

Phase 2 of NYC Taxi dataset Analysis

Group 35

Abhishek Gawali

Akhil Shetty

Deepika Kini

Shounak Desai

Phase II

1. Propose a document-oriented model for your dataset and compare it with your relational model. You should also provide code to load your data into this model.
2. Provide a program that issues at least five interesting SQL queries over the previous relational model and propose indexes to speed up query execution (report your timings).
3. Discover and explain functional dependencies and discuss normalization with respect to the relational model you provided in Phase I.
4. Documentation

Code files for the project:

- Mongo:
 - mongoimport : *mongo_import_commands.txt*
 - Cleaning tables in mongo: *mongo_cleaning_commands.txt*
- SQL :
 - Queries:
 - index:

1. Document oriented model:

We are using Mongo for this project phase as a document based nosql database

A) Comparison between mongo and sql database:

Mongo Advantages:

- unrequired fields can be removed for some objects (reduces storage)
- Different datatypes allowed for same column
- Useful if data is in json format and has subdocument (objects within objects)
- Primary key already

Issue using mongo for our dataset:

Ratecode id and are wrongly classified as float but are integers throughout (LocationIDs are correctly classified as Int)

SQL advantage: can set datatype and get an error if data quality issue in datatype

We have no major advantage of using mongo(nosql) since our data is structured

B) Steps to load data:

Step 1: created database in the mongo : NYC_Taxi

Step 2: imported the csv datasets (converted from parquet in phase1) of green, yellow and LV_Hire using mongoimport commands

Step 3: imported the static files: ratecode, payment, trip type, vendor,location, HL Vendor and Base file using mongoimport commands

Step 4: cleaned data:

Used below query to remove data for rows in columns with null/unnecessary values

Removed quotes from some columns in location_lookup

```
db.green_cab.updateMany({congestion_surcharge: 0.0}, { $unset : {  
congestion_surcharge: 0 }})
```

_type	congestion_surcharge
]1.0	[`0.0
]1.0	[`0.0
]1.0	[`2.75
]1.0	[`0.0
]1.0	[`0.0
]1.0	[`0.0
]1.0	[`0.0
]1.0	[`2.75
]1.0	[`2.75
]1.0	[`0.0
]1.0	[`0.0
]1.0	[`2.75
]1.0	[`0.0
]2.0	[`0.0
]1.0	[`0.0
]1.0	[`0.0
]2.0	[`0.0
]1.0	[`0.0
]1.0	[`0.0
Count Documents	⌚ 00:00:00.003

Before

trip_type	congestion_surcharge
]1.0	
]1.0	
]1.0	[`2.75
]1.0	
]1.0	
]1.0	
]1.0	[`2.75
]1.0	[`2.75
]1.0	
]1.0	
]1.0	[`2.75
]1.0	[`2.75
]1.0	
]1.0	
]1.0	[`2.0
]1.0	
]1.0	
]2.0	
]1.0	
]1.0	
]2.0	
]1.0	
]1.0	
Count Documents	⌚ 00:00:00.003

After

Similarly done for many other columns

```
1 > db.getCollection("green_cab").find({})
2 > db.green_cab.find({ehail_fee: {$exists:false}})
3 > db.green_cab.updateMany({ehail_fee: ""}, { $unset : { ehail_fee: 0 } })
4
```

Raw shell output | Shell Output (Documents) | 1 document selected

Step 5: changed datatype for some fields that were wrongly classified as decimal (foreign key fields)

Used the command:

```
db.green_cab.updateMany( {trip_type : { $type: 1 }},{[$set: { trip_type: { $toInt: "$trip_type" }}]})
```

Please check the txt files for all commands

After importing and cleaning, the files look as follows:

Green_cab

The screenshot shows the Studio 3T interface with the following details:

- Toolbar:** Connect, Collection, IntelliShell, SQL, Aggregate, Map-Reduce, Compare, Schema, Reschema, Tasks, Export, Import, Data Masking, SQL Migration, Users, Roles, Feedback, Go to Free.
- Query Bar:** hw6_final_genre > Aggregation: hw6... > IntelliShell: IMDB > IntelliShell: IMDB > location_lookup > lv_hire_vendor > yellow_cab > green_cab > 26
- Visual Query Builder:** Shows the current query structure: IMDB (localhost:27017) > NYC_Taxi > green_cab.
- Result Grid:** The 'green_cab' collection is selected. The results show 26 documents. The columns are:

_id	VendorID	lpep_pickup_datetime	lpep_dropoff_datetime	store_and_fwd_flag	RatecodeID	PULocationID	DOLocationID	passenger_count	trip_distance
642463e19874f	1	2022-01-01 00:4	2022-01-01 00:4	N	1	150	210	1.0	1.3
642463e19874f	2	2022-01-01 00:4	2022-01-01 00:4	N	1	112	80	1.0	1.3
642463e19874f	2	2022-01-01 00:4	2022-01-01 00:4	N	1	33	170	1.0	6.26
642463e19874f	2	2022-01-01 00:4	2022-01-01 00:1	N	1	181	181	1.0	1.69
642463e19874f	2	2022-01-01 00:4	2022-01-01 00:4	N	1	40	195	1.0	1.15
642463e19874f	1	2022-01-01 00:4	2022-01-01 00:4	N	1	116	41	1.0	2.1
642463e19874f	2	2022-01-01 00:4	2022-01-01 00:4	N	1	75	74	1.0	0.7
642463e19874f	2	2022-01-01 00:4	2022-01-01 01:0	C	1	256	186	1.0	4.75
642463e19874f	2	2021-12-31 23:4	2021-12-31 23:5	N	1	75	4	1.0	6.03
642463e19874f	2	2022-01-01 00:4	2022-01-01 00:4	N	1	41	116	1.0	2.82
642463e19874f	2	2022-01-01 00:1	2022-01-01 23:1	N	1	74	260	1.0	5.48
642463e19874f	2	2022-01-01 00:4	2022-01-01 01:2	N	1	74	231	1.0	7.54
642463e19874f	1	2022-01-01 00:4	2022-01-01 00:4	N	1	74	42	1.0	0.9
642463e19874f	2	2022-01-01 00:1	2022-01-01 00:4	N	5	213	174	1.0	5.57
642463e19874f	1	2022-01-01 00:4	2022-01-01 01:2	N	1	41	94	1.0	6.2
642463e19874f	2	2022-01-01 00:4	2022-01-01 00:4	N	1	51	185	1.0	1.87
642463e19874f	2	2022-01-01 00:4	2022-01-01 01:1	N	5	185	119	1.0	6.6
642463e19874f	2	2022-01-01 00:4	2022-01-01 00:4	N	1	129	260	1.0	0.26
642463e19874f	2	2022-01-01 01:0	2022-01-01 01:3	N	1	82	260	1.0	1.79
- Operations:** Shows import logs for 'green_cab' from CSV files on March 31, 2023, at 12:57:06 and 12:54:01, with a total of 265 documents imported.

Yellow_cab

Enjoy a 2-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial in My License (in the Help Menu).

Open connections

- _collections (9)
 - > LVHire_cab
 - > green_cab
 - > location_lookup
 - > lv_hire_vendor
 - > payment_lookup
 - > ratecode_lookup
 - > trip_type
 - > vendor_taxi_lookup
 - > yellow_cab
- GridFS Buckets (0)
- System (0)
- Views (0)
- admin
- Operations
- 2023-03-31 12:57:06: Imp Importing 1 unit finished: 1 Total time: 00:00:00.121 ✓ Import from CSV/Users/ Import finished 265 document(s) import Total time: 00:00:00.121
- ✓ Import from CSV/Users/ Import finished 265 document(s) import Total time: 00:00:00.090 ✓ Import from CSV/Users/ Import finished 265 document(s) import Total time: 00:00:00.085
- 2023-03-30 22:06:08: Imp

Result | Query Code | Explain | Table View

yellow_cab > VendorID

_id	VendorID	tppe_pickup_datetime	tppe_dropoff_datetime	passenger_count	trip_distance	RatecodeID	store_and_fwd_flag	PULocationID	DOLocationID
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 00:00:00	1.0	2.1	1	N	236	42
64246175a1187: 2	2	2022-01-01 00:00:00	2022-01-01 00:00:00	1.0	2.71	1	N	166	236
64246175a1187: 2	2	2022-01-01 00:00:00	2022-01-01 01:01:00	1.0	0.97	1	N	166	166
64246175a1187: 2	2	2022-01-01 00:00:00	2022-01-01 01:01:00	1.0	1.91	1	N	141	229
64246175a1187: 2	2	2022-01-01 00:00:00	2022-01-01 01:00:00	3.0	0.82	1	N	114	90
64246175a1187: 2	2	2022-01-01 00:00:00	2022-01-01 00:00:00	3.0	0.73	1	N	234	113
64246175a1187: 2	2	2022-01-01 00:00:00	2022-01-01 01:01:00	2.0	2.16	1	N	246	79
64246175a1187: 2	2	2022-01-01 00:00:00	2022-01-01 00:00:00	4.0	1.43	1	N	43	140
64246175a1187: 2	2	2022-01-01 00:00:00	2022-01-01 01:01:00	1.0	1.58	1	N	239	151
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 00:00:00	3.0	4.2	1	N	148	141
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 01:01:00	2.0	2.2	1	N	237	107
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 00:00:00	1.0	0.2	1	N	7	7
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 00:00:00	2.0	3.9	1	N	107	263
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 00:00:00	2.0	3.2	1	N	263	107
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 01:01:00	1.0	0.0	1	N	161	161
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 00:00:00	1.0	1.2	1	N	161	43
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 00:00:00	1.0	1.7	1	N	239	24
64246175a1187: 1	1	2022-01-01 00:00:00	2022-01-01 00:00:00	1.0	1.6	1	N	239	263
64246175a1187: 2	2	2022-01-01 00:00:00	2022-01-01 00:00:00	1.0	0.78	1	N	236	141

1 document selected | Count Documents | 00:00:00.003

LvHire_cab

Enjoy a 2-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial in My License (in the Help Menu).

Open connections

- _collections (9)
 - > IMDB
 - > NYC_Taxi
 - > LVHire_cab
 - > green_cab
 - > location_lookup
 - > lv_hire_vendor
 - > payment_lookup
 - > ratecode_lookup
 - > trip_type
 - > vendor_taxi_lookup
 - > yellow_cab
 - GridFS Buckets (0)
 - System (0)
 - Views (0)- Operations
- 2023-03-30 22:06:08: Imp Importing 1 unit finished: 1 Total time: 00:01:35.528 ✓ Import JSON done Target: IMDB Total time: 00:01:13.528
- ✓ Import JSON done Target: IMDB Total time: 00:01:13.528 ✓ Import JSON done Target: IMDB Total time: 00:01:13.528
- ✓ Import JSON done Target: IMDB Total time: 00:06:39.033 ✓ Import JSON done Target: IMDB Total time: 00:06:39.033

Result | Query Code | Explain | Table View

LVHire_cab > hvfh_license_num

trip_miles	pickup_datetime1	trip_time	base_passenger_fare	tolls	bcf	sales_tax	congestion_surcharge	airport_fee
4.51	2022-01-01T00:38:50.000Z	762	17.67	0.0	0.53	1.57	2.75	0.0
2.04	2022-01-01T00:32:25.000Z	710	10.64	0.0	0.32	0.94	2.75	0.0
8.79	2022-01-01T00:52:23.000Z	1507	107.56	0.0	0.83	2.45	2.75	0.0
1.65	2022-01-01T00:28:01.000Z	461	9.44	0.0	0.28	0.84	2.75	0.0
11.29	2022-01-01T00:17:02.000Z	1387	34.9	0.0	1.05	3.1	2.75	0.0
0.87	2022-01-01T00:43:20.000Z	251	7.91	0.0	0.24	0.7	0.0	0.0
1.89	2022-01-01T00:52:29.000Z	559	9.71	0.0	0.29	0.86	0.0	0.0
3.585	2022-01-01T00:45:34.000Z	810	27.02	0.0	0.81	2.4	2.75	0.0
0.92	2022-01-01T00:12:03.000Z	1384	30.37	0.0	0.91	2.7	2.75	0.0
1.88	2022-01-01T00:58:26.000Z	537	15.02	0.0	0.45	1.33	0.0	0.0
2.699	2022-01-01T00:34:59.000Z	916	20.47	0.0	0.61	1.82	2.75	0.0
1.65	2022-01-01T00:18:02.000Z	303	7.91	0.0	0.24	0.7	2.75	0.0
8.037	2022-01-01T00:58:09.000Z	1904	24.63	20.0	1.34	0.0	0.0	0.0
2.59	2022-01-01T00:56:30.000Z	476	14.87	0.0	0.45	1.32	0.0	0.0
1.71	2022-01-01T00:30:45.000Z	757	10.05	0.0	0.3	0.89	0.0	0.0
1.69	2022-01-01T00:28:15.000Z	637	9.81	0.0	0.29	0.87	0.0	0.0
0.87	2022-01-01T00:45:55.000Z	472	10.53	0.0	0.32	0.93	0.0	0.0
6.454	2022-01-01T00:52:25.000Z	1055	21.37	0.0	0.72	2.12	2.5	0.0
3.51	2022-01-01T00:47:30.000Z	1159	19.78	0.0	0.59	1.76	0.0	0.0

1 document selected | Count Documents | 00:00:00.010

Payment_lookup

The screenshot shows the Studio 3T interface with the following details:

- Toolbar:** Connect, Collection, IntelliShell, SQL, Aggregate, Map-Reduce, Compare, Schema, Reschema, Tasks, Export, Import, Data Masking, SQL Migration.
- Status Bar:** Enjoy a 2-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial in My License (in the Help Menu).
- Left Sidebar (Open connections):**
 - NYC_Taxi (selected)
 - Collections (9):
 - LVHire_cab
 - green_cab
 - location_lookup
 - lv_hire_vendor
 - payment_lookup (selected)
 - ratecode_lookup
 - trip_type
 - vendor_taxi_lookup
 - yellow_cab
 - GridFS Buckets (0)
 - System (0)
 - Views (0)
- Log Area:**
 - 2023-03-30 22:06:08: Importing 1 unit finished: 1
 - Total time: 00:01:13.528
 - Import JSON done
- Query Editor:** Shows the query `payment_lookup > id`. The results table has columns `_id`, `id`, and `typeofpayment`. The data is as follows:

_id	id	typeofpayment
6424696f2a30b	4	Dispute
6424696f2a30b	1	Credit card
6424696f2a30b	2	Cash
6424696f2a30b	6	Voided trip
6424696f2a30b	3	No charge
6424696f2a30b	5	Unknown

Ratecode_lookup

The screenshot shows the Studio 3T interface with the following details:

- Toolbar:** Connect, Collection, IntelliShell, SQL, Aggregate, Map-Reduce, Compare, Schema, Reschema, Tasks, Export, Import, Data Masking, SQL Migration, Users.
- Status Bar:** Enjoy a 2-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial in My License (in the Help Menu).
- Left Sidebar (Open connections):**
 - NYC_Taxi (selected)
 - Collections (9):
 - LVHire_cab
 - green_cab
 - location_lookup
 - lv_hire_vendor
 - payment_lookup
 - ratecode_lookup (selected)
 - trip_type
 - vendor_taxi_lookup
 - yellow_cab
 - GridFS Buckets (0)
 - System (0)
 - Views (0)
- Log Area:**
 - 2023-03-30 22:06:08: Importing 1 unit finished: 1
 - Total time: 00:01:13.528
 - Import JSON done
- Query Editor:** Shows the query `ratecode_lookup > id`. The results table has columns `_id`, `id`, and `typeservice`. The data is as follows:

_id	id	typeservice
6424695858e7	5	Negotiated fare
6424695858e7	99	Unknown
6424695858e7	2	JFK
6424695858e7	6	Group ride/Share
6424695858e7	1	Standard rate
6424695858e7	3	Newark
6424695858e7	4	Nassau or Westc

Trip_type

Connect Collection IntelliShell SQL Aggregate Map-Reduce Compare Schema Reschema Tasks Export Import

Enjoy a 2-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial.

Open connections

- IMDB
- NYC_Taxi
 - Collections (9)
 - LVHire_cab
 - green_cab
 - location_lookup
 - lv_hire_vendor
 - payment_lookup
 - ratecode_lookup
 - trip_type
 - vendor_taxi_lookup
 - yellow_cab
 - GridFS Buckets (0)
 - System (0)
 - Views (0)

Opera

2023-03-30 22:06:08: Importing 1 unit finished: 1
Total time: 00:01:13.528

hw6_final_genre x Aggregation: hw6... x LVHire_cab x IntelliShell: IMDB x In

IMDB (localhost:27017) > NYC_Taxi > trip_type

Run Load query Save query Query history Set default query Copy

Query {}
Projection {}
Skip

Result | Query Code | Explain

trip_type > id

_id	id	value
6424693b00418	2	Dispatch
6424693b00418	1	Street-Hail

Vendor_taxi_lookup

Connect Collection IntelliShell SQL Aggregate Map-Reduce Compare Schema Reschema Tasks Export Import

Enjoy a 2-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial.

Open connections

- IMDB
- NYC_Taxi
 - Collections (9)
 - LVHire_cab
 - green_cab
 - location_lookup
 - lv_hire_vendor
 - payment_lookup
 - ratecode_lookup
 - trip_type
 - vendor_taxi_lookup
 - yellow_cab
 - GridFS Buckets (0)
 - System (0)
 - Views (0)

Opera

2023-03-30 22:06:08: Importing 1 unit finished: 1
Total time: 00:01:13.528

Aggregation: hw6... x LVHire_cab x IntelliShell: IMDB* x IntelliShell: IMDB x In

IMDB (localhost:27017) > NYC_Taxi > vendor_taxi_lookup

Run Load query Save query Query history Set default query Copy

Query {}
Projection {}
Skip

Result | Query Code | Explain

vendor_taxi_lookup > id

_id	id	vendortype
642469128c293	2	VeriFone Inc.
642469128c293	1	Creative Mobile

Lv_hire_vendor

The screenshot shows the MongoDB Compass interface with the following details:

- Toolbar:** Connect, Collection, IntelliShell, SQL, Aggregate, Map-Reduce, Compare, Schema, Reschema, Tasks, Export.
- Top Bar:** Enjoy a 2-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial extension.
- Left Sidebar (Open connections):** IMDB localhost:27017 [direct], BioData, IMDB, NYC_Taxi (Collections: LVHire_cab, green_cab, location_lookup, lv_hire_vendor, payment_lookup, ratecode_lookup, trip_type, vendor_taxi_lookup, yellow_cab), GridFS Buckets (0).
- Current Collection:** lv_hire_vendor
- Query Bar:** Run, Load query, Save query, Query history, Set default query.
- Result Panel:** Shows the results of the query "lv_hire_vendor > high_volume_license_number". The table has columns: _id, high_volume_license_number, app_company_affiliation. The data is as follows:

_id	high_volume_license_number	app_company_affiliation
642469afdf4bd	HV0005	Lyft
642469afdf4bd	HV0004	Via
642469afdf4bd	HV0003	Uber
642469afdf4bd	HV0002	Juno

Location_lookup

Connect Collection IntelliShell SQL Aggregate Map-Reduce Compare Schema Reschema Tasks Export Import Data Masking SQL Migration Users Roles Feedback Go to Free

Enjoy a 2-day full product trial. Once the trial is complete, you will be switched to Studio 3T Free or you can switch now by disabling the trial in My License (in the Help Menu).

Open connections IMDB localhost:27017 [direct]

- BioData
- IMDB
- NYC_Taxi
 - Collections (9)
 - LV-hire_cab
 - green_cab
 - location_lookup
 - lv_hire_vendor
 - payment_lookup
 - ratecode_lookup
 - trip_type
 - vendor_taxi_lookup
 - yellow_cab
 - GridFS Buckets (0)
- Opera
- Operations

2023-03-31 12:57:06: Importing 1 unit finished: 1 Total time: 00:00:00.121 ✓ Import from CSV/Users/ Import finished 265 document(s) import Total time: 00:00:00.121 ✓ 2023-03-31 12:54:01: Importing 1 unit finished: 1 Total time: 00:00:00.090 ✓ Import from CSV/Users/ Import finished 265 document(s) import Total time: 00:00:00.085 ← 2023-03-30 22:06:08: Importing 1 unit finished: 1 Total time: 00:00:00.085

Aggregation: movies > hw6_final_genre > Aggregation: hw6... > IntelliShell: IMDB > IntelliShell: IMDB* > IntelliShell: IMDB > location_lookup > location_lookup > 25

Run Load query Save query Query history Set default query Copy Paste Visual Query Builder

Query Projection Sort Skip Limit Result Query Code Explain

location_lookup > locationid

_id	locationid	borough	zone	service_zone
642710e269a70 1	EWR	Newark Airport	EWR	
642710e269a70 2	Queens	Jamaica Bay	Boro Zone	
642710e269a70 3	Bronx	Allerton/Pelham	Boro Zone	
642710e269a70 4	Manhattan	Alphabet City	Yellow Zone	
642710e269a70 5	Staten Island	Arden Heights	Boro Zone	
642710e269a70 6	Staten Island	Arrochar/Fort W.	Boro Zone	
642710e269a70 7	Queens	Astoria	Boro Zone	
642710e269a70 8	Queens	Astoria Park	Boro Zone	
642710e269a70 9	Queens	Auburndale	Boro Zone	
642710e269a70 10	Queens	Baisley Park	Boro Zone	
642710e269a70 11	Brooklyn	Bath Beach	Boro Zone	
642710e269a70 12	Manhattan	Battery Park	Yellow Zone	
642710e269a70 13	Manhattan	Battery Park Cty	Yellow Zone	
642710e269a70 14	Brooklyn	Bay Ridge	Boro Zone	
642710e269a70 15	Queens	Bay Terrace/Fort	Boro Zone	
642710e269a70 16	Queens	Bayside	Boro Zone	
642710e269a70 17	Brooklyn	Bedford	Boro Zone	
642710e269a70 18	Bronx	Bedford Park	Boro Zone	
642710e269a70 19	Queens	Bellerose	Boro Zone	

1 document selected Count Documents 00:00:00.003

2. SQL queries and optimization using Indices:

1) Payment Analysis:

We have analyzed the payment options, total payment and total number of payments for every cab.

select * from

```
(select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'yellow' as cab , 'credit card' as typeofpayment  
from yellowcab y join paymentlookup p on p.id=y.payment_type
```

and y.payment_type=1

union

```
select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'yellow' as cab , 'cash' as typeofpayment from  
yellowcab y join paymentlookup p on p.id=y.payment_type
```

and y.payment_type=2

union

```
select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'yellow' as cab , 'No charge' as typeofpayment  
from yellowcab y join paymentlookup p on p.id=y.payment_type
```

and y.payment_type=3

union

```
select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'yellow' as cab , 'dispute' as typeofpayment from  
yellowcab y join paymentlookup p on p.id=y.payment_type
```

and y.payment_type=4

union

```
select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'yellow' as cab , 'unknown' as typeofpayment from  
yellowcab y join paymentlookup p on p.id=y.payment_type
```

and y.payment_type=5

union

```
select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'yellow' as cab , 'void' as typeofpayment from  
yellowcab y join paymentlookup p on p.id=y.payment_type
```

and y.payment_type=6

union

```
select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'green' as cab , 'credit card' as typeofpayment  
from greencab y join paymentlookup p on p.id=y.payment_type
```

```
and y.payment_type=1

union

select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'green' as cab , 'cash' as typeofpayment from
greencab y join paymentlookup p on p.id=y.payment_type

and y.payment_type=2

union

select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'green' as cab , 'No charge' as typeofpayment
from greencab y join paymentlookup p on p.id=y.payment_type

and y.payment_type=3

union

select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'green' as cab , 'dispute' as typeofpayment from
greencab y join paymentlookup p on p.id=y.payment_type

and y.payment_type=4

union

select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'green' as cab , 'Unknown' as typeofpayment from
greencab y join paymentlookup p on p.id=y.payment_type

and y.payment_type=5

union

select count(1) as cnt,sum(y.total_amount+y.tip_amount) as amt, 'green' as cab , 'void' as typeofpayment from
greencab y join paymentlookup p on p.id=y.payment_type

and y.payment_type=6

union

select count(1) ,sum(base_passenger_fare+tolls+bcf+sales_tax+congestion_surcharge+tips+airport_fee)as
amt,'forhire' as cab , 'credit' as typeofpayment from lvhiretaxi

union

select count(1) ,sum(base_passenger_fare+tolls+bcf+sales_tax+congestion_surcharge+tips+airport_fee)as
amt,'forhire_uber' as cab , 'credit' as typeofpayment from lvhiretaxi

where hvfhs_license_num ='HV0003'

union

select count(1) ,sum(base_passenger_fare+tolls+bcf+sales_tax+congestion_surcharge+tips+airport_fee)as
amt,'forhire_lyft' as cab , 'credit' as typeofpayment from lvhiretaxi

where hvfhs_license_num ='HV0005'

)x order by cab,typeofpayment
```

Output:

ABC cab	ABC paymentoption	123 totalpayment	123 countofpayment
forhire	credit	306,801,508.27	12,637,977
forhire_lyft	credit	82,285,840.68	3,394,738
forhire_uber	credit	224,515,667.59	9,243,239
green	cash	311,101.87	21,638
green	credit card	700,667.61	34,108
green	dispute	733.94	134
green	No charge	-84.63	319
green	Unknown	27.85	1
green	void	[NULL]	0
yellow	cash	3,872,255.87	233,407
yellow	credit card	18,648,504.68	804,905
yellow	dispute	369,933.07	4,875
yellow	No charge	44,037.39	5,388

Indexing:

The query initially ran for **53 seconds**. After applying the below index the query ran for **31 seconds**.

create index yellow_payment **on** yellowcab

```
(  
total_amount,  
tip_amount  
);
```

create index green_payment **on** greencab

```
(  
total_amount,  
tip_amount  
);
```

```

create index forhire_payment on lvhiretaxi
(
base_passenger_fare,
tolls,
bcf,
sales_tax,
congestion_surcharge,
tips,
airport_fee
);

```

2) Average time delay analysis:

We have analyzed the average time difference between the requested time and the pickup time for LVhire cabs for different locations.

```

select

case when Cab='HV0003' then 'Uber'
else 'Lyft'

end,

Borough,

Average_time_diff_in_minutes_bet_Req_and_Pickup

from

(select l.hvfhhs_license_num as Cab ,l2.borough as Borough,avg(extract(minute from pickup_datetime::time
-request_datetime::time)) as Average_time_diff_in_minutes_bet_Req_and_Pickup from lvhiretaxi |

join locationlookup l2

on

l.pulocationid =l2.locationid

group by l.hvfhhs_license_num ,l2.borough

)x;

```

Output:

ABC cab	ABC borough	123 average_time_diff_in_minutes_bet_req_and_pickup
Uber	"Bronx"	3.465647741
Uber	"Brooklyn"	3.3228500899
Uber	"Manhattan"	2.8668117711
Uber	"Queens"	3.626435703
Uber	"Staten Island"	4.7685436594
Uber	"Unknown"	6.4224137931
Lyft	"Bronx"	4.6021455779
Lyft	"Brooklyn"	4.0903635342
Lyft	"Manhattan"	3.4964322801
Lyft	"Queens"	4.4315833317
Lyft	"Staten Island"	5.7869834828
Lyft	"Unknown"	5.456445993

Indexing:

The query initially ran for **65 seconds**. After applying the below index the query ran for **45 seconds**.

```
create index lvhire_lookup on lvhiretaxi(  
hvhs_license_num,  
pickup_datetime,  
request_datetime,  
pulocationid  
)
```

3) Trip Distance analysis:

We have analyzed the total miles covered by all cabs.

```
select y.cab as cab ,y.dst as distance_in_miles from (select sum(trip_distance) as dst , 'yellow' as cab from yellowcab y  
union  
select sum(trip_distance) as dst,'green' as cab from greencab g
```

```

union

select sum(trip_miles) as dst,'forhire' as cab from lvhiretaxi l

union

select sum(trip_miles) as dst,'forhire_uber' as cab from lvhiretaxi l where hvfhs_license_num ='HV0003'

union

select sum(trip_miles) as dst,'forhire_lyft' as cab from lvhiretaxi l where hvfhs_license_num ='HV0005'

) y

order by dst desc,cab desc;

```

Output:

abc	cab	123	distance_in_miles
	forhire		58,797,246.035
	forhire_uber		42,491,370.94
	forhire_lyft		16,305,875.095
	green		4,859,525.73
	yellow		3,419,294.71

Indexing:

The query initially ran for **20 seconds**. After applying the below indexes the query ran for **16 seconds**.

```

create index trip_dist on greencab(trip_distance)

create index trip_dist_yellow on yellowcab(trip_distance)

create index trip_miles_lvhire on lvhiretaxi(trip_miles)

```

4) Pickup Location analysis:

We have analyzed the top 10 pickup locations of 'Uber' and 'Lyft' cabs.

```

select * from (select * from (select 'Uber' as CabType,pulocationid ,l2.borough,l2."zone" ,l2.service_zone ,count(1)
as cnt from lvhiretaxi l join locationlookup l2 on l2.locationid =l.pulocationid
where hvfhs_license_num='HV0003'
group by 2,3,4,5 order by cnt desc limit 10) x
union

```

```

select * from (select 'Lyft' as CabType,pulocationid ,l2.borough,l2."zone" ,l2.service_zone ,count(1) as cnt from
lvhiretaxi l join locationlookup l2 on l2.locationid =l.pulocationid

where hvfhs_license_num='HV0005'

group by 2,3,4,5 order by cnt desc limit 10) x

union

select * from (

select 'Juno' as CabType,pulocationid ,l2.borough,l2."zone" ,l2.service_zone ,count(1) as cnt from lvhiretaxi l join
locationlookup l2 on l2.locationid =l.pulocationid

where hvfhs_license_num='HV0002'

group by 2,3,4,5) x

) y order by cnt desc,cabtype desc

```

Output:

1	cabtype	123 pulocationid	borough	zone	service_zone	123 cnt
1	Uber	132	"Queens"	"JFK Airport"	"Airports"	139,968
2	Uber	79	"Manhattan"	"East Village"	"Yellow Zone"	137,397
3	Uber	61	"Brooklyn"	"Crown Heights North"	"Boro Zone"	127,363
4	Uber	138	"Queens"	"LaGuardia Airport"	"Airports"	115,764
5	Uber	231	"Manhattan"	"TriBeCa/Civic Center"	"Yellow Zone"	111,738
6	Uber	249	"Manhattan"	"West Village"	"Yellow Zone"	103,742
7	Uber	234	"Manhattan"	"Union Sq"	"Yellow Zone"	102,892
8	Uber	230	"Manhattan"	"Times Sq/Theatre District"	"Yellow Zone"	100,807
9	Uber	37	"Brooklyn"	"Bushwick South"	"Boro Zone"	98,707
10	Uber	68	"Manhattan"	"East Chelsea"	"Yellow Zone"	98,097
11	Lyft	61	"Brooklyn"	"Crown Heights North"	"Boro Zone"	59,312
12	Lyft	79	"Manhattan"	"East Village"	"Yellow Zone"	50,682
13	Lyft	132	"Queens"	"JFK Airport"	"Airports"	49,367
14	Lyft	37	"Brooklyn"	"Bushwick South"	"Boro Zone"	45,818
15	Lyft	138	"Queens"	"LaGuardia Airport"	"Airports"	43,905
16	Lyft	76	"Brooklyn"	"East New York"	"Boro Zone"	43,530
17	Lyft	7	"Queens"	"Astoria"	"Boro Zone"	40,375
18	Lyft	181	"Brooklyn"	"Park Slope"	"Boro Zone"	40,233
19	Lyft	42	"Manhattan"	"Central Harlem North"	"Boro Zone"	37,021

Indexing:

The query initially ran for **30 seconds**. After applying the below index the query ran for **11 seconds**.

```
create index location_index on locationlookup(locationid,borough);
```

5) Booking frequency analysis:

We have analyzed the numbers of cabs that are booked on weekdays compared to weekends.

```
select * from

(select 'Weekends_yellow Cab' cab, count(z.val_pickup::text ) from

(select to_Char(x.pickup,'Day')as val_pickup,to_char(x.dropoff,'Day') from

(select tpep_pickup_datetime::date as pickup,tpep_dropoff_datetime::date as dropoff from yellowcab y)x)z

where z.val_pickup like 'Sat%' or z.val_pickup ='Sun%'

union

select 'WeekDays_yellow Cab' cab,count(z.val_pickup::text )from

(select to_Char(x.pickup,'Day')as val_pickup,to_char(x.dropoff,'Day') from

(select tpep_pickup_datetime::date as pickup,tpep_dropoff_datetime::date as dropoff from yellowcab y)x)z

where z.val_pickup not like 'Sat%' and not z.val_pickup ='Sun%'

union

select 'Weekends_green Cab' cab, count(z.val_pickup::text ) from

(select to_Char(x.pickup,'Day')as val_pickup,to_char(x.dropoff,'Day') from

(select lpep_pickup_datetime::date as pickup,lpep_dropoff_datetime::date as dropoff from greencab y)x)z

where z.val_pickup like 'Sat%' or z.val_pickup ='Sun%'

union

select 'WeekDays_green_Cab' cab,count(z.val_pickup::text )from

(select to_Char(x.pickup,'Day')as val_pickup,to_char(x.dropoff,'Day') from

(select lpep_pickup_datetime::date as pickup,lpep_dropoff_datetime::date as dropoff from greencab y)x)z

where z.val_pickup not like 'Sat%' and not z.val_pickup ='Sun%'

union

select 'Weekends_for hire Cab' cab, count(z.val_pickup::text ) from

(select to_Char(x.pickup,'Day')as val_pickup,to_char(x.dropoff,'Day') from

(select pickup_datetime::date as pickup,dropoff_datetime::date as dropoff from lvhiretaxi y)x)z

where z.val_pickup like 'Sat%' or z.val_pickup ='Sun%'

union

select 'WeekDays_hire Cab' cab,count(z.val_pickup::text )from
```

```

(select to_Char(x.pickup,'Day')as val_pickup,to_char(x.dropoff,'Day') from
(select pickup_datetime::date as pickup,dropoff_datetime::date as dropoff from lvhiretaxi y)x)z
where z.val_pickup not like 'Sat%' and not z.val_pickup ='Sun%'
) outer_q

order by 1

```

Output:

cab	count
WeekDays_green_Cab	55,029
WeekDays_hire Cab	10,372,034
WeekDays_yellow Cab	885,385
Weekends_for hire Cab	2,265,943
Weekends_green Cab	7,466
Weekends_yellow Cab	163,190

Indexing:

The query initially ran for **170 seconds**. After applying the below indexes the query ran for **153 seconds**.

create index yellow_time **on** yellowcab(tpep_pickup_datetime,tpep_dropoff_datetime)

create index green_time **on** greencab(lpep_pickup_datetime,lpep_dropoff_datetime)

create index lvhire_time **on** lvhiretaxi(pickup_datetime,dropoff_datetime)

3. FDs and Normalization of Phase 1 data:

1. Payment Lookup

$Id \rightarrow typeofpayment$

$Typeofpayment \rightarrow id$

2. Vendortaxilookup

$Id \rightarrow vendortype$

Vendortype → id

3. Ratecodelookup

Id → typeofservice

Typeofservice → id

4.locationlookup

Location, borough, zone → service_zone

Service_zone → Location, borough, zone

5. Lvhiretaxivendor

high_volume_license_number → app_company_affiliation

App_company_affiliation → high_volume_license_number

6. Triptype

Id → value

Value → id

7. Greencab

passenger_count, trip_distance, fare_amount, extra, mta_tax, tip_amount,
tolls_amount, improvement_surcharge --> total_amount

Artificial_serial_key → rest of the table

8. Yellowcab

passenger_count, trip_distance, fare_amount, extra, mta_tax, tip_amount,
tolls_amount, improvement_surcharge --> total_amount

Artificial_serial_key → rest of the table

9. LvHireTaxi

Artificial_serial_key → rest of the table

1NF -

The data provided in phase 1, clearly all the columns in all the tables had atomic values. Hence the whole data was already in 1NF.

2NF -

During phase 1, there were columns which were categorical in nature but had complete string values inside them. So we created a lookup table for those string values with ids being integers and then included those integers in the main tables. Hence, these lookup tables had the ids as primary keys and the string values were functionally dependent on the ids i.e. primary key. Hence, all these tables were in 2NF format.

3NF -

The tables had no transitivity present, all tables had all non-key columns dependent only on the primary key and since these tables were already in 1NF and 2NF, those tables were also in 3NF. Hence, the complete data is in normalized format.