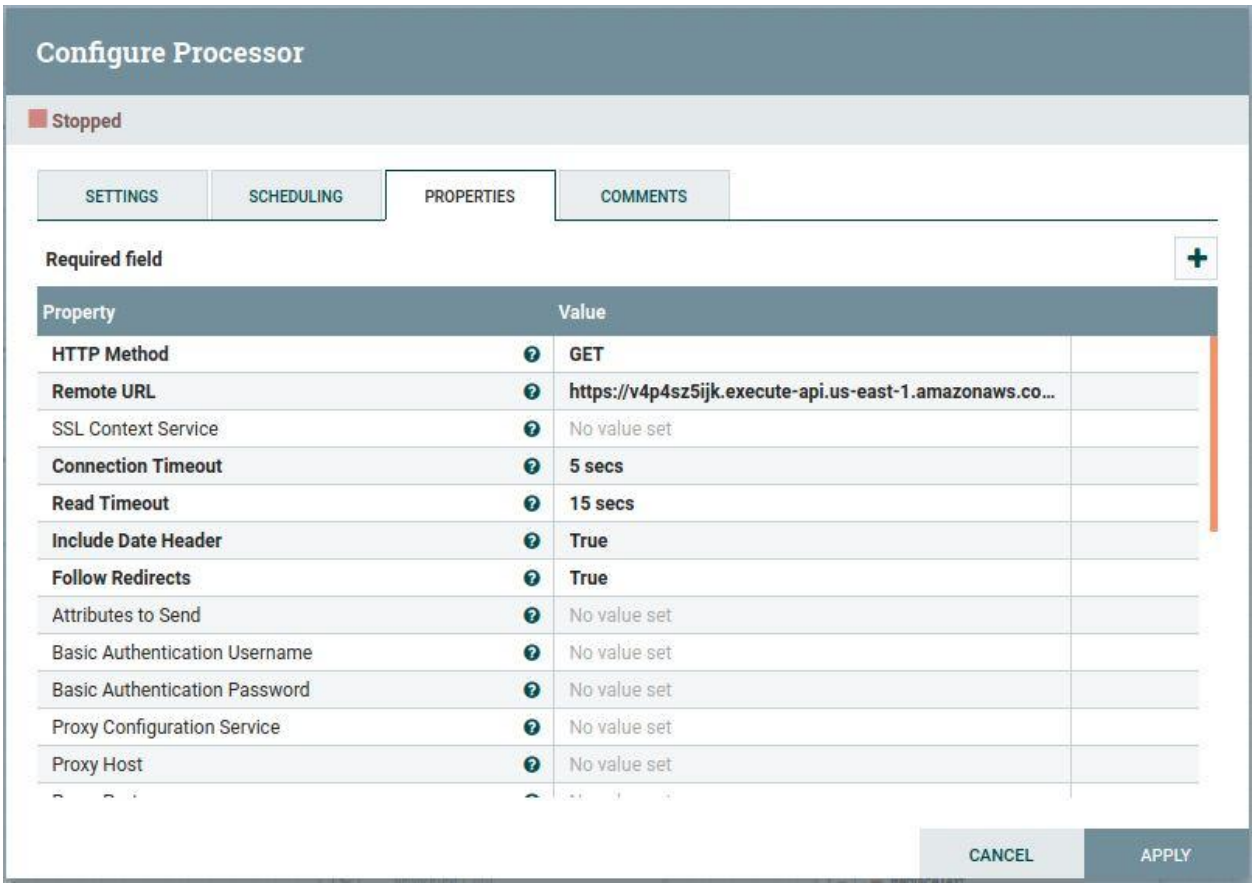
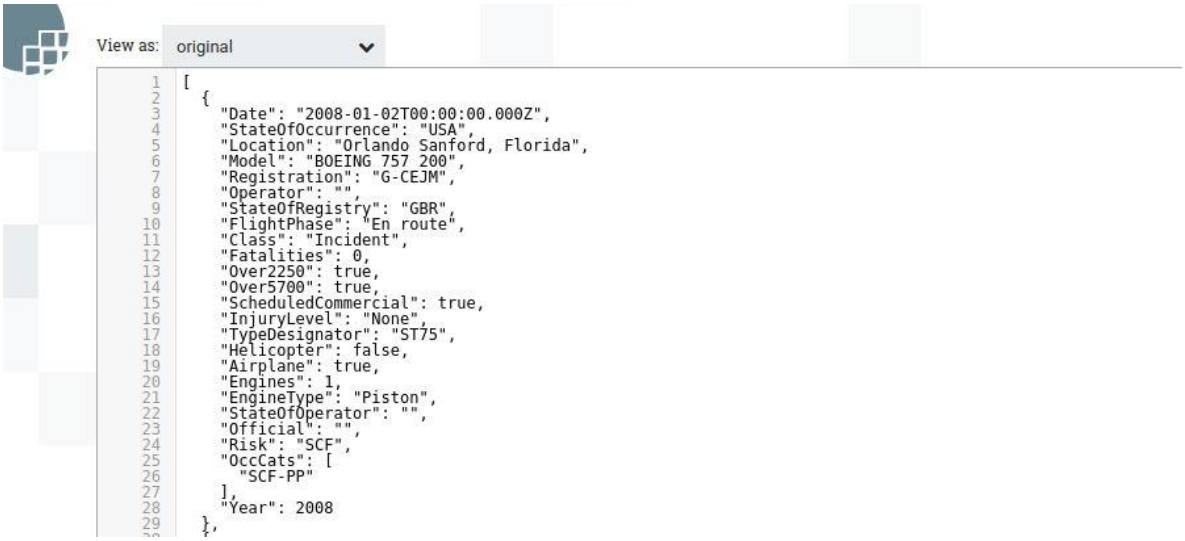
Project Execution Documentation

1. Data extraction is done using NiFi and store in HDFS and MySQL

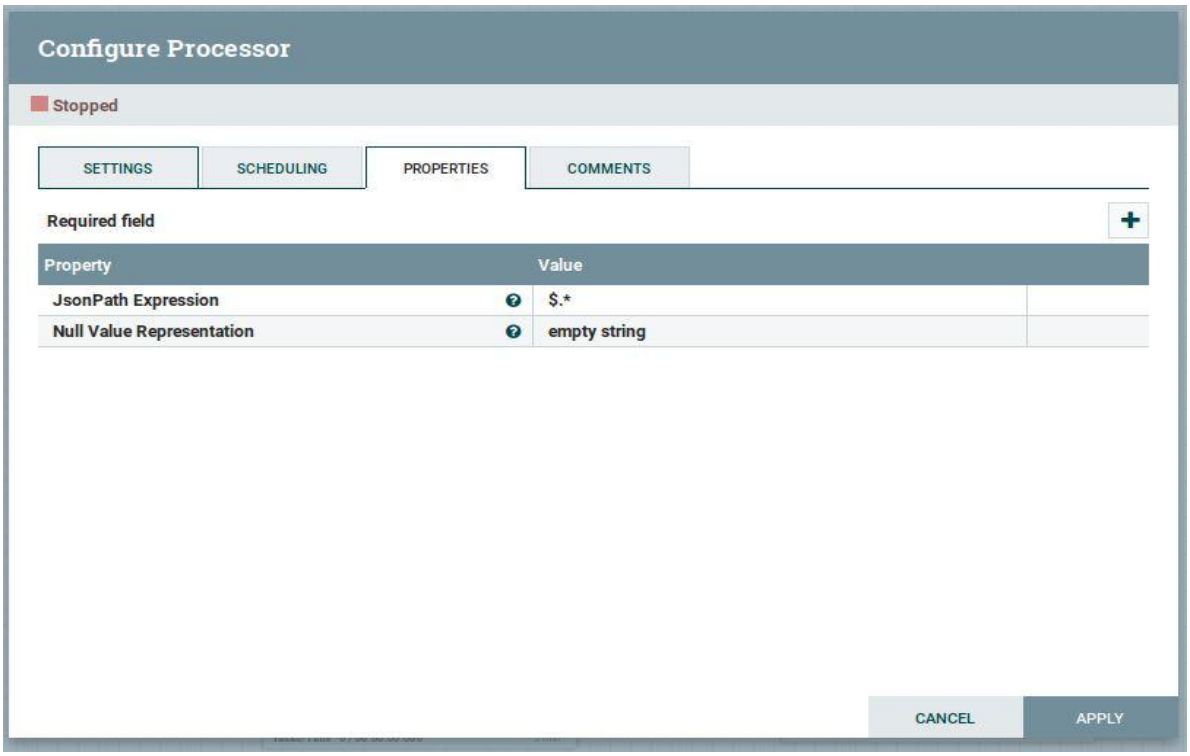
- Using **InvokeHttp processor we** will get the airline data from the API by specifying the API key generated from the invokeHTTP processor . which is followed by the below configurations in the processor.
- Data URL : https://www.icao.int/safety/istars/pages/api-data-service.aspx
- How to get API Key : Signup using the above link and get the api key and click on send api key https://v4p4sz5ijk.execute-api.us-east-1.amazonaws.com/anbdata/occurrences/class/incident



We will get response data as below .

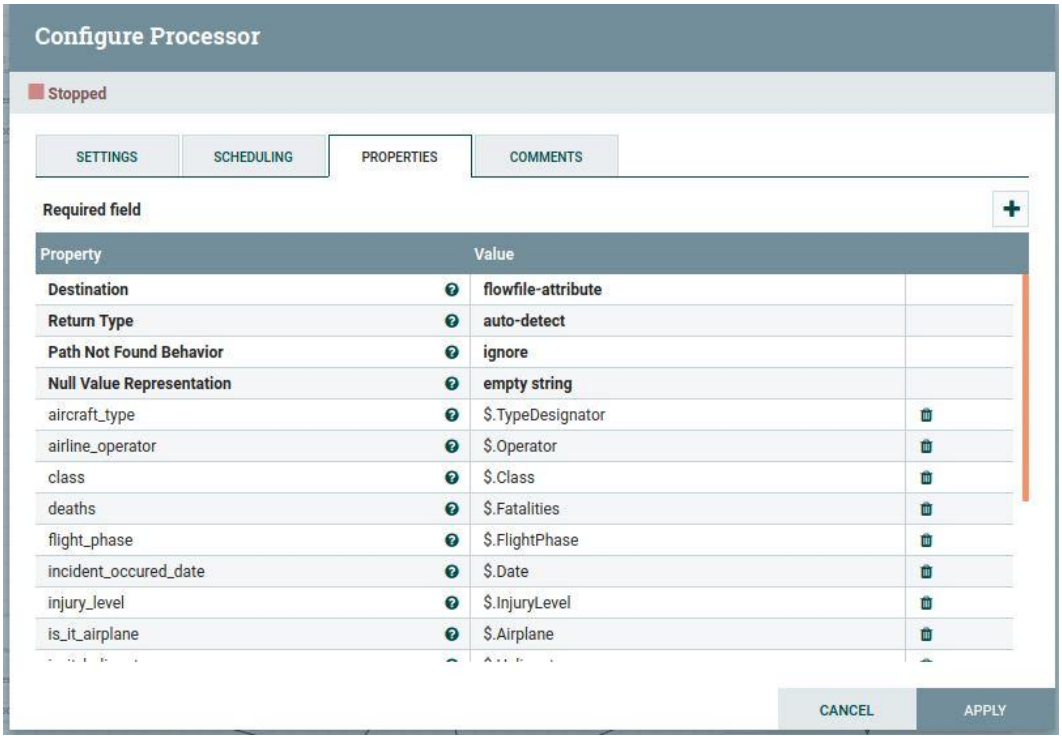


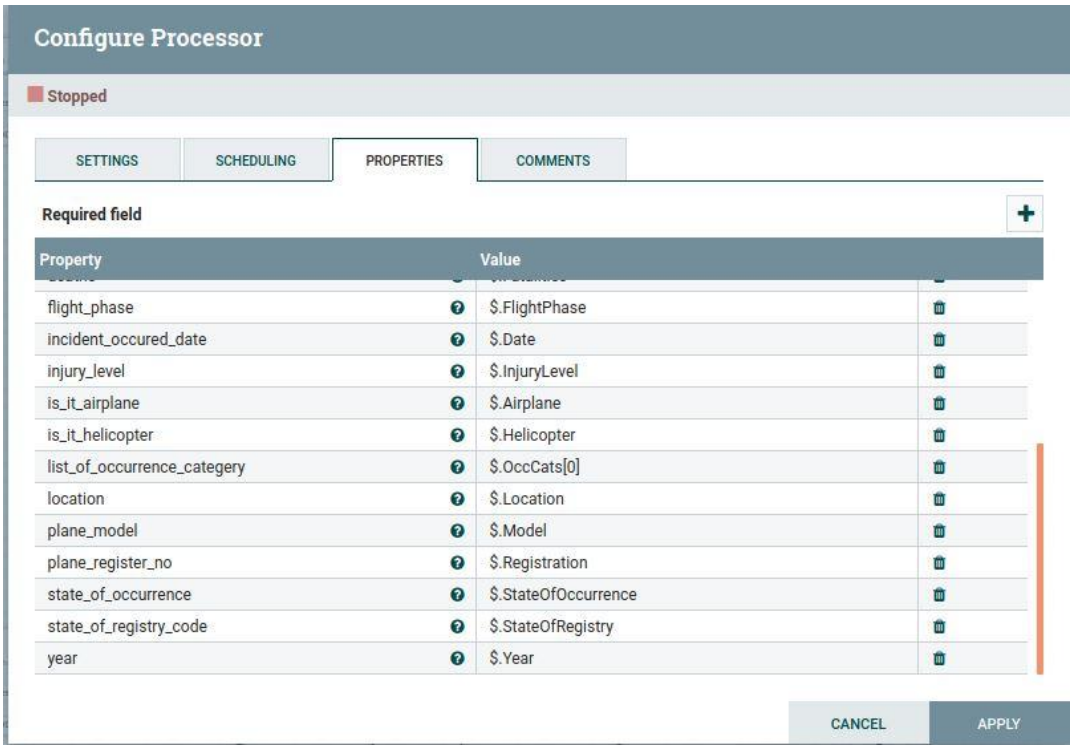
- Split the JSON array from the above data by configuring **SplitJson Processor** as shown below.



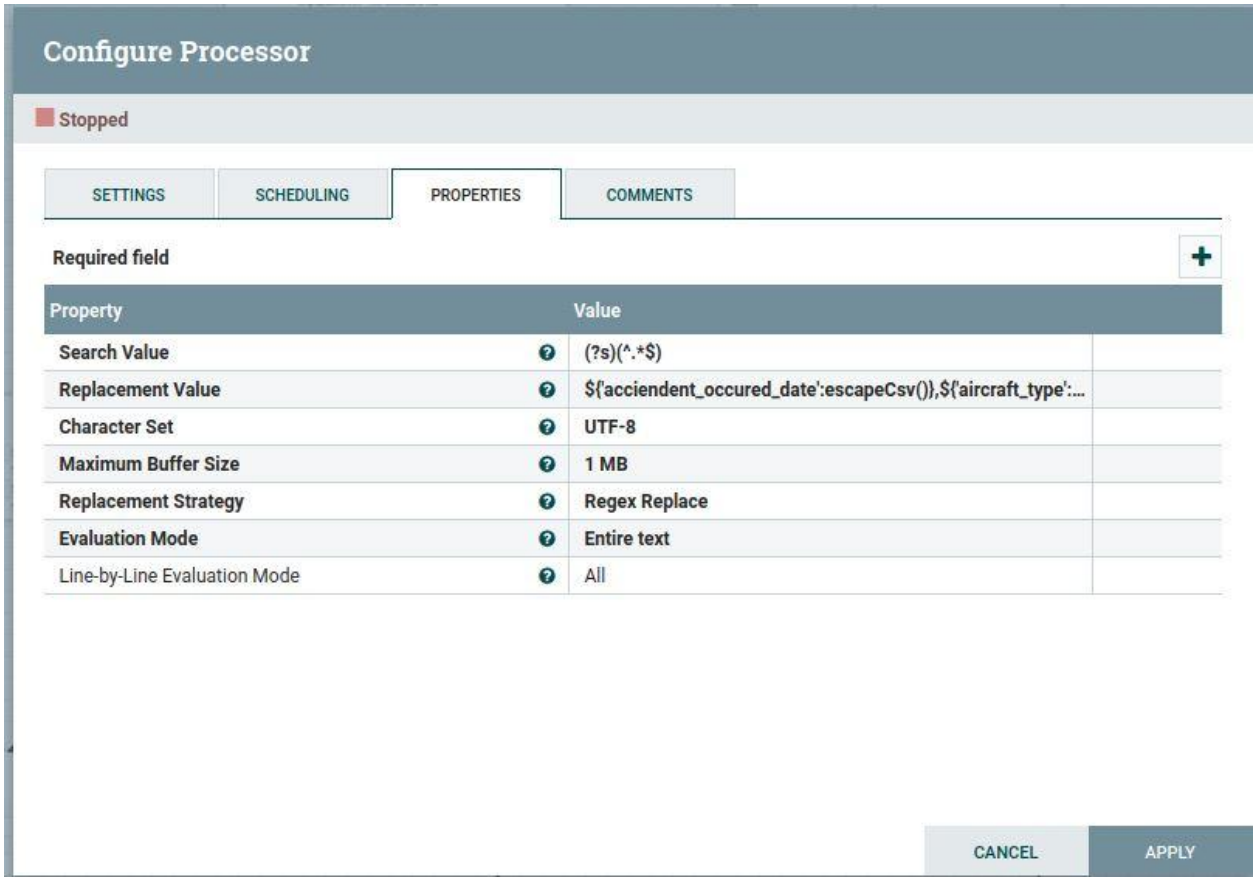
- Evaluate required fields from the data as below images using EvaluateJsonPath to flatten the complex data format to flat csv format to store in HDFS

.

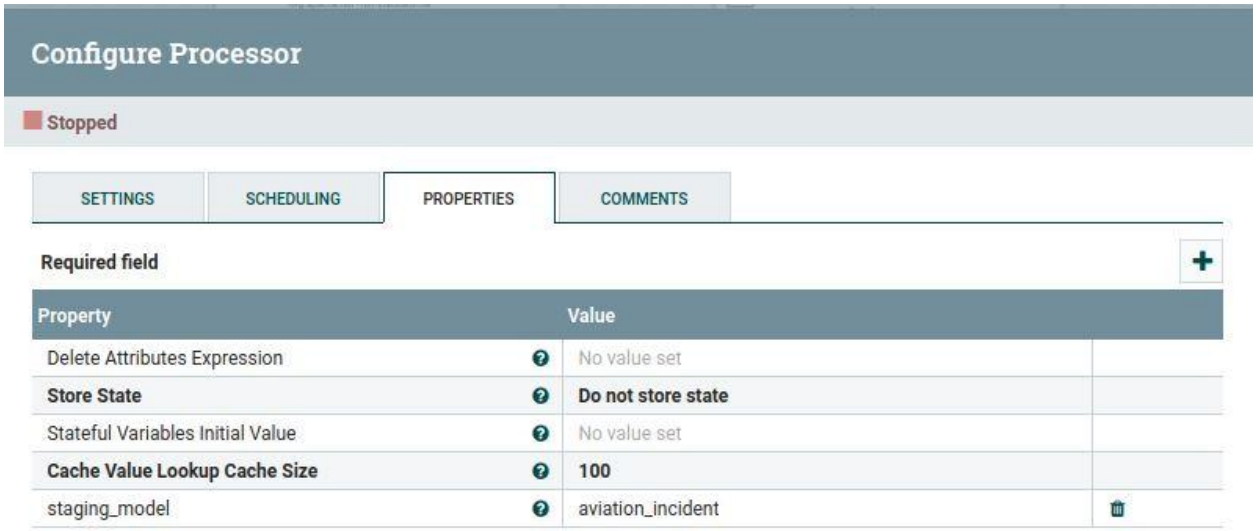




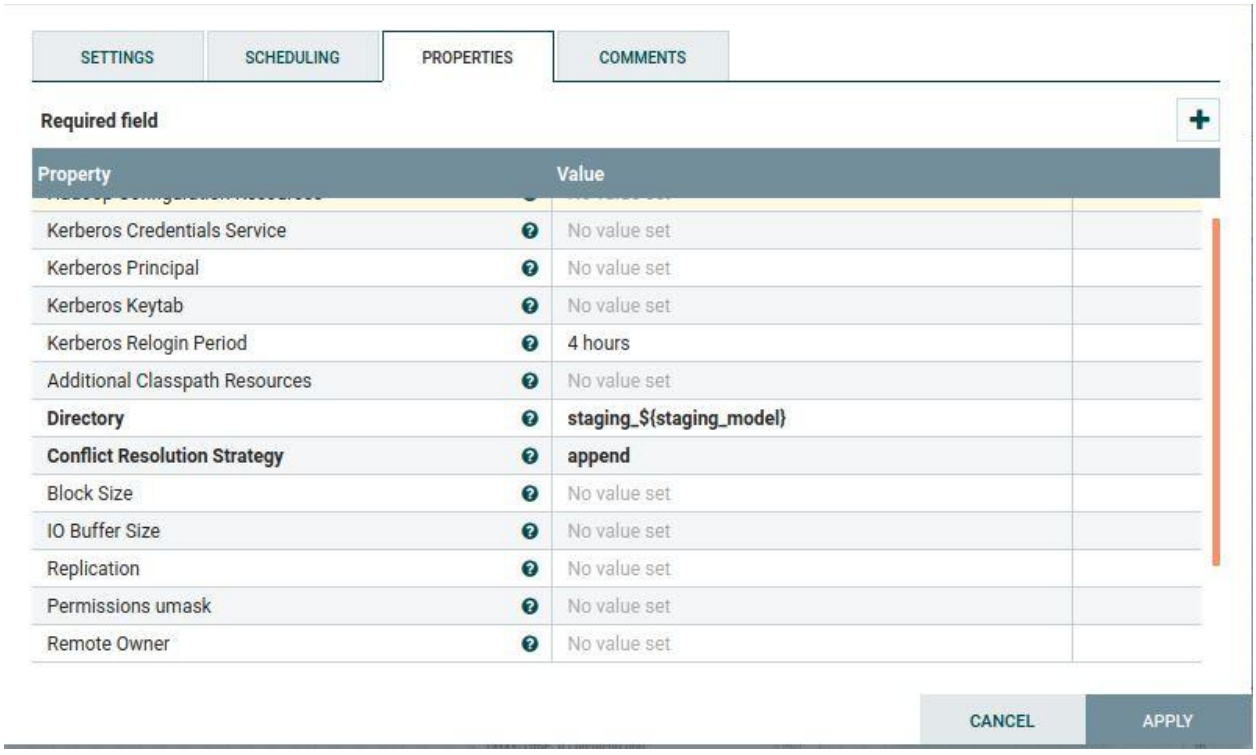
- After evaluating the required fields, convert into CSV format data by placing fields into an order using **ReplaceText** processor as below.



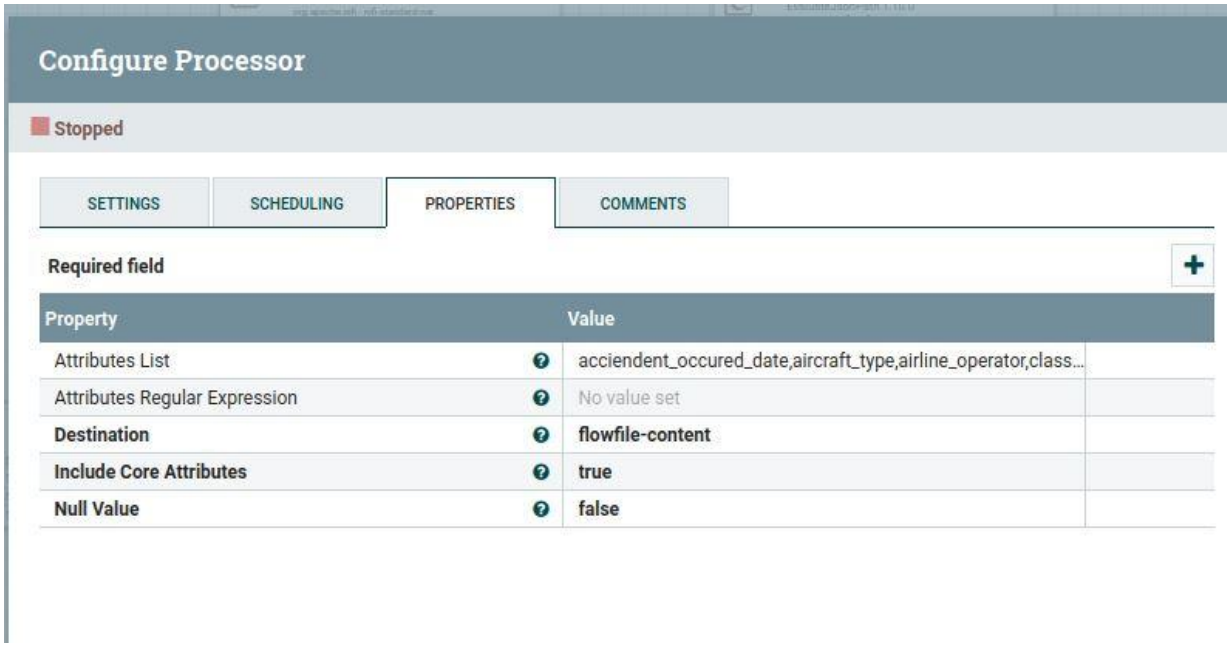
- Specify the name of staging_model by the table name using **UpdateAttribute** Processor



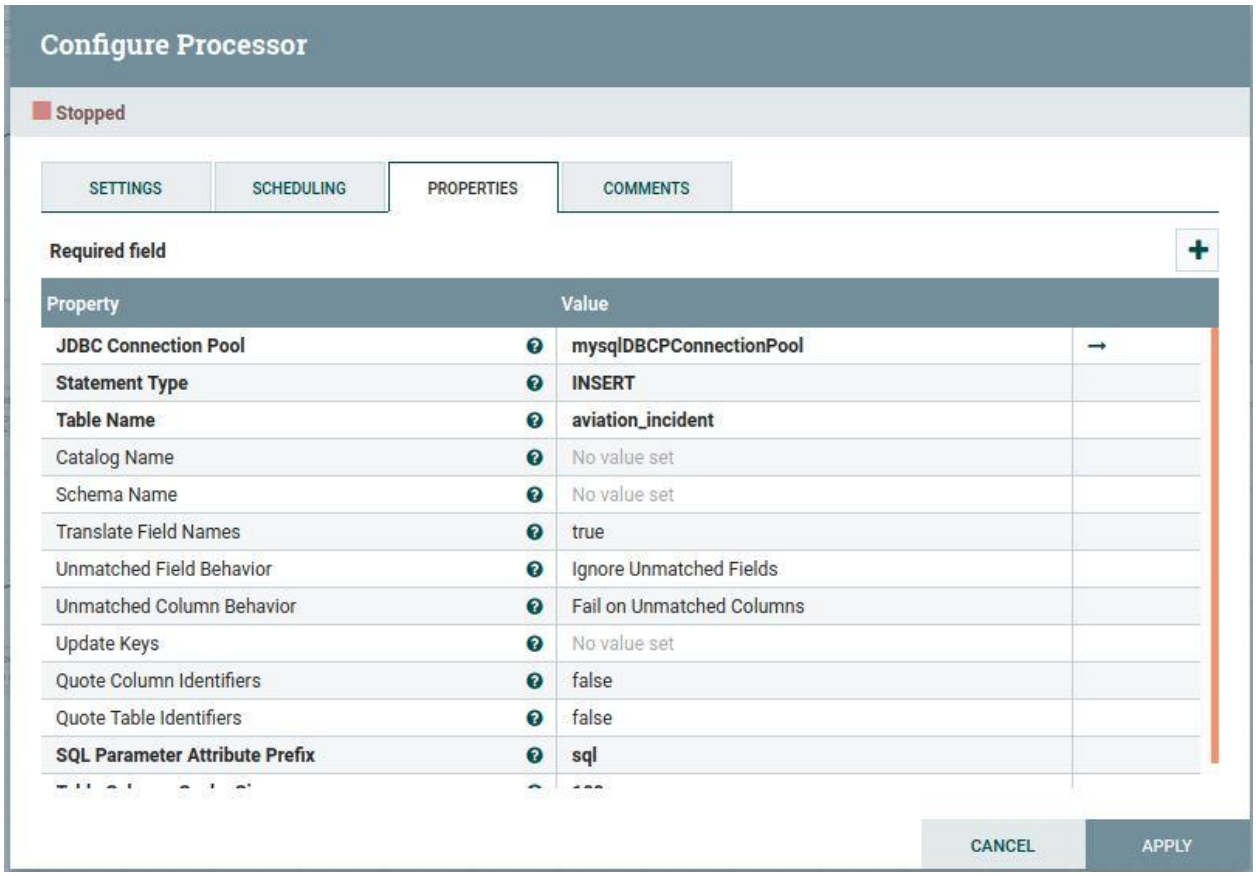
- Then store data into directory in the HDFS using PUTHDFS processor by using following configurations



- At the same time after giving the name to the staging_model we write attributes into flowfile content as json format by passing the AttributeList using **AttributestoJson** Processor.



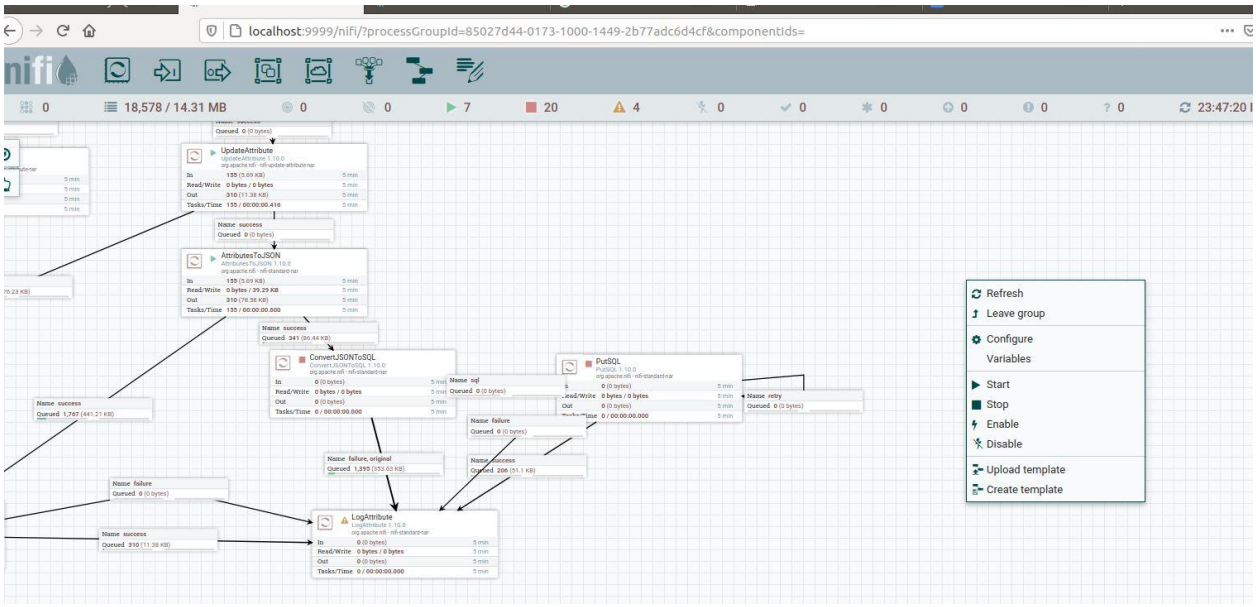
- Then we **convert** a **JSON**-formatted FlowFile into an INSERT **SQL** statement using ConvertJsonToSQL



Note : before loading data into the mysql we need to create table in HIVE

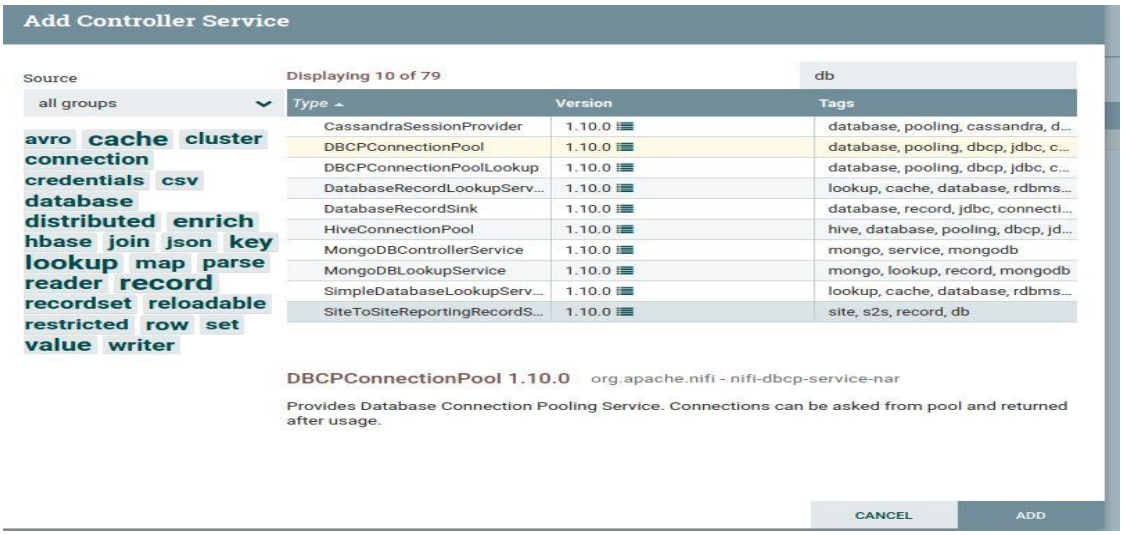
2. Create JDBConnectionPoolService in nifi (for mysql)

- Right click on the nifi Canvas select the “configure”

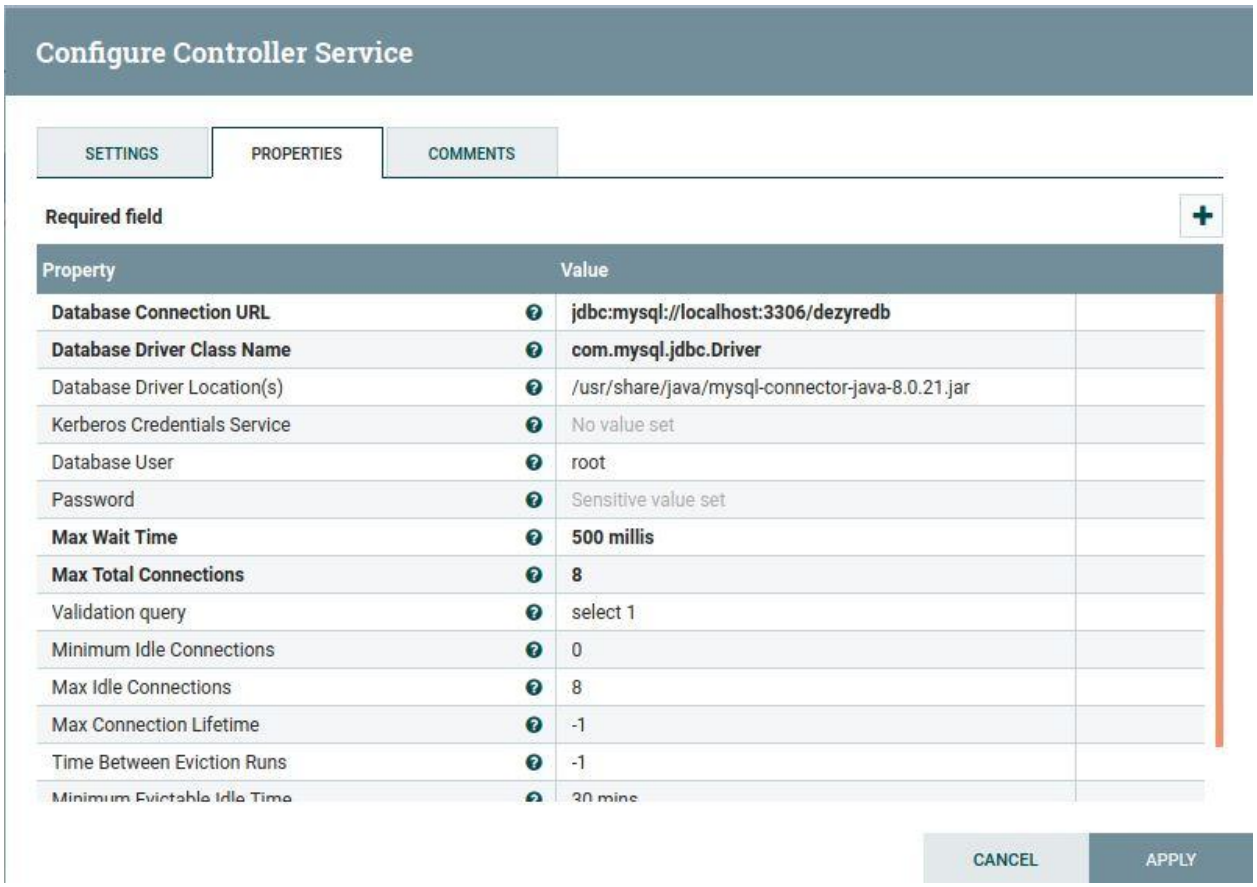


- Click on the “ + “ symbol at the right corner.

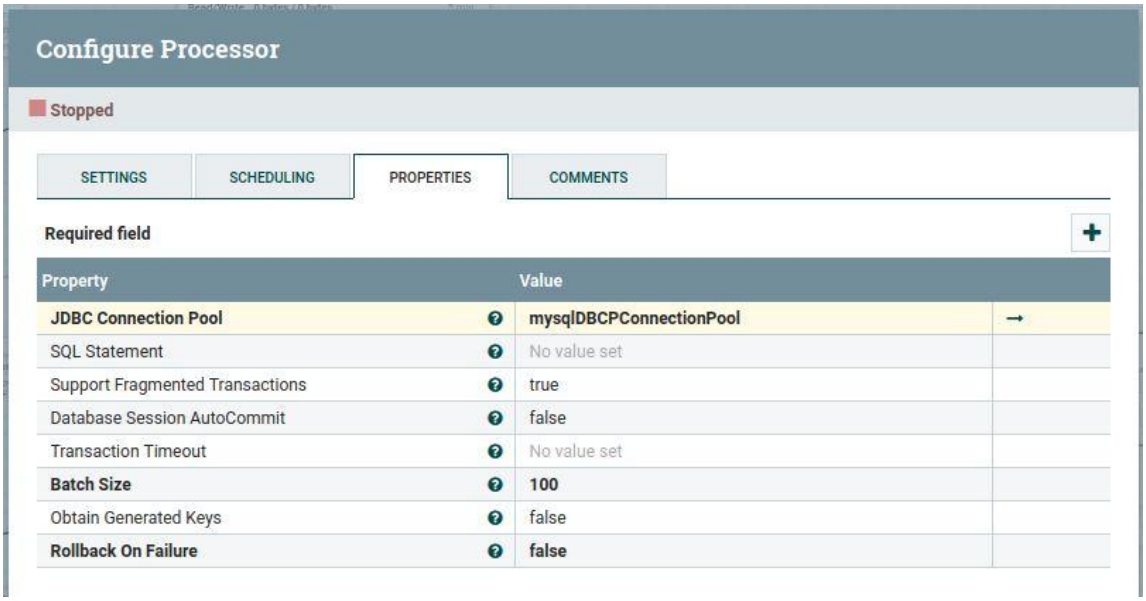
- Select the DBconnectionPool service then click on add



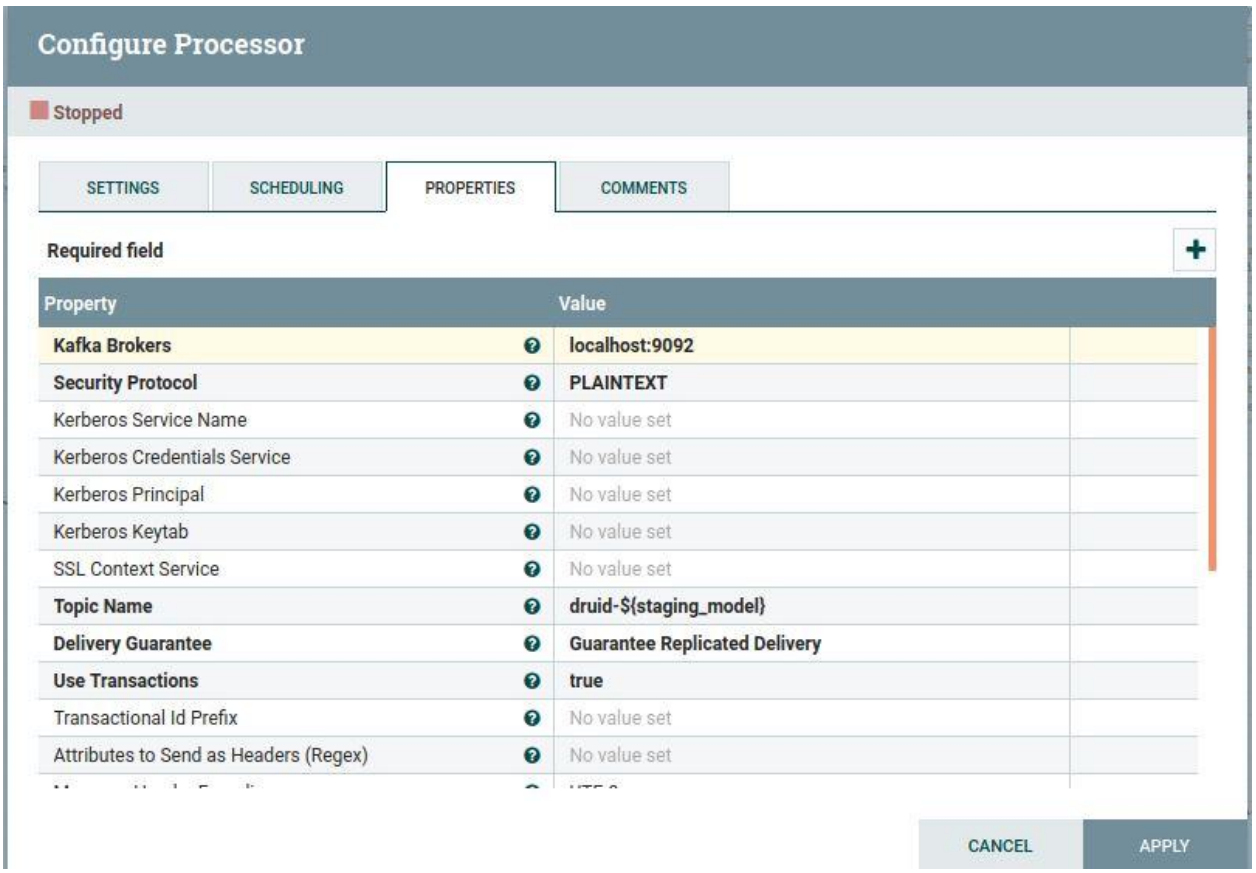
- Configured as in the below image for the mysql Localhost



- Executes a SQL INSERT command. The content of an incoming FlowFile is expected to be the SQL command to execute using PutSQL processor



- At the same time after writing attributes into flowfile content as json format by passing the AttributeList using **AttributestoJson** Processor.
- Publishing Data to Kafka topic through the Publish Kafka processor by configuring as below in the **PublishKafka** processor



- Once the data is extracted and stored in HDFS. We will next index the data from HDFS into Druid as follows .
- Open the druid UI using VNC server .
- Navigate to Load data tab

1. Data ingesting into DRUID from HDFS by indexing from hadoop

To configure HDFS to talk to druid , following configurations need to be made in /home/ubuntu/apache-druid-0.18.1/conf/druid/single-server/micro-quickstart/_common/common.runtime.properties file .

Configuration for HDFS

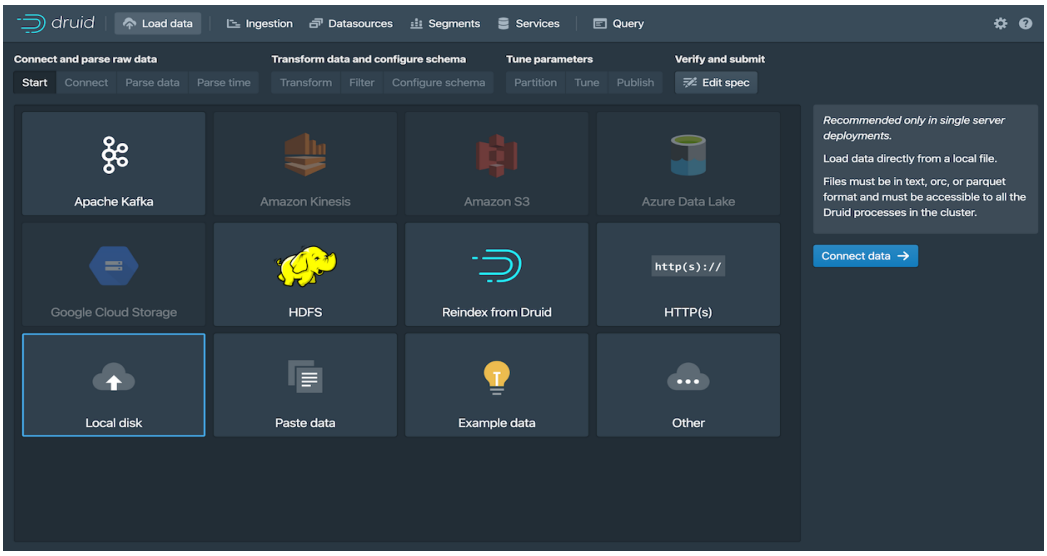
Property	Possible Values	Description	Default
druid.storage.type	hdfs		Must be set.
druid.storage.storageDirectory		Directory for storing segments.	Must be set.
druid.hadoop.security.kerberos.principal	druid@EXAMPLE.COM	Principal user name	empty
druid.hadoop.security.kerberos.keytab	/etc/security/keytabs/druid.headlessUser.keytab	Path to keytab file	empty

Besides the above settings, you also need to include all Hadoop configuration files (such as core-site.xml, hdfs-site.xml) in the Druid classpath. One way to do this is copying all those files under ${DRUID_HOME}/conf/_common.

Now we can load the data from HDFS to Druid .

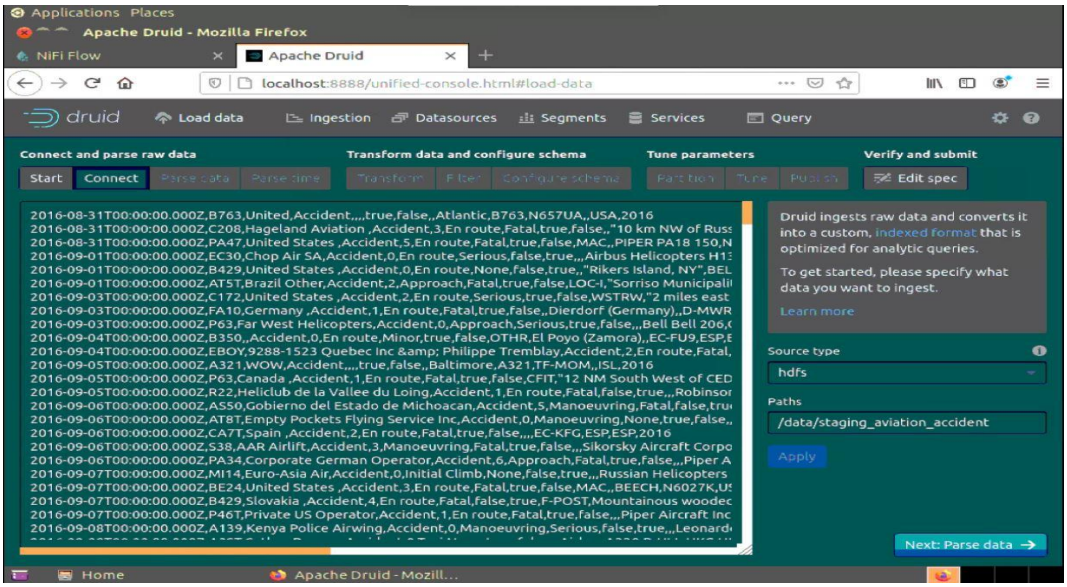
- Click Load data from the Druid console header (

).
- Select the HDFS tile and then click Connect data.



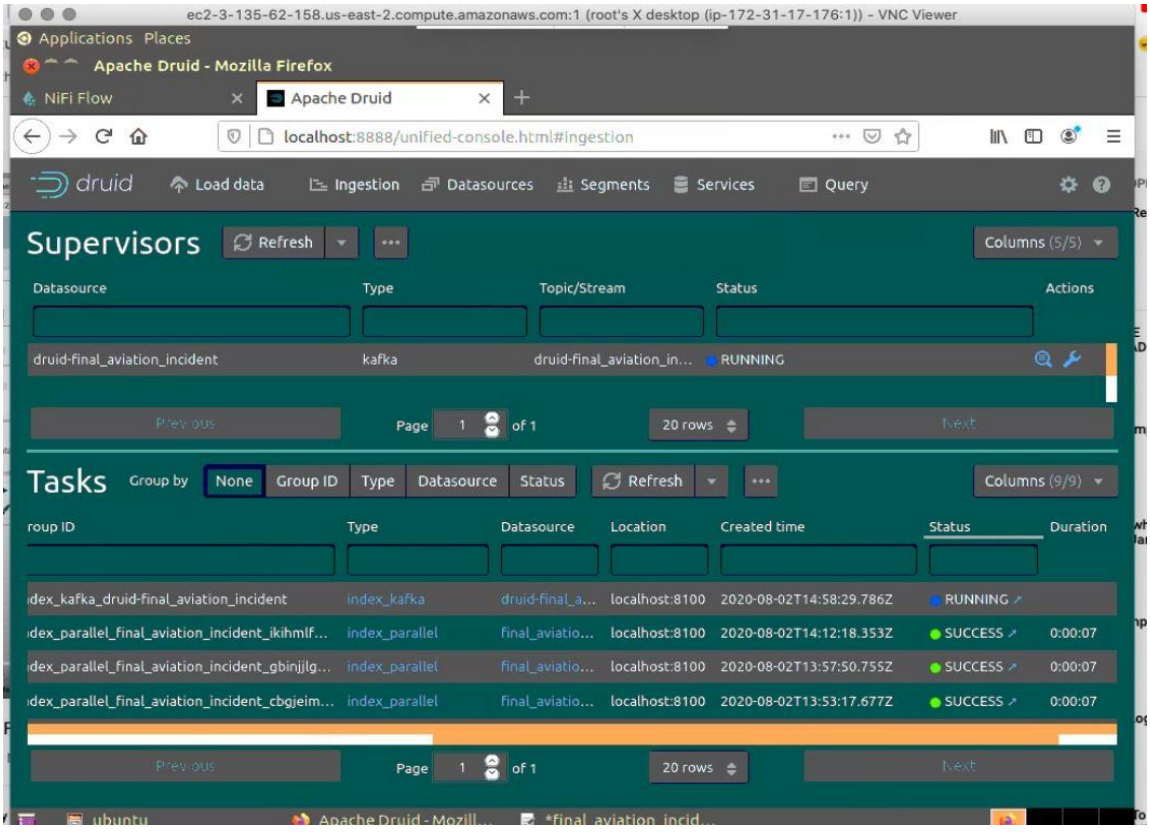
Enter the following values:
Select the source type **hdfs**

- Base directory: data/staging_aviation_incident and click on Apply.
- Data will be displayed in the console as shown in the figure.



3. Streaming Data Consume from the kafka topic into Druid data source

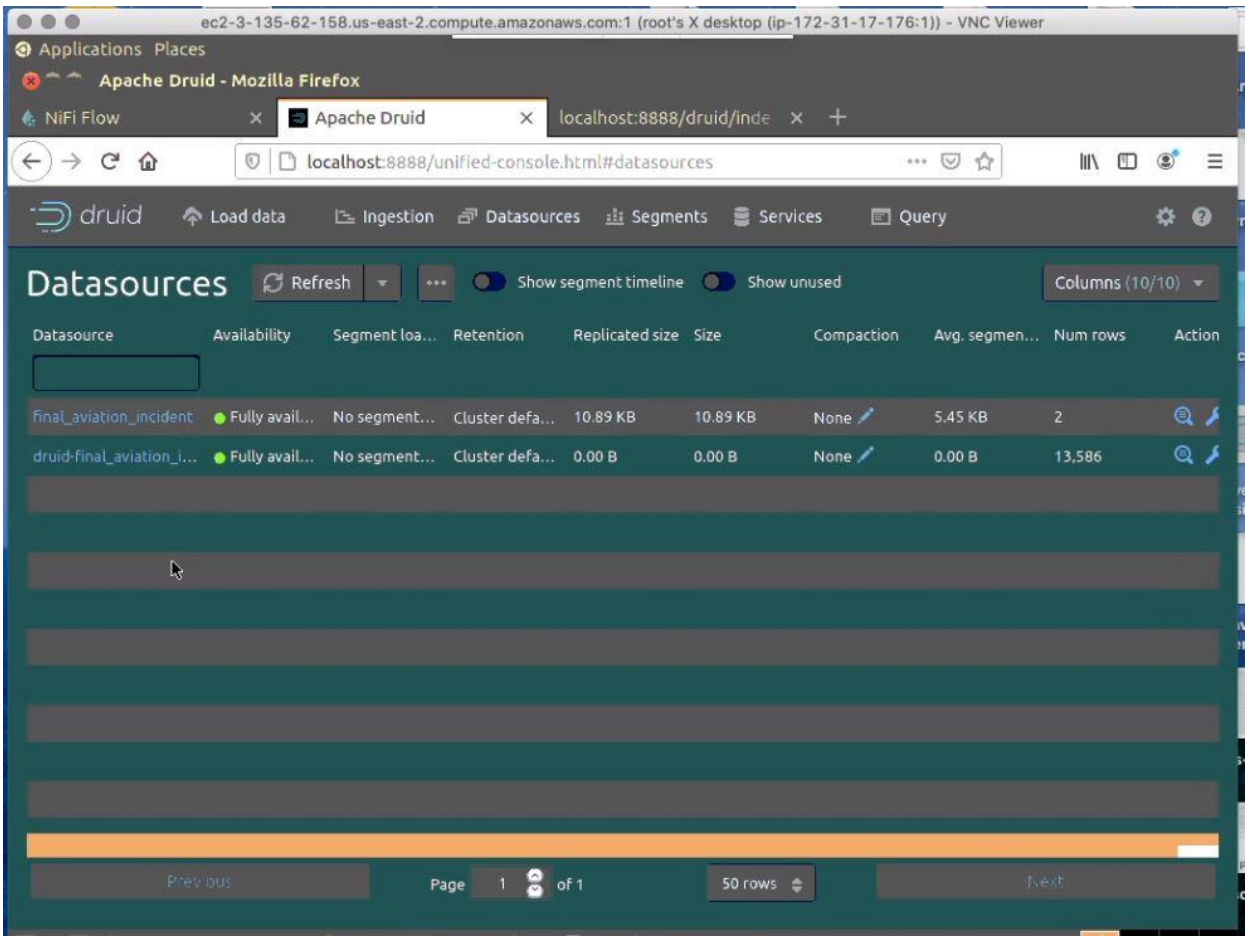
- Here we are consuming the data from the kafka topic by submitting the kafka-supervisor.json file
- Copy the code from supervisor.json file code into load data by configuring Kafka broker URL and topic name as localhost:9092 , topic name



4. Data Querying from Druid

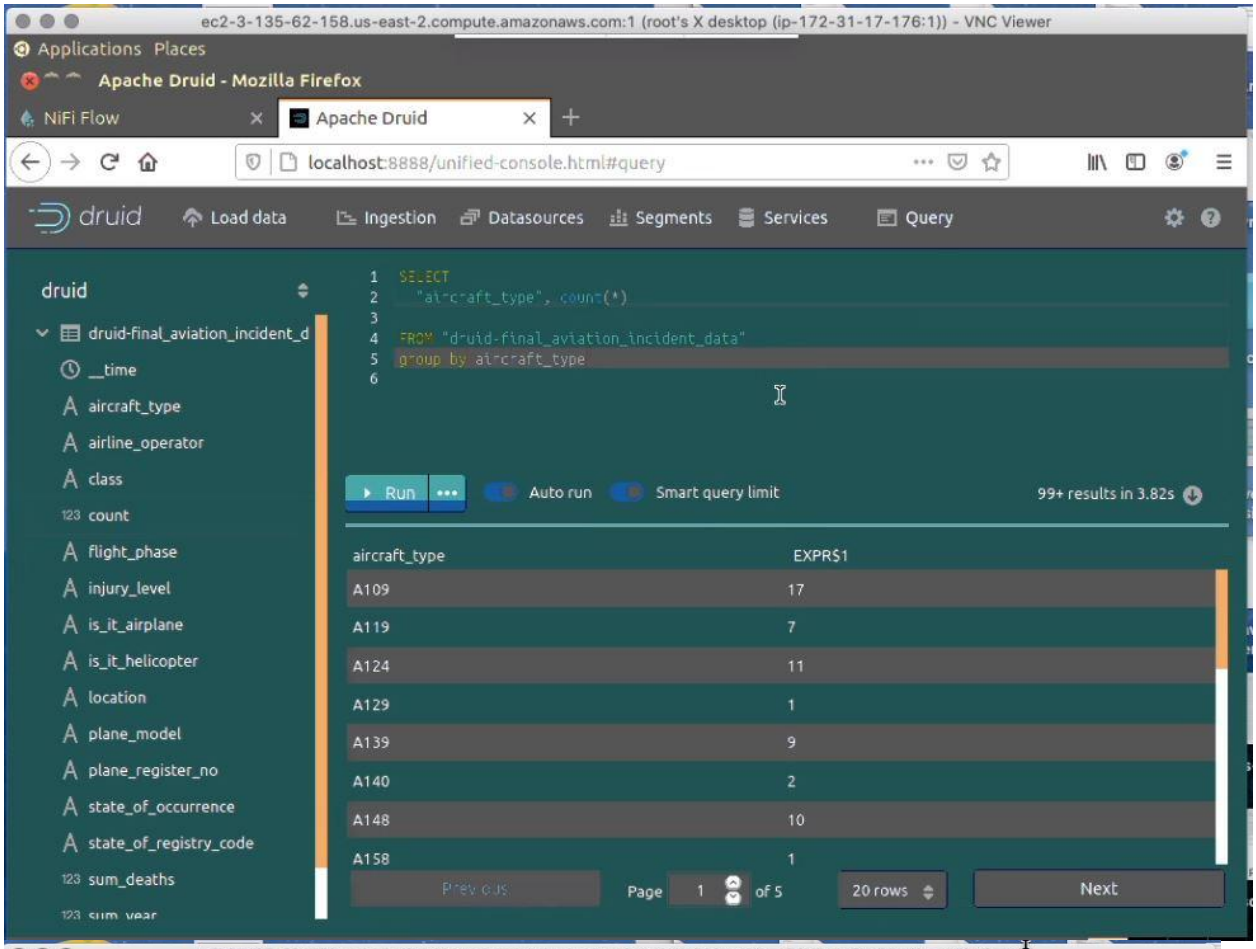
You can now see the data as a datasource in the console and try out a query, as follows:

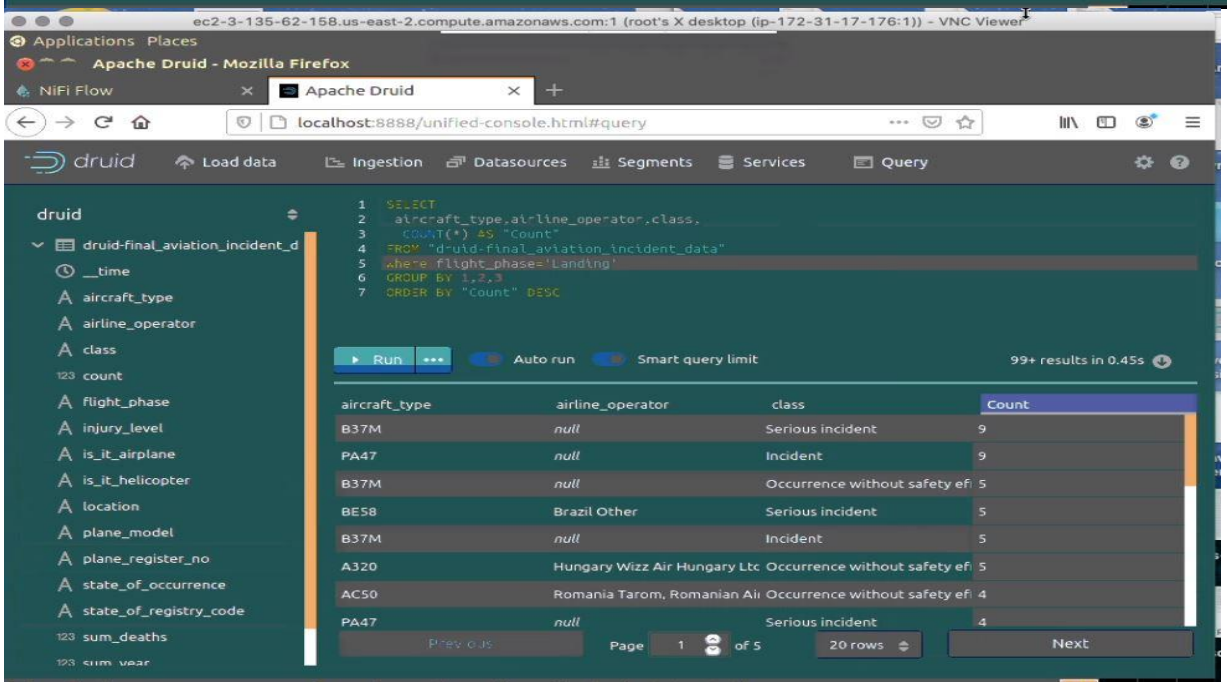
- Click Data Sources from the console header.
  If the wikipedia datasource doesn't appear, wait a few moments for the segment to finish loading. A datasource is queryable once it is shown to be "Fully available" in the Availability column.
- When the datasource is available, open the Actions menu ( ) for that datasource and choose Query with SQL.

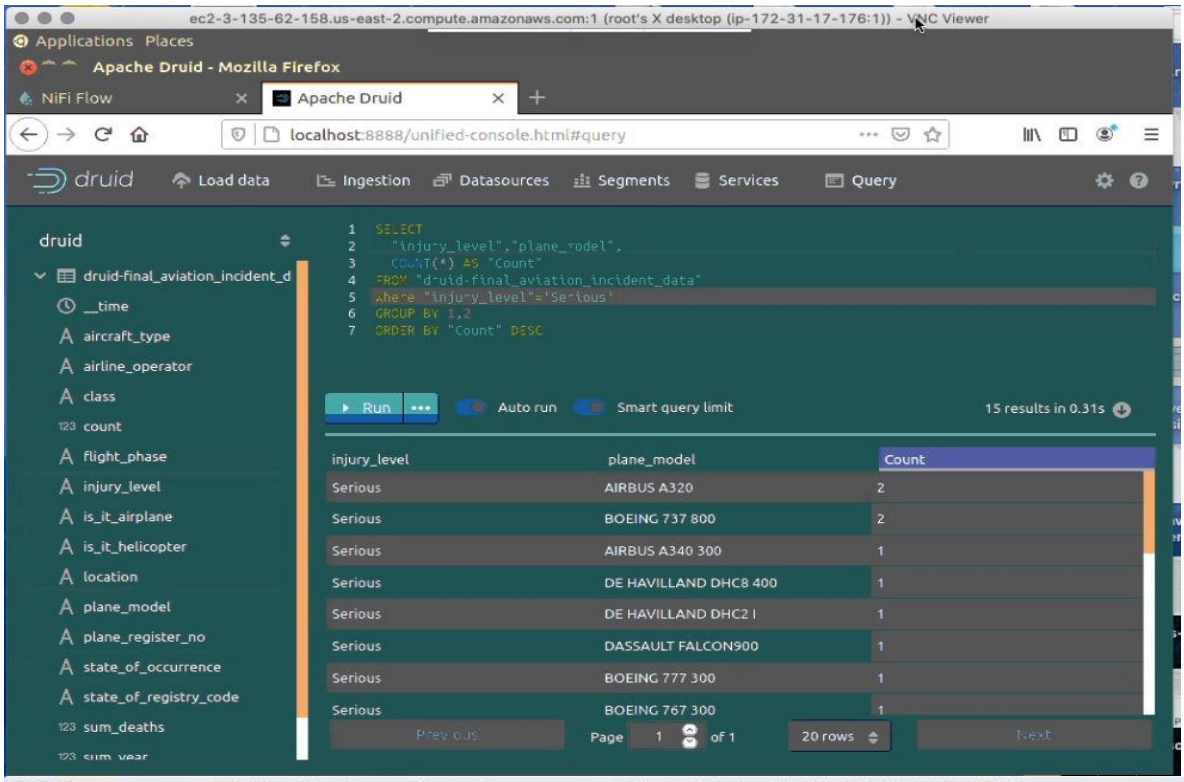


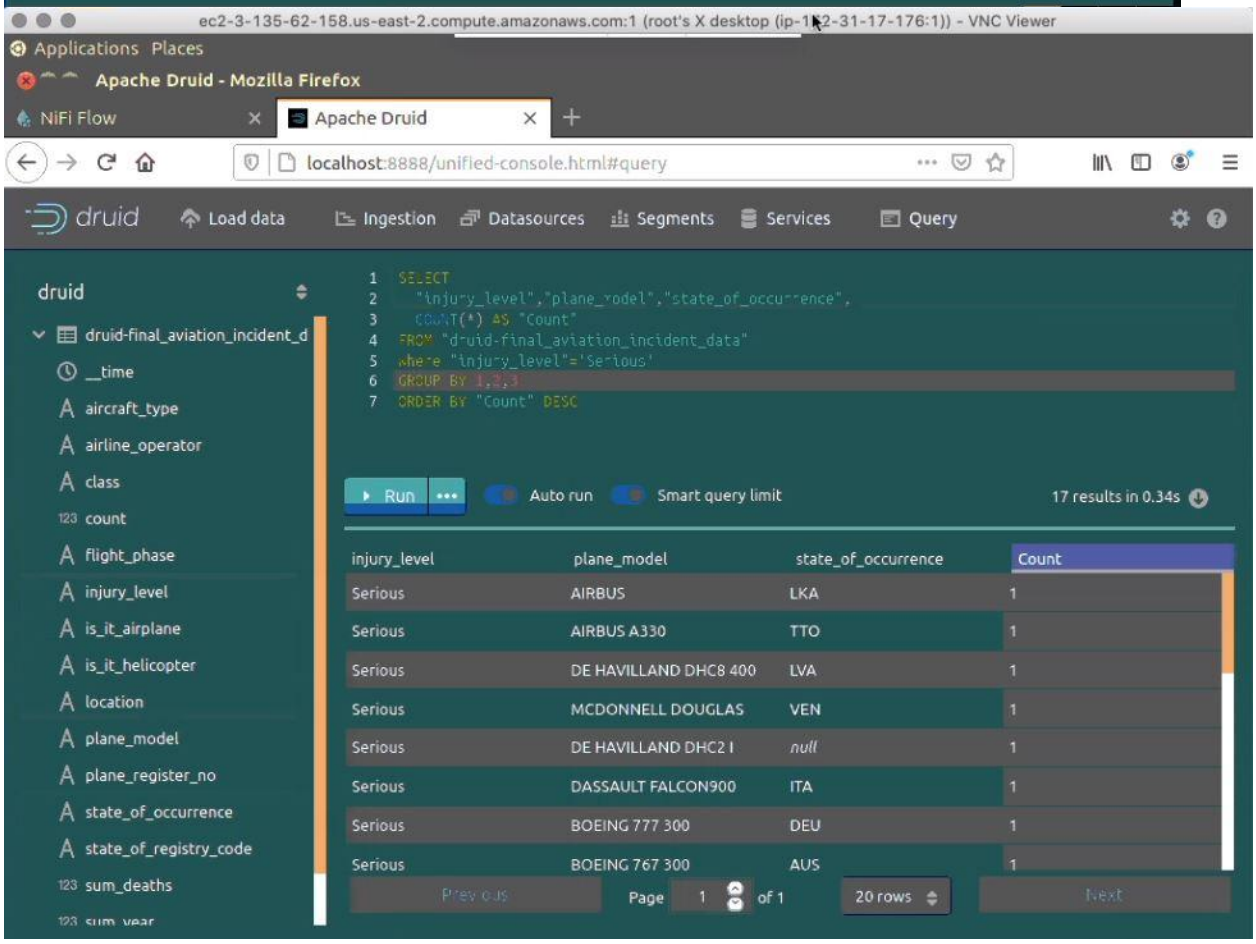
1. Run the pre populated query, SELECT * FROM "druid-final_aviation_incident" to see the results.
2. Then, run the queries given in the Druid SQL code

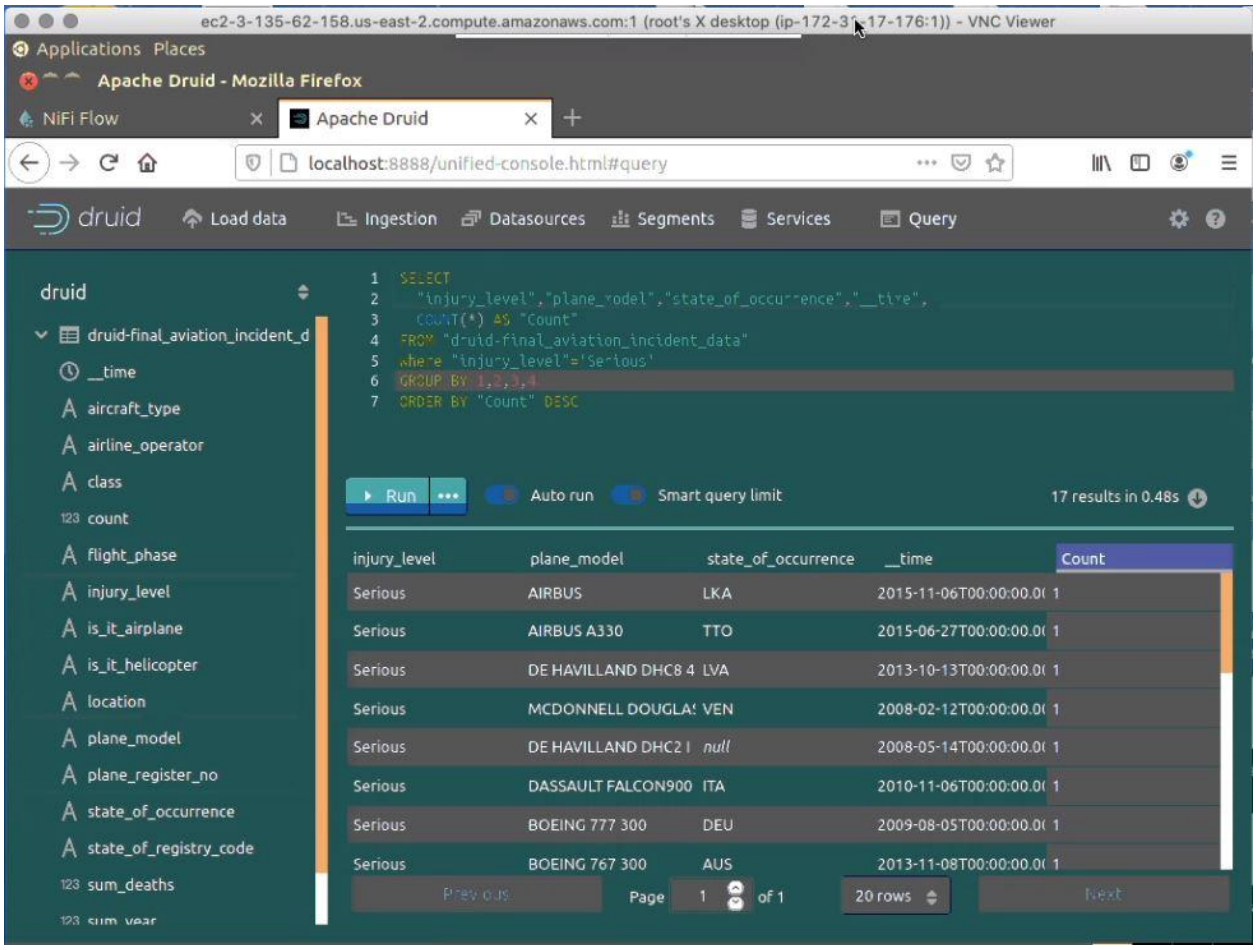
3.

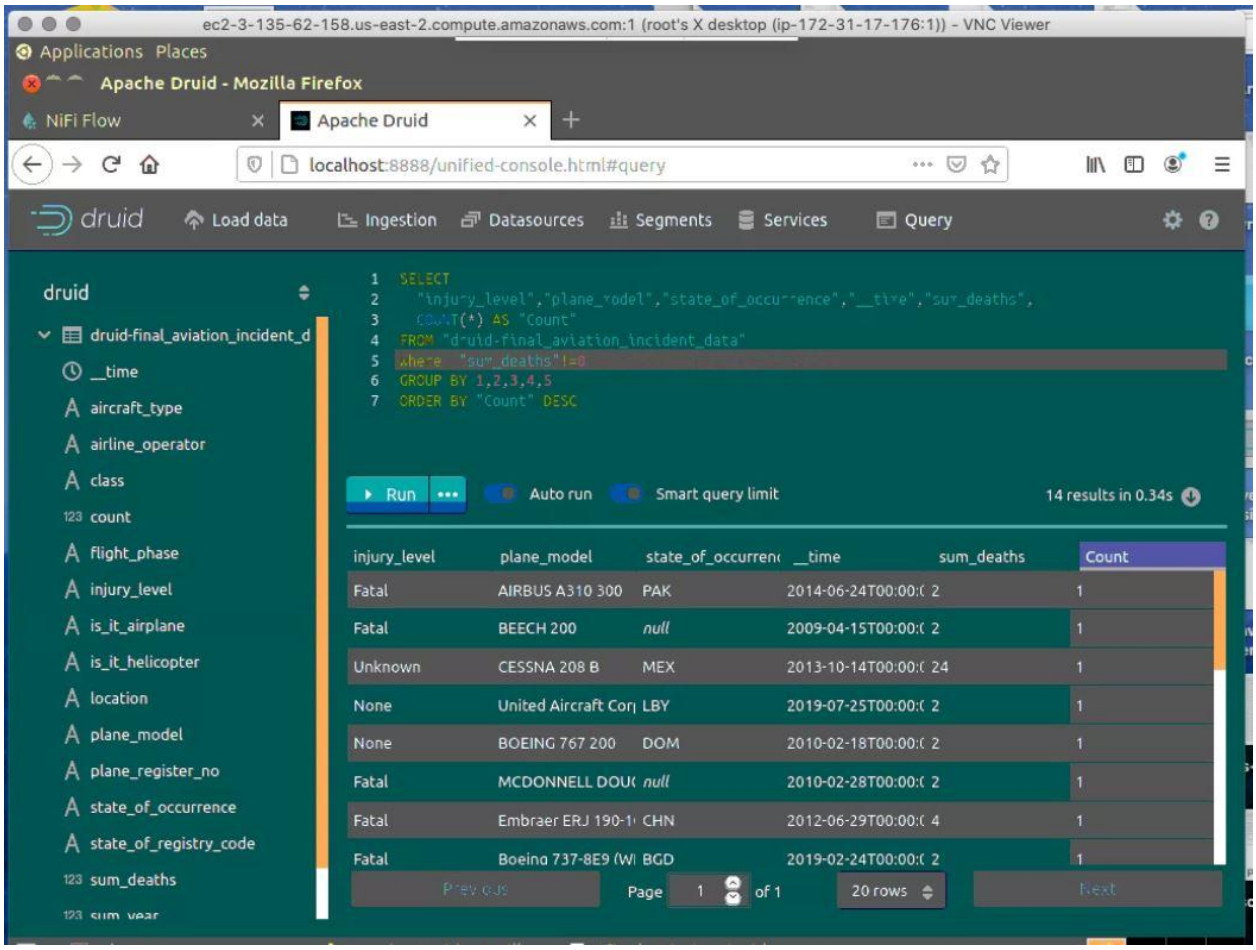

ec2-3-135-62-158.us-east-2.compute.amazonaws.com:1 (root's X desktop (ip-172-31-17-176:1)) - VNC Viewer
Applications Places
Apache Druid - Mozilla Firefox
NiFi Flow
Apache Druid
localhost:8888/unified-console.html#query
druid
Load data
Ingestion
Datasources
Segments
Services
Query
druid
druid-final_aviation_incident_d
__time
aircraft_type
airline_operator
class
count
flight_phase
injury_level
is_it_airplane
is_it_helicopter
location
plane_model
plane_register_no
state_of_occurrence
state_of_registry_code
sum_deaths
1 SELECT
2 "aircraft_type", count(*)
3
4 FROM "druid-final_aviation_incident_data"
5 group by aircraft_type
6
Run
Auto run
Smart query limit
99+ results in 3.82s
aircraft_type
EXPR$1
A109
17
A119
7
A124
11
A129
1
A139
9
A140
2
A148
10
A158
1
Prev
Page 1 of 5
20 rows
Next


ec2-3-135-62-158.us-east-2.compute.amazonaws.com:1 (root's X desktop (ip-172-31-17-176:1)) - VNC Viewer
Applications Places
Apache Druid - Mozilla Firefox
NiFi Flow
Apache Druid
localhost:8888/unified-console.html#query
druid
Load data
Ingestion
Datasources
Segments
Services
Query
druid
druid-final_aviation_incident_d
__time
aircraft_type
airline_operator
class
count
flight_phase
injury_level
is_it_airplane
is_it_helicopter
location
plane_model
plane_register_no
state_of_occurrence
state_of_registry_code
sum_deaths
1 SELECT
2 aircraft_type,airline_operator,class,
3 COUNT(*) AS "Count"
4 FROM "druid-final_aviation_incident_data"
5 where flight_phase='Landing'
6 GROUP BY 1,2,3
7 ORDER BY "Count" DESC
Run
Auto run
Smart query limit
99+ results in 0.45s
aircraft_type
airline_operator
class
Count
B37M
null
Serious incident
9
PA47
null
Incident
9
B37M
null
Occurrence without safety ef
5
BE58
Brazil Other
Serious incident
5
B37M
null
Incident
5
A320
Hungary Wizz Air Hungary Ltc
Occurrence without safety ef
5
AC50
Romania Tarom, Romanian Ai
Occurrence without safety ef
4
PA47
null
Serious incident
4
Prev
Page 1 of 5
20 rows
Next


ec2-3-135-62-158.us-east-2.compute.amazonaws.com:1 (root's X desktop (ip-172-31-17-176:1)) - VNC Viewer
Applications Places
Apache Druid - Mozilla Firefox
NiFi Flow
Apache Druid
localhost:8888/unified-console.html#query
druid
Load data
Ingestion
Datasources
Segments
Services
Query
druid
druid-final_aviation_incident_d
__time
aircraft_type
airline_operator
class
count
flight_phase
injury_level
is_it_airplane
is_it_helicopter
location
plane_model
plane_register_no
state_of_occurrence
state_of_registry_code
sum_deaths
1 SELECT
2 "injury_level","plane_model",
3 COUNT(*) AS "Count"
4 FROM "druid-final_aviation_incident_data"
5 where "injury_level"='Serious'
6 GROUP BY 1,2
7 ORDER BY "Count" DESC
Run
Auto run
Smart query limit
15 results in 0.31s
injury_level
plane_model
Count
Serious AIRBUS A320 2
Serious BOEING 737 800 2
Serious AIRBUS A340 300 1
Serious DE HAVILLAND DHC8 400 1
Serious DE HAVILLAND DHC2 I 1
Serious DASSAULT FALCON900 1
Serious BOEING 777 300 1
Serious BOEING 767 300 1
Previous
Page 1 of 1
20 rows
Next


ec2-3-135-62-158.us-east-2.compute.amazonaws.com:1 (root's X desktop (ip-172-31-17-176:1)) - VNC Viewer
Applications Places
Apache Druid - Mozilla Firefox
NiFi Flow
Apache Druid
localhost:8888/unified-console.html#query
druid
Load data
Ingestion
Datasources
Segments
Services
Query
druid
druid-final_aviation_incident_d
__time
aircraft_type
airline_operator
class
count
flight_phase
injury_level
is_it_airplane
is_it_helicopter
location
plane_model
plane_register_no
state_of_occurrence
state_of_registry_code
sum_deaths
1 SELECT
2 "injury_level","plane_model","state_of_occurrence",
3 COUNT(*) AS "Count"
4 FROM "druid-final_aviation_incident_data"
5 where "injury_level"='Serious'
6 GROUP BY 1,2,3
7 ORDER BY "Count" DESC
Run
Auto run
Smart query limit
17 results in 0.34s
injury_level
plane_model
state_of_occurrence
Count
Serious AIRBUS LKA 1
Serious AIRBUS A330 TTO 1
Serious DE HAVILLAND DHC8 400 LVA 1
Serious MCDONNELL DOUGLAS VEN 1
Serious DE HAVILLAND DHC2 I null 1
Serious DASSAULT FALCON900 ITA 1
Serious BOEING 777 300 DEU 1
Serious BOEING 767 300 AUS 1
Previous
Page 1 of 1
20 rows
Next


ec2-3-135-62-158.us-east-2.compute.amazonaws.com:1 (root's X desktop (ip-172-31-17-176:1)) - VNC Viewer
Applications Places
Apache Druid - Mozilla Firefox
NiFi Flow
Apache Druid
localhost:8888/unified-console.html#query
druid
Load data
Ingestion
Datasources
Segments
Services
Query
druid
druid-final_aviation_incident_d
__time
aircraft_type
airline_operator
class
count
flight_phase
injury_level
is_it_airplane
is_it_helicopter
location
plane_model
plane_register_no
state_of_occurrence
state_of_registry_code
sum_deaths
1 SELECT
2 "injury_level","plane_model","state_of_occurrence","__time",
3 COUNT(*) AS "Count"
4 FROM "druid-final_aviation_incident_data"
5 where "injury_level"='Serious'
6 GROUP BY 1,2,3,4
7 ORDER BY "Count" DESC
Run
Auto run
Smart query limit
17 results in 0.48s
injury_level plane_model state_of_occurrence __time Count
Serious AIRBUS LKA 2015-11-06T00:00:00.0(1
Serious AIRBUS A330 TTO 2015-06-27T00:00:00.0(1
Serious DE HAVILLAND DHC8 4 LVA 2013-10-13T00:00:00.0(1
Serious MCDONNELL DOUGLA! VEN 2008-02-12T00:00:00.0(1
Serious DE HAVILLAND DHC2 I null 2008-05-14T00:00:00.0(1
Serious DASSAULT FALCON900 ITA 2010-11-06T00:00:00.0(1
Serious BOEING 777 300 DEU 2009-08-05T00:00:00.0(1
Serious BOEING 767 300 AUS 2013-11-08T00:00:00.0(1
Previous
Page 1 of 1
20 rows
Next



5. Data Querying from HIVE

Login into Hive console (beeline) by giving the following commands

root@ip-172-31-17-176:/home/ubuntu/apache-hive-2.1.0-bin# bin/beeline
ls: cannot access '/home/ubuntu/apache-hive-2.1.0-bin/lib/hive-jdbc-*-standalone.jar': No such file or directory
Beeline version 2.1.0 by Apache Hive
beeline> !connect jdbc:hive2://
Connecting to jdbc:hive2://
Enter username for jdbc:hive2://: admin
Enter password for jdbc:hive2://: admin
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/ubuntu/apache-hive-2.1.0-bin/lib/log4j-slf4j-impl-2.4.1.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/ubuntu/hadoop-2.7.3/share/hadoop/common/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
20/08/02 18:50:57 [main]: WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
20/08/02 18:51:01 [main]: WARN session.SessionState: METASTORE_FILTER_HOOK will be ignored, since hive.security.authorization.manager is set to instance of HiveAuthorizerFactory.
Connected to: Apache Hive (version 2.1.0)
Driver: Hive JDBC (version 2.1.0)
20/08/02 18:51:01 [main]: WARN jdbc.HiveConnection: Request to set autoCommit to false; Hive does not support autoCommit=false.
Transaction isolation: TRANSACTION_REPEATABLE_READ
0: jdbc:hive2://>
Then give the following commands

0: jdbc:hive2://> SHOW DATABASES;
0: jdbc:hive2://>USE dezyredb ;
Then run the below queries from the HIVE queries code document :

```
0: jdbc:hive2:///> select aircraft_type,airline_operator,class,count(*) as count from staging_aviation_incident where flight_phase='Landing' group by aircra
ft_type,airline_operator,class  order by count desc;
20/08/02 18:16:07 [HiveServer2-Background-Pool: Thread-121]: WARN ql.Driver: Hive-on-MR is deprecated in Hive 2 and may not be available in the future vers
ions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez)
or using Hive 1.X releases.
Query ID = root_20200802181607_8c7f294e-b4fa-44dd-b826-3a8189b42ebd
Total jobs = 2
Launching Job 1 out of 2
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
20/08/02 18:16:07 [HiveServer2-Background-Pool: Thread-121]: WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implemen
t the Tool interface and execute your application with ToolRunner to remedy this.
Job running in-process (local Hadoop)
20/08/02 18:16:07 [LocalJobRunner Map Task Executor #0]: WARN lazy.LazyStruct: Extra bytes detected at the end of the row! Ignoring similar problems.
2020-08-02 18:16:08,670 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1066185543_0002
Launching Job 2 out of 2
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
20/08/02 18:16:08 [HiveServer2-Background-Pool: Thread-121]: WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implemen
t the Tool interface and execute your application with ToolRunner to remedy this.
Job running in-process (local Hadoop)
2020-08-02 18:16:09,894 Stage-2 map = 100%,  reduce = 100%
+----------+-----------------------------------+-----------------------------+--------+--+
| A319     | United Kingdom Easyjet Airline Company Ltd  | Incident                    | 1      |
| A319     | Turkey Pegasus Hava Tasimaciligi A.s. (Pegasus)  | Serious incident            | 1      |
| A319     | Turkey Mng Havayollari Ve Tasimacilik A.s.  | Serious incident            | 1      |
| A319     | Philippines Cebu Pacific Air                | Serious incident            | 1      |
| A319     | China China Eastern Airlines                | Serious incident            | 1      |
| A319     | Canada Air Canada                           | Serious incident            | 1      |
| A30B     | Singapore Singapore Airlines Limited        | Incident                    | 1      |
| A306     | Saudi Arabia Saudi Arabian Airlines         | Serious incident            | 1      |
| A306     | Australia Qantas Airways Limited            | Incident                    | 1      |
| A119     | Italy                                       | Serious incident            | 1      |
| A119     |                                             | Incident                    | 1      |
| A109     | Italy Other                                 | Serious incident            | 1      |
+----------+-----------------------------------+-----------------------------+--------+--+
433 rows selected (2.688 seconds)
0: jdbc:hive2:///>
```

```
Downloads — root@ip-172-31-17-176: /home/ubuntu/apache-hive-2.1.0-bin — ssh -i dezyre-keypair.pem ubuntu@ec2-3-135-62-158.us-east-2.compute.amazonaws.com — 155×37
0: jdbc:hive2:///> select aircraft_type,count(*) from staging_aviation_incident group by aircraft_type;
20/08/02 18:18:38 [HiveServer2-Background-Pool: Thread-215]: WARN ql.Driver: Hive-on-MR is deprecated in Hive 2 and may not be available in the future vers
ions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez)
or using Hive 1.X releases.
Query ID = root_20200802181838_5922fc55-1ac8-42dc-bb62-6301245abc29
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
20/08/02 18:18:38 [HiveServer2-Background-Pool: Thread-215]: WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implemen
t the Tool interface and execute your application with ToolRunner to remedy this.
Job running in-process (local Hadoop)
20/08/02 18:18:38 [LocalJobRunner Map Task Executor #0]: WARN lazy.LazyStruct: Extra bytes detected at the end of the row! Ignoring similar problems.
2020-08-02 18:18:39,350 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local34424947_0004
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 11419752 HDFS Write: 0 SUCCESS
```


Y141	1
YK40	3
YK42	9
YK55	1
Z42	1
ZA6	2
489 rows selected (1.352 seconds)
0: jdbc:hive2://> select injury_level,plane_model,count(*) as count from staging_aviation_incident where injury_level='Serious' group by injury_level,plane_model order by count desc;
20/08/02 18:20:17 [HiveServer2-Background-Pool: Thread-268]: WARN ql.Driver: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = root_20200802182017_c3aaf82a-c020-44ed-83b6-19d3c306ef18
Total jobs = 2
Launching Job 1 out of 2
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
set mapreduce.job.reduces=<number>
20/08/02 18:20:17 [HiveServer2-Background-Pool: Thread-268]: WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
Downloads — root@ip-172-31-17-176: /home/ubuntu/apache-hive-2.1.0-bin — ssh -i dezyre-keypair.pem ubuntu@ec2-3-135-62-158.us-east-2.compute.amazonaws.com — 155×37
0: jdbc:hive2://> select injury_level,plane_model,count(*) as count from staging_aviation_incident where injury_level='Serious' group by injury_level,plane_model order by count desc;
20/08/02 18:20:17 [HiveServer2-Background-Pool: Thread-268]: WARN ql.Driver: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = root_20200802182017_c3aaf82a-c020-44ed-83b6-19d3c306ef18
Total jobs = 2
Launching Job 1 out of 2
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
set mapreduce.job.reduces=<number>
20/08/02 18:20:17 [HiveServer2-Background-Pool: Thread-268]: WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your application with ToolRunner to remedy this.
Job running in-process (local Hadoop)
20/08/02 18:20:17 [LocalJobRunner Map Task Executor #0]: WARN lazy.LazyStruct: Extra bytes detected at the end of the row! Ignoring similar problems.
2020-08-02 18:20:18,446 Stage-1 map = 100%, reduce = 100%
Ended Job = job_local490050267_0005
Launching Job 2 out of 2
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):

```
+----------------+-------------------------+--------+--+
| injury_level   |       plane_model       | count  |
+----------------+-------------------------+--------+--+
| Serious        | BOEING 737 800 8AS      | 2      |
| Serious        | AIRBUS A320             | 2      |
| Serious        | BOEING 737 800          | 2      |
| Serious        | DE HAVILLAND DHC2 I     | 2      |
| Serious        | DE HAVILLAND DHC8 400   | 1      |
| Serious        | DASSAULT FALCON900      | 1      |
| Serious        | BOEING 767 300          | 1      |
| Serious        | BOEING 737 900          | 1      |
| Serious        | AIRBUS A340 300         | 1      |
| Serious        | AIRBUS A330             | 1      |
| Serious        | AIRBUS                  | 1      |
| Serious        |  Papua                  | 1      |
| Serious        |  Hessen"                | 1      |
+----------------+-------------------------+--------+--+
13 rows selected (2.559 seconds)
0: jdbc:hive2://>
```

```
0: jdbc:hive2://> select to_date(incident_occured_date) as date_occured, rtrim(plane_register_no) as pln_reg_num from staging_aviation_incident where flight_phase='Landing' and injury_level='Serious' limit 10 ;
OK
20/08/02 18:37:03 [d0c08354-0aff-401d-b019-ba9d168a4029 main]: WARN lazy.LazyStruct: Extra bytes detected at the end of the row! Ignoring similar problems.
+--------------+--------------+--+
| date_occured | pln_reg_num  |
+--------------+--------------+--+
| 2008-12-06   | VH-VQR       |
| 2015-11-06   | TC-JOA       |
| 2011-08-12   | RP-C8607     |
+--------------+--------------+--+
3 rows selected (0.104 seconds)
0: jdbc:hive2://>
```


```
0: jdbc:hive2://> select to_date(incident_occured_date) as date_occured, rtrim(plane_register_no) as pln_reg_num from staging_aviation_incident where flight_phase='Landing' and injury_level='Serious' limit 10 ;
OK
20/08/02 18:37:03 [d0c08354-0aff-401d-b019-ba9d168a4029 main]: WARN lazy.LazyStruct: Extra bytes detected at the end of the row! Ignoring similar problems.
+--------------+--------------+--+
| date_occured | pln_reg_num  |
+--------------+--------------+--+
| 2008-12-06   | VH-VQR       |
| 2015-11-06   | TC-JOA       |
| 2011-08-12   | RP-C8607     |
+--------------+--------------+--+
3 rows selected (0.104 seconds)
0: jdbc:hive2://>
```

```
0: jdbc:hive2://> select sum(deaths) as deaths,state_of_occurrence,flight_phase  from staging_aviation_incident where flight_phase='En route' group by state_of_occurrence,flight_phase order by deaths desc;
+---------+-------------------------+---------------+--+
| deaths  |  state_of_occurrence    | flight_phase  |
+---------+-------------------------+---------------+--+
| NULL    | C-GKEP                  | En route      |
| NULL    | B-18358                 | En route      |
| NULL    | EI-EKS                  | En route      |
| NULL    | D-AICF                  | En route      |
| NULL    | TCA                     | En route      |
| NULL    | IRN                     | En route      |
| NULL    | BOMBARDIER CL600 2B19   | En route      |
| NULL    | AIRBUS A321 200         | En route      |
+---------+-------------------------+---------------+--+
308 rows selected (2.497 seconds)
```

6. Data Querying from MySQL

Login to MYSQL using the following commands
root@ip-172-31-17-176:/home/ubuntu# sudo mysql -u root -p
Enter Password : root
mysql>
mysql> SHOW DATABASES;
mysql> USE dezyredb;

Then run the below queries according to MYSQL run book code :

```
mysql> select injury_level, plane_model from aviation_incident where injury_level='Serious' group by injury_level,plane_model;
+--------------+----------------------+
| injury_level | plane_model          |
+--------------+----------------------+
| Serious      | AIRBUS               |
| Serious      | AIRBUS A320          |
| Serious      | AIRBUS A340 300      |
| Serious      | BAE BAE146 200       |
| Serious      | BOEING 737 200       |
| Serious      | BOEING 737 800       |
| Serious      | BOEING 737 900       |
| Serious      | BOEING 767 300       |
| Serious      | BOEING 777 300       |
| Serious      | DASSAULT FALCON900   |
| Serious      | DE HAVILLAND DHC2 I  |
| Serious      | DE HAVILLAND DHC8 400 |
| Serious      | MCDONNELL DOUGLAS    |
+--------------+----------------------+
13 rows in set (0.00 sec)
```

```
mysql> select aircraft_type,plane_model, injury_level, count(deaths) from aviation_incident where flight_phase='Landing' group by aircraft_type,plane_model,injury_
level order by count(deaths) desc;
```

```
| A359          | AIRBUS A300 600                        | None         |               1 |
| E145          | EMBRAER EMB145 ER                      | None         |               1 |
| AN3           | Antonov An-24RV                        | None         |               1 |
| SC7           | SHORT SC7                              | Unknown      |               1 |
| A504          | BAE AVRO146RJ 300                      | None         |               1 |
| AVIN          | BAE AVRO146RJ 300                      | None         |               1 |
| A345          | AIRBUS A340 300                        | None         |               1 |
| GL5T          | BOMBARDIER BD700 1A10 NO SERIES EXISTS | None         |               1 |
| DH94          | DE HAVILLAND DHC8 300                  | None         |               1 |
| CL60          | CANADAIR CL600 1A11                    | Unknown      |               1 |
| U21           | BEECH 90 F90                           | None         |               1 |
| SJET          |                                        | Minor        |               1 |
| A310          | AIRBUS A310                            | None         |               1 |
| RC70          | ROCKWELL                               | None         |               1 |
| LJ55          | LEARJET 25 D                           | None         |               1 |
| B39M          | BOEING 737 900                         | None         |               1 |
| GLF5          | GULFSTREAM GV SP G550                  | None         |               1 |
| F28           | FOKKER F28 100                         | None         |               1 |
| F15           | MCDONNELL DOUGLAS                      |              |               1 |
| DH80          | DE HAVILLAND                           | None         |               1 |
| BE10          | BEECH B100                             | None         |               1 |
| BE9L          | BEECH 90                               | None         |               1 |
| MD83          | MCDONNELL DOUGLAS                      |              |               1 |
| A306          | Airbus A330-202                        |              |               1 |
| BE9L          | BEECH 90 F90                           | None         |               1 |
+---------------+----------------------------------------+--------------+-----------------+
304 rows in set (0.01 sec)

mysql>
```

```
mysql> select aircraft_type,plane_model, injury_level, count(deaths) from aviation_incident where flight_phase='Landing'and class='incident' group by aircraft_type
,plane_model,injury_level order by count(deaths) desc;
```

A359	AIRBUS A300 600	None	1
E145	EMBRAER EMB145 ER	None	1
AN3	Antonov An-24RV	None	1
SC7	SHORT SC7	Unknown	1
A504	BAE AVRO146RJ 300	None	1
AVIN	BAE AVRO146RJ 300	None	1
A345	AIRBUS A340 300	None	1
GL5T	BOMBARDIER BD700 1A10 NO SERIES EXISTS	None	1
DH94	DE HAVILLAND DHC8 300	None	1
CL60	CANADAIR CL600 1A11	Unknown	1
U21	BEECH 90 F90	None	1
SJET		Minor	1
A310	AIRBUS A310	None	1
RC70	ROCKWELL	None	1
LJ55	LEARJET 25 D	None	1
B39M	BOEING 737 900	None	1
GLF5	GULFSTREAM GV SP G550	None	1
F28	FOKKER F28 100	None	1
F15	MCDONNELL DOUGLAS		1
DH80	DE HAVILLAND	None	1
BE10	BEECH B100	None	1
BE9L	BEECH 90	None	1
MD83	MCDONNELL DOUGLAS		1
A306	Airbus A330-202		1
BE9L	BEECH 90 F90	None	1

304 rows in set (0.00 sec)

```
mysql> select aircraft_type,plane_model, injury_level, count(deaths) from aviation_incident where flight_phase='Landing'and class='incident' group by aircraft_type
,plane_model,injury_level order by count(deaths) desc;
```

A310	AIRBUS A310	None	1	
E145	EMBRAER EMB145 ER	None	1	
BE33	RAYTHEON	None	1	
VAMP	DE HAVILLAND	None	1	
E75L	EMBRAER EMB145 ER	None	1	
GL7T	BOMBARDIER BD700 1A10 NO SERIES EXISTS	None	1	
BE9L	BEECH 90	None	1	
CL60	CANADAIR CL600 1A11	Unknown	1	
MD82	MCDONNELL DOUGLAS	None	1	
WW23	IAI 1123		1	
B39M	BOEING 737 900	None	1	
PA47	PIPER PA23 250	Unknown	1	
C337	CESSNA	None	1	
A504	BAE AVRO146RJ 300	None	1	
GLF4	GULFSTREAM	None	1	
CRJX	BOMBARDIER	Unknown	1	
B37M	BOEING 747 400		1	
E55P	EMBRAER EMB505 PHENOM 300	Minor	1	
DH80	DE HAVILLAND	None	1	
F28	FOKKER F28 100	None	1	
AVIN	BAE AVRO146RJ 300	None	1	
DH94	DE HAVILLAND DHC8 300	None	1	
E190	EMBRAER ERJ190	None	1	
PA47	PIPER PA28	Unknown	1	

156 rows in set (0.01 sec)

```
mysql>
mysql>  select aircraft_type,plane_model, injury_level, count(deaths) from aviation_incident where flight_phase='Landing'and class!='incident' group by aircraft_ty
pe,plane_model,injury_level order by count(deaths) desc;
```

B37M	Boeing 737-330	None	1	
BE9L	BEECH 90 F90	None	1	
A345	AIRBUS A340 300	None	1	
SJET		Minor	1	
C525	CESSNA 525 A	None	1	
C404	CESSNA 404 NO SERIES EXISTS	None	1	
AN3	Antonov An-24RV	None	1	
C25M	CESSNA 525 A	None	1	
ZA6	MD HELICOPTER 902	None	1	
CL60	CANADAIR	None	1	
GLF4	GULFSTREAM	None	1	
FA10		Minor	1	
C04T	CESSNA 404 NO SERIES EXISTS	None	1	
MD81	MCDONNELL DOUGLAS	None	1	
MD90	MCDONNELL DOUGLAS MD90 30	None	1	
A359	AIRBUS A300 600	None	1	
GLF5	GULFSTREAM GV SP G550	None	1	
BE10	BEECH B100	None	1	
V10	ROCKWELL	None	1	
A306	Airbus A330-202		1	
COL3	KAVANAGH B350	None	1	
RC70	ROCKWELL	None	1	

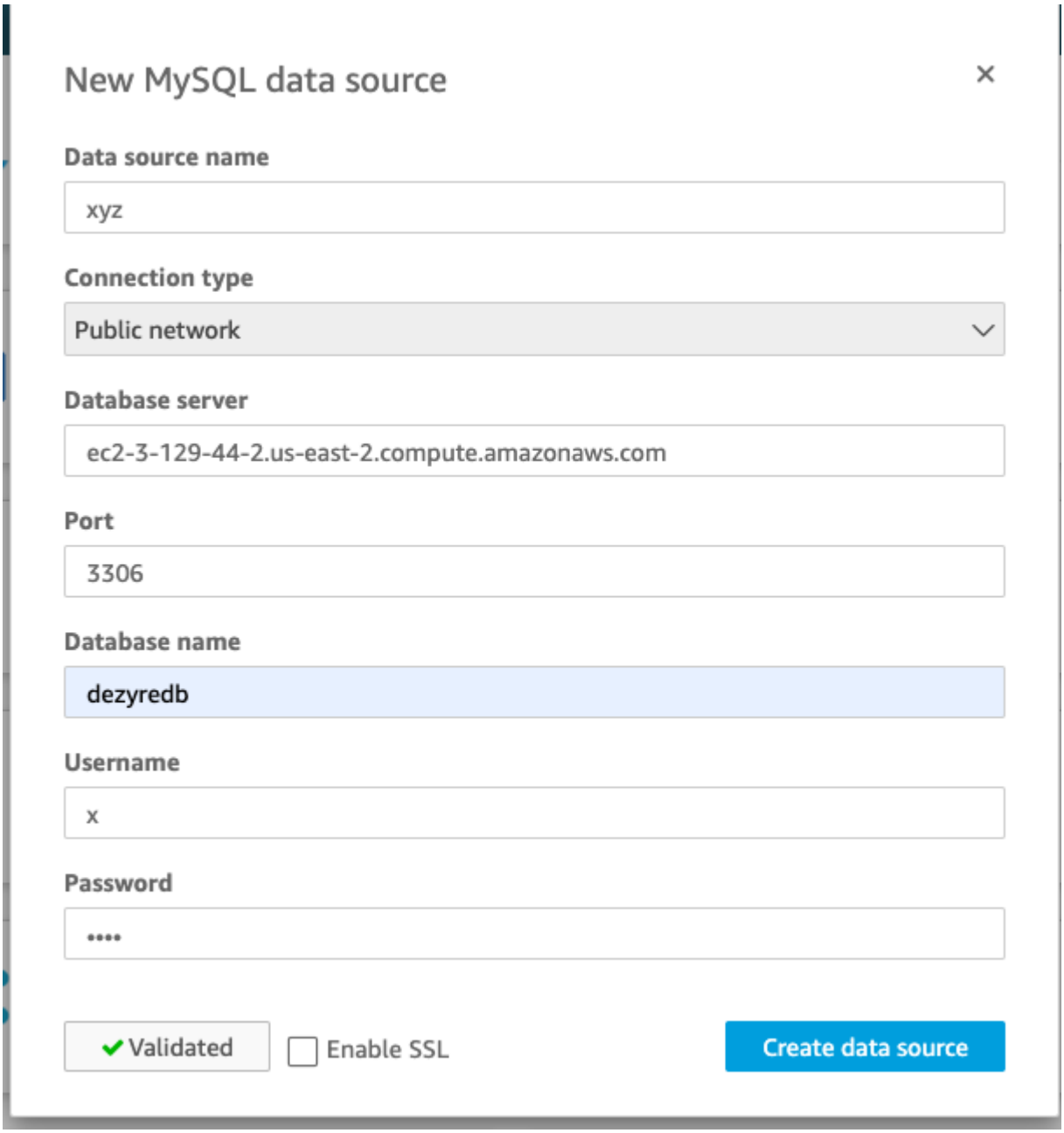
198 rows in set (0.00 sec)

7. Visualisation of data in Quicksight AWS

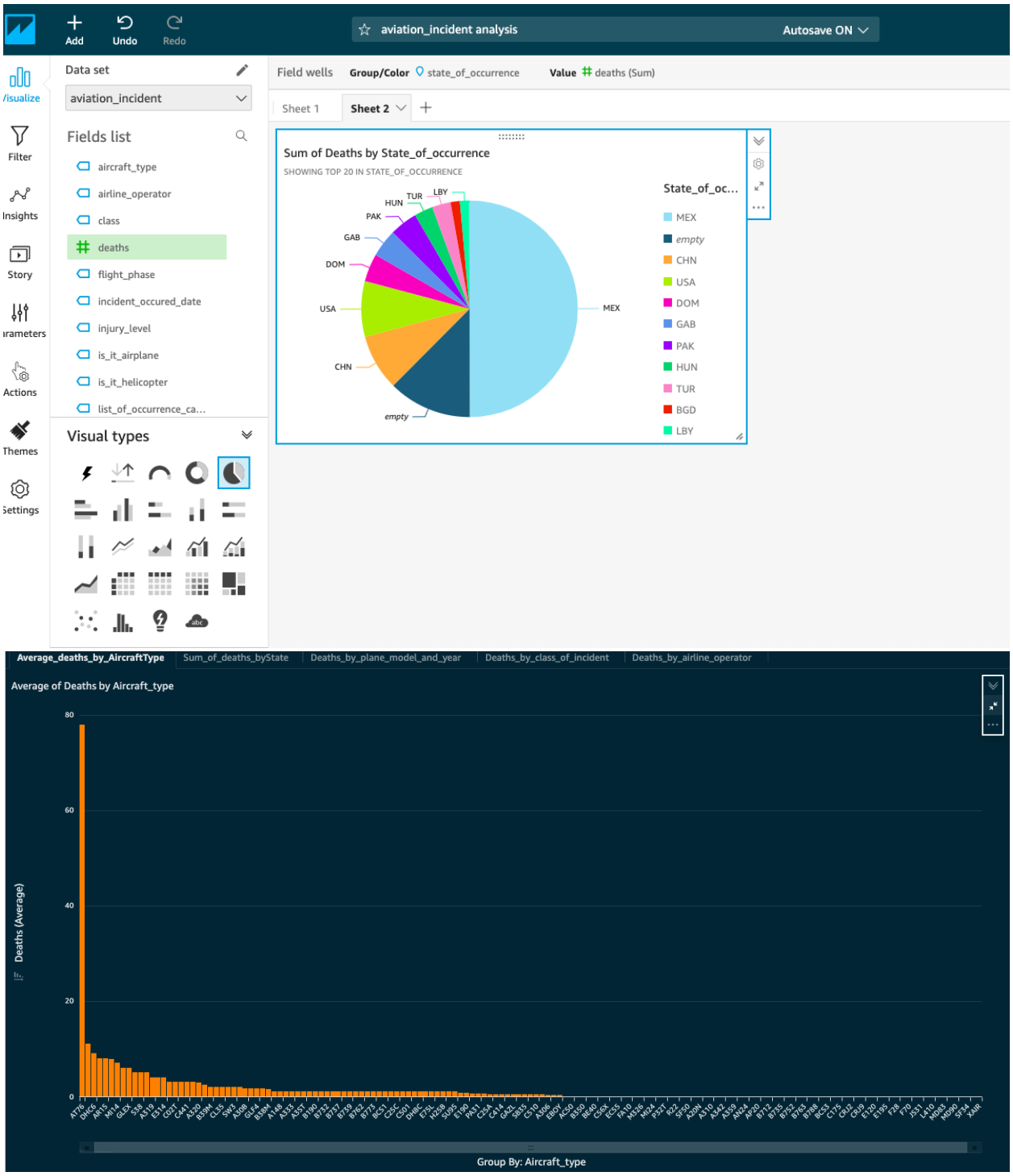
To provide access to Quicksight user from MYSQl follow the steps below
Login into MYSQL and create Quciksight user and grant required permissions

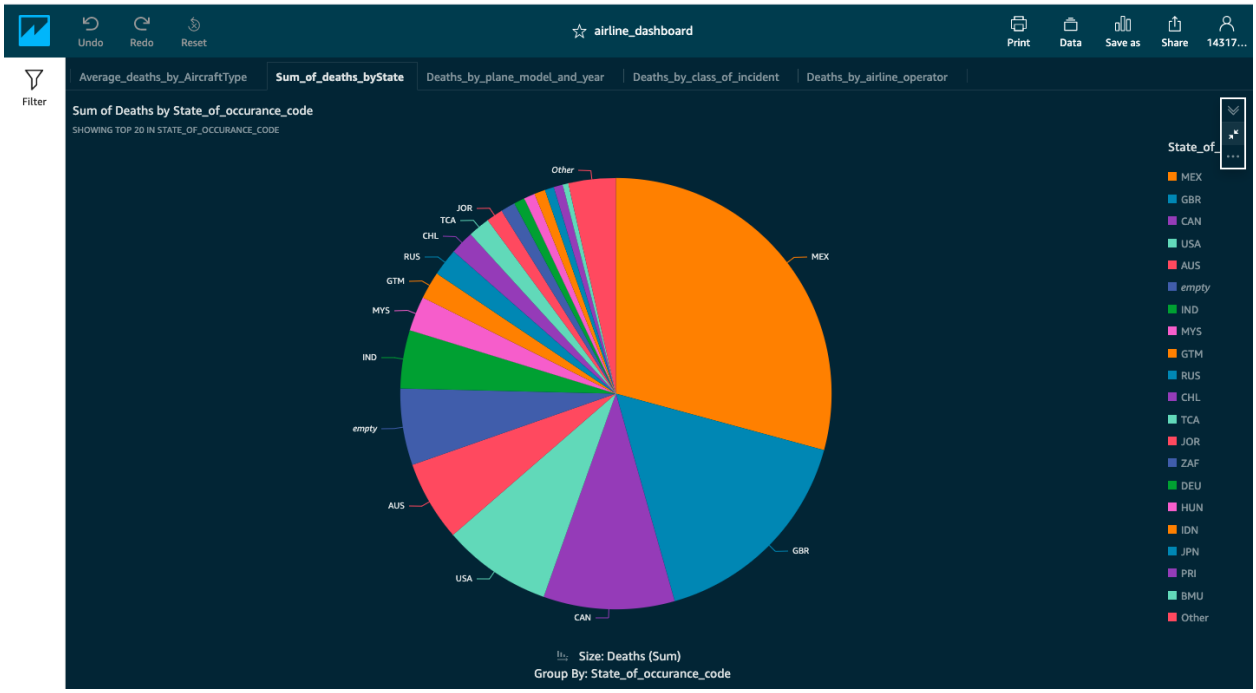
```
mysql> use mysql;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> flush privileges;
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'root'@'3.129.44.2' IDENTIFIED BY 'root';
Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql>
mysql> CREATE USER 'x'@'ec2-52-15-247-168.us-east-2.compute.amazonaws.com' IDENTIFIED BY 'root';
Query OK, 0 rows affected (0.00 sec)

mysql>
mysql> GRANT ALL PRIVILEGES ON *.* TO 'x'@'ec2-52-15-247-168.us-east-2.compute.amazonaws.com'
WITH GRANT OPTION;
Query OK, 0 rows affected (0.00 sec)

mysql> CREATE USER 'x'@'%' IDENTIFIED BY 'root';
Query OK, 0 rows affected (0.00 sec)

mysql> GRANT ALL PRIVILEGES ON *.* TO 'x'@'%' WITH GRANT OPTION;
Query OK, 0 rows affected (0.00 sec)

mysql>
```

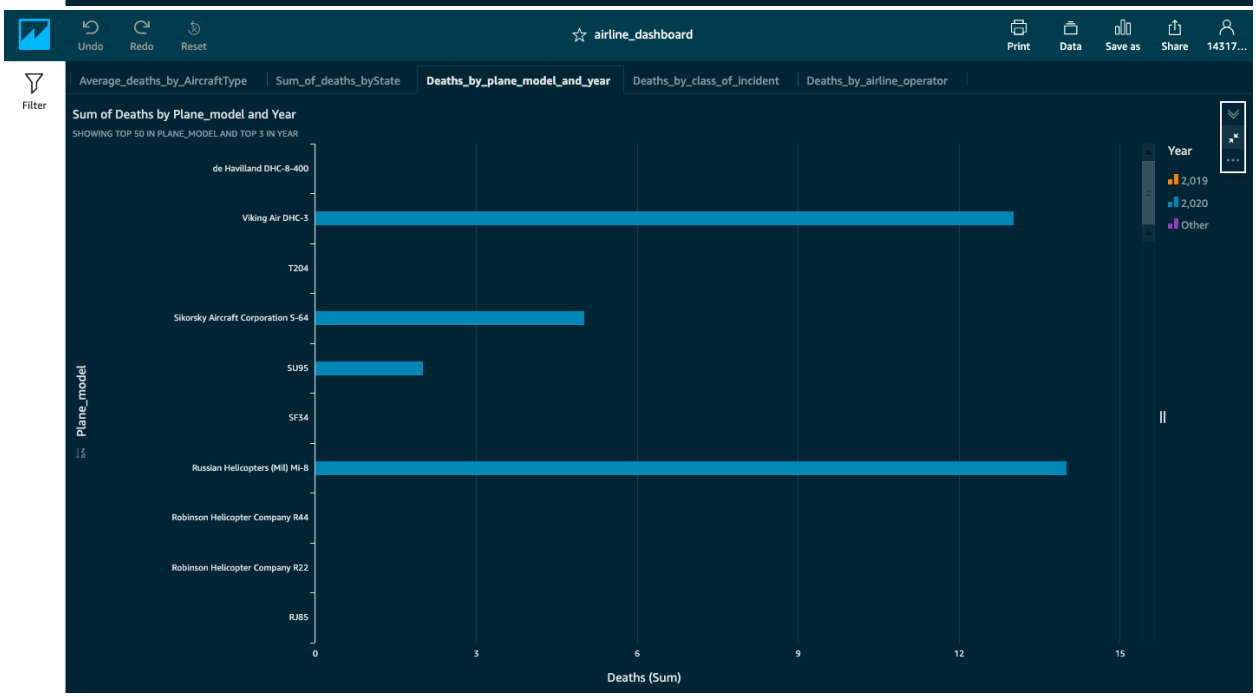
Then go to quicksight console and Create new dataset and select MYSQL and configure required parameters as follows :

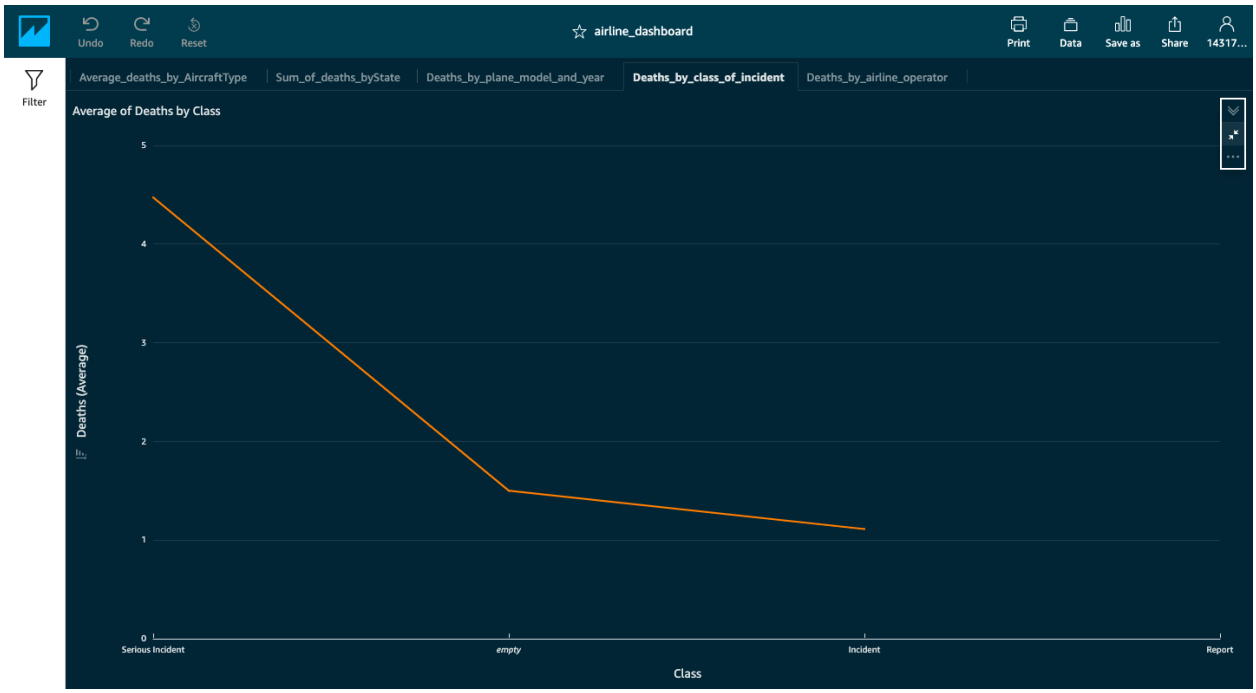


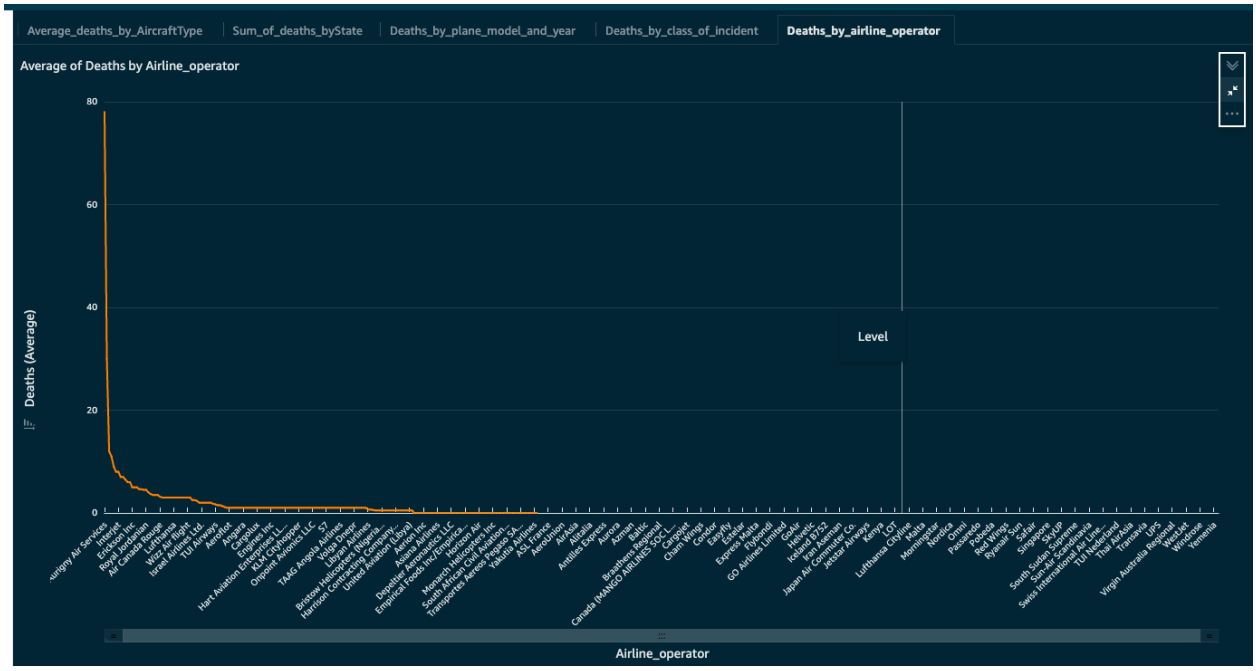
Once the connection is validated, click on Create datasource
Then it will show the list of databases available in MYSQL
Choose the aviation_incident database and click on continue .
Then it shows the data as follows from which we can create required visualisations


Add
Undo
Redo
aviation_incident analysis
Autosave ON
Data set
aviation_incident
Fields list
aircraft_type
airline_operator
class
deaths
flight_phase
incident_occured_date
injury_level
is_it_airplane
is_it_helicopter
list_of_occurrence_ca...
Visual types
Field wells
Group/Color state_of_occurrence
Value deaths (Sum)
Sheet 1
Sheet 2
Sum of Deaths by State_of_occurrence
SHOWING TOP 20 IN STATE_OF_OCCURRENCE
State_of_oc...
MEX
empty
CHN
USA
DOM
GAB
PAK
HUN
TUR
BGD
LBY
Average_deaths_by_AircraftType
Sum_of_deaths_byState
Deaths_by_plane_model_and_year
Deaths_by_class_of_incident
Deaths_by_airline_operator
Average of Deaths by Aircraft_type
Deaths (Average)
Group By: Aircraft_type


Undo
Redo
Reset
airline_dashboard
Print
Data
Save as
Share
14317...
Filter
Average_deaths_by_AircraftType
Sum_of_deaths_byState
Deaths_by_plane_model_and_year
Deaths_by_class_of_incident
Deaths_by_airline_operator
Sum of Deaths by State_of_occurance_code
SHOWING TOP 20 IN STATE_OF_OCCURANCE_CODE
Other
JOR
TCA
CHL
RUS
GTM
MYS
IND
empty
AUS
USA
CAN
GBR
MEX
State_of
MEX
GBR
CAN
USA
AUS
empty
IND
MYS
GTM
RUS
CHL
TCA
JOR
ZAF
DEU
HUN
IDN
JPN
PRI
BMU
Other
Size: Deaths (Sum)
Group By: State_of_occurance_code


Undo
Redo
Reset
airline_dashboard
Print
Data
Save as
Share
14317...
Filter
Average_deaths_by_AircraftType
Sum_of_deaths_byState
Deaths_by_plane_model_and_year
Deaths_by_class_of_incident
Deaths_by_airline_operator
Sum of Deaths by Plane_model and Year
SHOWING TOP 50 IN PLANE_MODEL AND TOP 3 IN YEAR
de Havilland DHC-8-400
Viking Air DHC-3
T204
Sikorsky Aircraft Corporation S-64
SU95
SF34
Russian Helicopters (Mil) Mi-8
Robinson Helicopter Company R44
Robinson Helicopter Company R22
RJ85
Plane_model
0
3
6
9
12
15
Deaths (Sum)
Year
2,019
2,020
Other





Extended project : We compare the performance of Hive/Druid/MySQL with variable workloads in project recordings.

Discuss on project extendibility using RDS and Sqoop