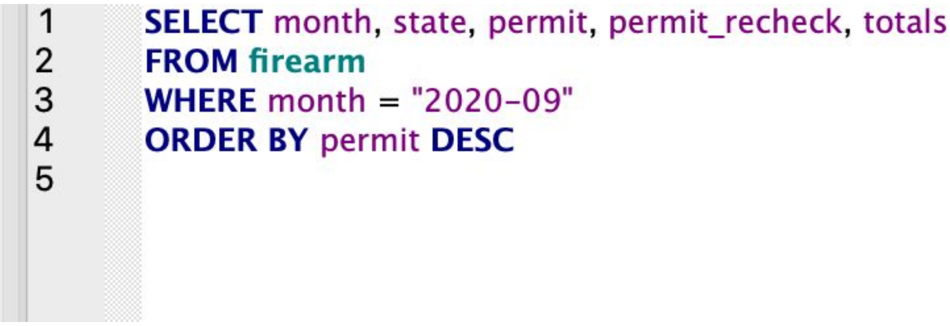


	state	totals	month
1	Alabama	80478	2020-09
2	Alaska	7897	2020-09
3	Arizona	51287	2020-09
4	Arkansas	24043	2020-09
5	California	139313	2020-09
6	Colorado	54479	2020-09
7	Connecticut	20091	2020-09
8	Delaware	6381	2020-09
9	District of Columbia	1469	2020-09
10	Florida	154982	2020-09

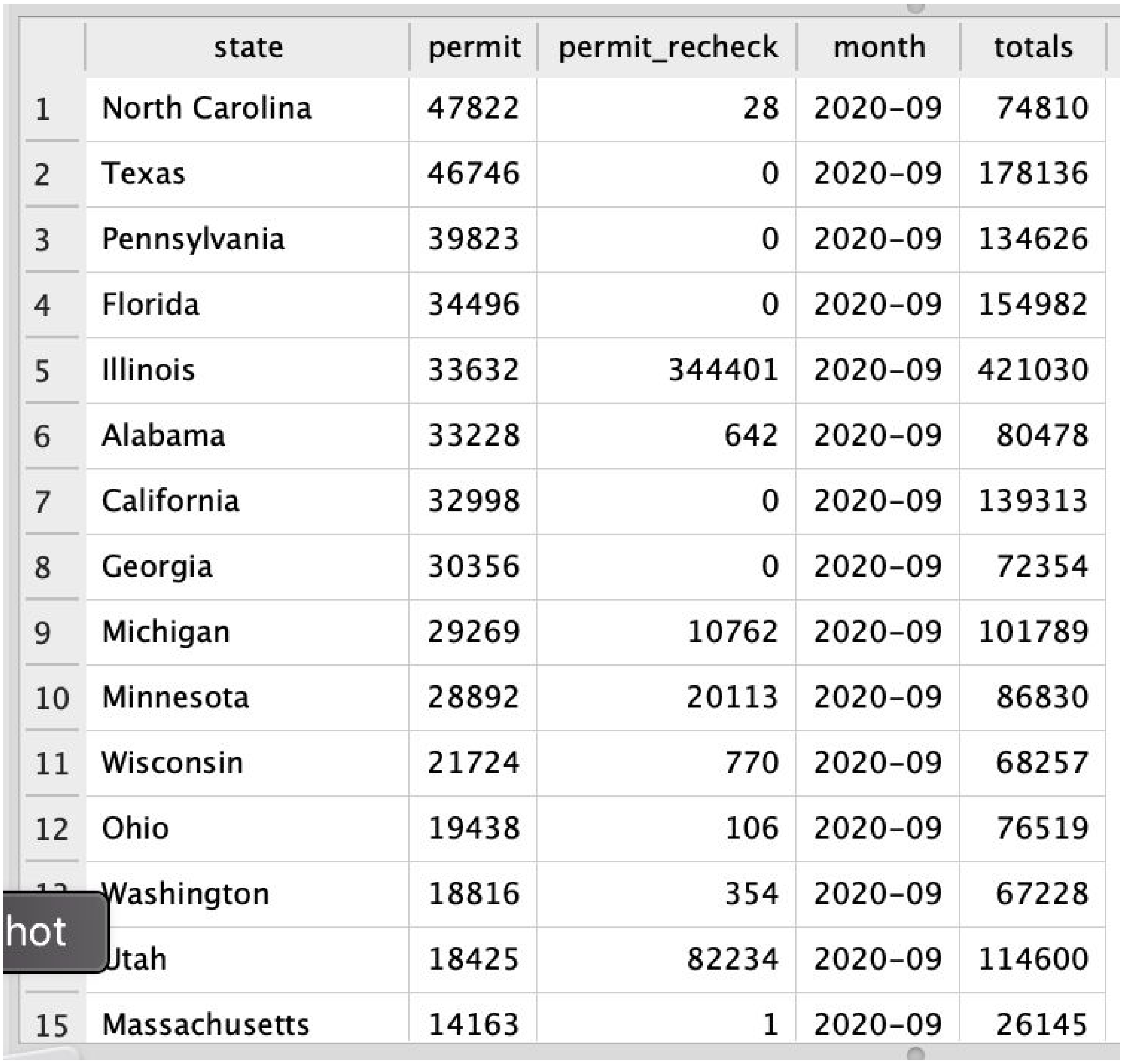


56	Alabama	19885	2000-09
57	Alaska	3553	2000-09
58	Arizona	12278	2000-09
59	Arkansas	15732	2000-09
60	California	59837	2000-09
61	Colorado	28974	2000-09
62	Connecticut	6702	2000-09
63	Delaware	1275	2000-09
64	District of Columbia	0	2000-09
65	Florida	21001	2000-09

This second query allows to look at different statistics in the data and order them by # of permits in September of 2020.

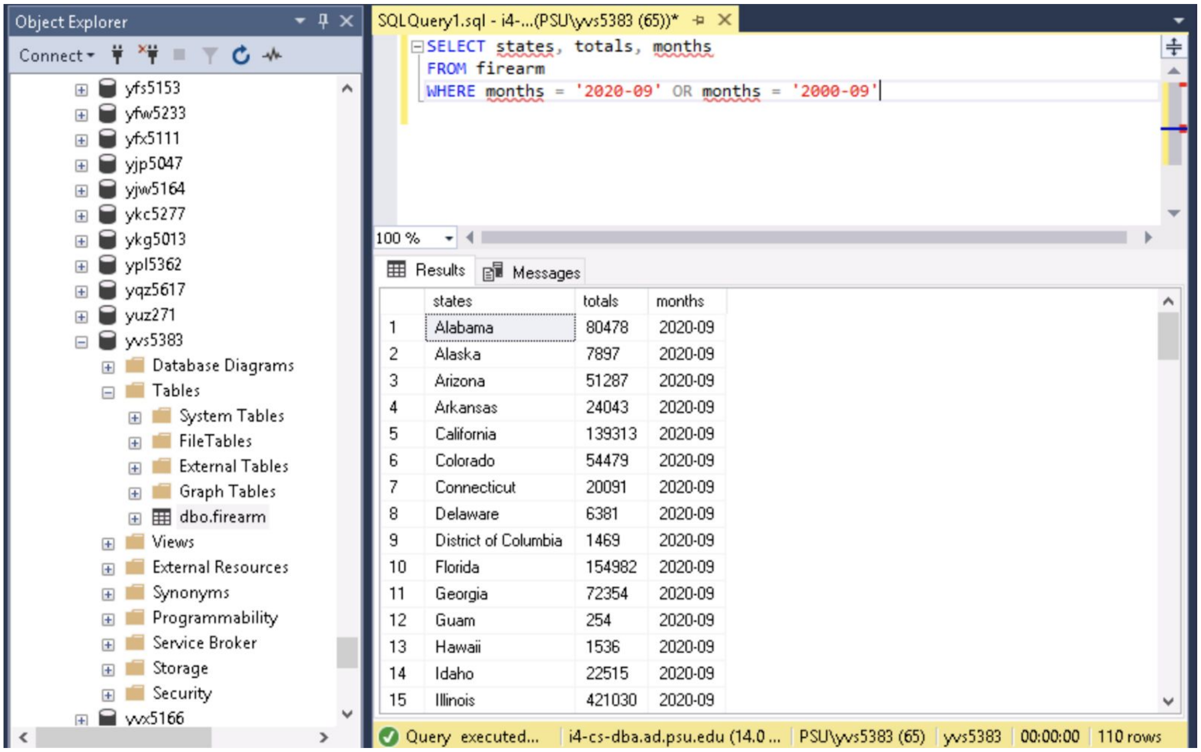


```
SELECT month, state, permit, permit_recheck, totals
FROM firearm
WHERE month = "2020-09"
ORDER BY permit DESC
```



	state	permit	permit_recheck	month	totals
1	North Carolina	47822	28	2020-09	74810
2	Texas	46746	0	2020-09	178136
3	Pennsylvania	39823	0	2020-09	134626
4	Florida	34496	0	2020-09	154982
5	Illinois	33632	344401	2020-09	421030
6	Alabama	33228	642	2020-09	80478
7	California	32998	0	2020-09	139313
8	Georgia	30356	0	2020-09	72354
9	Michigan	29269	10762	2020-09	101789
10	Minnesota	28892	20113	2020-09	86830
11	Wisconsin	21724	770	2020-09	68257
12	Ohio	19438	106	2020-09	76519
13	Washington	18816	354	2020-09	67228
[illegible]	Utah	18425	82234	2020-09	114600
15	Massachusetts	14163	1	2020-09	26145

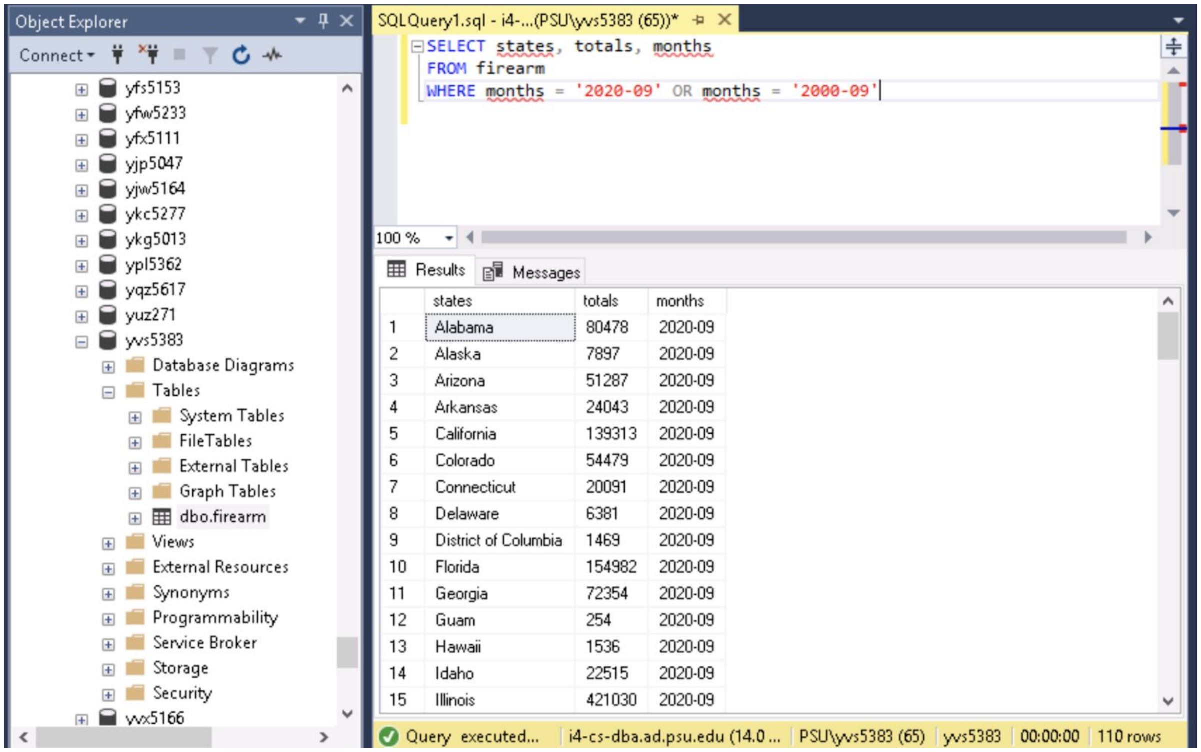
They do return the same results, but they have differences in terms of syntaxing.



```
SELECT states, totals, months
FROM firearm
WHERE months = '2020-09' OR months = '2000-09'
```

	states	totals	months
1	Alabama	80478	2020-09
2	Alaska	7897	2020-09
3	Arizona	51287	2020-09
4	Arkansas	24043	2020-09
5	California	139313	2020-09
6	Colorado	54479	2020-09
7	Connecticut	20091	2020-09
8	Delaware	6381	2020-09
9	District of Columbia	1469	2020-09
10	Florida	154982	2020-09
11	Georgia	72354	2020-09
12	Guam	254	2020-09
13	Hawaii	1536	2020-09
14	Idaho	22515	2020-09
15	Illinois	421030	2020-09

Query executed... | i4-cs-dba.ad.psu.edu (14.0 ... | PSU\yvs5383 (65) | yvs5383 | 00:00:00 | 110 rows



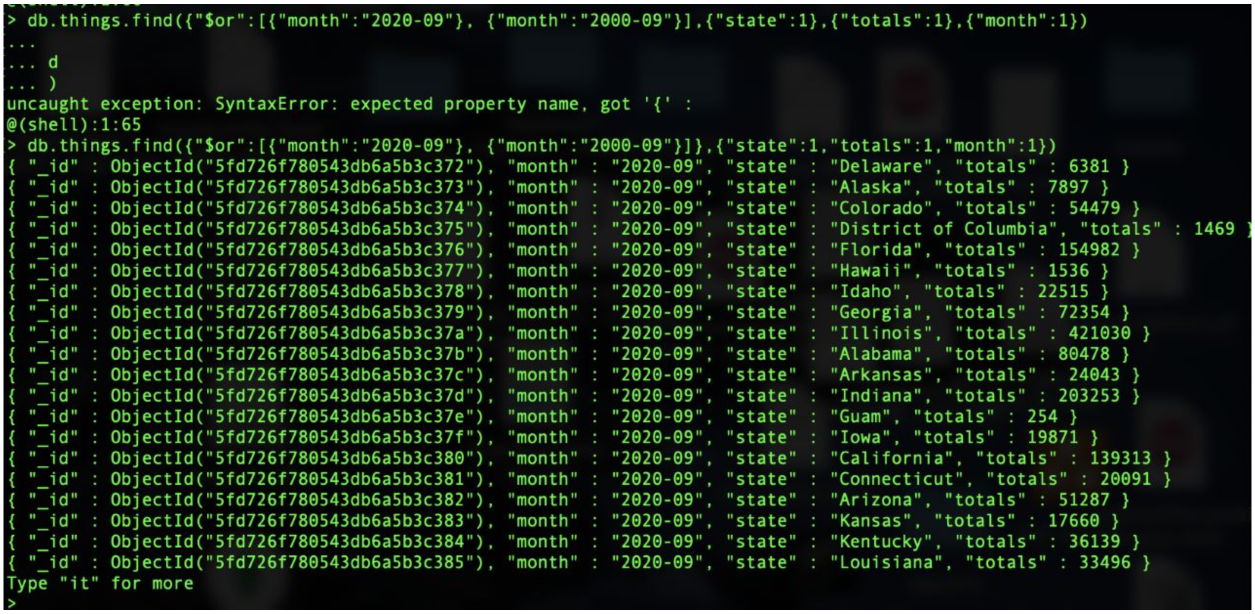
```
SELECT states, totals, months
FROM firearm
WHERE months = '2020-09' OR months = '2000-09'
```

	states	totals	months
1	Alabama	80478	2020-09
2	Alaska	7897	2020-09
3	Arizona	51287	2020-09
4	Arkansas	24043	2020-09
5	California	139313	2020-09
6	Colorado	54479	2020-09
7	Connecticut	20091	2020-09
8	Delaware	6381	2020-09
9	District of Columbia	1469	2020-09
10	Florida	154982	2020-09
11	Georgia	72354	2020-09
12	Guam	254	2020-09
13	Hawaii	1536	2020-09
14	Idaho	22515	2020-09
15	Illinois	421030	2020-09

Query executed... | i4-cs-dba.ad.psu.edu (14.0 ... | PSU\yvs5383 (65) | yvs5383 | 00:00:00 | 110 rows

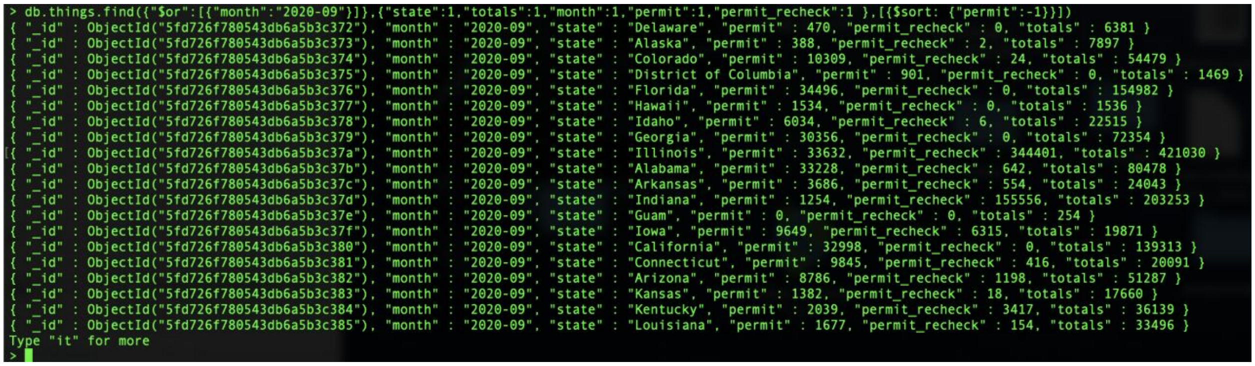
Using MongoDB for NoSQL

This query is using .find to call a state’s total in only the Septembers of 2000 and 2020, to see the rest of the results,



```
> db.things.find({"$or":[{"month":"2020-09"}, {"month":"2000-09"}],{"state":1},{"totals":1},{"month":1})
...
... d
... )
uncaught exception: SyntaxError: expected property name, got '{' :
@(shell):1:65
> db.things.find({"$or":[{"month":"2020-09"}, {"month":"2000-09"}]},{"state":1,"totals":1,"month":1})
{ "_id" : ObjectId("5fd726f780543db6a5b3c372"), "month" : "2020-09", "state" : "Delaware", "totals" : 6381 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c373"), "month" : "2020-09", "state" : "Alaska", "totals" : 7897 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c374"), "month" : "2020-09", "state" : "Colorado", "totals" : 54479 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c375"), "month" : "2020-09", "state" : "District of Columbia", "totals" : 1469 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c376"), "month" : "2020-09", "state" : "Florida", "totals" : 154982 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c377"), "month" : "2020-09", "state" : "Hawaii", "totals" : 1536 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c378"), "month" : "2020-09", "state" : "Idaho", "totals" : 22515 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c379"), "month" : "2020-09", "state" : "Georgia", "totals" : 72354 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37a"), "month" : "2020-09", "state" : "Illinois", "totals" : 421030 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37b"), "month" : "2020-09", "state" : "Alabama", "totals" : 80478 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37c"), "month" : "2020-09", "state" : "Arkansas", "totals" : 24043 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37d"), "month" : "2020-09", "state" : "Indiana", "totals" : 203253 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37e"), "month" : "2020-09", "state" : "Guam", "totals" : 254 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37f"), "month" : "2020-09", "state" : "Iowa", "totals" : 19871 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c380"), "month" : "2020-09", "state" : "California", "totals" : 139313 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c381"), "month" : "2020-09", "state" : "Connecticut", "totals" : 20091 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c382"), "month" : "2020-09", "state" : "Arizona", "totals" : 51287 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c383"), "month" : "2020-09", "state" : "Kansas", "totals" : 17660 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c384"), "month" : "2020-09", "state" : "Kentucky", "totals" : 36139 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c385"), "month" : "2020-09", "state" : "Louisiana", "totals" : 33496 }
Type "it" for more
>
```

Uses .find to get the totals,



```
> db.things.find({"$or":[{"month":"2020-09"}]},{"state":1,"totals":1,"month":1,"permit":1,"permit_recheck":1 },[{$sort: {"permit":-1}}])
{ "_id" : ObjectId("5fd726f780543db6a5b3c372"), "month" : "2020-09", "state" : "Delaware", "permit" : 470, "permit_recheck" : 0, "totals" : 6381 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c373"), "month" : "2020-09", "state" : "Alaska", "permit" : 388, "permit_recheck" : 2, "totals" : 7897 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c374"), "month" : "2020-09", "state" : "Colorado", "permit" : 10309, "permit_recheck" : 24, "totals" : 54479 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c375"), "month" : "2020-09", "state" : "District of Columbia", "permit" : 901, "permit_recheck" : 0, "totals" : 1469 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c376"), "month" : "2020-09", "state" : "Florida", "permit" : 34496, "permit_recheck" : 0, "totals" : 154982 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c377"), "month" : "2020-09", "state" : "Hawaii", "permit" : 1534, "permit_recheck" : 0, "totals" : 1536 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c378"), "month" : "2020-09", "state" : "Idaho", "permit" : 6034, "permit_recheck" : 6, "totals" : 22515 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c379"), "month" : "2020-09", "state" : "Georgia", "permit" : 30356, "permit_recheck" : 0, "totals" : 72354 }
[{ "_id" : ObjectId("5fd726f780543db6a5b3c37a"), "month" : "2020-09", "state" : "Illinois", "permit" : 33632, "permit_recheck" : 344401, "totals" : 421030 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37b"), "month" : "2020-09", "state" : "Alabama", "permit" : 33228, "permit_recheck" : 642, "totals" : 80478 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37c"), "month" : "2020-09", "state" : "Arkansas", "permit" : 3686, "permit_recheck" : 554, "totals" : 24043 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37d"), "month" : "2020-09", "state" : "Indiana", "permit" : 1254, "permit_recheck" : 155556, "totals" : 203253 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37e"), "month" : "2020-09", "state" : "Guam", "permit" : 0, "permit_recheck" : 0, "totals" : 254 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37f"), "month" : "2020-09", "state" : "Iowa", "permit" : 9649, "permit_recheck" : 6315, "totals" : 19871 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c380"), "month" : "2020-09", "state" : "California", "permit" : 32998, "permit_recheck" : 0, "totals" : 139313 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c381"), "month" : "2020-09", "state" : "Connecticut", "permit" : 9845, "permit_recheck" : 416, "totals" : 20091 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c382"), "month" : "2020-09", "state" : "Arizona", "permit" : 8786, "permit_recheck" : 1198, "totals" : 51287 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c383"), "month" : "2020-09", "state" : "Kansas", "permit" : 1382, "permit_recheck" : 18, "totals" : 17660 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c384"), "month" : "2020-09", "state" : "Kentucky", "permit" : 2039, "permit_recheck" : 3417, "totals" : 36139 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c385"), "month" : "2020-09", "state" : "Louisiana", "permit" : 1677, "permit_recheck" : 154, "totals" : 33496 }
Type "it" for more
>
```

Introduction or background

The firearms data basically shows a lot of different firearm statistics according to their state and their month/year. The data shows how many individual totals(firearms) and other neat statistics, including permits and permit rechecks. I think that the data is interesting because you’re able to see the difference in laws between states. The huge discrepancy in some of the data shows what is required with firearms in some states, and possibly the politics surrounding firearms in each state and their time period.

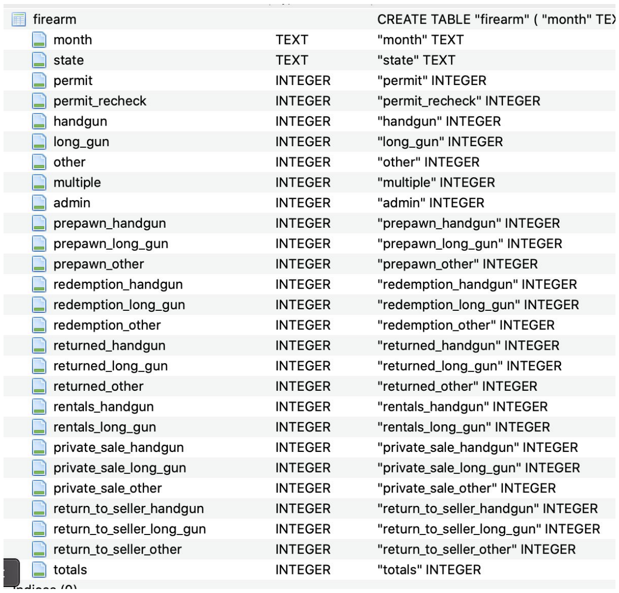
Dataset

We used the firearm permits and background checks data. The data goes by each state, and the month for that state. The data has over 20 years of data, 12 months for each year, and then every state and American territory. The data has some huge discrepancies, both over time and between states. You can see the certain laws or requirements from each state and that time period by looking at the numbers. Overall, the data is very good for analyzing firearm permits and background checking over the past 20 years and for each state. The data appears to have come from Buzzfeed and I think the only limitation in the dataset is that it doesn’t account per capita. The numbers can be slightly misleading due to population size differences, but it does very well with showing possible laws or the political environment of a time period and state.

Methods

First, I downloaded the required files and data including the csv data from the github repositories. After downloading the DB Browser for SQLite, I have imported the csv data into a new database table. The first row had a problem while executing the sql command as the database table inserted the columns as rows. After fixing the mistake SQL execution was working, and I was able to run SQL commands we have decided on.

SQL



firearm		CREATE TABLE "firearm" ("month" TE)
month	TEXT	"month" TEXT
state	TEXT	"state" TEXT
permit	INTEGER	"permit" INTEGER
permit_recheck	INTEGER	"permit_recheck" INTEGER
handgun	INTEGER	"handgun" INTEGER
long_gun	INTEGER	"long_gun" INTEGER
other	INTEGER	"other" INTEGER
multiple	INTEGER	"multiple" INTEGER
admin	INTEGER	"admin" INTEGER
prepawn_handgun	INTEGER	"prepawn_handgun" INTEGER
prepawn_long_gun	INTEGER	"prepawn_long_gun" INTEGER
prepawn_other	INTEGER	"prepawn_other" INTEGER
redemption_handgun	INTEGER	"redemption_handgun" INTEGER
redemption_long_gun	INTEGER	"redemption_long_gun" INTEGER
redemption_other	INTEGER	"redemption_other" INTEGER
returned_handgun	INTEGER	"returned_handgun" INTEGER
returned_long_gun	INTEGER	"returned_long_gun" INTEGER
returned_other	INTEGER	"returned_other" INTEGER
rentals_handgun	INTEGER	"rentals_handgun" INTEGER
rentals_long_gun	INTEGER	"rentals_long_gun" INTEGER
private_sale_handgun	INTEGER	"private_sale_handgun" INTEGER
private_sale_long_gun	INTEGER	"private_sale_long_gun" INTEGER
private_sale_other	INTEGER	"private_sale_other" INTEGER
return_to_seller_handgun	INTEGER	"return_to_seller_handgun" INTEGER
return_to_seller_long_gun	INTEGER	"return_to_seller_long_gun" INTEGER
return_to_seller_other	INTEGER	"return_to_seller_other" INTEGER
totals	INTEGER	"totals" INTEGER

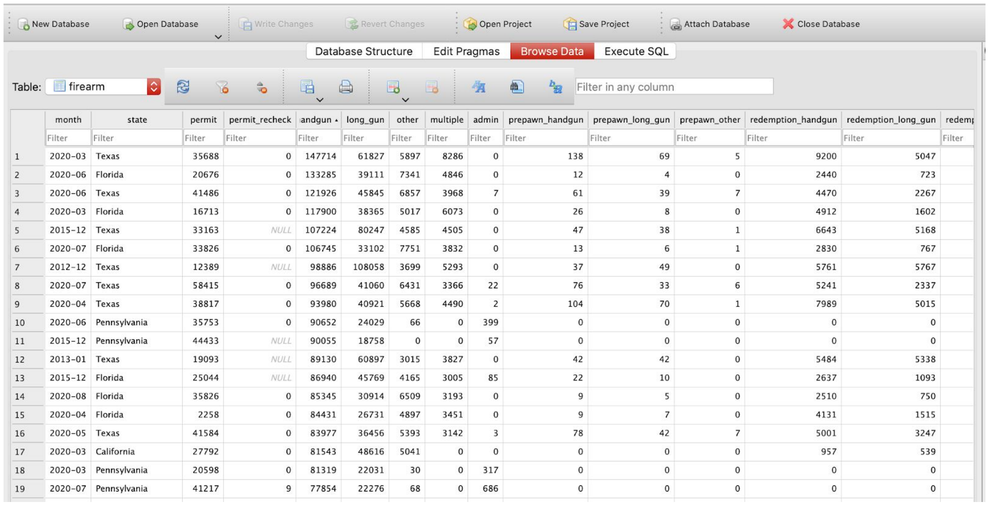
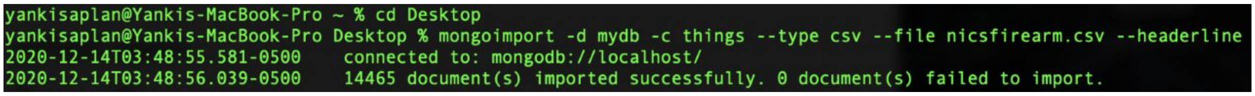


Table: firearm

	month	state	permit	permit_recheck	andgun ▴	long_gun	other	multiple	admin	prepawn_handgun	prepawn_long_gun	prepawn_other	redemption_handgun	redemption_long_gun	redemp
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	2020-03	Texas	35688	0	147714	61827	5897	8286	0	138	69	5	9200	5047	
2	2020-06	Florida	20676	0	133285	39111	7341	4846	0	12	4	0	2440	723	
3	2020-06	Texas	41486	0	121926	45845	6857	3968	7	61	39	7	4470	2267	
4	2020-03	Florida	16713	0	117900	38365	5017	6073	0	26	8	0	4912	1602	
5	2015-12	Texas	33163	NULL	107224	80247	4585	4505	0	47	38	1	6643	5168	
6	2020-07	Florida	33826	0	106745	33102	7751	3832	0	13	6	1	2830	767	
7	2012-12	Texas	12389	NULL	98886	108058	3699	5293	0	37	49	0	5761	5767	
8	2020-07	Texas	58415	0	96689	41060	6431	3366	22	76	33	6	5241	2337	
9	2020-04	Texas	38817	0	93980	40921	5668	4490	2	104	70	1	7989	5015	
10	2020-06	Pennsylvania	35753	0	90652	24029	66	0	399	0	0	0	0	0	
11	2015-12	Pennsylvania	44433	NULL	90055	18758	0	0	57	0	0	0	0	0	
12	2013-01	Texas	19093	NULL	89130	60897	3015	3827	0	42	42	0	5484	5338	
13	2015-12	Florida	25044	NULL	86940	45769	4165	3005	85	22	10	0	2637	1093	
14	2020-08	Florida	35826	0	85345	30914	6509	3193	0	9	5	0	2510	750	
15	2020-04	Florida	2258	0	84431	26731	4897	3451	0	9	7	0	4131	1515	
16	2020-05	Texas	41584	0	83977	36456	5393	3142	3	78	42	7	5001	3247	
17	2020-03	California	27792	0	81543	48616	5041	0	0	0	0	0	957	539	
18	2020-03	Pennsylvania	20598	0	81319	22031	30	0	317	0	0	0	0	0	
19	2020-07	Pennsylvania	41217	9	77854	22276	68	0	686	0	0	0	0	0	

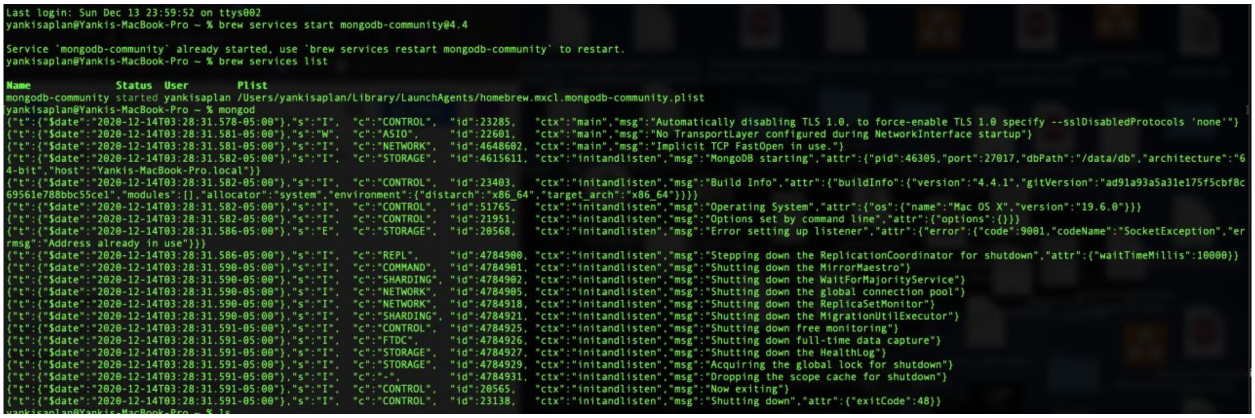
NoSQL

Importing to the MongoDB terminal.



```
yankisaplan@Yankis-MacBook-Pro ~ % cd Desktop
yankisaplan@Yankis-MacBook-Pro Desktop % mongoimport -d mydb -c things --type csv --file nicsfirearm.csv --headerline
2020-12-14T03:48:55.581-0500    connected to: mongodb://localhost/
2020-12-14T03:48:56.039-0500    14465 document(s) imported successfully. 0 document(s) failed to import.
```

An example on running MongoDB.

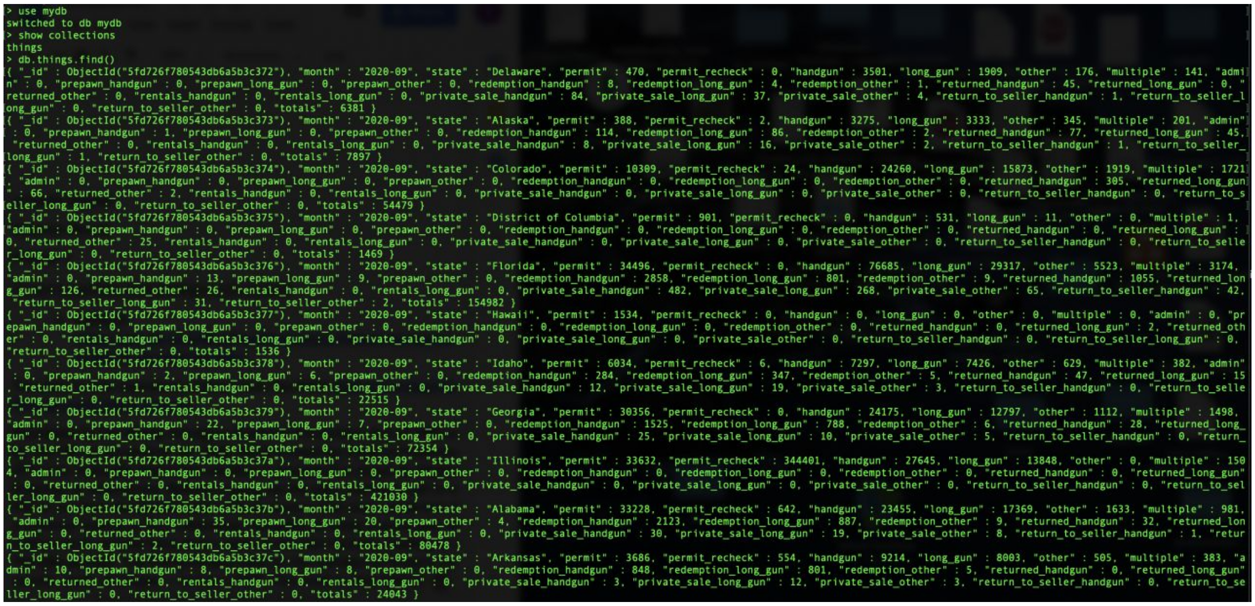


```
Last login: Sun Dec 13 23:59:52 on ttys002
yankisaplan@Yankis-MacBook-Pro ~ % brew services start mongodb-community@4.4

Service `mongodb-community` already started, use `brew services restart mongodb-community` to restart.
yankisaplan@Yankis-MacBook-Pro ~ % brew services list

Name              Status  User        Plist
mongodb-community started yankisaplan /Users/yankisaplan/Library/LaunchAgents/homebrew.mxcl.mongodb-community.plist
yankisaplan@Yankis-MacBook-Pro ~ % mongod
{"t":{"$date":"2020-12-14T03:28:31.578-05:00"},"s":"I",  "c":"CONTROL",  "id":23285,   "ctx":"main","msg":"Automatically disabling TLS 1.0, to force-enable TLS 1.0 specify --sslDisabledProtocols 'none'"}
{"t":{"$date":"2020-12-14T03:28:31.581-05:00"},"s":"W",  "c":"ASIO",     "id":22601,   "ctx":"main","msg":"No TransportLayer configured during NetworkInterface startup"}
{"t":{"$date":"2020-12-14T03:28:31.581-05:00"},"s":"I",  "c":"NETWORK",  "id":4648602, "ctx":"main","msg":"Implicit TCP FastOpen in use."}
{"t":{"$date":"2020-12-14T03:28:31.582-05:00"},"s":"I",  "c":"STORAGE",  "id":4615611, "ctx":"initandlisten","msg":"MongoDB starting","attr":{"pid":46305,"port":27017,"dbPath":"/data/db","architecture":"6
4-bit","host":"Yankis-MacBook-Pro.local"}}
{"t":{"$date":"2020-12-14T03:28:31.582-05:00"},"s":"I",  "c":"CONTROL",  "id":23403,   "ctx":"initandlisten","msg":"Build Info","attr":{"buildInfo":{"version":"4.4.1","gitVersion":"ad91a93a5a31e175f5cbf8c
69561e788bbc55ce1","modules":[],"allocator":"system","environment":{"distarch":"x86_64","target_arch":"x86_64"}}}}
{"t":{"$date":"2020-12-14T03:28:31.582-05:00"},"s":"I",  "c":"CONTROL",  "id":51765,   "ctx":"initandlisten","msg":"Operating System","attr":{"os":{"name":"Mac OS X","version":"19.6.0"}}}
{"t":{"$date":"2020-12-14T03:28:31.582-05:00"},"s":"I",  "c":"CONTROL",  "id":21951,   "ctx":"initandlisten","msg":"Options set by command line","attr":{"options":{}}}
{"t":{"$date":"2020-12-14T03:28:31.586-05:00"},"s":"E",  "c":"STORAGE",  "id":20568,   "ctx":"initandlisten","msg":"Error setting up listener","attr":{"error":{"code":9001,"codeName":"SocketException","er
rmsg":"Address already in use"}}}
{"t":{"$date":"2020-12-14T03:28:31.586-05:00"},"s":"I",  "c":"REPL",     "id":4784900, "ctx":"initandlisten","msg":"Stepping down the ReplicationCoordinator for shutdown","attr":{"waitTimeMillis":10000}}
{"t":{"$date":"2020-12-14T03:28:31.590-05:00"},"s":"I",  "c":"COMMAND",  "id":4784901, "ctx":"initandlisten","msg":"Shutting down the MirrorMaestro"}
{"t":{"$date":"2020-12-14T03:28:31.590-05:00"},"s":"I",  "c":"SHARDING", "id":4784902, "ctx":"initandlisten","msg":"Shutting down the WaitForMajorityService"}
{"t":{"$date":"2020-12-14T03:28:31.590-05:00"},"s":"I",  "c":"NETWORK",  "id":4784905, "ctx":"initandlisten","msg":"Shutting down the global connection pool"}
{"t":{"$date":"2020-12-14T03:28:31.590-05:00"},"s":"I",  "c":"NETWORK",  "id":4784918, "ctx":"initandlisten","msg":"Shutting down the ReplicaSetMonitor"}
{"t":{"$date":"2020-12-14T03:28:31.590-05:00"},"s":"I",  "c":"SHARDING", "id":4784921, "ctx":"initandlisten","msg":"Shutting down the MigrationUtilExecutor"}
{"t":{"$date":"2020-12-14T03:28:31.591-05:00"},"s":"I",  "c":"CONTROL",  "id":4784925, "ctx":"initandlisten","msg":"Shutting down free monitoring"}
{"t":{"$date":"2020-12-14T03:28:31.591-05:00"},"s":"I",  "c":"FTDC",     "id":4784926, "ctx":"initandlisten","msg":"Shutting down full-time data capture"}
{"t":{"$date":"2020-12-14T03:28:31.591-05:00"},"s":"I",  "c":"STORAGE",  "id":4784927, "ctx":"initandlisten","msg":"Shutting down the HealthLog"}
{"t":{"$date":"2020-12-14T03:28:31.591-05:00"},"s":"I",  "c":"STORAGE",  "id":4784929, "ctx":"initandlisten","msg":"Acquiring the global lock for shutdown"}
{"t":{"$date":"2020-12-14T03:28:31.591-05:00"},"s":"I",  "c":"-",        "id":4784931, "ctx":"initandlisten","msg":"Dropping the scope cache for shutdown"}
{"t":{"$date":"2020-12-14T03:28:31.591-05:00"},"s":"I",  "c":"CONTROL",  "id":20565,   "ctx":"initandlisten","msg":"Now exiting"}
{"t":{"$date":"2020-12-14T03:28:31.591-05:00"},"s":"I",  "c":"CONTROL",  "id":23138,   "ctx":"initandlisten","msg":"Shutting down","attr":{"exitCode":48}}
yankisaplan@Yankis-MacBook-Pro ~ % ls
```

Using mydb, able to switch to our database and return everything within the database.



```
> use mydb
switched to db mydb
> show collections
things
> db.things.find()
{ "_id" : ObjectId("5fd726f780543db6a5b3c372"), "month" : "2020-09", "state" : "Delaware", "permit" : 470, "permit_recheck" : 0, "handgun" : 3501, "long_gun" : 1909, "other" : 176, "multiple" : 141, "admi
n" : 0, "prepawn_handgun" : 0, "prepawn_long_gun" : 0, "prepawn_other" : 0, "redemption_handgun" : 8, "redemption_long_gun" : 4, "redemption_other" : 1, "returned_handgun" : 45, "returned_long_gun" : 0, "
returned_other" : 0, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 84, "private_sale_long_gun" : 37, "private_sale_other" : 4, "return_to_seller_handgun" : 1, "return_to_seller_l
ong_gun" : 0, "return_to_seller_other" : 0, "totals" : 6381 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c373"), "month" : "2020-09", "state" : "Alaska", "permit" : 388, "permit_recheck" : 2, "handgun" : 3275, "long_gun" : 3333, "other" : 345, "multiple" : 201, "admin"
 : 0, "prepawn_handgun" : 1, "prepawn_long_gun" : 0, "prepawn_other" : 0, "redemption_handgun" : 114, "redemption_long_gun" : 86, "redemption_other" : 2, "returned_handgun" : 77, "returned_long_gun" : 45,
 "returned_other" : 0, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 8, "private_sale_long_gun" : 16, "private_sale_other" : 2, "return_to_seller_handgun" : 1, "return_to_seller_
long_gun" : 1, "return_to_seller_other" : 0, "totals" : 7897 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c374"), "month" : "2020-09", "state" : "Colorado", "permit" : 10309, "permit_recheck" : 24, "handgun" : 24260, "long_gun" : 15873, "other" : 1919, "multiple" : 1721
, "admin" : 0, "prepawn_handgun" : 0, "prepawn_long_gun" : 0, "prepawn_other" : 0, "redemption_handgun" : 0, "redemption_long_gun" : 0, "redemption_other" : 0, "returned_handgun" : 305, "returned_long_gun
" : 66, "returned_other" : 2, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 0, "private_sale_long_gun" : 0, "private_sale_other" : 0, "return_to_seller_handgun" : 0, "return_to_s
eller_long_gun" : 0, "return_to_seller_other" : 0, "totals" : 54479 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c375"), "month" : "2020-09", "state" : "District of Columbia", "permit" : 901, "permit_recheck" : 0, "handgun" : 531, "long_gun" : 11, "other" : 0, "multiple" : 1,
"admin" : 0, "prepawn_handgun" : 0, "prepawn_long_gun" : 0, "prepawn_other" : 0, "redemption_handgun" : 0, "redemption_long_gun" : 0, "redemption_other" : 0, "returned_handgun" : 0, "returned_long_gun" :
0, "returned_other" : 25, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 0, "private_sale_long_gun" : 0, "private_sale_other" : 0, "return_to_seller_handgun" : 0, "return_to_selle
r_long_gun" : 0, "return_to_seller_other" : 0, "totals" : 1469 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c376"), "month" : "2020-09", "state" : "Florida", "permit" : 34496, "permit_recheck" : 0, "handgun" : 76685, "long_gun" : 29317, "other" : 5523, "multiple" : 3174,
"admin" : 0, "prepawn_handgun" : 13, "prepawn_long_gun" : 9, "prepawn_other" : 0, "redemption_handgun" : 2858, "redemption_long_gun" : 801, "redemption_other" : 9, "returned_handgun" : 1055, "returned_lon
g_gun" : 126, "returned_other" : 26, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 482, "private_sale_long_gun" : 268, "private_sale_other" : 65, "return_to_seller_handgun" : 42,
 "return_to_seller_long_gun" : 31, "return_to_seller_other" : 2, "totals" : 154982 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c377"), "month" : "2020-09", "state" : "Hawaii", "permit" : 1534, "permit_recheck" : 0, "handgun" : 0, "long_gun" : 0, "other" : 0, "multiple" : 0, "admin" : 0, "pr
epawn_handgun" : 0, "prepawn_long_gun" : 0, "prepawn_other" : 0, "redemption_handgun" : 0, "redemption_long_gun" : 0, "redemption_other" : 0, "returned_handgun" : 0, "returned_long_gun" : 2, "returned_oth
er" : 0, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 0, "private_sale_long_gun" : 0, "private_sale_other" : 0, "return_to_seller_handgun" : 0, "return_to_seller_long_gun" : 0,
"return_to_seller_other" : 0, "totals" : 1536 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c378"), "month" : "2020-09", "state" : "Idaho", "permit" : 6034, "permit_recheck" : 6, "handgun" : 7297, "long_gun" : 7426, "other" : 629, "multiple" : 382, "admin"
 : 0, "prepawn_handgun" : 2, "prepawn_long_gun" : 6, "prepawn_other" : 0, "redemption_handgun" : 284, "redemption_long_gun" : 347, "redemption_other" : 5, "returned_handgun" : 47, "returned_long_gun" : 15
, "returned_other" : 1, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 12, "private_sale_long_gun" : 19, "private_sale_other" : 3, "return_to_seller_handgun" : 0, "return_to_selle
r_long_gun" : 0, "return_to_seller_other" : 0, "totals" : 22515 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c379"), "month" : "2020-09", "state" : "Georgia", "permit" : 30356, "permit_recheck" : 0, "handgun" : 24175, "long_gun" : 12797, "other" : 1112, "multiple" : 1498,
"admin" : 0, "prepawn_handgun" : 22, "prepawn_long_gun" : 7, "prepawn_other" : 0, "redemption_handgun" : 1525, "redemption_long_gun" : 788, "redemption_other" : 6, "returned_handgun" : 28, "returned_long_
gun" : 0, "returned_other" : 0, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 25, "private_sale_long_gun" : 10, "private_sale_other" : 5, "return_to_seller_handgun" : 0, "return_
to_seller_long_gun" : 0, "return_to_seller_other" : 0, "totals" : 72354 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37a"), "month" : "2020-09", "state" : "Illinois", "permit" : 33632, "permit_recheck" : 344401, "handgun" : 27645, "long_gun" : 13848, "other" : 0, "multiple" : 150
4, "admin" : 0, "prepawn_handgun" : 0, "prepawn_long_gun" : 0, "prepawn_other" : 0, "redemption_handgun" : 0, "redemption_long_gun" : 0, "redemption_other" : 0, "returned_handgun" : 0, "returned_long_gun"
 : 0, "returned_other" : 0, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 0, "private_sale_long_gun" : 0, "private_sale_other" : 0, "return_to_seller_handgun" : 0, "return_to_sel
ler_long_gun" : 0, "return_to_seller_other" : 0, "totals" : 421030 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37b"), "month" : "2020-09", "state" : "Alabama", "permit" : 33228, "permit_recheck" : 642, "handgun" : 23455, "long_gun" : 17369, "other" : 1633, "multiple" : 981,
 "admin" : 0, "prepawn_handgun" : 35, "prepawn_long_gun" : 20, "prepawn_other" : 4, "redemption_handgun" : 2123, "redemption_long_gun" : 887, "redemption_other" : 9, "returned_handgun" : 32, "returned_lon
g_gun" : 0, "returned_other" : 0, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 30, "private_sale_long_gun" : 19, "private_sale_other" : 8, "return_to_seller_handgun" : 1, "retur
n_to_seller_long_gun" : 2, "return_to_seller_other" : 0, "totals" : 80478 }
{ "_id" : ObjectId("5fd726f780543db6a5b3c37c"), "month" : "2020-09", "state" : "Arkansas", "permit" : 3686, "permit_recheck" : 554, "handgun" : 9214, "long_gun" : 8003, "other" : 505, "multiple" : 383, "a
dmin" : 10, "prepawn_handgun" : 8, "prepawn_long_gun" : 8, "prepawn_other" : 0, "redemption_handgun" : 848, "redemption_long_gun" : 801, "redemption_other" : 5, "returned_handgun" : 0, "returned_long_gun"
 : 0, "returned_other" : 0, "rentals_handgun" : 0, "rentals_long_gun" : 0, "private_sale_handgun" : 3, "private_sale_long_gun" : 12, "private_sale_other" : 3, "return_to_seller_handgun" : 0, "return_to_se
ller_long_gun" : 0, "return_to_seller_other" : 0, "totals" : 24043 }
```

Results

I was able to pull out the same output, although I have received a lot of errors at first. Firstly, loading the csv data into my database took some time, even though I reloaded it for 30 minutes. After I solved that problem, there was a problem when I executed my commands, first one was that sql did not understand ‘state’ and ‘month’ as a column, instead executed as an inbuilt function which made me change the column names to ‘states’ and ‘months’. Another error that took me so long to understand was that IST SQL accepts single quotes where my SQLite accepts double quotes. The images can be found at page number 4 and 5.

Discussion

First, both of us both agreed that we like using the DB Browser for SQLite because it felt a lot cleaner and less clunky than other database methods used in the past. This program was very easily navigable and simple to upload our data into. It was also a lot faster and smoother, I would assume because we aren’t “logged into” a PSU computer. I think for circumstances involving the organization of data, the DB Browser would be better due to a simple infrastructure and easy usage once the data is imported.

The obvious differences between the NoSQL tool and the relational tool is the look and how the data is interpreted. I’d argue that the relational tool has a very clean cut look to it that helps look at the data. The SQL tool also had many buttons, or different methods of using the program, it was much simpler to use. The SQL section of the project took a much shorter duration than its NoSQL counterpart. The NoSQL data tool was all coding and a lot less user friendly. Taking a lot more time with this tool and its less-inviting infrastructure sums up the time using MongoDB.

Collaboration report

Both Spencer and Yanki

- On Zoom call while writing the queries and discussing what we need to do
- Figured out what we wanted the queries to be and what we want from the data
- Decided which dataset to use
- Zoom call used for the entire assignment, good communication with each other on what we need to do before the deadlines

Spencer

- Questions about discussion, dataset, and background
- Did first query
- Formatting of the project
- Difference of SQL and NoSQL data tools

Yanki

- Executed first, and second query
- Questions about results and methods
- Downloaded repo from github created the database, columns, and rows
- Executed SQL commands both DB SQLite, IST vlabs, and MongoDB