

Flight Delay Analysis



Poojith Shankar Shetty
001821661
4/26/2019

AGENDA

- Introduction
- Dataset Overview
- List of Analysis
- Representation using Tableau
- Analysis Outputs
- Technologies and Patterns used
- Appendix
- Additional Screenshots

INTRODUCTION

- Flight Cancellation Analysis
 - Based on Airlines, month, State etc.
 - Reason for the Delay.
- Flight Delay Analysis
 - Time of Delay
 - Based on Airlines, month and State
 - Reason for the Delay
- Dataset Link
 - https://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=236&DB_Short_Name=On-Time
- Flight Distance Travelled
 - Details about how much distance the different flight travelled

DATA SET OVERVIEW

Columns of Dataset:

YEAR	ORIGIN_CITY_NAME	DEP_DELAY_NEW	CRS_ELAPSED_TIME
QUARTER	ORIGIN_STATE_ABR	TAXI_OUT	ACTUAL_ELAPSED_TIME
MONTH	ORIGIN_STATE_NM	WHEELS_ON	AIR_TIME
DAY_OF_MONTH	DEST_AIRPORT_ID	TAXI_IN	FLIGHTS
DAY_OF_WEEK	DEST	CRS_ARR_TIME	DISTANCE
FL_DATE	DEST_CITY_NAME	ARR_TIME	DISTANCE_GROUP
OP_UNIQUE_CARRIER	DEST_STATE_ABR	ARR_DELAY	CARRIER_DELAY
TAIL_NUM	DEST_STATE_NM	ARR_DELAY_NEW	WEATHER_DELAY
OP_CARRIER_FL_NUM	CRS_DEP_TIME	CANCELLED	NAS_DELAY
ORIGIN_AIRPORT_ID	DEP_TIME	CANCELLATION_CODE	SECURITY_DELAY
ORIGIN	DEP_DELAY	DIVERTED	LATE_AIRCRAFT_DELAY

DATA SET OVERVIEW

- Cancellation Code: These are the reasons for Flight cancellation
 - A – Carrier
 - B – Weather
 - C – National Air System
 - D – Security
- All the time in the data set are in Minutes.(Delay related columns).

LIST OF ANALYSIS

1. Used PutMerge to Merge the Data.
2. Used AWS Cloud EMR instance for running MapReduce Jobs.
3. Used counting with Counters to determine Number of Flights in different Distance Groups
4. Used Memory-Conscious Median and Standard Deviation of Distance travelled based on Airliner used separate Combiner to simplify the input to Reducer.
5. Calculated the each Cancellation of Flight based on Airliner each month.
6. Calculated the each Delay of Flight based on Airliner each month.
7. Did Secondary Sorting of the States and the available Airports in that state to know the list of Airports Present in US.(Partitioner is used in this case)
8. Calculated Average Delay by the airliner each month used Custom Writable class for this technique.

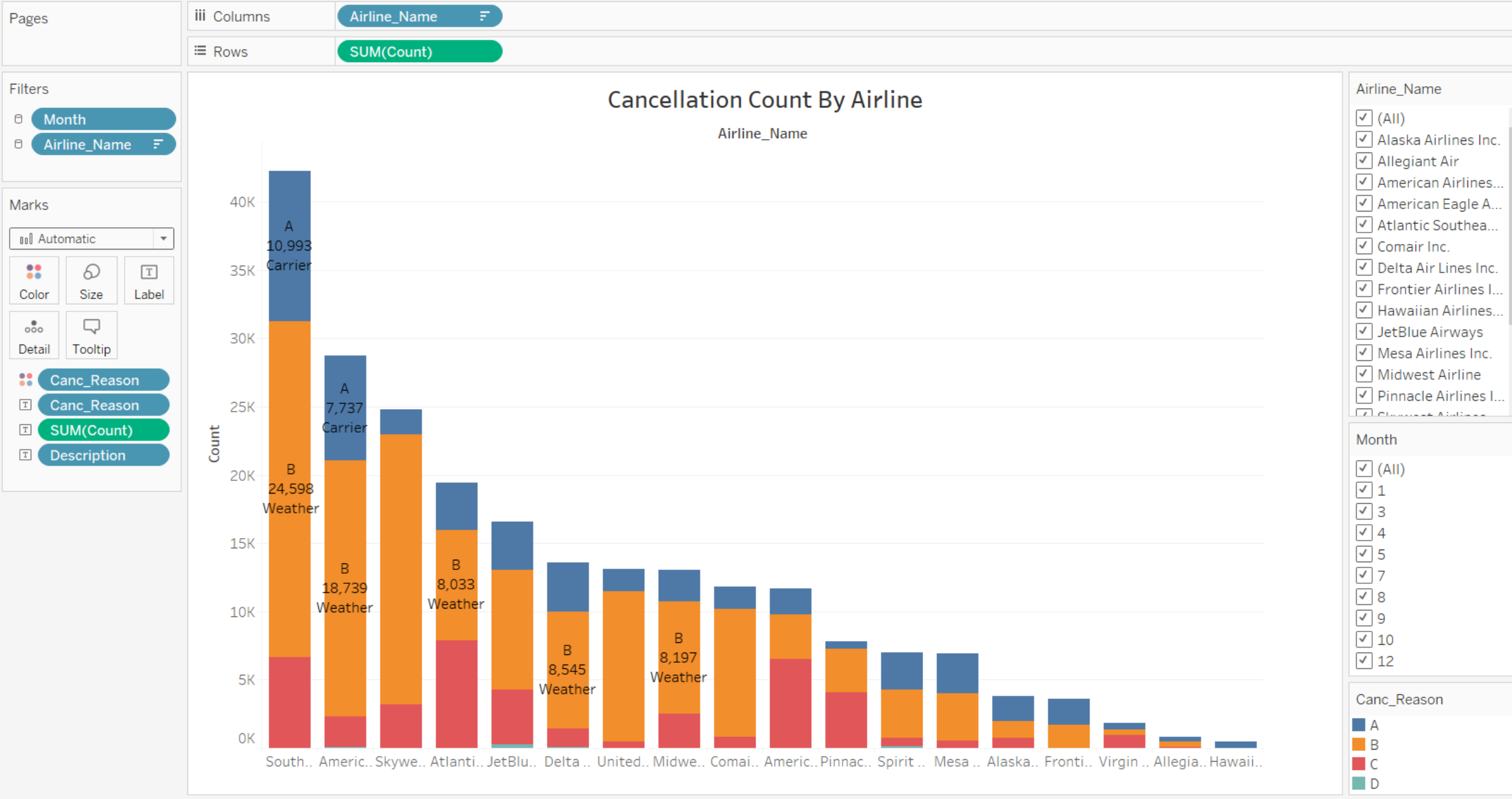
LIST OF ANALYSIS

9. Calculated Total Average Flight Delay by different Airliner.
10. Used Shuffling Technique to shuffle the Data.
11. Calculated Total Cancelled Flights based on Different Reason.
12. Calculated Total Delay of Flight each month
13. Determined Top 10 flight data based on longest Distance.
14. Determined Delay Median based on Airliner
15. Determined Distinct Airline in the Dataset.
16. Separated the data based on Cancellation Reason by using Binning Technique.

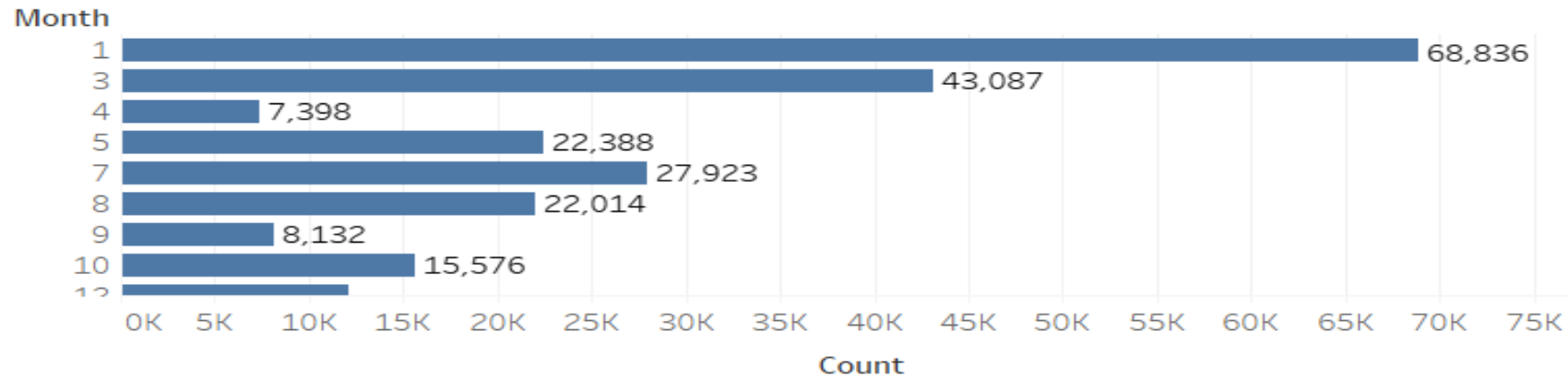
LIST OF ANALYSIS

17. Used Bloom Filter to Filter the data based on the Airport input.
18. Used Filtered output using Chaining and determined the Flights from the Airport using Inverted Index Method.
19. Based on the Carrier Names from a different Carrier CSV found the Full name of the Flights present in the original Data.
20. Determined the Minimum and Maximum Distance travelled by flight based on each Airliner.
21. Calculated Flight cancelled statewise in each month to determine which state has most flight cancelled.
22. Determined Flight delayed state wise in each month.
23. Found Top 10 Airline with Most Flights using Pig.
24. Filtered the Latest Data based on Year using Pig.
25. Joined Airline Data based on Airline Name using Pig.

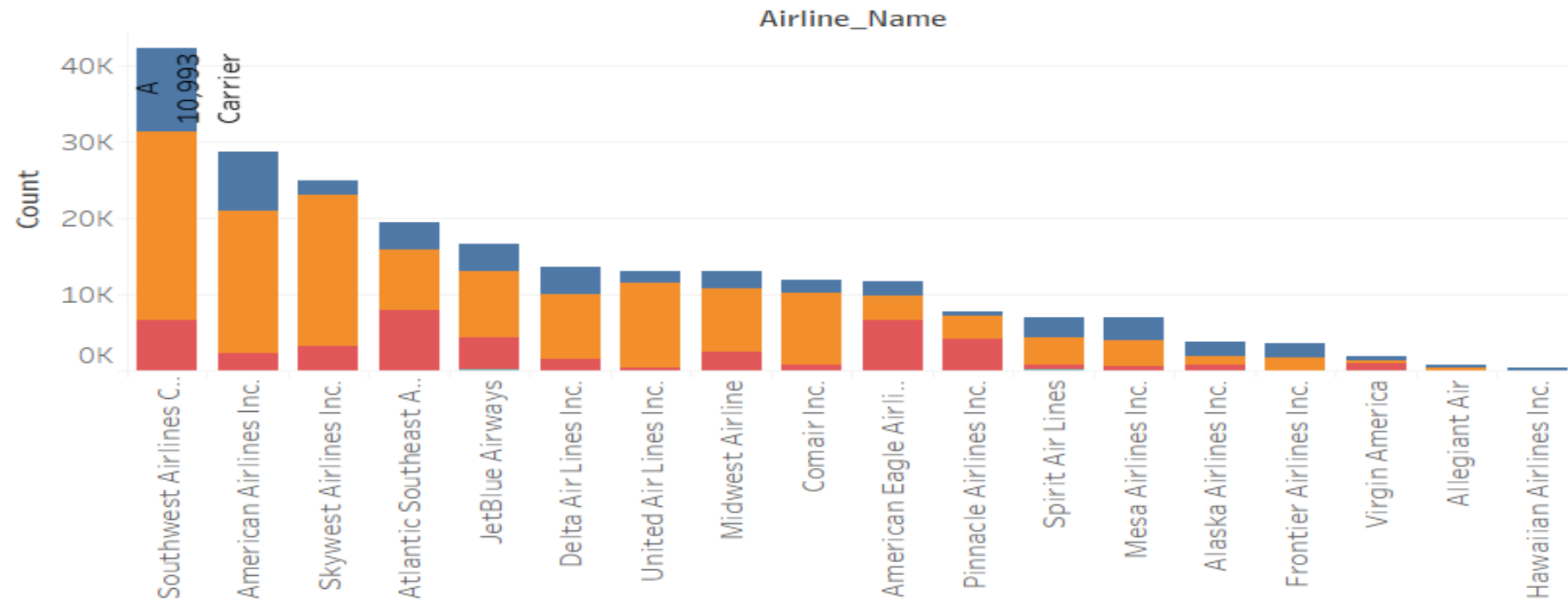
Analysis using Tableau



Count of Cancellation By Month



Cancellation Count By Airline



Pages

Columns

AirlineDesc

Rows

SUM(Count)

Filters

Month

AirlineDesc

Marks

Automatic

Color

Size

Label

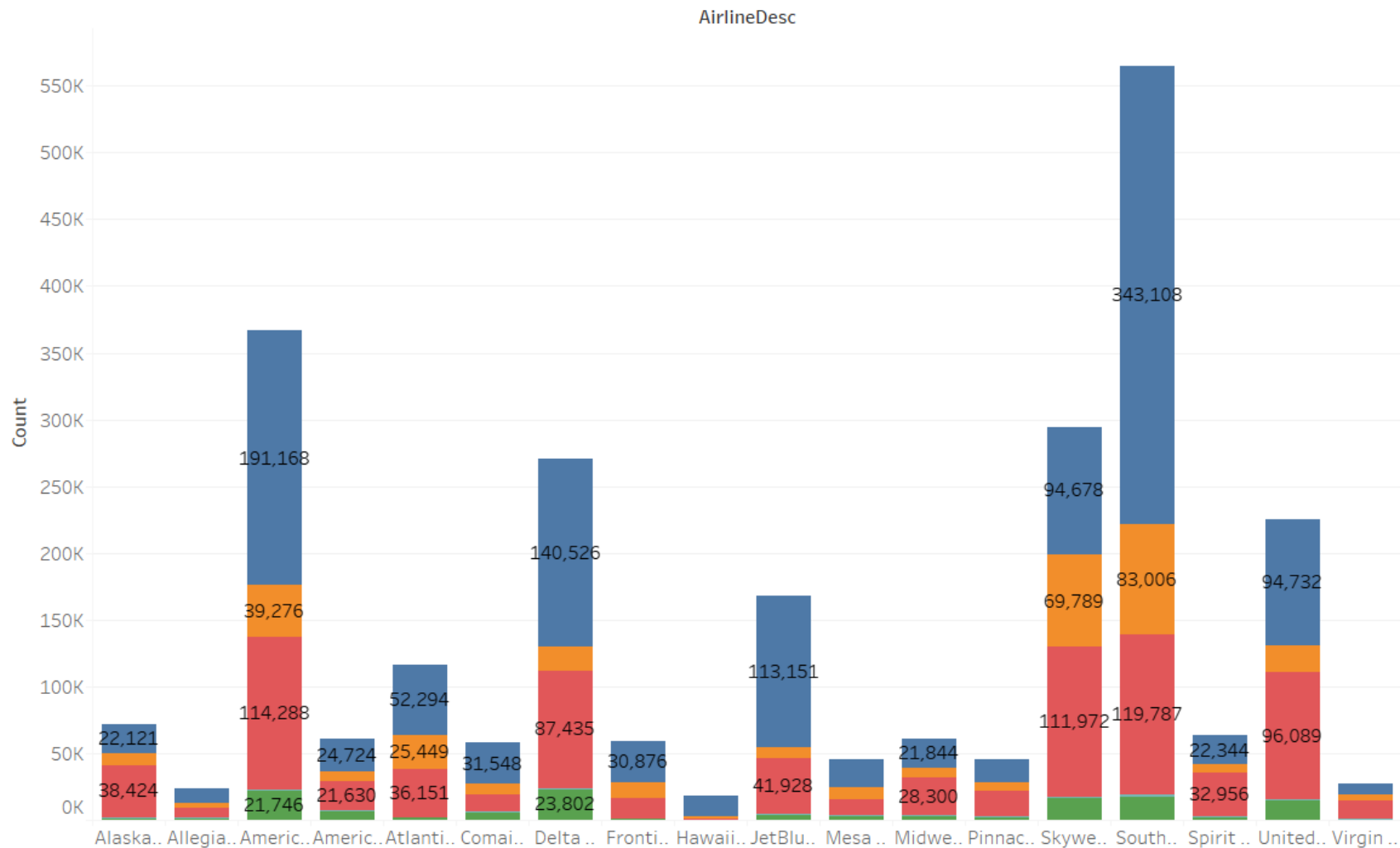
Detail

Tooltip

Delay_reason

SUM(Count)

Delay Reason by airline count



AirlineDesc

- ☒ (All)
- ☒ Alaska Airlines Inc.
- ☒ Allegiant Air
- ☒ American Airlines...
- ☒ American Eagle A...
- ☒ Atlantic Southea...
- ☒ Comair Inc.
- ☒ Delta Air Lines Inc.
- ☒ Frontier Airlines I...
- ☒ Hawaiian Airlines...
- ☒ JetBlue Airways
- ☒ Mesa Airlines Inc.
- ☒ Midwest Airline
- ☒ Pinnacle Airlines

Month

- ☒ (All)
- ☒ 1
- ☒ 3
- ☒ 4
- ☒ 5
- ☒ 7
- ☒ 8
- ☒ 9
- ☒ 10
- ☒ 12

Delay_reason

- ☒ CARRIER_DELAY
- ☒ LATE_AIRCRAFT_DEL..
- ☒ NAS_DELAY
- ☒ SECURITY_DELAY
- ☒ WEATHER_DELAY

Filters

Month

AirlineDesc

Marks

Automatic

Color

Size

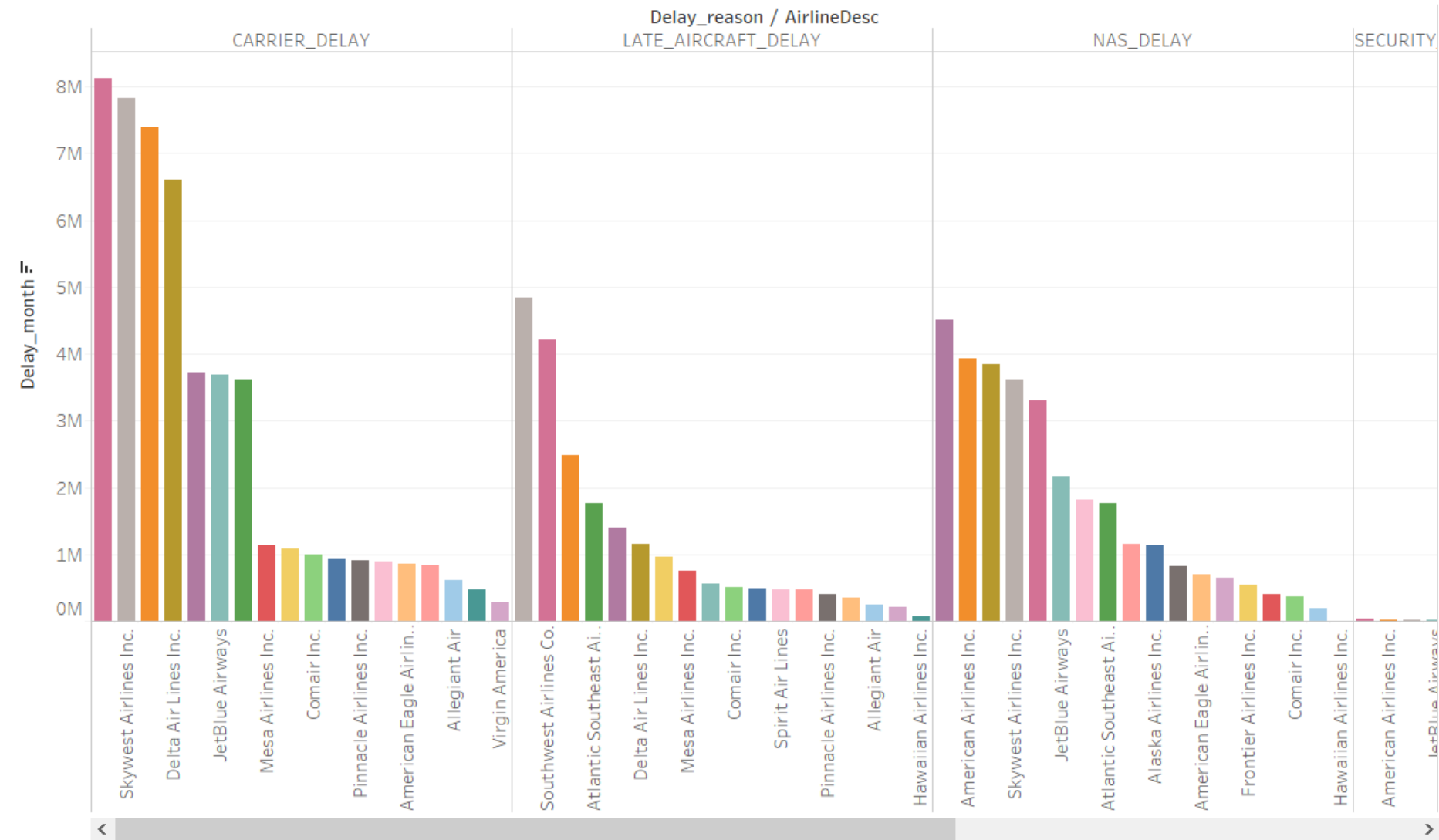
Label

Detail

Tooltip

AirlineDesc

Delay By airline and reason



AirlineDesc

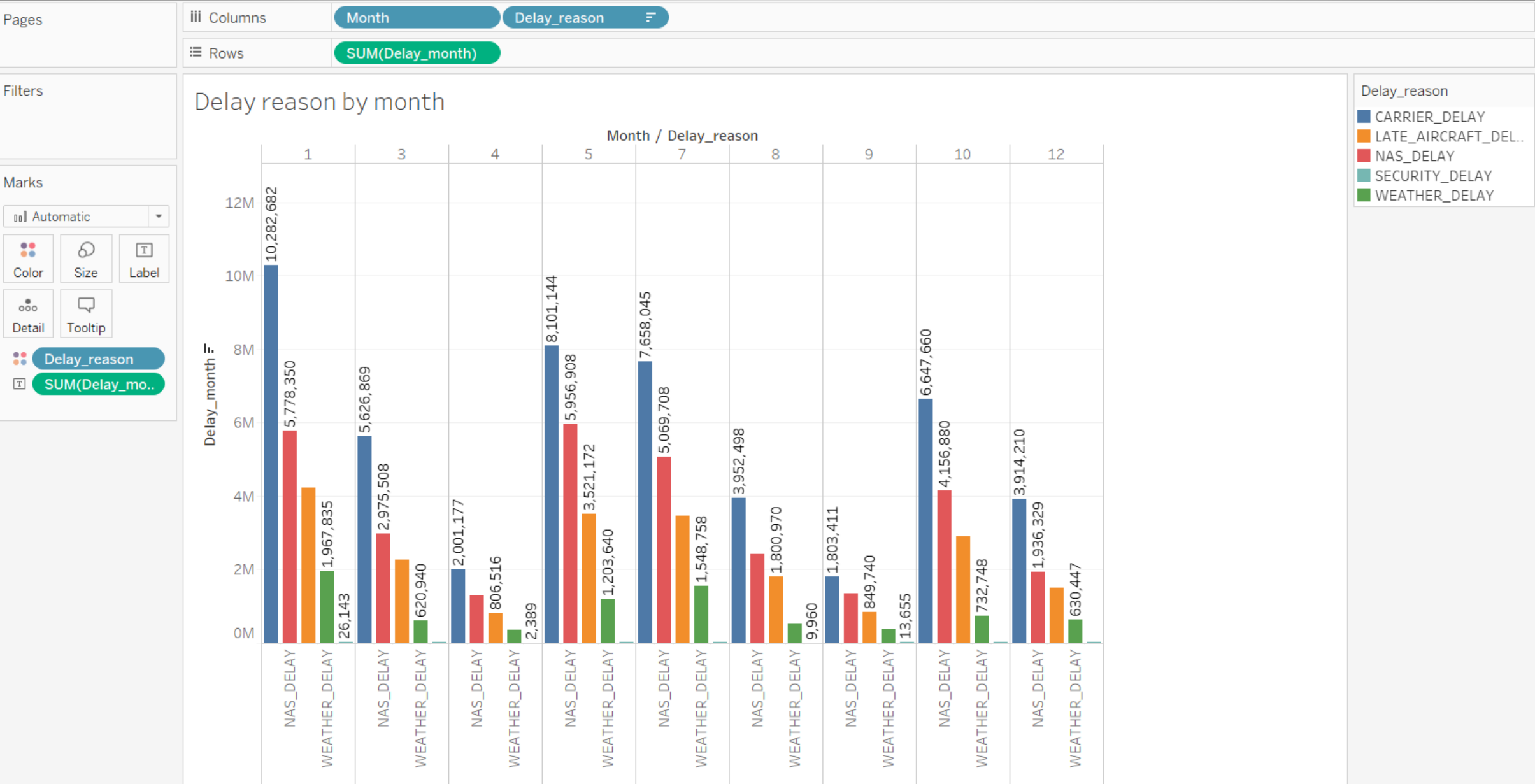
- ☒ (All)
- ☒ Alaska Airlines Inc.
- ☒ Allegiant Air
- ☒ American Airlines...
- ☒ American Eagle A...
- ☒ Atlantic Southea...
- ☒ Comair Inc.
- ☒ Delta Air Lines Inc.
- ☒ Frontier Airlines I...

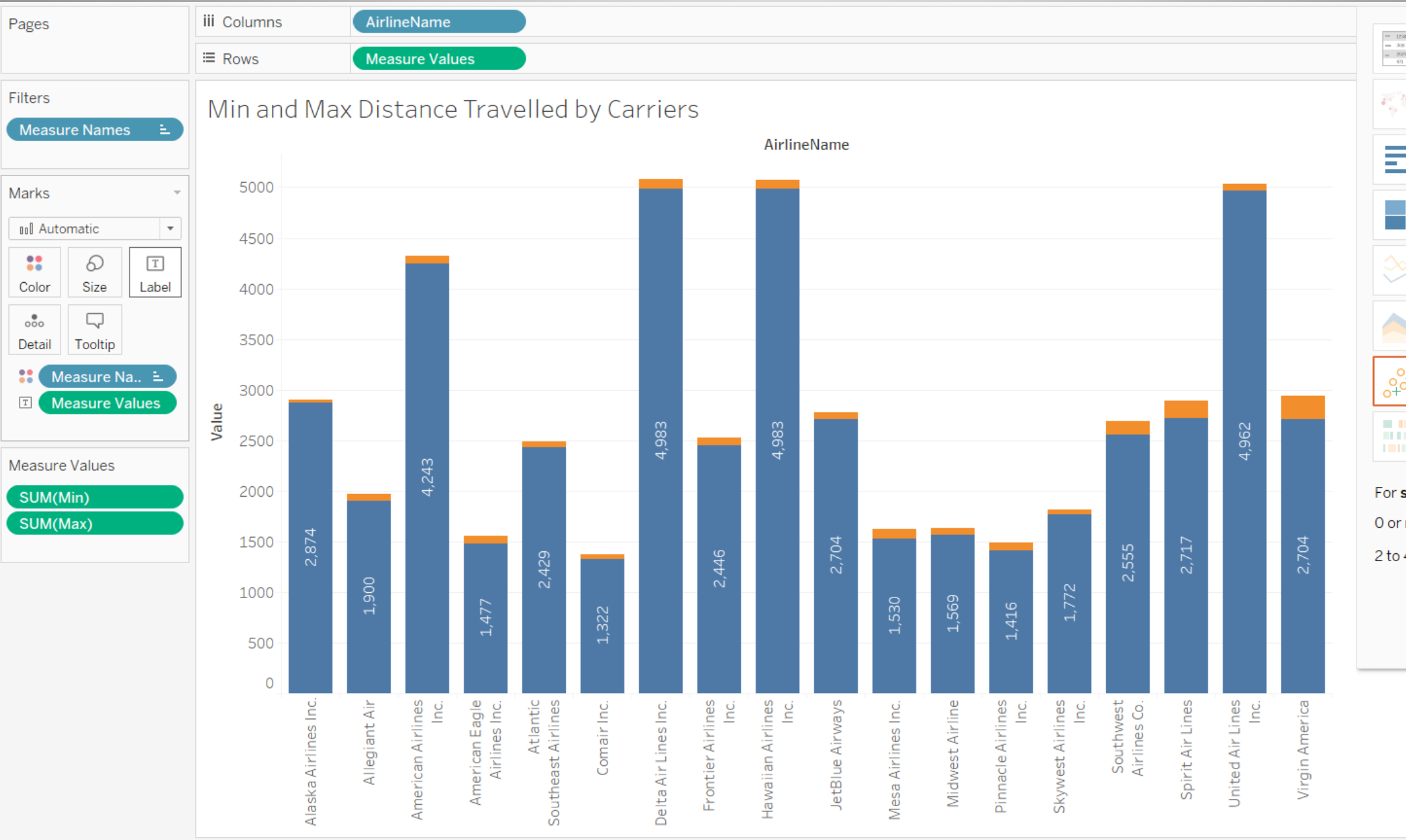
Month

- ☒ (All)
- ☒ 1
- ☒ 3
- ☒ 4
- ☒ 5
- ☒ 7
- ☒ 8
- ☒ 9
- ☒ 10

AirlineDesc

- ☒ Alaska Airlines Inc.
- ☒ Allegiant Air
- ☒ American Airlines I...
- ☒ American Eagle Air...
- ☒ Atlantic Southeast ...
- ☒ Comair Inc.
- ☒ Delta Air Lines Inc.
- ☒ Frontier Airlines Inc.
- ☒ Hawaiian Airlines I...
- ☒ JetBlue Airways
- ☒ Mesa Airlines Inc.





Pages

Columns

State

Rows

SUM(Count)

Filters

Month

Marks

Automatic

Color

Size

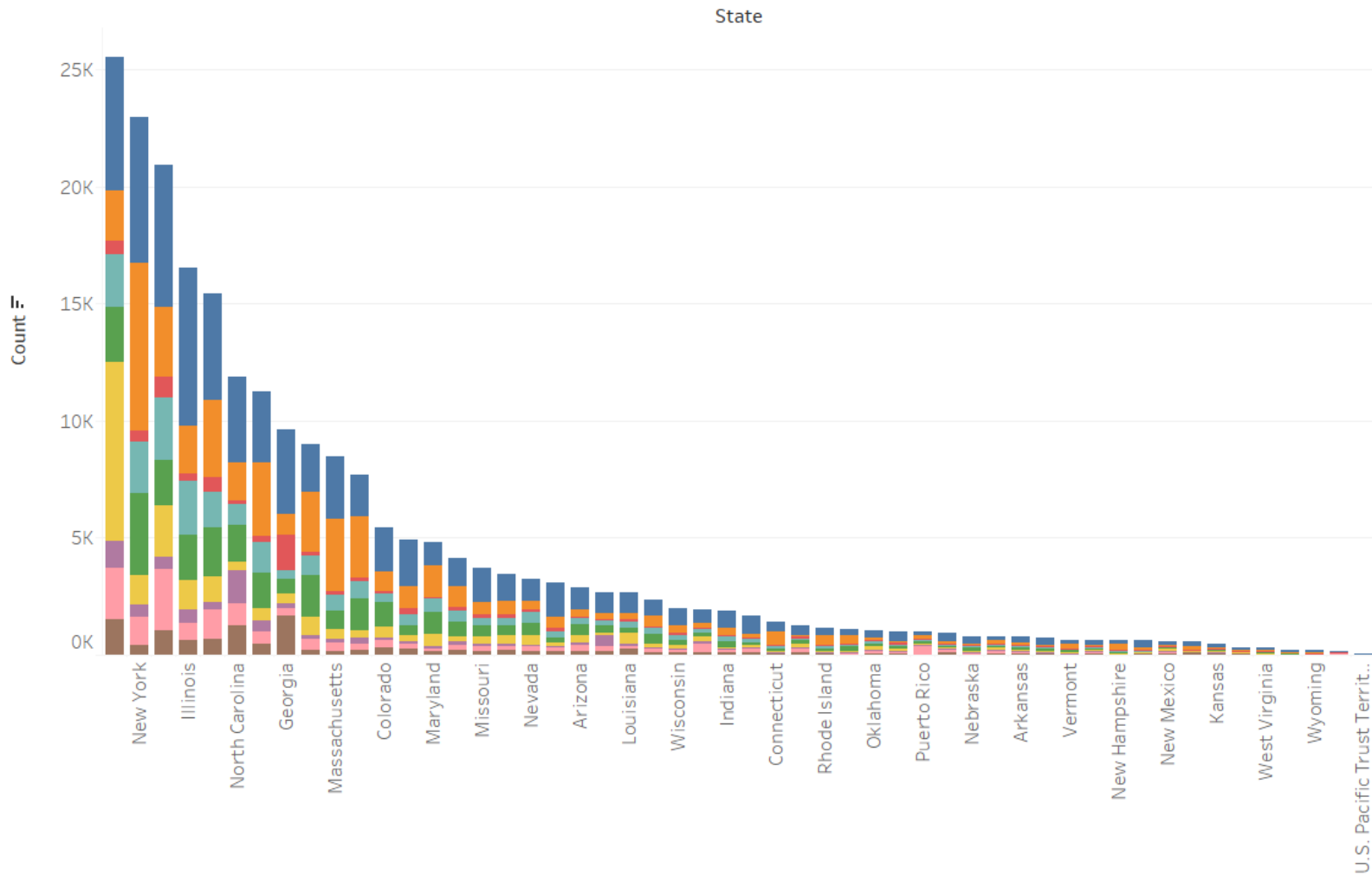
Label

Detail

Tooltip

Month

Statewise Cancellation Monthly



Month

☒ (All)

☒ 1

☒ 3

☒ 4

☒ 5

☒ 7

☒ 8

☒ 9

☒ 10

☒ 12

Month

1

3

4

5

7

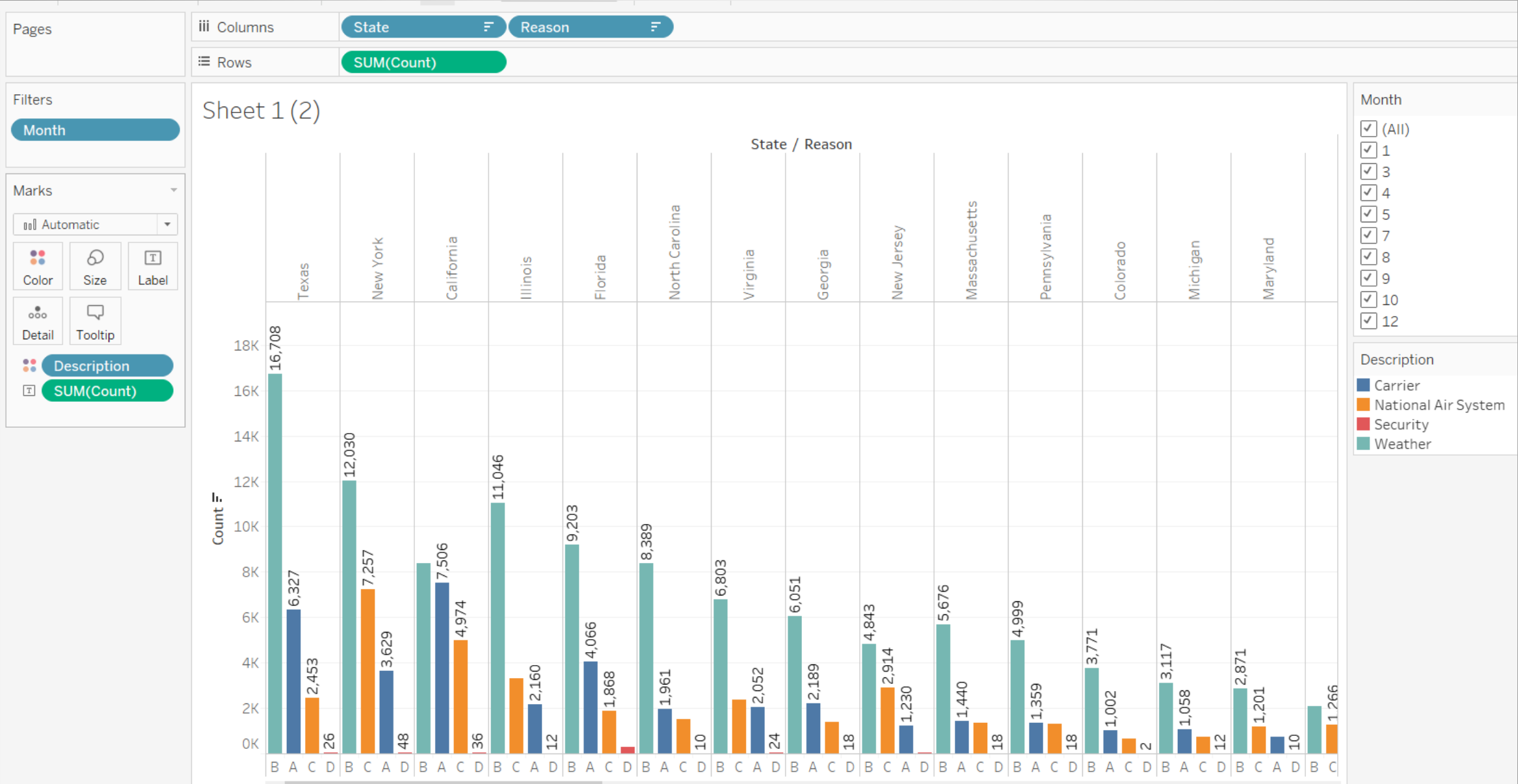
8

9

10

12

STATEWISE BASED CANCELLATION COUNT



Pages

Columns

State

Rows

SUM(Total_Delay)

Filters

Month

Marks

Automatic

Color

Size

Label

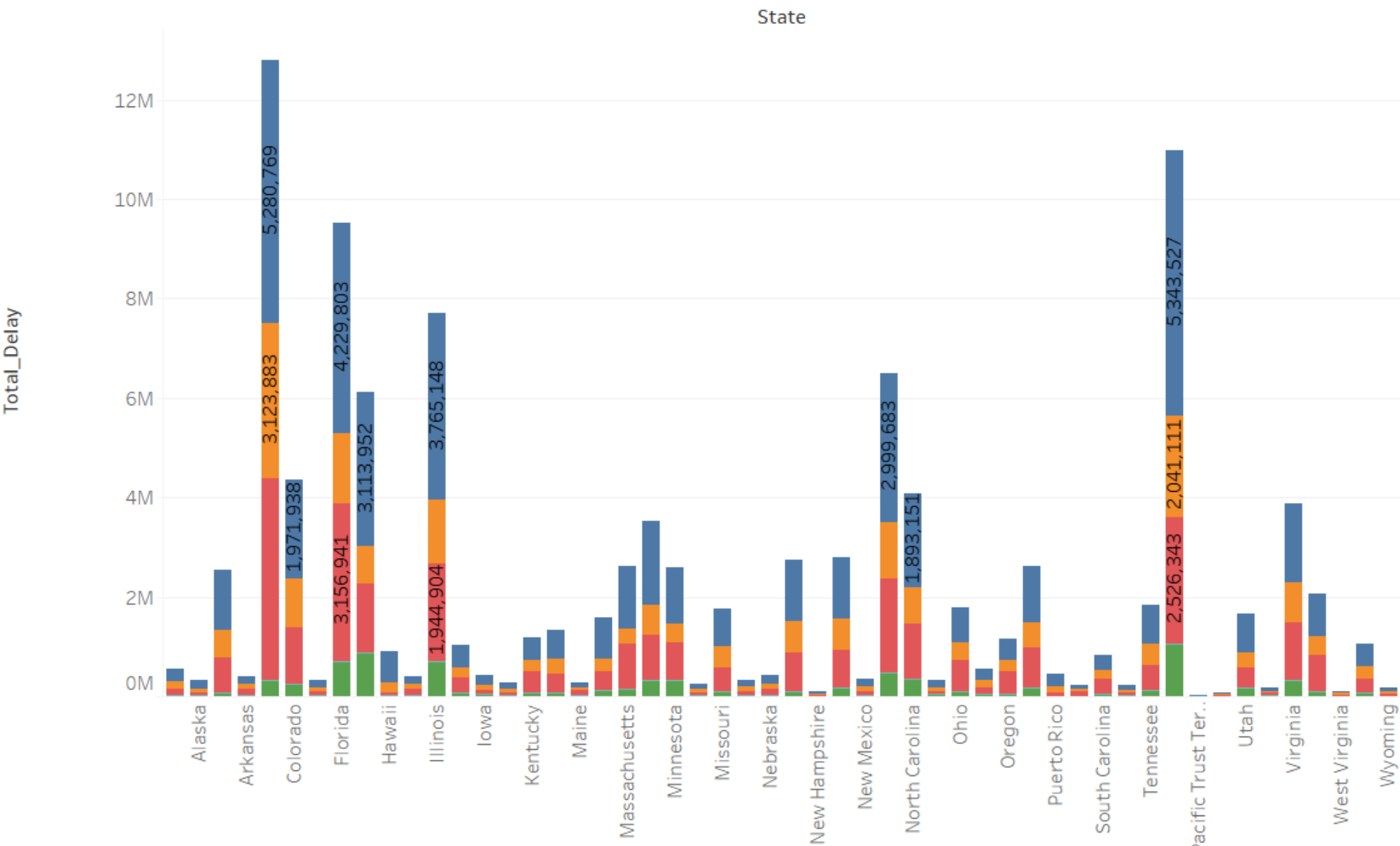
Detail

Tooltip

Delay Reason

SUM(Total_Del..

Statewise Delay Total



Delay Reason

- CARRIER_DELAY
- LATE_AIRCRAFT_DEL..
- NAS_DELAY
- SECURITY_DELAY
- WEATHER_DELAY

ANALYSIS OUTPUTS

PutMerge:

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

FlightDelay OMerge Merge

Project

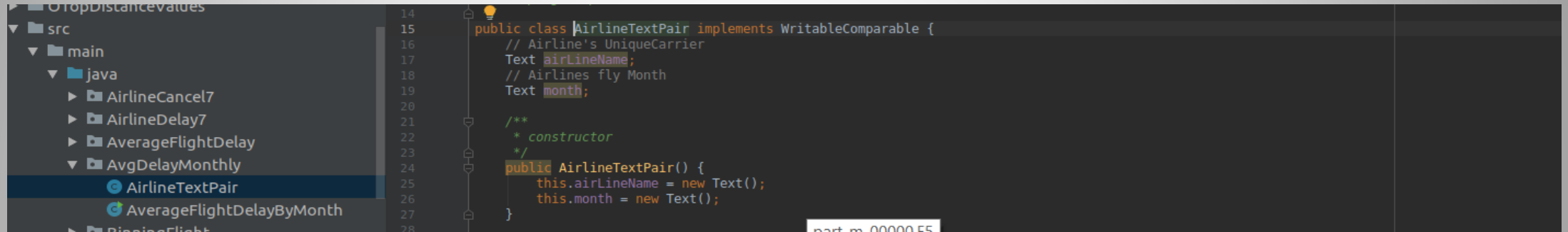
- OAirlineCancel
- OAirlineDelay
- OAvgDelayMonthlyW
- OAvgFlightDelay
- OCancellationCount
- OCounters
- ODelayCount
- ODelayFMedian
- ODistinct
- OFlightCancellation
- OInvertedIndex
- OInvertedIndex_1
- OJoins
- OMaxMin
- OMerge
 - .Merge.crc
 - Merge
- OMSDMedian
- OSecSort

The file is too large: 121.25 MB. Showing a read-only preview of the first 2.56 MB. [Hide notification](#) [Don't show again](#)

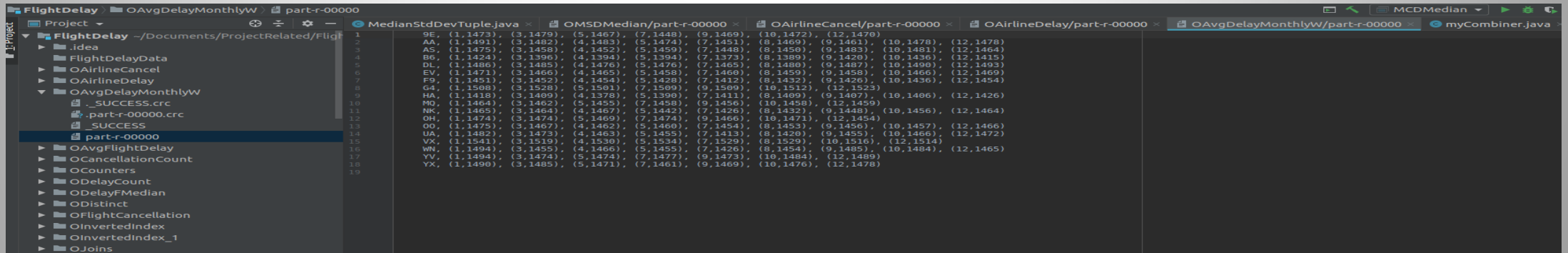
1	YEAR,QUARTER,MONTH,DAY_OF_MONTH,DAY_OF_WEEK,FL_DATE,OP_UNIQUE_CARRIER,TAIL_NUM,OP_CARRIER_FL_NUM,ORIGIN_AIRPORT_ID,ORIGIN,ORIGIN_CITY_NAME,ORIGIN_STATE_ABR,ORIGIN_STATE_NM,DEST_AIRPORT_ID,DEST,
2	2018,1,3,1,4,2018-03-01,F9,N201FR,1680,12889,LAS,"Las Vegas, NV",NV,Nevada,13204,MCO,"Orlando, FL",FL,Florida,30,221,111,111,18,1011,8,756,1019,143,143,0,,0,266,298,272,1,2039,9,20,0,32,0,91
3	2018,1,3,1,4,2018-03-01,F9,N201FR,681,13204,MCO,"Orlando, FL",FL,Florida,11292,DEN,"Denver, CO",CO,Colorado,900,1108,128,128,12,1302,8,1115,1310,115,115,0,,0,255,242,222,1,1546,7,4,0,0,0,111
4	2018,1,3,1,4,2018-03-01,F9,N201FR,681,11292,DEN,"Denver, CO",CO,Colorado,14747,SEA,"Seattle, WA",WA,Washington,1215,1358,103,103,13,1534,6,1413,1540,87,87,0,,0,178,162,143,1,1024,5,0,0,0,0,87
5	2018,1,3,1,4,2018-03-01,F9,N201FR,144,14747,SEA,"Seattle, WA",WA,Washington,11292,DEN,"Denver, CO",CO,Colorado,1514,1621,67,67,11,1940,8,1855,1948,53,53,0,,0,161,147,128,1,1024,5,0,0,0,0,53
6	2018,1,3,1,4,2018-03-01,F9,N201FR,122,11292,DEN,"Denver, CO",CO,Colorado,11298,DFW,"Dallas/Fort Worth, TX",TX,Texas,1955,2035,40,40,9,2307,10,2252,2317,25,25,0,,0,117,102,83,1,641,3,0,0,0,0,25
7	2018,1,3,1,4,2018-03-01,F9,N202FR,1138,12889,LAS,"Las Vegas, NV",NV,Nevada,10423,AUS,"Austin, TX",TX,Texas,845,836,-9,0,14,1306,10,1331,1316,-15,0,0,,0,166,160,136,1,1090,5,,,,,
8	2018,1,3,1,4,2018-03-01,F9,N202FR,1141,10423,AUS,"Austin, TX",TX,Texas,12889,LAS,"Las Vegas, NV",NV,Nevada,1421,1416,-5,0,12,1505,6,1525,1511,-14,0,0,,0,184,175,157,1,1090,5,,,,,
9	2018,1,3,1,4,2018-03-01,F9,N202FR,1106,12889,LAS,"Las Vegas, NV",NV,Nevada,12266,IAH,"Houston, TX",TX,Texas,1615,1611,-4,0,31,2108,7,2117,2115,-2,0,0,,0,182,184,146,1,1222,5,,,,,
10	2018,1,3,1,4,2018-03-01,F9,N202FR,1111,12266,IAH,"Houston, TX",TX,Texas,12889,LAS,"Las Vegas, NV",NV,Nevada,2207,2200,-7,0,20,2314,7,2342,2321,-21,0,0,,0,215,201,174,1,1222,5,,,,,
11	2018,1,3,1,4,2018-03-01,F9,N203FR,1236,13487,MSP,"Minneapolis, MN",MN,Minnesota,15304,TPA,"Tampa, FL",FL,Florida,704,700,-4,0,50,1132,8,1120,1140,20,20,0,,0,196,220,162,1,1306,6,0,0,20,0,0
12	2018,1,3,1,4,2018-03-01,F9,N203FR,1201,15304,TPA,"Tampa, FL",FL,Florida,12339,IND,"Indianapolis, IN",IN,Indiana,1210,1225,15,15,13,1431,10,1430,1441,11,11,0,,0,140,136,113,1,837,4,,,,,
13	2018,1,3,1,4,2018-03-01,F9,N203FR,1204,12339,IND,"Indianapolis, IN",IN,Indiana,15304,TPA,"Tampa, FL",FL,Florida,1520,1622,62,62,16,1831,6,1733,1837,64,64,0,,0,133,135,113,1,837,4,56,0,8,0,0
14	2018,1,3,1,4,2018-03-01,F9,N203FR,1209,15304,TPA,"Tampa, FL",FL,Florida,13342,MKE,"Milwaukee, WI",WI,Wisconsin,1825,1939,74,74,17,2125,5,2017,2130,73,73,0,,0,172,171,149,1,1075,5,16,0,0,0,57
15	2018,1,3,1,4,2018-03-01,F9,N203FR,1099,13342,MKE,"Milwaukee, WI",WI,Wisconsin,12889,LAS,"Las Vegas, NV",NV,Nevada,2107,2214,67,67,19,6,7,2259,13,74,74,0,,0,232,239,213,1,1524,7,0,0,7,0,67
16	2018,1,3,1,4,2018-03-01,F9,N205FR,1977,13204,MCO,"Orlando, FL",FL,Florida,12339,IND,"Indianapolis, IN",IN,Indiana,935,935,0,0,13,1149,7,1207,1156,-11,0,0,,0,152,141,121,1,829,4,,,,,
17	2018,1,3,1,4,2018-03-01,F9,N205FR,1977,12339,IND,"Indianapolis, IN",IN,Indiana,12889,LAS,"Las Vegas, NV",NV,Nevada,1259,1306,7,7,34,1424,9,1414,1433,19,19,0,,0,255,267,224,1,1590,7,7,0,12,0,0
18	"YEAR","QUARTER","MONTH","DAY_OF_MONTH","DAY_OF_WEEK","FL_DATE","OP_UNIQUE_CARRIER","TAIL_NUM","OP_CARRIER_FL_NUM","ORIGIN_AIRPORT_ID","ORIGIN","ORIGIN_CITY_NAME","ORIGIN_STATE_ABR","ORIGIN_STA
19	2017,2,4,1,6,2017-04-01,"DL","N3755D","2","12478","JFK","New York, NY","NY","New York",14679,"SAN","San Diego, CA","CA","California","1925","1925",0.00,0.00,36.00,"2216",4.00,"2240","2220",-20.00
20	2017,2,4,1,6,2017-04-01,"DL","N3741S","4","12889,"LAS","Las Vegas, NV","NV","Nevada",12478,"JFK","New York, NY","NY","New York",1024,"1129",65.00,65.00,13.00,"1912",8.00,"1829","1920",51.00,51
21	2017,2,4,1,6,2017-04-01,"DL","N3743H","7","12889,"LAS","Las Vegas, NV","NV","Nevada",12892,"LAX","Los Angeles, CA","CA","California","0802","0801",-1.00,0.00,12.00,"0852",10.00,"0919","0902",-17
22	2017,2,4,1,6,2017-04-01,"DL","N910DN","11",13487,"MSP","Minneapolis, MN","MN","Minnesota",11292,"DEN","Denver, CO","CO","Colorado",1800,"1800",0.00,0.00,13.00,"1855",9.00,"1920","1904",-16.00
23	2017,2,4,1,6,2017-04-01,"DL","N658DL","15",10397,"ATL","Atlanta, GA","GA","Georgia",14683,"SAT","San Antonio, TX","TX","Texas",1835,"1832",-3.00,0.00,16.00,"1953",6.00,"2011","1959",-12.00,0.
24	2017,2,4,1,6,2017-04-01,"DL","N591NW","16",13204,"MCO","Orlando, FL","FL","Florida",11433,"DTW","Detroit, MI","MI","Michigan",1400,"1358",-2.00,0.00,15.00,"1617",7.00,"1642","1624",-18.00,0.6
25	2017,2,4,1,6,2017-04-01,"DL","N592NW","17",11433,"DTW","Detroit, MI","MI","Michigan",13204,"MCO","Orlando, FL","FL","Florida",1517,"1522",5.00,5.00,12.00,"1741",11.00,"1753","1752",-1.00,0.00
26	2017,2,4,1,6,2017-04-01,"DL","N948AT","18",11433,"DTW","Detroit, MI","MI","Michigan",13108,"MCT","Kansas City, MO","MO","Missouri",2018,"2000",0.00,0.00,15.00,"2050",5.00,"2122","2104",-18.6

AIRLINE DELAY:

Calculated Average Delay by the airliner each month used Custom Writable class for this technique.



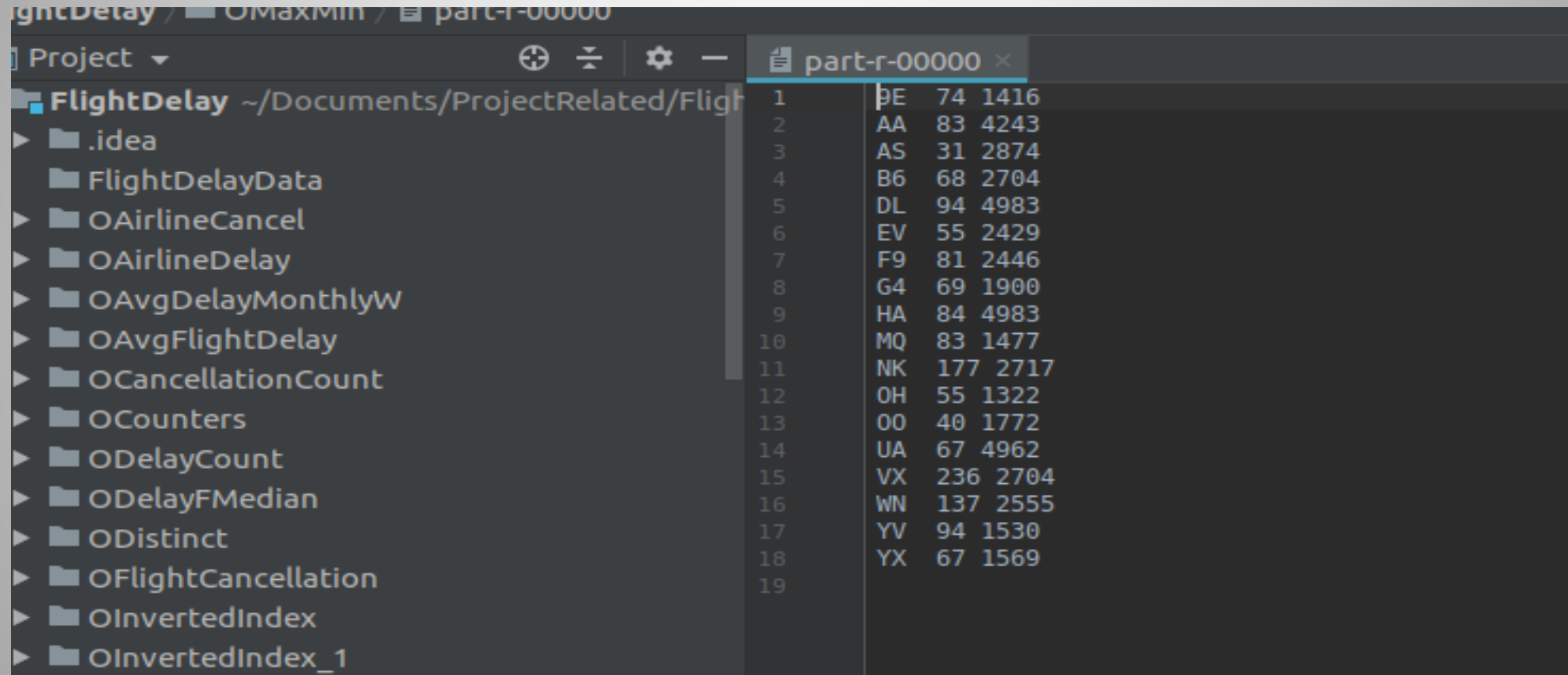
```
14
15 public class AirlineTextPair implements WritableComparable {
16     // Airline's UniqueCarrier
17     Text airlineName;
18     // Airlines fly Month
19     Text month;
20
21     /**
22     * constructor
23     */
24     public AirlineTextPair() {
25         this.airlineName = new Text();
26         this.month = new Text();
27     }
28
```



Key	Value
9E	(1, 1473), (3, 1479), (5, 1467), (7, 1448), (9, 1469), (10, 1472), (12, 1470)
AA	(1, 1491), (3, 1482), (4, 1483), (5, 1474), (7, 1451), (8, 1469), (9, 1461), (10, 1478), (12, 1478)
AS	(1, 1475), (3, 1458), (4, 1452), (5, 1459), (7, 1448), (8, 1450), (9, 1483), (10, 1481), (12, 1464)
B6	(1, 1424), (3, 1396), (4, 1394), (5, 1394), (7, 1373), (8, 1380), (9, 1420), (10, 1436), (12, 1415)
DL	(1, 1486), (3, 1485), (4, 1476), (5, 1476), (7, 1465), (8, 1480), (9, 1487), (10, 1490), (12, 1493)
EV	(1, 1471), (3, 1466), (4, 1465), (5, 1458), (7, 1460), (8, 1459), (9, 1458), (10, 1466), (12, 1469)
F9	(1, 1451), (3, 1452), (4, 1454), (5, 1428), (7, 1412), (8, 1432), (9, 1426), (10, 1436), (12, 1454)
G4	(1, 1508), (3, 1528), (5, 1501), (7, 1509), (9, 1509), (10, 1512), (12, 1523)
HA	(1, 1418), (3, 1469), (4, 1378), (5, 1390), (7, 1411), (8, 1409), (9, 1407), (10, 1406), (12, 1426)
MO	(1, 1464), (3, 1462), (5, 1455), (7, 1458), (9, 1450), (10, 1458), (12, 1459)
NK	(1, 1465), (3, 1464), (4, 1467), (5, 1442), (7, 1426), (8, 1432), (9, 1448), (10, 1456), (12, 1464)
OH	(1, 1474), (3, 1474), (5, 1469), (7, 1474), (9, 1466), (10, 1471), (12, 1454)
OO	(1, 1475), (3, 1467), (4, 1462), (5, 1460), (7, 1454), (8, 1453), (9, 1456), (10, 1457), (12, 1466)
UA	(1, 1482), (3, 1473), (4, 1463), (5, 1455), (7, 1413), (8, 1420), (9, 1455), (10, 1466), (12, 1472)
VX	(1, 1541), (3, 1519), (4, 1530), (5, 1534), (7, 1529), (8, 1529), (10, 1516), (12, 1514)
WN	(1, 1494), (3, 1455), (4, 1466), (5, 1455), (7, 1426), (8, 1454), (9, 1485), (10, 1484), (12, 1465)
YV	(1, 1494), (3, 1474), (5, 1474), (7, 1477), (9, 1473), (10, 1484), (12, 1489)
YX	(1, 1490), (3, 1485), (5, 1471), (7, 1461), (9, 1469), (10, 1476), (12, 1478)

MIN-MAX DELAY:

Determined the Minimum and Maximum Distance travelled by flight based on each Airliner.



The screenshot shows an IDE interface. On the left is a project explorer for 'FlightDelay' with various subfolders. On the right is a data table titled 'part-r-00000' with 19 rows of data.

1	BE	74	1416	
2	AA	83	4243	
3	AS	31	2874	
4	B6	68	2704	
5	DL	94	4983	
6	EV	55	2429	
7	F9	81	2446	
8	G4	69	1900	
9	HA	84	4983	
10	MQ	83	1477	
11	NK	177	2717	
12	OH	55	1322	
13	OO	40	1772	
14	UA	67	4962	
15	VX	236	2704	
16	WN	137	2555	
17	YV	94	1530	
18	YX	67	1569	
19				

MEDIAN

Determined Delay Median based on Airliner

```
protected void reduce(Text key, Iterable<Text> values, Context context) throws IOException, InterruptedException {
    Integer median=0;
    //int sum = 0;
    //int count = 0;

    //Integer temp= (int) Float.parseFloat(key.toString());
    try {
        for (Text value:values) {
            Integer temp = (int) Float.parseFloat(value.toString());
            if (temp != 0){
                array.add(temp);
            }
            //context.write(new Text(key), NullWritable.get());
        }
        Collections.sort(array);
        int len=array.size();
        if(array.size()%2==0){
            // median= array.get((int) len/2 - 1)+array.get((int) len/2);
            median= (array.get((int) len/2 - 1)+array.get((int) len/2)) /2;
        }
        else{
            median= array.get(len/2);
        }

        context.write(key,new Text(key.toString()+median+""));
    }
}
```

The screenshot shows an IDE window with the following components:

- Project Explorer:** Displays the project structure for 'FlightDelay'. The 'src/main/java' directory is expanded, showing subdirectories like 'AirlineCancel7', 'AirlineDelay7', 'AverageFlightDelay', 'AvgDelayMonthly', 'BinningFlight', and 'CancelFlightCancel7'. The 'part-r-00000' directory is also visible.
- Run Configuration:** Shows 'App' as the selected configuration.
- Output Console:** Displays a list of airline delay medians and standard deviations. The data is as follows:

Airline	Median	Std	Div
9E	422.0	11.182264	
AA	918.5	10.709496	
AS	972.5	19.131805	
B6	992.0	13.007162	
DL	679.5	12.000799	
EV	435.0	8.557014	
F9	978.5	18.91692	
G4	895.0	20.08316	
HA	142.0	24.758072	
MQ	372.0	8.99425	
NK	975.0	13.585728	
OH	356.0	6.303177	
OO	398.0	9.116984	
UA	991.0	14.888375	
VX	1476.0	19.074	
WN	632.0	8.721409	
YV	485.5	9.903636	
YX	542.0	12.116253	

MEMORY CONSCIOUS MEDIAN

Used Memory-Conscious Median and Standard Deviation of Distance travelled based on Airliner used separate Combiner to simplify the input to Reducer.

```
3 import ...
4
5 public class myCombiner extends Reducer<Text, SortedMapWritable, Text, SortedMapWritable> {
6     @Override
7     protected void reduce(Text key, Iterable<SortedMapWritable> values, Context context) throws IOException, InterruptedException {
8         // int sum=0;
9         SortedMapWritable map = new SortedMapWritable();
10         for(SortedMapWritable v:values) {
11             for(Object entry : v.entrySet() ) {
12                 if(entry instanceof Map.Entry) {
13                     Map.Entry entry1 = (Map.Entry) entry;
14                     IntWritable count = (IntWritable) map.get(entry1.getKey());
15                     FloatWritable mapKey = ((FloatWritable) entry1.getKey());
16                     Integer mapValue = ((IntWritable) entry1.getValue()).get();
17                     if (count != null) {
18                         map.put(mapKey, new IntWritable( value: count.get() + ((IntWritable) entry1.getValue()).get()));
19                     } else {
20                         map.put(mapKey, new IntWritable(((IntWritable) entry1.getValue()).get()));
21                     }
22                 }
23             }
24             v.clear();
25         }
26         context.write(key,map);
27     }
28 }
```

```
DistanceMax/Driver.java × OMaxMin/part-r-00000 × MedianStdDevTuple.java × OMSDMedian/part-r-00000 × myCombiner.java ×
1 9E median:422.0 Std_Div:11.182264
2 AA median:918.5 Std_Div:10.709496
3 AS median:972.5 Std_Div:19.131805
4 B6 median:992.0 Std_Div:13.007162
5 DL median:679.5 Std_Div:12.000799
6 EV median:435.0 Std_Div:8.557014
7 F9 median:979.5 Std_Div:18.91692
8 G4 median:895.0 Std_Div:20.08316
9 HA median:142.0 Std_Div:24.758072
10 MQ median:372.0 Std_Div:8.99425
11 NK median:975.0 Std_Div:13.585728
12 OH median:356.0 Std_Div:6.303177
13 OO median:398.0 Std_Div:9.116984
14 UA median:991.0 Std_Div:14.888375
15 VX median:1476.0 Std_Div:19.074
16 WN median:632.0 Std_Div:8.721409
17 YV median:485.5 Std_Div:9.903636
18 YX median:542.0 Std_Div:12.116253
19
```


COUNTING WITH COUNTERS

Used counting with Counters to determine Number of Flights in different Distance Groups

The screenshot displays an IDE with a project structure on the left, a code editor in the center, and a run console at the bottom.

Project Structure (Left):

- .idea
- FlightDelayData
- OAirlineCancel
- OAirlineDelay
- OAvgDelayMonthlyW
- OAvgFlightDelay
- OCancellationCount
- OCounters
- OFlightCancellation
- src
 - main
 - java
 - AirlineCancel7
 - AirlineDelay7
 - AverageFlightDelay
 - AvgDelayMonthly
 - BinningFlight
 - CancellationCount2
 - CountersDistanceGroup

Code Editor (Center):

```
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
  
job.setMapperClass(MyMapper.class);  
  
job.setOutputKeyClass(NullWritable.class);  
job.setOutputValueClass(NullWritable.class);  
  
int code = job.waitForCompletion(verbose: true)? 1 : 0;  
if (code==1) {  
    System.out.print("the counter output:\n");  
    for(Counter counter: job.getCounters().getGroup(groupName: "DistanceCounter")) {  
        //  
        System.out.println(counter.getDisplayName()+"\t"+counter.getValue());  
        // System.out.println("the counter values is "+counter.getValue());  
    }  
}
```

Run Console (Bottom):

Run: CountingCounters x

File Output Format Counters
Bytes Written=8

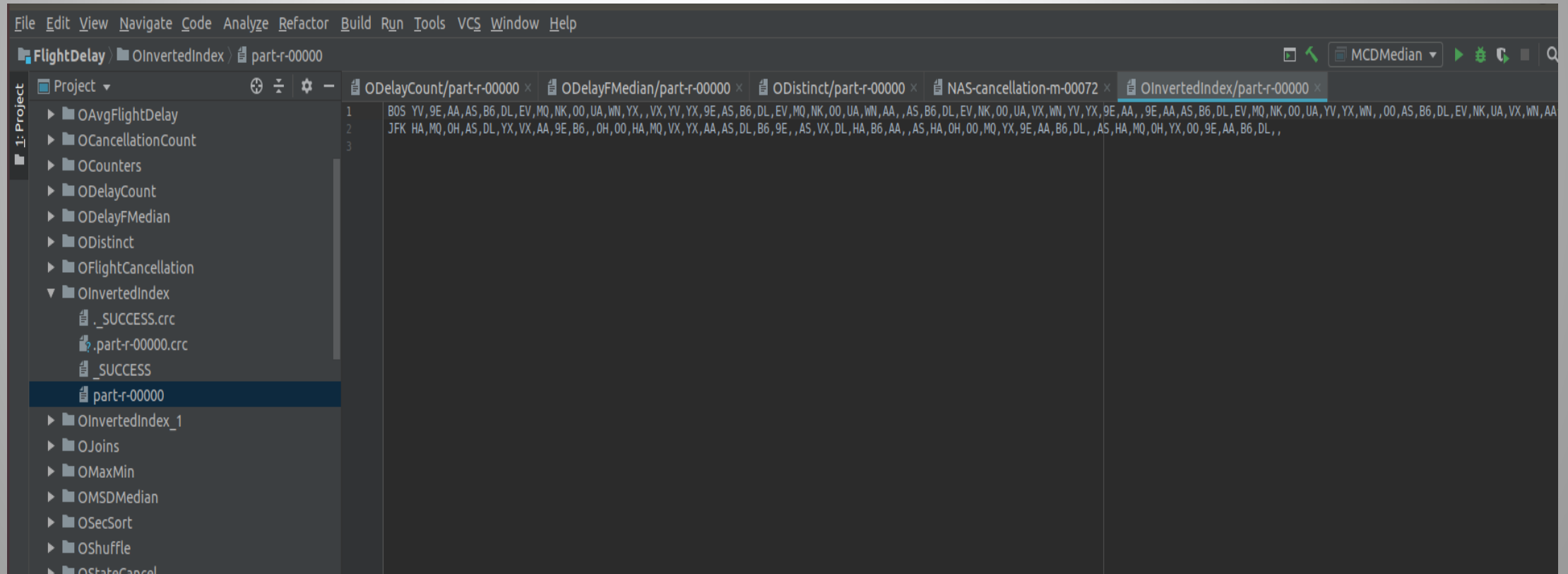
the counter output:

```
1: 1731883  
10: 385284  
11: 282962  
2: 3285052  
3: 2669652  
4: 2092525  
5: 1431105  
6: 604557  
7: 631685  
8: 325164  
9: 227472
```

Process finished with exit code 0

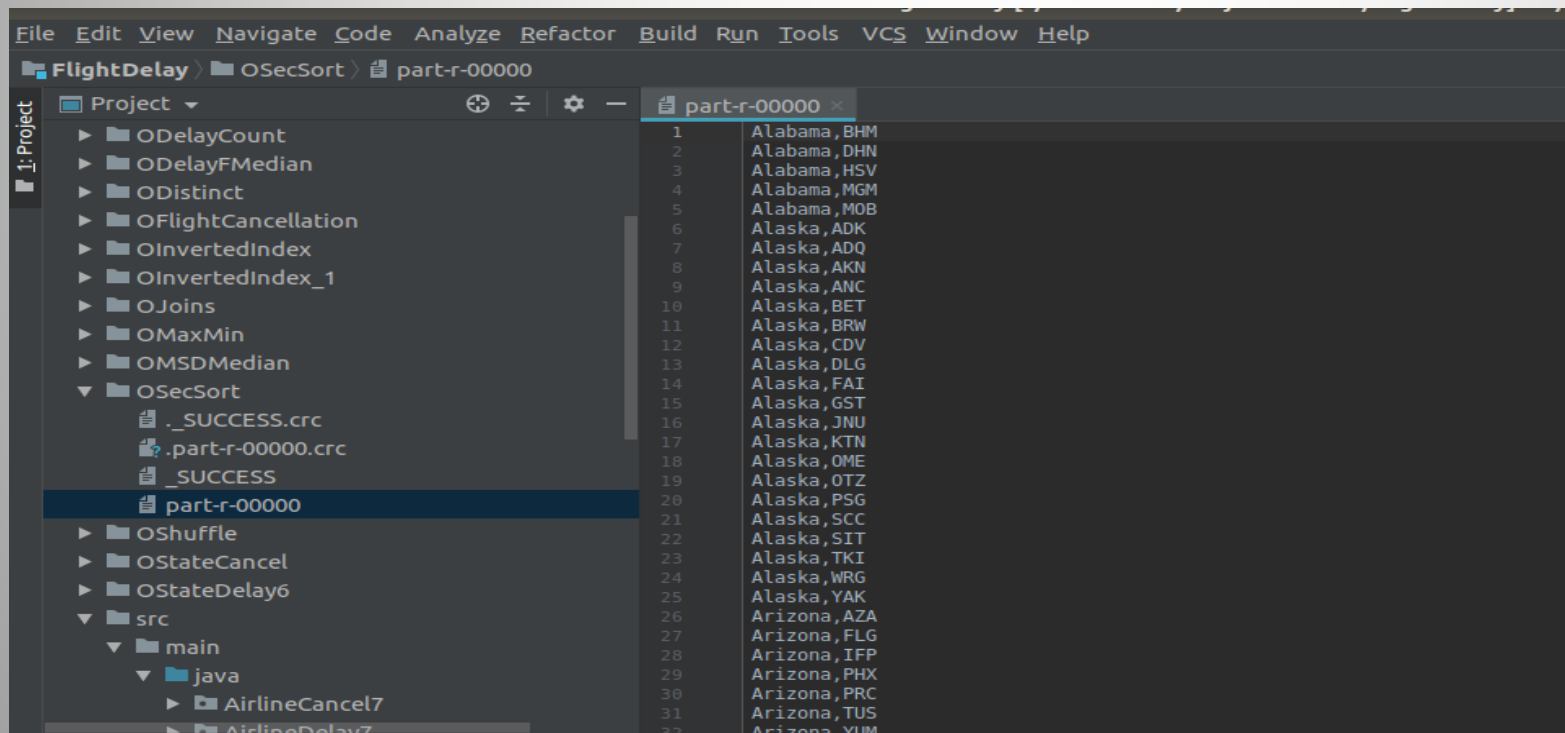
INVERTED INDEX:

Used Filtered output and determined the Flights from the Airport using Inverted Index Method.



SECONDARY SORTING:

Did Secondary Sorting of the States and the available Airports in that state to know the list of Airports Present in US.(Partitioner is used in this case)



The screenshot shows an IDE window with the following components:

- Menu Bar:** File, Edit, View, Navigate, Code, Analyze, Refactor, Build, Run, Tools, VCS, Window, Help.
- Breadcrumb:** FlightDelay > OSecSort > part-r-00000
- Project Explorer (Left):**
 - Project
 - ODelayCount
 - ODelayFMedian
 - ODistinct
 - OFlightCancellation
 - OInvertedIndex
 - OInvertedIndex_1
 - OJoins
 - OMaxMin
 - OMSDMedian
 - OSecSort
 - ._SUCCESS.crc
 - .part-r-00000.crc
 - ._SUCCESS
 - part-r-00000
 - OShuffle
 - OStateCancel
 - OStateDelay6
 - src
 - main
 - java
 - AirlineCancel7
 - AirlineDelay7

- Editor (Right):** A file named 'part-r-00000' is open, displaying a list of airports sorted by state. The list is as follows:

Line Number	State, Airport
1	Alabama, BHM
2	Alabama, DHN
3	Alabama, HSV
4	Alabama, MGM
5	Alabama, MOB
6	Alaska, ADK
7	Alaska, ADQ
8	Alaska, AKN
9	Alaska, ANC
10	Alaska, BET
11	Alaska, BRW
12	Alaska, CDV
13	Alaska, DLG
14	Alaska, FAI
15	Alaska, GST
16	Alaska, JNU
17	Alaska, KTN
18	Alaska, OME
19	Alaska, OTZ
20	Alaska, PSG
21	Alaska, SCC
22	Alaska, SIT
23	Alaska, TKI
24	Alaska, WRG
25	Alaska, YAK
26	Arizona, AZA
27	Arizona, FLG
28	Arizona, IFP
29	Arizona, PHX
30	Arizona, PRC
31	Arizona, TUS
32	Arizona, YUM

JOINS:

Based on the Carrier Names from a different Carrier CSV joined the Full name of the Flights present in the original Data.

Code	Description
02Q	Titan Airways
04Q	Tradewind Aviation
05Q	Comlux Aviation, AG
06Q	Master Top Linhas Aereas Ltd.
07Q	Flair Airlines Ltd.
09Q	Swift Air, LLC
0BQ	DCA
0CQ	ACM AIR CHARTER GmbH
0FQ	Maine Aviation Aircraft Charter, LLC
0GQ	Inter Island Airways, d/b/a Inter Island Air
0HQ	Polar Airlines de Mexico d/b/a Nova Air
0J	JetClub AG
0JQ	Vision Airlines
0KQ	Mokulele Flight Services, Inc.
0LQ	Metropix UK, LLP
0MQ	Multi-Aero, Inc. d/b/a Air Choice One
0Q	Flying Service N.V.
16	PSA Airlines Inc.
17	Piedmont Airlines
1I	Sky Trek Int'l Airlines
2E	Smokey Bay Air Inc.
2F	Frontier Flying Service
2M	Midway Express Airlines
2O	Island Air Service
2R	Regal Air
2T	Canada 3000 Airlines Ltd.
2U	Valley Air Express Inc.
37	Zeal 320
3C	Regions Air, Inc.
3F	Pacific Airways, Inc.
3M	Gulfstream Int
3Z	Tatonduk Flying Service
4B	Olson Air Service
4E	Tanana Air Service

FlightDelay > OJoins > part-r-00000

Project

OAvgFlightDelay

OCancellationCount

OCounters

ODelayCount

ODelayFMedian

ODistinct

OFlightCancellation

OInvertedIndex

OInvertedIndex_1

OJoins

part-r-00000

OMaxMin

OMSDMedian

OSecSort

OShuffle

OStateCancel

OStateDelay6

src

ODelayCount/part-r-00000

ODelayFMedian/p

1 9E Pinnacle Airlines Inc.

2 AA American Airlines Inc.

3 AS Alaska Airlines Inc.

4 B6 JetBlue Airways

5 DL Delta Air Lines Inc.

6 EV Atlantic Southeast Airlines

7 F9 Frontier Airlines Inc.

8 G4 Allegiant Air

9 HA Hawaiian Airlines Inc.

10 MQ American Eagle Airlines Inc.

11 NK Spirit Air Lines

12 OH Comair Inc.

13 OO Skywest Airlines Inc.

14 UA United Air Lines Inc.

15 VX Virgin America

16 WN Southwest Airlines Co.

17 YV Mesa Airlines Inc.

18 YX Midwest Airline

19

BLOOM FILTER:

Used Bloom Filter to Filter the data based on the Airport and used Chaining

```
private BloomFilter<String> origin;

Funnel<String> p = (from, into) -> {
    into.putString(from, Charsets.UTF_8);
};

@Override
protected void setup(Mapper<LongWritable, Text, Text, NullWritable>.Context context) throws IOException, InterruptedException {
    this.origin = BloomFilter.create(p, expectedInsertions: 300000, falsePositiveProbability: 0.1);

    String p1 = "BOS";
    String p2 = "JFK";
    // Person p2 = new Person("Jamie", "Scott");

    ArrayList<String> originList = new ArrayList<>();
    originList.add(p1);
    originList.add(p2);

    for (String ps : originList) {
        origin.put(ps);
    }
}
```

```
File Edit View Navigate Code Analyze Refactor Build Run Tools VSJ Windows Help
FlightDelay | Olmvertedindex_1 | part=00000 [MCDMedian] Q G C Q
Project ODayCount/part=00000 ODelayMedian/part=00000 ODistinct/part=00000 NAScancellation-m00072 Onvertedindex_1/part=00000
OlavFlightDelay
OCancellationCount
OCounters
ODelayCount
ODelayMedian
ODistinct
OFlightCancellation
OInvertedIndex
OInvertedIndex_1 SUCCESS.crc
part=00000.crc
SUCCESS
part=00000
SUCCESS
Joins
OMaxMin
OMSMedian
OSort
OShuffle
OSateCancel
OSateDelay
src
```

```

job.setOutputFormatClass(TextOutputFormat.class);
job.setMapOutputValueClass(NullWritable.class);
job.setMapOutputKeyClass(Text.class);
job.setInputFormatClass(TextInputFormat.class);

// job.setNumReduceTasks(1);

Boolean a=job.waitForCompletion(verbose: true);

if(a==true)
{
    Job job2 = Job.getInstance();

    job2.setJarByClass(Driver.class);

    // job2.setGroupingComparatorClass(GroupComparator.class);
    // job2.setSortComparatorClass(SecondarySortComparator.class);
    //job2.setPartitionerClass(KeyPartition.class);

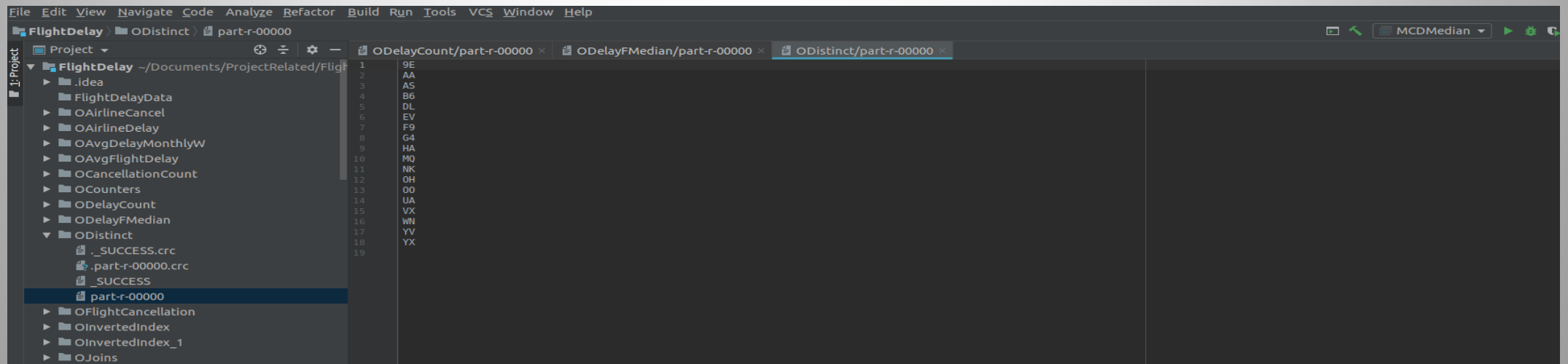
    FileInputFormat.addInputPath(job2, new Path(pathString: args[1]+"_1"));
    Path outDir1 = new Path(args[1]);
}

```


DISTINCT:

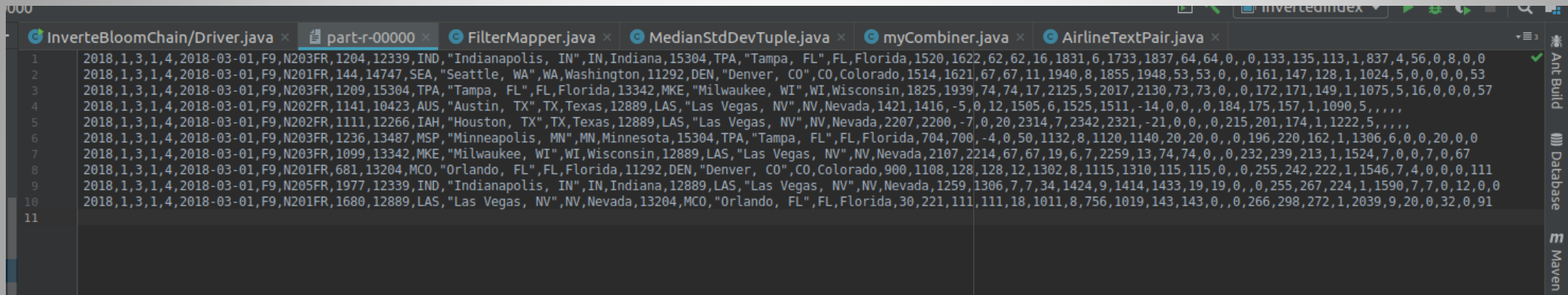
Determined Distinct Airline in the Dataset. (Used NLineInputFormat)

```
FileInputFormat.addInputPath(job2, new Path(args[0]));  
Path outDir1 = new Path(args[1]);  
FileOutputFormat.setOutputPath(job2, outDir1);  
  
job2.setMapperClass(DistinctMapper.class);  
job2.setReducerClass(DistinctReducer.class);  
job2.setCombinerClass(DistinctReducer.class);  
  
job2.setOutputFormatClass(TextOutputFormat.class);  
job2.setMapOutputValueClass(NullWritable.class);  
job2.setMapOutputKeyClass(Text.class);  
job2.setInputFormatClass(NLineInputFormat.class);  
NLineInputFormat.setNumLinesPerSplit(job2, numLines: 10000);  
job2.setNumReduceTasks(1);
```



TOP K:

Determined Top 10 flight data based on longest Distance.

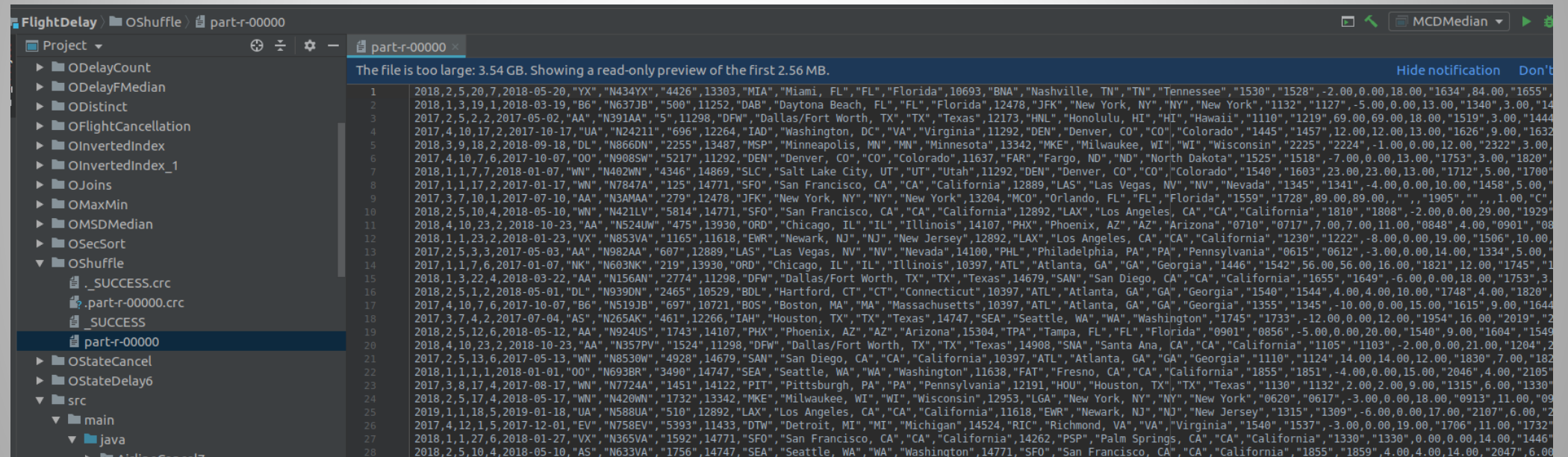


The screenshot shows an IDE with several open files: InverteBloomChain/Driver.java, part-r-00000, FilterMapper.java, MedianStdDevTuple.java, myCombiner.java, and AirlineTextPair.java. The 'part-r-00000' file is active and displays a list of 11 flight records. Each record is a long string of numbers and text, representing flight data. The records are sorted by distance, with the longest distance record at the top. The IDE interface includes a sidebar on the right with icons for 'Ant Build', 'Database', and 'Maven'.

```
1 2018,1,3,1,4,2018-03-01,F9,N203FR,1204,12339,IND,"Indianapolis, IN",IN,Indiana,15304,TPA,"Tampa, FL",FL,Florida,1520,1622,62,62,16,1831,6,1733,1837,64,64,0,,0,133,135,113,1,837,4,56,0,8,0,0
2 2018,1,3,1,4,2018-03-01,F9,N201FR,144,14747,SEA,"Seattle, WA",WA,Washington,11292,DEN,"Denver, CO",CO,Colorado,1514,1621,67,67,11,1940,8,1855,1948,53,53,0,,0,161,147,128,1,1024,5,0,0,0,0,53
3 2018,1,3,1,4,2018-03-01,F9,N203FR,1209,15304,TPA,"Tampa, FL",FL,Florida,13342,MKE,"Milwaukee, WI",WI,Wisconsin,1825,1939,74,74,17,2125,5,2017,2130,73,73,0,,0,172,171,149,1,1075,5,16,0,0,0,57
4 2018,1,3,1,4,2018-03-01,F9,N202FR,1141,10423,AUS,"Austin, TX",TX,Texas,12889,LAS,"Las Vegas, NV",NV,Nevada,1421,1416,-5,0,12,1505,6,1525,1511,-14,0,0,,0,184,175,157,1,1090,5,,,,,
5 2018,1,3,1,4,2018-03-01,F9,N202FR,1111,12266,IAH,"Houston, TX",TX,Texas,12889,LAS,"Las Vegas, NV",NV,Nevada,2207,2200,-7,0,20,2314,7,2342,2321,-21,0,0,,0,215,201,174,1,1222,5,,,,,
6 2018,1,3,1,4,2018-03-01,F9,N203FR,1236,13487,MSP,"Minneapolis, MN",MN,Minnesota,15304,TPA,"Tampa, FL",FL,Florida,704,700,-4,0,50,1132,8,1120,1140,20,20,0,,0,196,220,162,1,1306,6,0,0,20,0,0
7 2018,1,3,1,4,2018-03-01,F9,N203FR,1099,13342,MKE,"Milwaukee, WI",WI,Wisconsin,12889,LAS,"Las Vegas, NV",NV,Nevada,2107,2214,67,67,19,6,7,2259,13,74,74,0,,0,232,239,213,1,1524,7,0,0,7,0,67
8 2018,1,3,1,4,2018-03-01,F9,N201FR,681,13204,MCO,"Orlando, FL",FL,Florida,11292,DEN,"Denver, CO",CO,Colorado,900,1108,128,128,12,1302,8,1115,1310,115,115,0,,0,255,242,222,1,1546,7,4,0,0,0,111
9 2018,1,3,1,4,2018-03-01,F9,N205FR,1977,12339,IND,"Indianapolis, IN",IN,Indiana,12889,LAS,"Las Vegas, NV",NV,Nevada,1259,1306,7,7,34,1424,9,1414,1433,19,19,0,,0,255,267,224,1,1590,7,7,0,12,0,0
10 2018,1,3,1,4,2018-03-01,F9,N201FR,1680,12889,LAS,"Las Vegas, NV",NV,Nevada,13204,MCO,"Orlando, FL",FL,Florida,30,221,111,111,18,1011,8,756,1019,143,143,0,,0,266,298,272,1,2039,9,20,0,32,0,91
11
```

SHUFFLING:

Used Shuffling Technique to shuffle the Data.

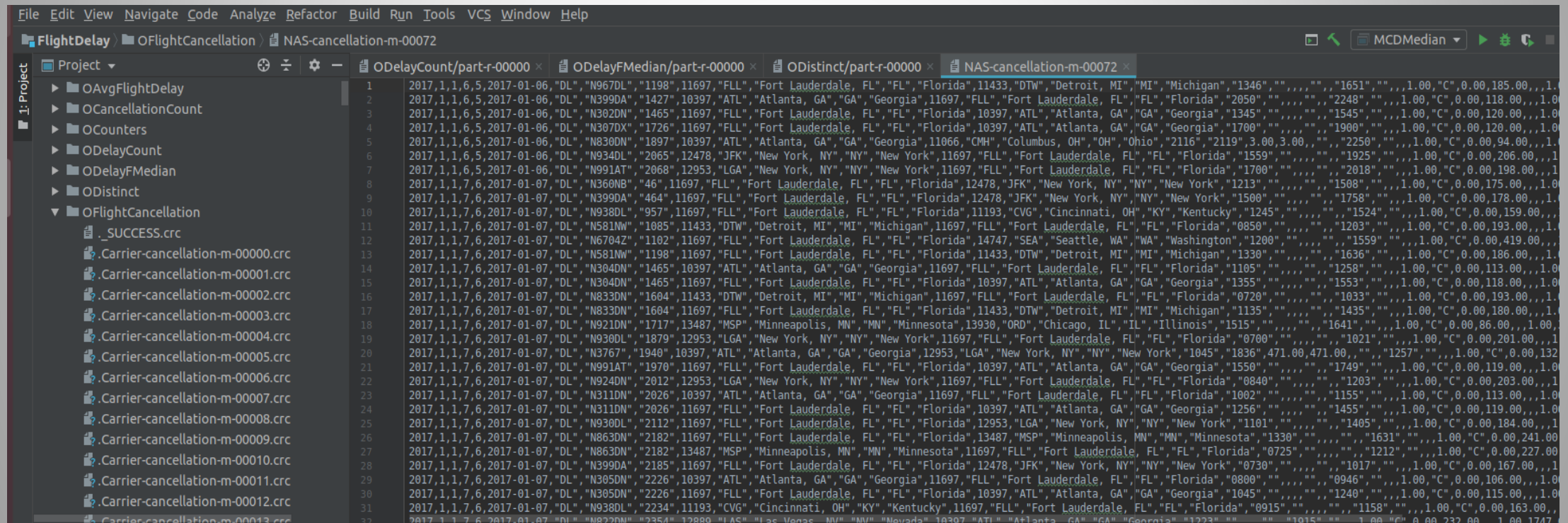


```
FlightDelay \ OShuffle \ part-r-00000
Project
  ODelayCount
  ODelayFMedian
  ODistinct
  OFlightCancellation
  OInvertedIndex
  OInvertedIndex_1
  OJoins
  OMaxMin
  OMSDMedian
  OSecSort
  OShuffle
    _SUCCESS.crc
    .part-r-00000.crc
    _SUCCESS
    part-r-00000
  OStateCancel
  OStateDelay6
  src
    main
      java
        ...

part-r-00000
The file is too large: 3.54 GB. Showing a read-only preview of the first 2.56 MB.
1 2018,2,5,20,7,2018-05-20,"YX","N434YX","4426",13303,"MIA","Miami, FL","FL","Florida",10693,"BNA","Nashville, TN","TN","Tennessee",1530,"1528",-2.00,0.00,18.00,"1634",84.00,"1655",
2 2018,1,3,19,1,2018-03-19,"B6","N637JB","500",11252,"DAB","Daytona Beach, FL","FL","Florida",12478,"JFK","New York, NY","NY","New York",1132,"1127",-5.00,0.00,13.00,"1340",3.00,"14
3 2017,2,5,2,2,2017-05-02,"AA","N391AA","5",11298,"DFW","Dallas/Fort Worth, TX","TX","Texas",12173,"HNL","Honolulu, HI","HI","Hawaii",1110,"1219",69.00,69.00,18.00,"1519",3.00,"1444
4 2017,4,10,17,2,2017-10-17,"UA","N24211","696",12264,"IAD","Washington, DC","VA","Virginia",11292,"DEN","Denver, CO","CO","Colorado",1445,"1457",12.00,12.00,13.00,"1626",9.00,"1632
5 2018,3,9,18,2,2018-09-18,"DL","N866DN","2255",13487,"MSP","Minneapolis, MN","MN","Minnesota",13342,"MKE","Milwaukee, WI","WI","Wisconsin",2225,"2224",-1.00,0.00,12.00,"2322",3.00,
6 2017,4,10,7,6,2017-10-07,"OO","N908SW","5217",11292,"DEN","Denver, CO","CO","Colorado",11637,"FAR","Fargo, ND","ND","North Dakota",1525,"1518",-7.00,0.00,13.00,"1753",3.00,"1820"
7 2018,1,1,7,7,2018-01-07,"WN","N402WN","4346",14869,"SLC","Salt Lake City, UT","UT","Utah",11292,"DEN","Denver, CO","CO","Colorado",1540,"1603",23.00,23.00,13.00,"1712",5.00,"1700"
8 2017,1,1,17,2,2017-01-17,"WN","N7847A","125",14771,"SFO","San Francisco, CA","CA","California",12889,"LAS","Las Vegas, NV","NV","Nevada",1345,"1341",-4.00,0.00,10.00,"1458",5.00,"
9 2017,3,7,10,1,2017-07-10,"AA","N3AMAA","279",12478,"JFK","New York, NY","NY","New York",13204,"MCO","Orlando, FL","FL","Florida",1559,"1728",89.00,89.00,"",,"1905","",,"1.00","C",
10 2018,2,5,10,4,2018-05-10,"WN","N421LV","5814",14771,"SFO","San Francisco, CA","CA","California",12892,"LAX","Los Angeles, CA","CA","California",1810,"1808",-2.00,0.00,29.00,"1929",
11 2018,4,10,23,2,2018-10-23,"AA","N524UW","475",13930,"ORD","Chicago, IL","IL","Illinois",14107,"PHX","Phoenix, AZ","AZ","Arizona",0710,"0717",7.00,7.00,11.00,"0848",4.00,"0901","08
12 2018,1,1,23,2,2018-01-23,"VX","N853VA","1165",11618,"EWR","Newark, NJ","NJ","New Jersey",12892,"LAX","Los Angeles, CA","CA","California",1230,"1222",-8.00,0.00,19.00,"1506",10.00,
13 2017,2,5,3,3,2017-05-03,"AA","N982AA","607",12889,"LAS","Las Vegas, NV","NV","Nevada",14100,"PHL","Philadelphia, PA","PA","Pennsylvania",0615,"0612",-3.00,0.00,14.00,"1334",5.00,"
14 2017,1,1,7,6,2017-01-07,"NK","N603NK","219",13930,"ORD","Chicago, IL","IL","Illinois",10397,"ATL","Atlanta, GA","GA","Georgia",1446,"1542",56.00,56.00,16.00,"1821",12.00,"1745",1
15 2018,1,3,22,4,2018-03-22,"AA","N156AN","2774",11298,"DFW","Dallas/Fort Worth, TX","TX","Texas",14679,"SAN","San Diego, CA","CA","California",1655,"1649",-6.00,0.00,18.00,"1753",3.
16 2018,2,5,1,2,2018-05-01,"DL","N939DN","2465",10529,"BDL","Hartford, CT","CT","Connecticut",10397,"ATL","Atlanta, GA","GA","Georgia",1540,"1544",4.00,4.00,10.00,"1748",4.00,"1820",
17 2017,4,10,7,6,2017-10-07,"B6","N519JB","697",10721,"BOS","Boston, MA","MA","Massachusetts",10397,"ATL","Atlanta, GA","GA","Georgia",1355,"1345",-10.00,0.00,15.00,"1615",9.00,"1644
18 2017,3,7,4,2,2017-07-04,"AS","N265AK","461",12266,"IAH","Houston, TX","TX","Texas",14747,"SEA","Seattle, WA","WA","Washington",1745,"1733",-12.00,0.00,12.00,"1954",16.00,"2019",2
19 2018,2,5,12,6,2018-05-12,"AA","N924US","1743",14107,"PHX","Phoenix, AZ","AZ","Arizona",15304,"TPA","Tampa, FL","FL","Florida",0901,"0856",-5.00,0.00,20.00,"1540",9.00,"1604",1549
20 2018,4,10,23,2,2018-10-23,"AA","N357PV","1524",11298,"DFW","Dallas/Fort Worth, TX","TX","Texas",14908,"SNA","Santa Ana, CA","CA","California",1105,"1103",-2.00,0.00,21.00,"1204",2
21 2017,2,5,13,6,2017-05-13,"WN","N8530W","4928",14679,"SAN","San Diego, CA","CA","California",10397,"ATL","Atlanta, GA","GA","Georgia",1110,"1124",14.00,14.00,12.00,"1830",7.00,"182
22 2018,1,1,1,1,2018-01-01,"OO","N693BR","3490",14747,"SEA","Seattle, WA","WA","Washington",11638,"FAT","Fresno, CA","CA","California",1855,"1851",-4.00,0.00,15.00,"2045",4.00,"2105",
23 2017,3,8,17,4,2017-08-17,"WN","N772AA","1451",14122,"PIT","Pittsburgh, PA","PA","Pennsylvania",12191,"HOU","Houston, TX","TX","Texas",1130,"1132",2.00,2.00,9.00,"1315",6.00,"1330"
24 2018,2,5,17,4,2018-05-17,"WN","N420WN","1732",13342,"MKE","Milwaukee, WI","WI","Wisconsin",12953,"LGA","New York, NY","NY","New York",0620,"0617",-3.00,0.00,18.00,"0913",11.00,"09
25 2019,1,1,18,5,2019-01-18,"UA","N588UA","510",12892,"LAX","Los Angeles, CA","CA","California",11618,"EWR","Newark, NJ","NJ","New Jersey",1315,"1309",-6.00,0.00,17.00,"2107",6.00,"2
26 2017,4,12,1,5,2017-12-01,"EV","N758EV","5393",11433,"DTW","Detroit, MI","MI","Michigan",14524,"RIC","Richmond, VA","VA","Virginia",1540,"1537",-3.00,0.00,19.00,"1706",11.00,"1732"
27 2018,1,1,27,6,2018-01-27,"VX","N365VA","1592",14771,"SFO","San Francisco, CA","CA","California",14262,"PSP","Palm Springs, CA","CA","California",1330,"1330",0.00,0.00,14.00,"1446",
28 2018,2,5,10,4,2018-05-10,"AS","N633VA","1756",14747,"SEA","Seattle, WA","WA","Washington",14771,"SFO","San Francisco, CA","CA","California",1855,"1859",4.00,4.00,14.00,"2047",6.00
```


BINNING:

Separated the data based on Cancellation Reason by using Binning Technique.

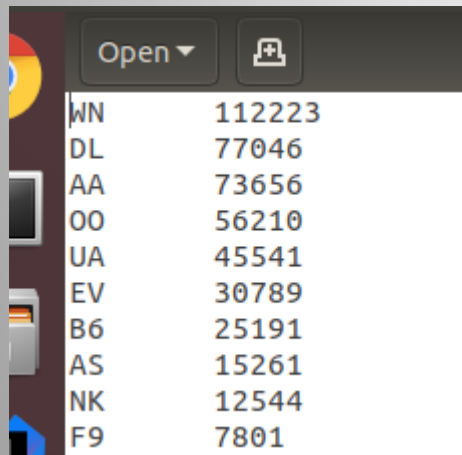


The screenshot shows an IDE with a project named 'FlightDelay' and a sub-project 'OFlightCancellation'. The project structure includes folders for 'OAvGFlightDelay', 'OCancellationCount', 'OCounters', 'ODelayCount', 'ODelayFMedian', 'ODistinct', and 'OFlightCancellation'. The 'OFlightCancellation' folder contains a file named '_SUCCESS.crc' and a series of files named 'Carrier-cancellation-m-00000.crc' through 'Carrier-cancellation-m-00013.crc'. The main editor window displays the content of 'NAS-cancellation-m-00072', which is a large table of flight cancellation data. The table has 32 columns, with the first column being an index from 1 to 32. The data rows contain flight details such as date, time, origin, destination, carrier, and cancellation reason. For example, the first row (index 1) shows a flight on 2017-01-06 from Fort Lauderdale, FL to Detroit, MI, with a cancellation reason of 'DL', 'N967DL', '1198', '11697', 'FLL', 'Fort Lauderdale, FL', 'FL', 'Florida', '11433', 'DTW', 'Detroit, MI', 'MI', 'Michigan', '1346', '1651', '1.00', 'C', '0.00', '185.00', '1.00'. The table continues with similar data for indices 2 through 32.

PIG: TOP K

Found Top 10 Airline with Most Flights using Pig.

```
flights = LOAD 'apr17.csv' using PigStorage(',');
grouped = GROUP flights BY $6;
summed = FOREACH grouped GENERATE group, COUNT(flights) AS cntd;
sorted = ORDER summed BY cntd DESC;
top25 = LIMIT sorted 10;
Dump top25;
STORE top25 into 'top10Carriers';
```



A screenshot of a terminal window displaying the output of a Pig script. The window has a dark header bar with an 'Open' button and a file icon. The output is a list of airline codes and their corresponding flight counts, sorted in descending order. The airlines listed are WN, DL, AA, OO, UA, EV, B6, AS, NK, and F9.

WN	112223
DL	77046
AA	73656
OO	56210
UA	45541
EV	30789
B6	25191
AS	15261
NK	12544
F9	7801

PIG: JOIN

Joined Airline csv and unique Airline names using Pig.

```
uniqueid = LOAD 'distinct' using PigStorage(',') as (uid);
carriers = LOAD 'carriers.csv' using PigStorage(',') as (id,name);
joined = JOIN uniqueid BY uid, carriers BY id;
summed = FOREACH joined GENERATE $0,$2;
STORE summed into 'joinoutput';
```

"9E"	"Pinnacle Airlines Inc."
"AA"	"American Airlines Inc."
"AS"	"Alaska Airlines Inc."
"B6"	"JetBlue Airways"
"DL"	"Delta Air Lines Inc."
"EV"	"Atlantic Southeast Airlines"
"F9"	"Frontier Airlines Inc."
"G4"	"Allegiant Air"
"HA"	"Hawaiian Airlines Inc."
"MQ"	"American Eagle Airlines Inc."
"NK"	"Spirit Air Lines"
"OH"	"Comair Inc."
"OO"	"Skywest Airlines Inc."
"UA"	"United Air Lines Inc."
"VX"	"Virgin America"
"WN"	"Southwest Airlines Co."
"YV"	"Mesa Airlines Inc."
"YX"	"Midwest Airline"

PIG FILTER AND MERGE:

Filtered the Latest Data based on Year using Pig.

```
flights = LOAD 'Merge.csv' using PigStorage(',');
filter1 = FILTER flights BY $0 > 2017;
filter2 = FOREACH filter1 GENERATE $0,$1,$2,$3,$4,$5,$6,$7,$8,$9,'LatestData';
STORE filter2 into 'LatestData';
```

part-m-00000										
/usr/local/pig-0.17.0/bin/Late										
2018	1	3	1	4	2018-03-01	F9	N201FR	1680	12889	Latest Data
2018	1	3	1	4	2018-03-01	F9	N201FR	681	13204	Latest Data
2018	1	3	1	4	2018-03-01	F9	N201FR	681	11292	Latest Data
2018	1	3	1	4	2018-03-01	F9	N201FR	144	14747	Latest Data
2018	1	3	1	4	2018-03-01	F9	N201FR	122	11292	Latest Data
2018	1	3	1	4	2018-03-01	F9	N202FR	1138	12889	Latest Data
2018	1	3	1	4	2018-03-01	F9	N202FR	1141	10423	Latest Data
2018	1	3	1	4	2018-03-01	F9	N202FR	1106	12889	Latest Data
2018	1	3	1	4	2018-03-01	F9	N202FR	1111	12266	Latest Data
2018	1	3	1	4	2018-03-01	F9	N203FR	1236	13487	Latest Data
2018	1	3	1	4	2018-03-01	F9	N203FR	1201	15304	Latest Data
2018	1	3	1	4	2018-03-01	F9	N203FR	1204	12339	Latest Data
2018	1	3	1	4	2018-03-01	F9	N203FR	1209	15304	Latest Data
2018	1	3	1	4	2018-03-01	F9	N203FR	1099	13342	Latest Data

Used AWS Cloud EMR instance for running MapReduce Jobs.

AWS Services Resource Groups

Amazon EMR

- Clusters
- Security configurations
- VPC subnets
- Events
- Notebooks
- Help
- What's new

Cluster: FlightCluster Waiting Cluster ready after last step completed.

Summary Application history Monitoring Hardware Configurations Events Steps Bootstrap actions

Add step Clone step Cancel step

Steps

Filter: All steps Filter steps ... 4 steps (all loaded)

	ID	Name	Status	Start time (UTC-4)	Elapsed time	Log files	Actions
<input type="radio"/>	s-2KLGKR4N9VF0Z	Custom JAR	Completed	2019-04-24 19:45 (UTC-4)	4 minutes	View logs	View jobs

JAR location : s3://flightmapreduce/jar/FlightDelay.jar
Main class : None
Arguments : AirlineCancel7.Driver s3://flightmapreduce/input s3://flightmapreduce/output/file
Action on failure: Continue

[View all interactive jobs](#) | [View all jobs](#)

Amazon S3

> ... > 000000 > job_1556143697838_0001-1556149535584-hadoop-FlightDelay.jar-1556149810763-60-1-SUCCEEDED-default-1556149546461.jhist.gz

job_1556143697838_0001-1556149535584-hadoop-Flig... Latest version ▾

Overview

Properties

Permissions

Select from

Open

Download

Download as

Make public

Copy path

Owner

shetty.poo

Last modified

Apr 24, 2019 7:58:17 PM GMT-0400

Etag

ee493fe385f7a1d2785881c5bcd44eb4

Storage class

Standard

Server-side encryption

AES-256

Size

21.2 KB

Key

logs/j-266838CXTTDT1/hadoop-mapreduce/history/2019/04/24/000000/job_1556143697838_0001-1556149535584-hadoop-FlightDelay.jar-1556149810763-60-1-SUCCEEDED-default-1556149546461.jhist.gz

Object URL

https://s3.amazonaws.com/flightmapreduce/logs/j-266838CXTTDT1/hadoop-mapreduce/history/2019/04/24/000000/job_1556143697838_0001-1556149535584-hadoop-FlightDelay.jar-1556149810763-60-1-SUCCEEDED-default-1556149546461.jhist.gz

Open ▾				parc1-00000
				~/Downloads
9E	A	1	13	
9E	A	10	12	
9E	A	12	23	
9E	A	3	494	
9E	A	5	8	
9E	A	9	7	
9E	B	1	364	
9E	B	10	106	
9E	B	12	28	
9E	B	3	2086	
9E	B	5	350	
9E	B	7	12	
9E	B	9	36	
9E	C	1	1077	
9E	C	10	60	
9E	C	12	65	
9E	C	3	486	
9E	C	5	84	
9E	C	7	1302	
9E	C	9	109	
AA	A	1	1008	
AA	A	10	642	
AA	A	12	553	
AA	A	3	621	
AA	A	4	274	
AA	A	5	1180	
AA	A	7	2335	
AA	A	8	608	
AA	A	9	333	
AA	B	1	3671	
AA	B	10	1168	
AA	B	12	1132	
AA	B	3	5525	
AA	B	4	356	
AA	B	5	1322	
AA	B	7	2290	
AA	B	8	974	
AA	B	9	1021	
AA	C	1	214	
AA	C	10	266	
AA	C	12	13	
AA	C	3	52	

TECHNOLOGIES AND PATTERNS USED

- Max Min
- Count
- Average
- Median
- Memory Conscious Median
- Inverted Index
- Counting with Counters
- Input Format
- Custom Combiner
- Partitioner

- Bloom Filter
- Top K
- Distinct
- Joins
- Shuffling
- Binning
- Chaining
- AWS Cloud
- PIG Top K
- PIG Filter and Merge
- PIG Joins