Title: - NTD_Public Transit Improvement

Introduction:

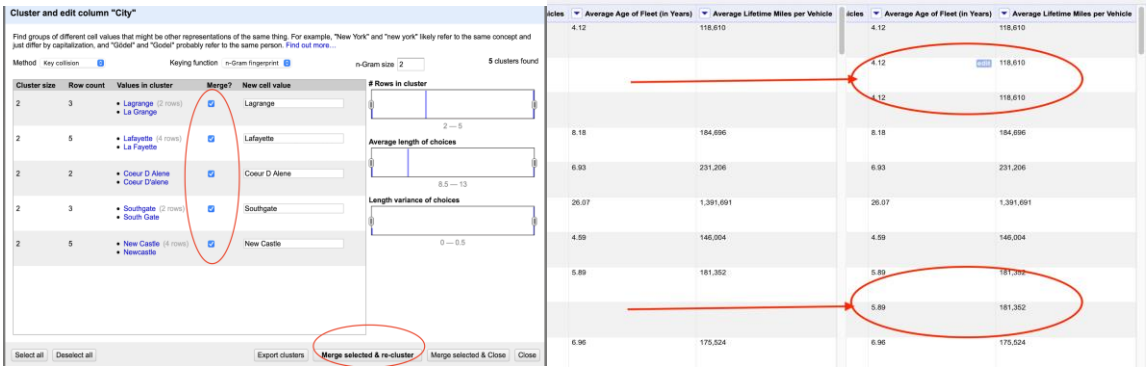
Dataset Overview:

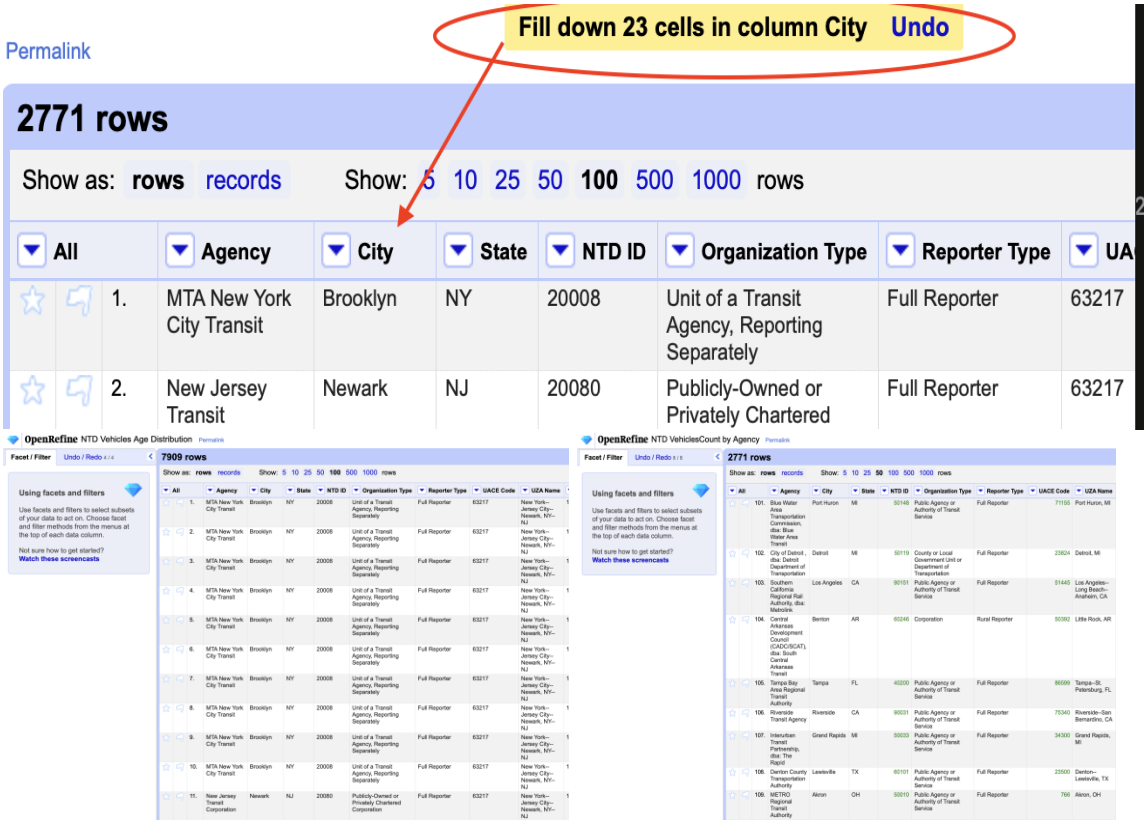
The dataset, "2022 NTD Annual Data – Vehicle Age Distribution & Vehicles Type Count by Agency", likely contains comprehensive information about public transit systems, vehicle types and ages including types of vehicles used by various transit agencies in a particular year. Such datasets typically include:

- Vehicle type (buses & it's types etc.)
- Quantities or capacities
- Operating agencies
- Geographical information
- Operational data (like usage statistics)

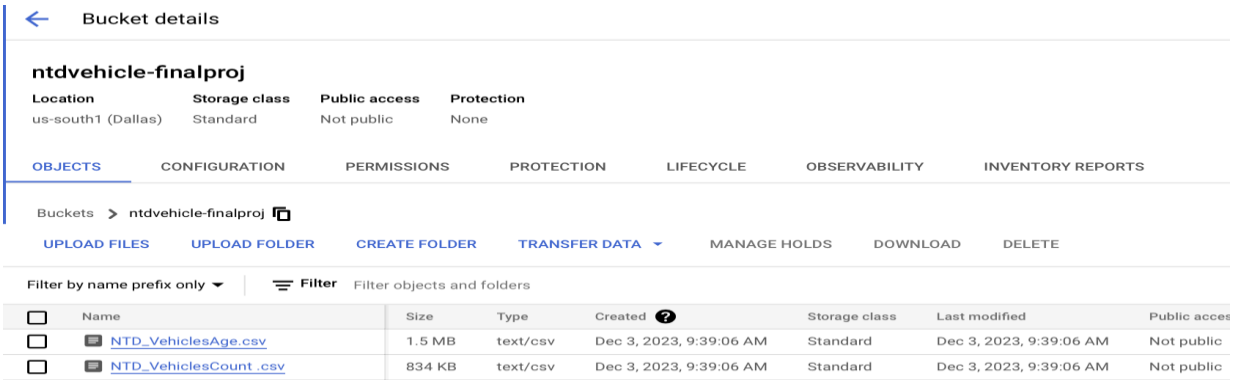
Data Life Cycle:

1. **Data Processing Using Open Refine tool:** It is used to enhance ethe quality of the data like cleaning and transforming. We used our both static and streaming data source in this tool for better refinement and finally ensured that the data is accurate, consistent, and suitable for the intended analysis.

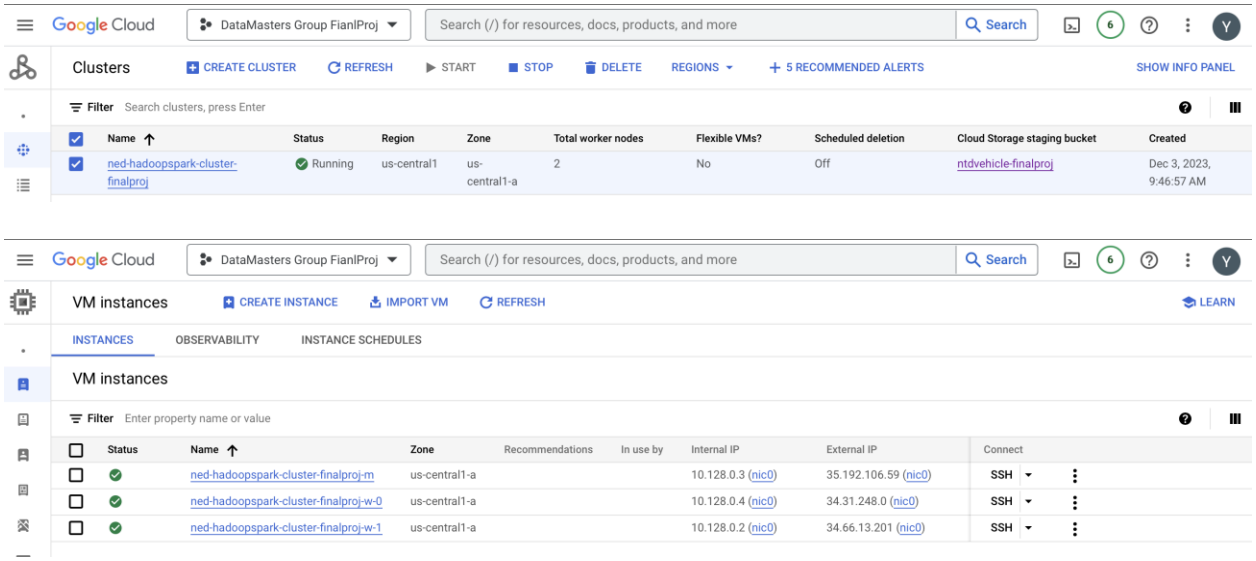




2. **Set up Storage:** Secure and scalable storage is essential for managing large datasets. Having this data in Cloud Storage allows for easy accessibility and manipulation as needed, crucial for transit agencies looking to analyze their fleet and operational efficiency.



3. **Set Up Hadoop Ecosystem (Clusters) using Dataproc:** In this phase, we need to enable the GCP API's called Compute Engine API & Cloud Dataproc API. We created a Cluster in Dataproc and the nodes with 1 Manager & 2 Worker Nodes in Compute Engine→VM Instances. This gives transit agencies real-time insights into their operations, enabling quick decision-making for issues like vehicle deployment and route adjustments.



4. **Creating Sample Queries:** Performing moderate to complex queries and analytics using BigQuery, Hive, and Spark.

   a. Big Query Studio (Running fast, ad-hoc queries on large datasets for operational reports.)

   - For this we enabled Big query API and created a table from both the datasets directly from the cloud storage bucket we created.
   - Performed some queries in Big Query Studio
   - Below are the examples for both the datasets.

Big Queries for Data set 1:

Total Vehicles by state

RUN SAVE QUERY

```
SELECT State, SUM(`Total_Vehicles`) AS TotalVehicles
FROM `VehiclesAge_2.VehiclesAge`
GROUP BY State
ORDER BY TotalVehicles DESC
LIMIT 5;
```

JOB INFORMATION | RESULTS | CHART PREVIEW

Row	State	TotalVehicles
1	CA	26187
2	NY	24551
3	IL	11408
4	TX	10030
5	WA	9222

Top Cities with more than 1...les

RUN

```
SELECT City, SUM(`Total_Vehicles`) AS TotalVehicles
FROM `VehiclesAge_2.VehiclesAge`
GROUP BY City
HAVING SUM(`Total_Vehicles`) > 1000
ORDER BY SUM(`Total_Vehicles`) DESC
LIMIT 5;
```

Row	City	TotalVehicles
1	Brooklyn	14358
2	Chicago	6137
3	Los Angeles	5615
4	Newark	5487
5	Washington	4764

Big Studio Queries for Dataset 2:

List of Agenci... RUN

```
1  SELECT Agency, City
2  FROM `VehiclesCount_1.VehiclesCount`
3  WHERE City = 'Los Angeles';
```

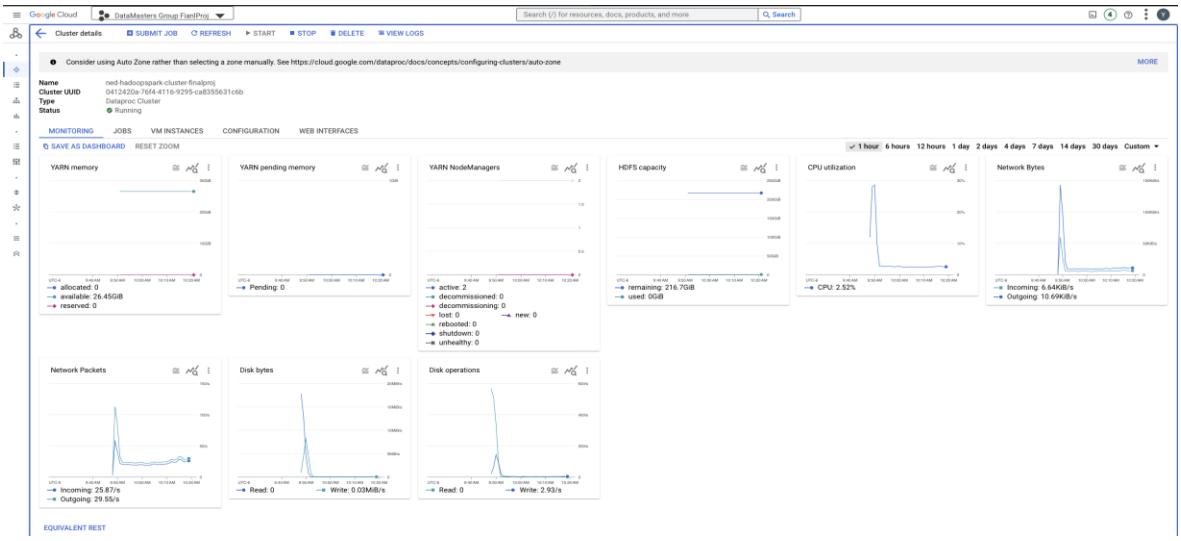
Row	Agency	City
1	City of Los Angeles, dba: City of Los Angeles Department of Transportation	Los Angeles
2	Los Angeles County Metropolitan Transportation Authority , dba: Metro	Los Angeles
3	Southern California Regional R...	Los Angeles

Count of Agencies by State

```
1  SELECT State, COUNT(*) as AgencyCount
2  FROM `VehiclesCount_1.VehiclesCount`
3  GROUP BY State
4  ORDER BY AgencyCount DESC
5  LIMIT 5;
6
```

Row	State	AgencyCount
1	CA	218
2	MI	197
3	WI	163
4	OH	153
5	NC	101

After this we can monitor the cluster and it's performance if needed.



b. Hive (Managing and querying structured data for detailed analysis, like vehicle utilization rates.)

- Accessing Hive through SSH on the created Dataproc cluster.
- Use the Hive command line to execute Hive queries.
- Below are the 3 example's for both the datasets.
  We use the following Hive command: beeline -u jdbc:hive2://localhost:10000

```
t@ntd-hadoopspark-cluster-finalproj-m:~$ beeline -u jdbc:hive2://localhost:10000
Connecting to jdbc:hive2://localhost:10000
Connected to: Apache Hive (version 3.1.3)
Driver: Hive JDBC (version 3.1.3)
Transaction isolation: TRANSACTION_REPEATABLE_READ
Beeline version 3.1.3 by Apache Hive
0: jdbc:hive2://localhost:10000>
```

Queries & Output in Hive: -

3 Hive Queries for Dataset-1 :

```
0: jdbc:hive2://localhost:10000> CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehicles
. . . . . .> (ntdvehicles string)
. . . . . .> ROW FORMAT DELIMITED
. . . . . .> STORED AS TEXTFILE
. . . . . .> LOCATION '/user/yashwanthjilla_unt/data/NTD_VehiclesCount/';
INFO  : Compiling command(queryId=hive_20231203194153_d45e8c8b-b373-4777-9ce6-e6aa8619ed66): CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehicles
(ntdvehicles string)
ROW FORMAT DELIMITED
STORED AS TEXTFILE
LOCATION '/user/yashwanthjilla_unt/data/NTD_VehiclesCount/'
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:null, properties:null)
INFO  : Completed compiling command(queryId=hive_20231203194153_d45e8c8b-b373-4777-9ce6-e6aa8619ed66); Time taken: 0.216 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Executing command(queryId=hive_20231203194153_d45e8c8b-b373-4777-9ce6-e6aa8619ed66): CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehicles
(ntdvehicles string)
ROW FORMAT DELIMITED
STORED AS TEXTFILE
LOCATION '/user/yashwanthjilla_unt/data/NTD_VehiclesCount/'
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20231203194153_d45e8c8b-b373-4777-9ce6-e6aa8619ed66); Time taken: 1.877 seconds
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manager
No rows affected (2.787 seconds)
0: jdbc:hive2://localhost:10000>
```

```
0: jdbc:hive2://localhost:10000>
0: jdbc:hive2://localhost:10000> SELECT * FROM ntd_vehicles LIMIT 1;
INFO  : Compiling command(queryId=hive_20231203194616_902bd8b8-a2fb-402c-a8cf-59aa96d0be04): SELECT * FROM ntd_vehicles LIMIT 1
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name:ntd_vehicles.ntdvehicles, type:string, comment:null)], properties:null)
INFO  : Completed compiling command(queryId=hive_20231203194616_902bd8b8-a2fb-402c-a8cf-59aa96d0be04); Time taken: 2.323 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Executing command(queryId=hive_20231203194616_902bd8b8-a2fb-402c-a8cf-59aa96d0be04): SELECT * FROM ntd_vehicles LIMIT 1
INFO  : Query ID = hive_20231203194616_902bd8b8-a2fb-402c-a8cf-59aa96d0be04
INFO  : Total jobs = 1
INFO  : Launching Job 1 out of 1
INFO  : Starting task [Stage-1:MAPRED] in serial mode
INFO  : Subscribed to counters: [] for queryId: hive_20231203194616_902bd8b8-a2fb-402c-a8cf-59aa96d0be04
INFO  : Tez session hasn't been created yet. Opening session
INFO  : Dag name: SELECT * FROM ntd_vehicles LIMIT 1 (Stage-1)
INFO  : Status: Running (Executing on YARN cluster with App id application_1701629121542_0001)

INFO  : Completed executing command(queryId=hive_20231203194616_902bd8b8-a2fb-402c-a8cf-59aa96d0be04); Time taken: 22.704 seconds
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manager
----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 01/01  [==========================>>] 100%  ELAPSED TIME: 12.94 s
----------------------------------------------------------------------------------------------
+----------------------------------------------------+
|                  ntd_vehicles.ntdvehicles          |
+----------------------------------------------------+
| Agency,City,State,NTD ID,Organization Type,Reporter Type,UACE Code,UZA Name,Primary UZA Population,Agency VOMS,Bus,Bus with ULB Reported,Bus >= ULB,Articulated Bus,Articulated Bu
s with ULB Reported,Articulated Bus >= ULB,Over-the-Road Bus,Over-the-Road Bus with ULB Reported,Over-the-Road Bus >= ULB,Double Decker Bus,Double Decker Bus with ULB Reported,Doub
le Decker Bus >= ULB,School Bus,SchoolBus with ULB Reported,School Bus >= ULB,Van,Van with ULB Reported,Van >= ULB,Cutaway,Cutaway with ULB Reported,Cutaway >= ULB,Automobile,Autom
obile with ULB Reported,Automobile >= ULB,Minivan,Minivan with ULB Reported,Minivan >= ULB,Sport Utility Vehicle,Sport Utility Vehicle with ULB Reported,Sport Utility Vehicle >= UL
B,Trolleybus,TrolleyBus with ULB Reported,Trolleybus >= ULB,Heavy Rail Passenger Car,Heavy Rail Passenger Car with ULB Reported,Heavy Rail Passenger Car >= ULB,Light Rail Vehicle,L
ight Rail Vehicle with ULB Reported,Light Rail Vehicle >= ULB,Commuter Rail Passenger Coach,Commuter Rail Passenger Coach with ULB Reported,Commuter Rail Passenger Coach >= ULB,Com
muter Rail Self-Propelled Passenger Car,Commuter Rail Self-Propelled Passenger Car with ULB Reported,Commuter Rail Self-Propelled Passenger Car >= ULB,Locomotive,Locomotive with UL
B Reported,Locomotive >= ULB,Automated Guideway Vehicle,Automated Guideway Vehicle with ULB Reported,Automated Guideway Vehicle >= ULB,Vintage/Historic Trolley,Vintage/Historic Tro
lley with ULB Reported,Vintage/Historic Trolley >= ULB,Streetcar,Streetcar with ULB Reported,Streetcar >= ULB,Aerial Tramway,Aerial Tramway with ULB Reported,Aerial Tramway >= ULB,
Monorail,Monorail >= ULB,Cable Car,Cable Car with ULB Reported,Cable Car >= ULB,Inclined Plane,Inclined Plane with ULB Reported,Inclined Plane >= ULB,Ferryboat,Ferryboat with ULB R
eported,Ferryboat >= ULB,Other,Other with ULB Reported,Other >= ULB,Total Revenue Vehicles,Total with ULB Reported,Total Revenue Vehicles >= ULB,Automobiles,Automobiles >= ULB,Truc
ks and Other Rubber Tire Vehicles,Trucks and Other Rubber Tire Vehicles >= ULB,Steel Wheel Vehicles,Steel Wheel Vehicles >= ULB,Total Service Vehicles,Total Service Vehicles >= ULB
 |
+----------------------------------------------------+
1 row selected (33.105 seconds)
0: jdbc:hive2://localhost:10000>
```

```
INFO  : Dag name: SELECT * FROM ntd_vehicles LIMIT 5 (Stage-1)
INFO  : Status: Running (Executing on YARN cluster with App id application_1701629121542_0001)

----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 01/01  [==========================>>] 100%  ELAPSED TIME: 10.17 s
----------------------------------------------------------------------------------------------
INFO  : Completed executing command(queryId=hive_20231203195106_58b7a662-e605-4de3-82b1-262eba4d0ec7); Time taken: 10.89 seconds
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manager
+----------------------------------------------------+
|              ntd_vehicles.ntdvehicles              |
+----------------------------------------------------+
| Agency,City,State,NTD ID,Organization Type,Reporter Type,UACE Code,UZA Name,Primary UZA Population,Agency VOMS,Bus,Bus with ULB Reported,Bus >= ULB,Articulated Bus,Articulated Bu
s with ULB Reported,Articulated Bus >= ULB,Over-the-Road Bus,Over-the-Road Bus with ULB Reported,Over-the-Road Bus >= ULB,Double Decker Bus,Double Decker Bus with ULB Reported,Doub
le Decker Bus >= ULB,School Bus,SchoolBus with ULB Reported,School Bus >= ULB,Van,Van with ULB Reported,Van >= ULB,Cutaway,Cutaway with ULB Reported,Cutaway >= ULB,Automobile,Autom
obile with ULB Reported,Automobile >= ULB,Minivan,Minivan with ULB Reported,Minivan >= ULB,Sport Utility Vehicle,Sport Utility Vehicle with ULB Reported,Sport Utility Vehicle >= UL
B,Trolleybus,TrolleyBus with ULB Reported,Trolleybus >= ULB,Heavy Rail Passenger Car,Heavy Rail Passenger Car with ULB Reported,Heavy Rail Passenger Car >= ULB,Light Rail Vehicle,L
ight Rail Vehicle with ULB Reported,Light Rail Vehicle >= ULB,Commuter Rail Passenger Coach,Commuter Rail Passenger Coach with ULB Reported,Commuter Rail Passenger Coach >= ULB,Com
muter Rail Self-Propelled Passenger Car,Commuter Rail Self-Propelled Passenger Car with ULB Reported,Commuter Rail Self-Propelled Passenger Car >= ULB,Locomotive,Locomotive with UL
B Reported,Locomotive >= ULB,Automated Guideway Vehicle,Automated Guideway Vehicle with ULB Reported,Automated Guideway Vehicle >= ULB,Vintage/Historic Trolley,Vintage/Historic Tro
lley with ULB Reported,Vintage/Historic Trolley >= ULB,Streetcar,Streetcar with ULB Reported,Streetcar >= ULB,Aerial Tramway,Aerial Tramway with ULB Reported,Aerial Tramway >= ULB,
Monorail,Monorail >= ULB,Cable Car,Cable Car with ULB Reported,Cable Car >= ULB,Inclined Plane,Inclined Plane with ULB Reported,Inclined Plane >= ULB,Ferryboat,Ferryboat with ULB R
eported,Ferryboat >= ULB,Other,Other with ULB Reported,Other >= ULB,Total Revenue Vehicles,Total with ULB Reported,Total Revenue Vehicles >= ULB,Automobiles,Automobiles >= ULB,Truc
ks and Other Rubber Tire Vehicles,Trucks and Other Rubber Tire Vehicles >= ULB,Steel Wheel Vehicles,Steel Wheel Vehicles >= ULB,Total Service Vehicles,Total Service Vehicles >= ULB
 |
| MTA New York City Transit,Brooklyn,NY,20008,"Unit of a Transit Agency, Reporting Separately",Full Reporter,63217,"New York--Jersey City--Newark, NY--NJ","19,426,449","10,019","3,
131","3,131",367,"1,021","1,021",83,520,520,7,0,0,0,0,0,0,48,48,4,822,822,121,0,0,0,0,0,0,318,318,260,0,0,0,"6,453","4,153",0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,"12,313","10,013",842,434,211,"1,429",480,182,127,"2,045",818 |
| New Jersey Transit Corporation,Newark,NJ,20080,Publicly-Owned or Privately Chartered Corporation,Full Reporter,63217,"New York--Jersey City--Newark, NY--NJ","19,426,449","3,289",
"1,243","1,243",488,110,110,0,"1,106","1,106",230,0,0,0,0,0,0,14,14,0,403,403,254,122,122,49,31,31,0,0,0,0,0,0,0,0,0,0,0,92,92,0,948,883,0,157,157,0,186,171,14,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,"4,412","4,332","1,035",65,36,856,548,69,26,990,610 |
| Washington Metropolitan Area Transit Authority,Washington,DC,30030,Public Agency or Authority of Transit Service,Full Reporter,92242,"Washington--Arlington, DC--VA--MD","5,174,75
9","2,807","1,514","1,514",97,77,77,2,0,0,0,0,0,0,0,0,0,0,588,588,588,0,0,0,177,177,0,0,0,0,0,0,0,0,0,0,0,634,634,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,"2,990","2,990",687,138,76,"1,376",620,86,23,"1,600",719 |
| "Los Angeles County Metropolitan Transportation Authority , dba: Metro",Los Angeles,CA,90154,Public Agency or Authority of Transit Service,Full Reporter,51445,"Los Angeles--Long
Beach--Anaheim, CA","12,237,376","2,509","1,841","1,841",28,268,268,95,0,0,0,0,0,0,0,0,0,0,806,127,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,84,84,14,247,247,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,"3,246","2,567",137,459,171,970,390,10,2,"1,439",563 |
+----------------------------------------------------+
5 rows selected (11.205 seconds)
0: jdbc:hive2://localhost:10000>
```

3 Queries for Dataset-2 in Hive:

```
. . . . . . . . . . . . . . . . .>
. . . . . . . . . . . . . . . . .> CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehiclesAge
. . . . . . . . . . . . . . . . .> (ntdvehiclesAge string)
. . . . . . . . . . . . . . . . .> ROW FORMAT DELIMITED
. . . . . . . . . . . . . . . . .> STORED AS TEXTFILE
. . . . . . . . . . . . . . . . .> LOCATION '/user/yashwanthjilla_unt/data/NTD_VehiclesAge/';
Error: Error while compiling statement: FAILED: ParseException line 2:23 missing EOF at 'ashwanthjilla_unt' near ')' (state=42000,code=40000)
0: jdbc:hive2://localhost:10000> CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehiclesAge
. . . . . . . . . . . . . . . . .> (ntdvehiclesAge STRING)
. . . . . . . . . . . . . . . . .> ROW FORMAT DELIMITED
. . . . . . . . . . . . . . . . .> STORED AS TEXTFILE
. . . . . . . . . . . . . . . . .> LOCATION '/user/yashwanthjilla_unt/data/NTD_VehiclesAge/';
INFO  : Compiling command(queryId=hive_20231203200155_027c678d-fa2f-4731-9877-3d4796397070): CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehiclesAge
(ntdvehiclesAge STRING)
ROW FORMAT DELIMITED
STORED AS TEXTFILE
LOCATION '/user/yashwanthjilla_unt/data/NTD_VehiclesAge/'
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:null, properties:null)
INFO  : Completed compiling command(queryId=hive_20231203200155_027c678d-fa2f-4731-9877-3d4796397070); Time taken: 0.03 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Executing command(queryId=hive_20231203200155_027c678d-fa2f-4731-9877-3d4796397070): CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehiclesAge
(ntdvehiclesAge STRING)
ROW FORMAT DELIMITED
STORED AS TEXTFILE
LOCATION '/user/yashwanthjilla_unt/data/NTD_VehiclesAge/'
INFO  : Starting task [Stage-0:DDL] in serial mode
INFO  : Completed executing command(queryId=hive_20231203200155_027c678d-fa2f-4731-9877-3d4796397070); Time taken: 0.098 seconds
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manager
No rows affected (0.145 seconds)
0: jdbc:hive2://localhost:10000>
```

```
0: jdbc:hive2://localhost:10000> SELECT COUNT(*) FROM ntd_vehiclesAge;
INFO  : Compiling command(queryId=hive_20231203200839_a6d5140a-43db-41eb-b070-f7731f0f411c): SELECT COUNT(*) FROM ntd_vehiclesAge
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name:_c0, type:bigint, comment:null)], properties:null)
INFO  : Completed compiling command(queryId=hive_20231203200839_a6d5140a-43db-41eb-b070-f7731f0f411c); Time taken: 0.728 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Executing command(queryId=hive_20231203200839_a6d5140a-43db-41eb-b070-f7731f0f411c): SELECT COUNT(*) FROM ntd_vehiclesAge
INFO  : Query ID = hive_20231203200839_a6d5140a-43db-41eb-b070-f7731f0f411c
INFO  : Total jobs = 1
INFO  : Launching Job 1 out of 1
INFO  : Starting task [Stage-1:MAPRED] in serial mode
INFO  : Subscribed to counters: [] for queryId: hive_20231203200839_a6d5140a-43db-41eb-b070-f7731f0f411c
INFO  : Session is already open
INFO  : Dag name: SELECT COUNT(*) FROM ntd_vehiclesAge (Stage-1)
INFO  : Tez session was closed. Reopening...
INFO  : Session re-established.
INFO  : Session re-established.
INFO  : Status: Running (Executing on YARN cluster with App id application_1701629121542_0002)

INFO  : Completed executing command(queryId=hive_20231203200839_a6d5140a-43db-41eb-b070-f7731f0f411c); Time taken: 15.106 seconds
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manager
----------------------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
----------------------------------------------------------------------------------------------
Map 1 .......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
----------------------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 7.93 s
----------------------------------------------------------------------------------------------
+-------+
|  _c0  |
+-------+
| 7910  |
+-------+
1 row selected (17.672 seconds)
0: jdbc:hive2://localhost:10000>
```

```
0: jdbc:hive2://localhost:10000> SELECT AVG(LENGTH(ntdvehiclesAge)) AS avg_length FROM ntd_vehiclesAge;
INFO  : Compiling command(queryId=hive_20231203201239_d017061f-c6d6-4f26-a527-a2933781ff67): SELECT AVG(LENGTH(ntdvehiclesAge)) AS avg_length FROM ntd_vehiclesAge
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Semantic Analysis Completed (retrial = false)
INFO  : Returning Hive schema: Schema(fieldSchemas:[FieldSchema(name:avg_length, type:double, comment:null)], properties:null)
INFO  : Completed compiling command(queryId=hive_20231203201239_d017061f-c6d6-4f26-a527-a2933781ff67); Time taken: 0.214 seconds
INFO  : Concurrency mode is disabled, not creating a lock manager
INFO  : Executing command(queryId=hive_20231203201239_d017061f-c6d6-4f26-a527-a2933781ff67): SELECT AVG(LENGTH(ntdvehiclesAge)) AS avg_length FROM ntd_vehiclesAge
INFO  : Query ID = hive_20231203201239_d017061f-c6d6-4f26-a527-a2933781ff67
INFO  : Total jobs = 1
INFO  : Launching Job 1 out of 1
INFO  : Starting task [Stage-1:MAPRED] in serial mode
INFO  : Subscribed to counters: [] for queryId: hive_20231203201239_d017061f-c6d6-4f26-a527-a2933781ff67
INFO  : Session is already open
INFO  : Dag name: SELECT AVG(LENGTH(ntdvehic...ntd_vehiclesAge (Stage-1)
INFO  : Status: Running (Executing on YARN cluster with App id application_1701629121542_0002)

--------------------------------------------------------------------------------
        VERTICES      MODE        STATUS  TOTAL  COMPLETED  RUNNING  PENDING  FAILED  KILLED
--------------------------------------------------------------------------------
Map 1 ......... container     SUCCEEDED      1          1        0        0       0       0
Reducer 2 ...... container     SUCCEEDED      1          1        0        0       0       0
--------------------------------------------------------------------------------
VERTICES: 02/02  [==========================>>] 100%  ELAPSED TIME: 5.41 s
--------------------------------------------------------------------------------
INFO  : Completed executing command(queryId=hive_20231203201239_d017061f-c6d6-4f26-a527-a2933781ff67); Time taken: 6.2 seconds
INFO  : OK
INFO  : Concurrency mode is disabled, not creating a lock manager
+--------------------+
|     avg_length     |
+--------------------+
| 201.6211125158028  |
+--------------------+
1 row selected (6.467 seconds)
0: jdbc:hive2://localhost:10000>
```

c. Spark (Advanced analytics, possibly including predictive modeling for future fleet requirements.)

- We used the Spark shell for data processing and analysis.

```
yashwanthjilla_unt@ned-hadoopspark-cluster-finalproj-m:~$ spark-sql
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
ivysettings.xml file not found in HIVE_HOME or HIVE_CONF_DIR,/etc/hive/conf.dist/ivysettings.xml will be used
23/12/03 20:17:24 INFO org.apache.spark.SparkEnv: Registering MapOutputTracker
23/12/03 20:17:24 INFO org.apache.spark.SparkEnv: Registering BlockManagerMaster
23/12/03 20:17:25 INFO org.apache.spark.SparkEnv: Registering BlockManagerMasterHeartbeat
23/12/03 20:17:25 INFO org.apache.spark.SparkEnv: Registering OutputCommitCoordinator
Spark master: yarn, Application Id: application_1701629121542_0003
spark-sql> show tables;
default ntd_vehicles    false
default ntd_vehiclesage false
Time taken: 3.862 seconds, Fetched 2 row(s)
spark-sql>
```

For Dataset -1 (3 Queries & output's in Spark)

```
== SQL ==
CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehicles
 (ntdvehicles string)                    /data/NTD_VehiclesCount/`
----------------------^^^

spark-sql> CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehicles
         > (
         >   ntdvehicles STRING
         > )
         > ROW FORMAT DELIMITED
         > STORED AS TEXTFILE
         > LOCATION '/user/                          /NTD_VehiclesCount/';
23/12/03 20:22:10 WARN org.apache.hadoop.hive.ql.session.SessionState: M
thorizerFactory.
Time taken: 0.575 seconds
spark-sql>
```

```
spark-sql>
         > SELECT * FROM ntd_vehicles LIMIT 1;
Agency,City,State,NTD ID,Organization Type,Reporter Type,UACE Code,UZA N
with ULB Reported,Articulated Bus >= ULB,Over-the-Road Bus,Over-the-Road
 Decker Bus >= ULB,School Bus,SchoolBus with ULB Reported,School Bus >=
ile with ULB Reported,Automobile >= ULB,Minivan,Minivan with ULB Reporte
Trolleybus,TrolleyBus with ULB Reported,Trolleybus >= ULB,Heavy Rail Pas
ht Rail Vehicle with ULB Reported,Light Rail Vehicle >= ULB,Commuter Rai
ter Rail Self-Propelled Passenger Car,Commuter Rail Self-Propelled Passe
Reported,Locomotive >= ULB,Automated Guideway Vehicle,Automated Guideway
ey with ULB Reported,Vintage/Historic Trolley >= ULB,Streetcar,Streetcar
norail,Monorail >= ULB,Cable Car,Cable Car with ULB Reported,Cable Car >
orted,Ferryboat >= ULB,Other,Other with ULB Reported,Other >= ULB,Total
 and Other Rubber Tire Vehicles,Trucks and Other Rubber Tire Vehicles >=
Time taken: 5.717 seconds, Fetched 1 row(s)
spark-sql>
```

```
spark-sql> SELECT * FROM ntd_vehicles LIMIT 5;
23/12/03 20:24:32 WARN org.apache.hadoop.util.concurrent.ExecutorHelper: Thread (Thread[GetFileInfo #1,5,main]) interrupted:
java.lang.InterruptedException
        at com.google.common.util.concurrent.AbstractFuture.get(AbstractFuture.java:510)
        at com.google.common.util.concurrent.FluentFuture$TrustedFuture.get(FluentFuture.java:88)
        at org.apache.hadoop.util.concurrent.ExecutorHelper.logThrowableFromAfterExecute(ExecutorHelper.java:48)
        at org.apache.hadoop.util.concurrent.HadoopThreadPoolExecutor.afterExecute(HadoopThreadPoolExecutor.java:90)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1157)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:750)
Agency,City,State,NTD ID,Organization Type,Reporter Type,UACE Code,UZA Name,Primary UZA Population,Agency VOMS,Bus,Bus with ULB Reported,Bus >= ULB,Articulated Bus,Articulated Bus
with ULB Reported,Articulated Bus >= ULB,Over-the-Road Bus,Over-the-Road Bus with ULB Reported,Over-the-Road Bus >= ULB,Double Decker Bus,Double Decker Bus with ULB Reported,Double
 Decker Bus >= ULB,School Bus,SchoolBus with ULB Reported,School Bus >= ULB,Van,Van with ULB Reported,Van >= ULB,Cutaway,Cutaway with ULB Reported,Cutaway >= ULB,Automobile,Automob
ile with ULB Reported,Automobile >= ULB,Minivan,Minivan with ULB Reported,Minivan >= ULB,Sport Utility Vehicle,Sport Utility Vehicle with ULB Reported,Sport Utility Vehicle >= ULB,
Trolleybus,TrolleyBus with ULB Reported,Trolleybus >= ULB,Heavy Rail Passenger Car,Heavy Rail Passenger Car with ULB Reported,Heavy Rail Passenger Car >= ULB,Light Rail Vehicle,Lig
ht Rail Vehicle with ULB Reported,Light Rail Vehicle >= ULB,Commuter Rail Passenger Coach,Commuter Rail Passenger Coach with ULB Reported,Commuter Rail Passenger Coach >= ULB,Commu
ter Rail Self-Propelled Passenger Car,Commuter Rail Self-Propelled Passenger Car with ULB Reported,Commuter Rail Self-Propelled Passenger Car >= ULB,Locomotive,Locomotive with ULB
Reported,Locomotive >= ULB,Automated Guideway Vehicle,Automated Guideway Vehicle with ULB Reported,Automated Guideway Vehicle >= ULB,Vintage/Historic Trolley,Vintage/Historic Troll
ey with ULB Reported,Vintage/Historic Trolley >= ULB,Streetcar,Streetcar with ULB Reported,Streetcar >= ULB,Aerial Tramway,Aerial Tramway with ULB Reported,Aerial Tramway >= ULB,Mo
norail,Monorail >= ULB,Cable Car,Cable Car with ULB Reported,Cable Car >= ULB,Inclined Plane,Inclined Plane with ULB Reported,Inclined Plane >= ULB,Ferryboat,Ferryboat with ULB Rep
orted,Ferryboat >= ULB,Other,Other with ULB Reported,Other >= ULB,Total Revenue Vehicles,Total with ULB Reported,Total Revenue Vehicles >= ULB,Automobiles,Automobiles >= ULB,Trucks
 and Other Rubber Tire Vehicles,Trucks and Other Rubber Tire Vehicles >= ULB,Steel Wheel Vehicles,Steel Wheel Vehicles >= ULB,Total Service Vehicles,Total Service Vehicles >= ULB
MTA New York City Transit,Brooklyn,NY,20008,"Unit of a Transit Agency, Reporting Separately",Full Reporter,63217,"New York--Jersey City--Newark, NY--NJ","19,426,449","10,019","3,13
1","3,131",367,"1,021","1,021",83,520,520,7,0,0,0,0,0,0,48,48,4,822,822,121,0,0,0,0,0,0,318,318,260,0,0,0,"6,453","4,153",0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,"12,313","10,013",842,434,211,"1,429",480,182,127,"2,045",818
New Jersey Transit Corporation,Newark,NJ,20080,Publicly-Owned or Privately Chartered Corporation,Full Reporter,63217,"New York--Jersey City--Newark, NY--NJ","19,426,449","3,289","1
,243","1,243",488,110,110,0,"1,106","1,106",230,0,0,0,0,0,0,14,14,0,403,403,254,122,122,49,31,31,0,0,0,0,0,0,0,0,0,0,92,92,0,948,883,0,157,157,0,186,171,14,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,"4,412","4,332","1,035",65,36,856,548,69,26,990,610
Washington Metropolitan Area Transit Authority,Washington,DC,30030,Public Agency or Authority of Transit Service,Full Reporter,92242,"Washington--Arlington, DC--VA--MD","5,174,759"
,"2,807","1,514","1,514",97,77,77,2,0,0,0,0,0,0,0,0,0,588,588,588,0,0,0,177,177,0,0,0,0,0,0,0,0,0,0,634,634,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,"2,990","2,990",687,138,76,"1,376",620,86,23,"1,600",719
"Los Angeles County Metropolitan Transportation Authority , dba: Metro",Los Angeles,CA,90154,Public Agency or Authority of Transit Service,Full Reporter,51445,"Los Angeles--Long Be
ach--Anaheim, CA","12,237,376","2,509","1,841","1,841",28,268,268,95,0,0,0,0,0,0,0,0,0,806,127,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,84,84,14,247,247,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,"3,246","2,567",137,459,171,970,390,10,2,"1,439",563
Time taken: 0.339 seconds, Fetched 5 row(s)
spark-sql>
```

For Dataset-2 (3 Queries & outputs in Spark)

```
LOCATION '/user/yashwanthjilla_unt/data/NTD_VehiclesAge/'

spark-sql> CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehiclesAge
         > (
         >   ntdvehiclesAge STRING
         > )
         > ROW FORMAT DELIMITED
         > STORED AS TEXTFILE
         > LOCATION '/user/yashwanthjilla_unt/data/NTD_VehiclesAge/';
Time taken: 0.12 seconds
spark-sql> SELECT COUNT(*) FROM ntd_vehiclesAge;
23/12/03 20:27:09 WARN org.apache.hadoop.util.concurrent.ExecutorHelper: Thread (Thread[GetFileInfo #1,5,main]) interrupted:
java.lang.InterruptedException
        at com.google.common.util.concurrent.AbstractFuture.get(AbstractFuture.java:510)
        at com.google.common.util.concurrent.FluentFuture$TrustedFuture.get(FluentFuture.java:88)
        at org.apache.hadoop.util.concurrent.ExecutorHelper.logThrowableFromAfterExecute(ExecutorHelper.java:48)
        at org.apache.hadoop.util.concurrent.HadoopThreadPoolExecutor.afterExecute(HadoopThreadPoolExecutor.java:90)
        at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1157)
        at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:624)
        at java.lang.Thread.run(Thread.java:750)
7910
Time taken: 1.706 seconds, Fetched 1 row(s)
spark-sql>
```

```
spark-sql> CREATE EXTERNAL TABLE IF NOT EXISTS ntd_vehiclesAge
         > (
         >   ntdvehiclesAge STRING
         > )
         > ROW FORMAT DELIMITED
         > STORED AS TEXTFILE
         > LOCATION '/user/[redacted]data/NTD_VehiclesAge/';
Time taken: 0.12 seconds
spark-sql>
```

```
spark-sql> SELECT AVG(LENGTH(ntdvehiclesAge)) AS avg_length FROM ntd_vehiclesAge;
201.6211125158028
Time taken: 0.923 seconds, Fetched 1 row(s)
spark-sql>
```

Comparison of Hive & Spark Execution Time: -

Dataset	Query Description	Hive Execution Time (sec)	Spark Execution Time (sec)
Dataset-1	CREATE EXTERNAL TABLE	2.787	0.575
Dataset-1	Selecting 1 row	33.105	5.717
Dataset-1	Selecting 5 rows	11.205	0.339
Dataset-2	CREATE EXTERNAL TABLE in VehiclesAge	0.145	0.12
Dataset-2	Select Count all from Vehicle Age	17.672	1.706
Dataset-2	Select Average length	6.467	0.923

Key- Analysis: -

- Hive is suitable for traditional data warehousing and SQL queries.
- Spark's in-memory processing makes it a superior choice for tasks involving iterative algorithms and real-time analytics.
- In comparing Hive and Spark based on the provided data and queries, Spark consistently outperforms Hive with faster execution times, particularly for data processing and analytics.
- Developing robust data management through complex queries, deeper multidimensional analysis

Conclusion:

Data experts can uncover patterns and relationships to better understand transit usage. This allows for data-driven forecasting and infrastructure planning to efficiently address current problems and proactively meet future needs.

- Predict optimal accessible stop locations via modeling.
- Forecast peak demand periods with predictive analytics.

- Collect and analyze emissions, mileage, fuel efficiency data.
- Model route optimization scenarios to increase sustainability.

References:

1. https://spark.apache.org/
2. https://hive.apache.org/
3. https://cloud.google.com/
4. https://www.transit.dot.gov/ntd/data-product/2022-vehicles