**#Create a new database in hive.**

hive> create database airlines;

**#Check if the new database is created in hive.**

hive> show databases;

**#Making a new folder DelayedFlights and loading the dataset in the hdfs.**

[cloudera@quickstart ~]$ hadoop fs -mkdir /user/cloudera/DelayedFlights

[cloudera@quickstart ~]$ hadoop fs -put DelayedFlights.csv /user/cloudera/DelayedFlights

**#Loading hive**

[cloudera@quickstart ~]$ hive

**#Create a new table in the airlines database in hive.**

hive> create table airlines.DelayedFlights (ID int, Year int, Month int, DayofMonth int, DayOfWeek int, DepTime float, CRSDepTime int, ArrTime float, CRSArrTime int, UniqueCarrier string, FlightNum int, TailNum string, ActualElapsedTime float, CRSElapsedTime float, AirTime float, ArrDelay float, DepDelay float, Origin string, Dest string, Distance int, TaxiIn float, TaxiOut float, Cancelled int, CancellationCode string, Diverted int, CarrierDelay float, WeatherDelay float, NASDelay float, SecurityDelay float, LateAircraftDelay float) row format delimited fields terminated by ',' stored as textfile location '/user/hive/warehouse/airlines.db/DelayedFlights';

**#Using the airlines database.**

hive> use airlines;

**#Checking if the new table was created.**

hive> show tables;

**#Loading the records of the data in the table created in the airlines database.**

hive> load data local inpath 'DelayedFlights.csv' overwrite into table DelayedFlights;

**#This code makes sure that the headers of the table are visible.**

hive> set hive.cli.print.header=true;

**Hive Queries**

**Q1.) What is the average of the delays that can be controlled?**

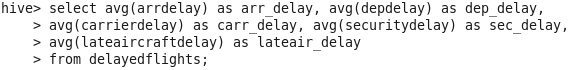
**Code:**

hive> select avg(arrdelay) as arr\_delay, avg(depdelay) as dep\_delay,

> avg(carrierdelay) as carr\_delay, avg(securitydelay) as sec\_delay,

> avg(lateaircraftdelay) as lateair\_delay

> from delayedflights;



**Output:**

arr\_delay dep\_delay carr\_delay

42.19988477321014 43.185176464999756 19.179398920069772

sec\_delay lateair\_delay

0.09013713959573158 25.296466178432176



**Q2. Which is the month having the most arrival delay and departure delay (in minutes)?**

**Code:**

hive> select month, avg(arrdelay) as arr\_delay, avg(depdelay) as dep\_delay

> from delayedflights

> group by month;



**Output:**

month arr\_delay dep\_delay

1 42.801492496514776 43.54567992720417

2 45.00612326043738 44.98844006880032

3 41.91476943372844 42.8852232102847

4 38.83530491074601 39.864115313272876

5 37.59357217227312 39.0415752064388

6 46.5327283278636 45.825925520371904

7 45.99513643774256 46.65988685123944

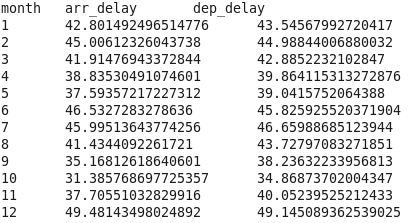
8 41.4344092261721 43.72797083271851

9 35.16812618640601 38.23632233956813

10 31.385768697725357 34.86873702004347

11 37.70551032829916 40.05239525212433

12 49.48143498024892 49.145089362539025



On an average, the maximum number of arrival delays and departure delays occurs in the month of December with 49.48 minutes and 49.15 minutes, respectively.

**Q3. Which are the top 10 destinations that has been visited the most?**

**Code:**

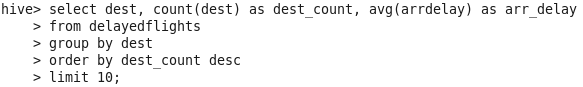
hive> select dest, count(dest) as dest\_count, avg(arrdelay) as arr\_delay

> from delayedflights

> group by dest

> order by dest\_count desc

> limit 10;



**Output:**

dest dest\_count arr\_delay

ORD 108984 60.028153142751584

ATL 106898 47.898826731743945

DFW 70657 43.1663983129337

DEN 63003 37.5066984884646

LAX 59969 36.752268435740184

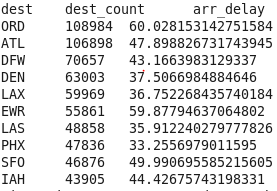
EWR 55861 59.87794637064802

LAS 48858 35.912240279777826

PHX 47836 33.2556979011595

SFO 46876 49.990695585215605

IAH 43905 44.42675743198331



The top 10 destinations that have been visited the most are

**Q4. What are the top 10 origins from which the most number of flights have departed?**

**Code:**

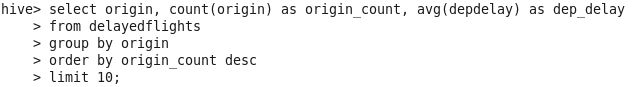
hive> select origin, count(origin) as origin\_count, avg(depdelay) as dep\_delay

> from delayedflights

> group by origin

> order by origin\_count desc

> limit 10;



**Output:**

origin origin\_count dep\_delay

ATL 131613 40.89324002948037

ORD 125979 50.53116789306154

DFW 95414 38.34061039260486

DEN 74323 37.69886845256516

LAX 58772 38.17499829850949

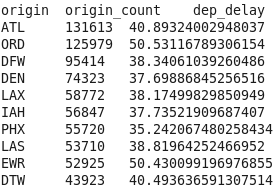
IAH 56847 37.73521909687407

PHX 55720 35.242067480258434

LAS 53710 38.81964252466952

EWR 52925 50.430099196976855

DTW 43923 40.493636591307514



**Q5. Which day of the week has the most arrival and departure delay (in minutes)?**

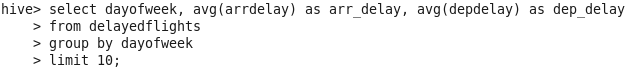
**Code:**

hive> select dayofweek, avg(arrdelay) as arr\_delay, avg(depdelay) as dep\_delay

> from delayedflights

> group by dayofweek

> limit 10;



**Output:**

dayofweek arr\_delay dep\_delay

1 41.731355797696516 43.22763316639914

2 43.80667914817554 44.07560271783493

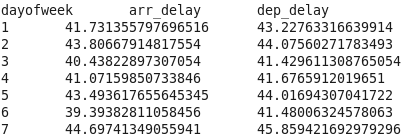
3 40.43822897307054 41.429611308765054

4 41.07159850733846 41.6765912019651

5 43.493617655645345 44.01694307041722

6 39.39382811058456 41.48006324578063

7 44.69741349055941 45.859421692979296



**Q6. Which are the top 10 flights having the highest arrival and departure delay?**

**Code:**

**For arrival delay,**

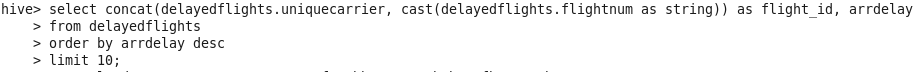
hive> select concat(delayedflights.uniquecarrier, cast(delayedflights.flightnum as string)) as

flight\_id, arrdelay

> from delayedflights

> order by arrdelay desc

> limit 10;



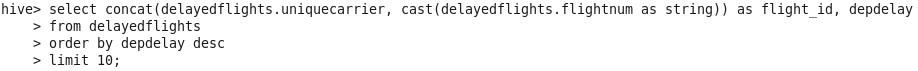
**For departure delay,**

hive> select concat(delayedflights.uniquecarrier, cast(delayedflights.flightnum as string)) as flight\_id, depdelay

> from delayedflights

> order by depdelay desc

> limit 10;



**Output:**

**For arrival delay,**

NW808 2461.0

NW1699 2453.0

NW1107 1951.0

MQ3538 1707.0

NW357 1655.0

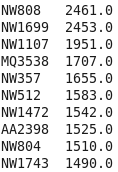
NW512 1583.0

NW1472 1542.0

AA2398 1525.0

NW804 1510.0

NW1743 1490.0



**For departure delay,**

NW1699 2467.0

NW808 2457.0

NW1107 1952.0

MQ3538 1710.0

NW357 1597.0

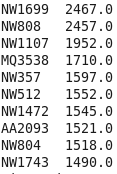
NW512 1552.0

NW1472 1545.0

AA2093 1521.0

NW804 1518.0

NW1743 1490.0



**Q7. In each month, how many flights have been cancelled?**

**Code:**

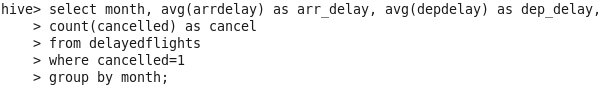
hive> select month, avg(arrdelay) as arr\_delay, avg(depdelay) as dep\_delay,

> count(cancelled) as cancel

> from delayedflights

> where cancelled=1

> group by month;



**Output:**

month arr\_delay dep\_delay cancel

10 NULL 76.72881355932203 59

11 NULL 83.95744680851064 94

12 NULL 95.50625 480

