Data

Air Travel Consumer Report Period From June 2003 to October 2020

Airline Delay Causes Database:

Airport delay statistics (Bureau of Transportation Statistics): Dataset obtained from: Dataset link

carrier information:

- carrier: Airline code.
- carrier_name: Airline name.
- airport: Airport code.
- airport_name: Airport name.

Airport statistics:

- arr_flights: Number of flights which arrived at the airport.
- arr_del15: Number of flights delayed (>= 15minutes late).
- carrier_ct: Number of flights delayed due to air carrier (e.g. maintenance or crew problems, aircraft cleaning, baggage loading, fueling, etc.).
- weather_ct: Number of flights delayed due to weather.
- nas_ct: Number of flights delayed due to National Aviation System (e.g. non-extreme weather conditions, airport operations, heavy traffic volume, and air traffic control).
- security_ct: Number of flights delayed due to security (e.g. evacuation of a terminal or concourse, re-boarding of aircraft because of security breach, inoperative screening equipment and/or long lines in excess of 29 minutes at screening areas).
- late_aircraft_ct: Number of flights delayed due to a previous flight using the same aircraft being late.
- arr_cancelled: Number of cancelled flights.
- arr_diverted: Number of diverted flights.
- arr_delay: Total time (minutes) of delayed flights.
- carrier_delay: Total time (minutes) of delayed flights due to air carrier.
- weather_delay: Total time (minutes) of delayed flights due to weather.
- nas_delay: Total time (minutes) of delayed flights due to National Aviation System.
- security_delay: Total time (minutes) of delayed flights due to security.
- late_aircraft_delay: Total time (minutes) of delayed flights due to a previous flight using the same aircraft being late.

Airport Database:

Dataset obtained from: Aviation Support Tables

airports.csv describes the locations of US airports, with the fields:

- AirportID: An identification number assigned by US DOT to identify a unique airport. Use this field for airport analysis across a range of years because an airport can change its airport code and airport codes can be reused.
- Airport: A three character alpha-numeric code issued by the U.S. Department of Transportation which is the official designation of the airport. The airport code is not always unique to a specific airport because airport codes can change or can be reused
- AirportName:Airport Name.
- AirportCityName: Airport City Name with either U.S. State or Country"EX: NY =New York"
- AirportCountryName: Country Name for the Physical Location of the Airport (USA).
- AirportStateName: State Name for the Physical Location of the Airport.
- Latitude: Decimal degrees, usually to six significant digits. Negative is South, positive is North.
- Longitude: Decimal degrees, usually to six significant digits. Negative is West, positive is East.

This majority of this data comes from the FAA, but a few extra airports (mainly military bases and US protectorates) were collected from other web sources by Ryan Hafen and Hadley Wickham.

Types of Delay

Carrier Delay

Carrier delay is within the control of the air carrier. Examples of occurrences that may determine carrier delay are: aircraft cleaning, aircraft damage, awaiting the arrival of connecting passengers or crew, baggage, bird strike, cargo loading, catering, computer, outage-carrier equipment, crew legality (pilot or attendant rest), damage by hazardous goods, engineering inspection, fueling, handling disabled passengers, late crew, lavatory servicing, maintenance, oversales, potable water servicing, removal of unruly passenger, slow boarding or seating, stowing carry-on baggage, weight and balance delays.

Late Arrival Delay

Arrival delay at an airport due to the late arrival of the same aircraft at a previous airport. The ripple effect of an earlier delay at downstream airports is referred to as delay propagation.

NAS Delay

Delay that is within the control of the National Airspace System (NAS) may include: non-extreme weather conditions, airport operations, heavy traffic volume, air traffic control, etc. Delays that occur after Actual Gate Out are usually attributed to the NAS and are also reported through OPSNET.

Security Delay

Security delay is caused by evacuation of a terminal or concourse, re-boarding of aircraft because of security breach, inoperative screening equipment and/or long lines in excess of 29 minutes at screening areas.

Weather Delay Weather delay is caused by extreme or hazardous weather conditions that are forecasted or manifest themselves on point of departure, enroute, or on point of arrival.

OPSNET Delay Cause

Delays to Instrument Flight Rules (IFR) traffic of 15 minutes or more, experienced by individual flights, which result from the ATC system detaining an aircraft at the gate, short of the runway, on the runway, on a taxiway, and/or in a holding configuration anywhere en route.

Such delays include delays due to weather conditions at airports and en route (Weather), FAA and non-FAA equipment malfunctions (Equipment), the volume of traffic at an airport (Volume), reduction to runway capacity (Runway), and other factors (Others). Flight delays of less than 15 minutes are not reported in OPSNET. ASPM reports the most dominant OPSNET delay cause for any flight with an ASQP Reported NAS Delay.

flight information

Diverted Flight

A diverted flight is one that has been routed from its original arrival destination to a new, typically temporary, arrival destination. The leg of the flight that is routed back to the original arrival destination is called the recovery leg.

When you are viewing flight information for a diverted flight, you will see the diversion leg. The recovery leg will be displayed only if available.

Bureau of Transportation Statistics

Air Travel Consumer Report Period From June 2003 to October 2020

Types of Delay

1.Carrier Delay

2.Late Arrival Delay

3.NAS Delay

4.Security Delay Cause

5.Weather Delay Cause

Analysis layer

1.Airline

2.Airport

2.State

In [1]:

```
# Necessary imports

import os              #this module routines for NT or Posix depending on what system
we're on.
import pandas as pd    # this module
pd.set_option('display.max_columns', None)

import numpy as np     # library used for working with arrays. It also has functions for
working in domain of linear algebra
import types
import csv
import seaborn as sb   # provides a high-level interface for drawing attractive and
informative statistical graphics.
import time            # This module provides various functions to manipulate time values.
import pandocfilters   # This Functions to aid writing python scripts that process the
pandoc AST serialized as JSON.
import nbconvert       # This module converting notebooks to and from different formats
import pyppeteer       # Generate screenshots and PDFs of pages
import pip

print ("libarary-Imported")
```

```
libarary-Imported
```

In [2]:

```
# Matplotlib and associated plotting modules

import matplotlib.cm as cm
import matplotlib.colors as colors
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
%matplotlib inline
import plotly as px

print ("libarary-Imported")
```

```
libarary-Imported
```

In [3]:

```
#folium makes it easy to visualize data that’s been manipulated in Python on an
interactive leaflet map
import folium
from folium import plugins
import plotly.express as px  #Plotly Express is a terse, consistent, high-level API for
creating figures
```

```
print ("libarary-Imported")
```

```
libarary-Imported
```

In [4]:

```
#geospatial data visualization library

import geoplot as gplt  #geoplot is a geospatial data visualization library designed for data scientists and geospatial analysts
import geopandas as gpd  #GeoPandas depends for its spatial functionality (GEOS, GDAL, PROJ).
import geoplot.crs as gcrs

print ("libarary-Imported")
```

```
libarary-Imported
```

In [5]:

```
#converting notebooks to different format

from IPython.display import HTML
from timeit import default_timer as timer
from pandas.core.tools.datetimes import to_datetime
from pandas.core.tools.timedeltas import to_timedelta
from nbconvert import LatexExporter
from nbconvert import PDFExporter
from nbconvert import webpdf
from nbconvert import nbconvertapp
from nbconvert import templates

print ("functions-Imported")
```

```
functions-Imported
```

In [6]:

```
#Determine current working directory

import os
os.getcwd()
```

Out[6]:

```
'C:\\Users\\Abdelrazek\\PycharmProjects\\Flight'
```

In [7]:

```
# Read the csv file, and check its top 5 rows airports data frame

airports = pd.read_csv('airports_Master_Coordinate.csv')
print(airports.shape)
airports.head()
```

```
(18109, 10)
```

Out[7]:

	AIRPORT_ID	AIRPORT	DISPLAY_AIRPORT_NAME	DISPLAY_AIRPORT_CITY_NAME_FULL	AIRPORT_COUNTRY_NAME	AIRPORT_STAT
0	10001	01A	Afognak Lake Airport	Afognak Lake, AK	United States	
1	10003	03A	Bear Creek Mining Strip	Granite Mountain, AK	United States	

2	10004	04A	Lik Mining Camp	Lik, AK	United States
3	10005	05A	Little Squaw Airport	Little Squaw, AK	United States
4	10006	06A	Kizhuyak Bay	Kizhuyak, AK	United States

In [8]:

```
#check COUNTRY_NAME

airports['AIRPORT_COUNTRY_NAME'].values
```

Out[8]:

```
array(['United States', 'United States', 'United States', ..., 'Somalia',
       'South Sudan', 'United States'], dtype=object)
```

In [9]:

```
#filter airports data frame to get usa_airports

usa_airports=airports.loc[airports['AIRPORT_COUNTRY_NAME'] =='United States']
print(usa_airports.shape)
usa_airports.head()
```

```
(6931, 10)
```

Out[9]:

	AIRPORT_ID	AIRPORT	DISPLAY_AIRPORT_NAME	DISPLAY_AIRPORT_CITY_NAME_FULL	AIRPORT_COUNTRY_NAME	AIRPORT_STAT
0	10001	01A	Afognak Lake Airport	Afognak Lake, AK	United States	
1	10003	03A	Bear Creek Mining Strip	Granite Mountain, AK	United States	
2	10004	04A	Lik Mining Camp	Lik, AK	United States	
3	10005	05A	Little Squaw Airport	Little Squaw, AK	United States	
4	10006	06A	Kizhuyak Bay	Kizhuyak, AK	United States	

In [10]:

```
#create usa_airports csv file

usa_airports.to_csv (r'C:\Users\Abdelrazek\PycharmProjects\Flight\usa_airports.csv', index
= False, header=True)
usa_airports_df = pd.read_csv('usa_airports.csv')
usa_airports_df.head(2)
```

Out[10]:

	AIRPORT_ID	AIRPORT	DISPLAY_AIRPORT_NAME	DISPLAY_AIRPORT_CITY_NAME_FULL	AIRPORT_COUNTRY_NAME	AIRPORT_STAT
0	10001	01A	Afognak Lake Airport	Afognak Lake, AK	United States	
1	10003	03A	Bear Creek Mining Strip	Granite Mountain, AK	United States	

In [11]:

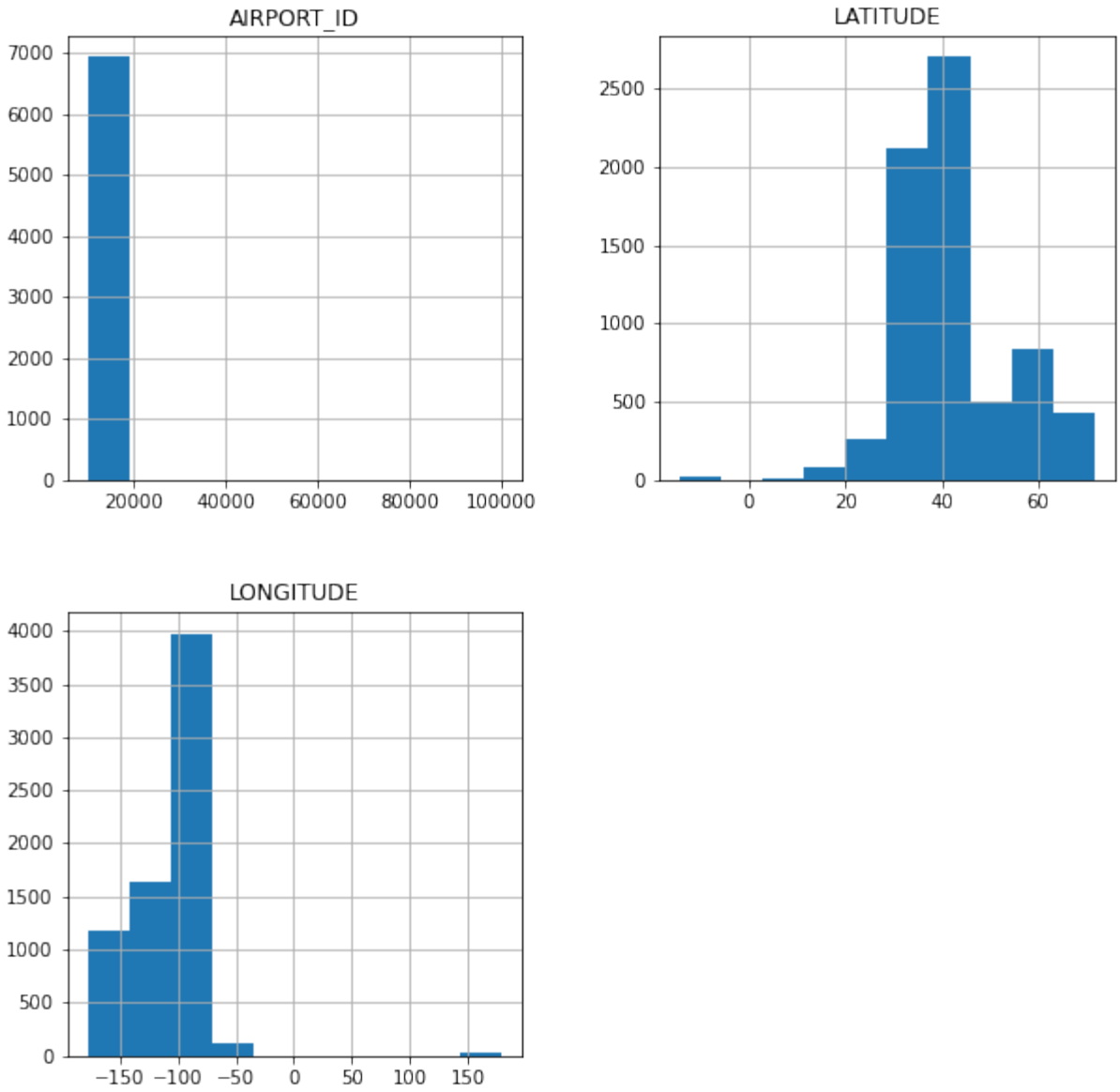
```
#drop columan without value

usa_airports_df_1=usa_airports_df.drop(columns=['Unnamed: 9'])
usa_airports_df_1.head(2)
```

Out[11]:

	AIRPORT_ID	AIRPORT	DISPLAY_AIRPORT_NAME	DISPLAY_AIRPORT_CITY_NAME_FULL	AIRPORT_COUNTRY_NAME	AIRPORT_STAT
0	10001	01A	Afognak Lake Airport	Afognak Lake, AK	United States	
1	10003	03A	Bear Creek Mining Strip	Granite Mountain, AK	United States	

In [12]:

```
#Explore what the histogram of the data looks like
usa_airports_df_1.hist(figsize=(10,10));
```



In [13]:

```
# display a summary of the dataframe
```

```
usa_airports_df_1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6931 entries, 0 to 6930
Data columns (total 9 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   AIRPORT_ID                      6931 non-null   int64  
 1   AIRPORT                         6931 non-null   object 
 2   DISPLAY_AIRPORT_NAME            6931 non-null   object 
 3   DISPLAY_AIRPORT_CITY_NAME_FULL  6931 non-null   object 
 4   AIRPORT_COUNTRY_NAME            6931 non-null   object 
 5   AIRPORT_STATE_NAME              6931 non-null   object 
 6   DISPLAY_CITY_MARKET_NAME_FULL   6931 non-null   object 
 7   LATITUDE                        6930 non-null   float64
 8   LONGITUDE                       6930 non-null   float64
dtypes: float64(2), int64(1), object(6)
memory usage: 487.5+ KB
```

In [14]:

```
# Read the csv file, and check its top 2 rows

airline_delay_causes = pd.read_csv('airline_delay_causes.csv')
print(airline_delay_causes.shape)
airline_delay_causes.head(2)
```

```
(289637, 22)
```

Out[14]:

	year	month	carrier	carrier_name	airport	airport_name	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	late
0	2004	1	DL	Delta Air Lines Inc.	PBI	West Palm Beach/Palm Beach, FL: Palm Beach Int...	650.0	126.0	21.06	6.44	51.58	1.0	
1	2004	1	DL	Delta Air Lines Inc.	PDX	Portland, OR: Portland International	314.0	61.0	14.09	2.61	34.25	0.0	

In [15]:

```
# display a summary of the dataframe

airline_delay_causes.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 289637 entries, 0 to 289636
Data columns (total 22 columns):
 #   Column               Non-Null Count   Dtype  
---  ------               --------------   -----  
 0   year                 289637 non-null  int64  
 1    month               289637 non-null  int64  
 2   carrier              289637 non-null  object 
 3   carrier_name         289637 non-null  object 
 4   airport              289637 non-null  object 
 5   airport_name         289637 non-null  object 
 6   arr_flights          289193 non-null  float64
 7   arr_del15            288968 non-null  float64
 8   carrier_ct           289193 non-null  float64
 9    weather_ct          289193 non-null  float64
```

```
 10  nas_ct               289193 non-null  float64
 11  security_ct          289193 non-null  float64
 12  late_aircraft_ct     289193 non-null  float64
 13  arr_cancelled        289193 non-null  float64
 14  arr_diverted         289193 non-null  float64
 15   arr_delay           289193 non-null  float64
 16   carrier_delay       289193 non-null  float64
 17  weather_delay        289193 non-null  float64
 18  nas_delay            289193 non-null  float64
 19  security_delay       289193 non-null  float64
 20  late_aircraft_delay  289193 non-null  float64
 21  Unnamed: 21          0 non-null       float64
dtypes: float64(16), int64(2), object(4)
memory usage: 48.6+ MB
```

In [16]:

```
#rename columns strat with space

airline_delay_causes.rename(columns={'airport' : 'AIRPORT',' month' : 'month','
weather_ct': 'weather_ct',' arr_delay' : 'arr_delay',' carrier_delay' : 'carrier_delay'},
inplace=True)
```

In [17]:

```
# display a summary of the dataframe

airline_delay_causes.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 289637 entries, 0 to 289636
Data columns (total 22 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   year                 289637 non-null  int64
 1   month                289637 non-null  int64
 2   carrier              289637 non-null  object
 3   carrier_name         289637 non-null  object
 4   AIRPORT              289637 non-null  object
 5   airport_name         289637 non-null  object
 6   arr_flights          289193 non-null  float64
 7   arr_del15            288968 non-null  float64
 8   carrier_ct           289193 non-null  float64
 9   weather_ct           289193 non-null  float64
 10  nas_ct               289193 non-null  float64
 11  security_ct          289193 non-null  float64
 12  late_aircraft_ct     289193 non-null  float64
 13  arr_cancelled        289193 non-null  float64
 14  arr_diverted         289193 non-null  float64
 15  arr_delay            289193 non-null  float64
 16  carrier_delay        289193 non-null  float64
 17  weather_delay        289193 non-null  float64
 18  nas_delay            289193 non-null  float64
 19  security_delay       289193 non-null  float64
 20  late_aircraft_delay  289193 non-null  float64
 21  Unnamed: 21          0 non-null       float64
dtypes: float64(16), int64(2), object(4)
memory usage: 48.6+ MB
```

In [18]:

```
#get all coulman names

airline_delay_causes.columns
```

Out[18]:

```
Index(['year', 'month', 'carrier', 'carrier_name', 'AIRPORT', 'airport_name',
```

```
       'arr_flights', 'arr_del15', 'carrier_ct', 'weather_ct', 'nas_ct',
       'security_ct', 'late_aircraft_ct', 'arr_cancelled', 'arr_diverted',
       'arr_delay', 'carrier_delay', 'weather_delay', 'nas_delay',
       'security_delay', 'late_aircraft_delay', 'Unnamed: 21'],
      dtype='object')
```

In [19]:

```
# drop duplicates of the dataframe
print(airline_delay_causes.duplicated().sum)
```

```
<bound method NDFrame._add_numeric_operations.<locals>.sum of 0         False
1         False
2         False
3         False
4         False
          ...
289632    False
289633    False
289634    False
289635    False
289636    False
Length: 289637, dtype: bool>
```

In [20]:

```
#drop columan without value
df_airline_delay_causes=airline_delay_causes.drop(columns=['Unnamed: 21'])
df_airline_delay_causes.head(2)
```

Out[20]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	la
0	2004	1	DL	Delta Air Lines Inc.	PBI	West Palm Beach/Palm Beach, FL: Palm Beach Int...	650.0	126.0	21.06	6.44	51.58	1.0	
1	2004	1	DL	Delta Air Lines Inc.	PDX	Portland, OR: Portland International	314.0	61.0	14.09	2.61	34.25	0.0	

In [21]:

```
# display a static summary of the dataframe
df_airline_delay_causes.describe()
```

Out[21]:

	year	month	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	late_
count	289637.000000	289637.000000	289193.000000	288968.000000	289193.000000	289193.000000	289193.000000	289193.000000	289
mean	2011.562369	6.516215	389.580585	75.004270	21.097864	2.675956	25.127567	0.173963	
std	5.161139	3.426717	1042.744693	203.225578	46.928801	10.173334	88.057001	0.829669	
min	2003.000000	1.000000	1.000000	0.000000	0.000000	0.000000	-0.010000	0.000000	
25%	2007.000000	4.000000	60.000000	10.000000	3.230000	0.000000	1.950000	0.000000	
50%	2011.000000	7.000000	124.000000	24.000000	8.530000	0.620000	5.920000	0.000000	
75%	2016.000000	9.000000	279.000000	58.000000	20.000000	2.080000	16.120000	0.000000	

max	2020.000000	12.000000	21977.000000	6377.000000	1792.070000	717.940000	4091.270000	80.560000	1

In [22]:

```
#Calculating the average for late_aircraft_dela
avg_late_aircraft_delay=df_airline_delay_causes.late_aircraft_delay.mean()
avg_late_aircraft_delay
```

Out[22]:

```
1638.9782878562069
```

In [23]:

```
# index
df_airline_delay_causes.set_index(['year', 'month', 'carrier', 'carrier_name', 'AIRPORT',
'airport_name',
       'arr_flights', 'arr_del15', 'carrier_ct', 'weather_ct', 'nas_ct',
       'security_ct', 'late_aircraft_ct', 'arr_cancelled', 'arr_diverted',
       'arr_delay', 'carrier_delay', 'weather_delay', 'nas_delay',
       'security_delay', 'late_aircraft_delay'])
df_airline_delay_causes.index
```

Out[23]:

```
RangeIndex(start=0, stop=289637, step=1)
```

In [24]:

```
#Calculating Difference from Average Delay
#add a new column in the dataframe name 'def_avg_Delay'
df_airline_delay_causes.reset_index
df_avg_delay_causes = df_airline_delay_causes.copy()
df_avg_delay_causes
df_avg_delay_causes['def_avg_Delay'] = df_avg_delay_causes['late_aircraft_delay']
-avg_late_aircraft_delay
df_avg_delay_causes.head(2)
```

Out[24]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	la
0	2004	1	DL	Delta Air Lines Inc.	PBI	West Palm Beach/Palm Beach, FL: Palm Beach Int...	650.0	126.0	21.06	6.44	51.58	1.0	
1	2004	1	DL	Delta Air Lines Inc.	PDX	Portland, OR: Portland International	314.0	61.0	14.09	2.61	34.25	0.0	

In [25]:

```
#Explore what the histogram of the data looks like
df_avg_delay_causes.hist(figsize=(25,25));
```

In [26]:

```
# display a sum of zero value in the dataframe for each columan
# we will drop zero value as we will level up report data in additional steps

print((df_avg_delay_causes == 0).sum())
```

```
year                  0
month                 0
carrier               0
carrier_name          0
AIRPORT               0
airport_name          0
arr_flights           0
```

```
arr_del15                7595
carrier_ct              14808
weather_ct             112636
nas_ct                  23600
security_ct            251295
late_aircraft_ct        33045
arr_cancelled          115905
arr_diverted           207682
arr_delay                7820
carrier_delay           14806
weather_delay          112614
nas_delay               23575
security_delay         251287
late_aircraft_delay     33040
def_avg_Delay               0
dtype: int64
```

In [27]:

```
#level up and calculate the Delay in (min/flight) for each airport and carrier
#add a new column in the dataframe name 'aircraft_delay(min/flight)'
df_avg_delay_causes_1 = df_avg_delay_causes.copy()
df_avg_delay_causes_1
df_avg_delay_causes_1['aircraft_delay(min/flight)'] =
df_avg_delay_causes_1['late_aircraft_delay'] /df_avg_delay_causes_1['late_aircraft_ct']
df_avg_delay_causes_1.drop(['late_aircraft_ct'],axis = 1, inplace = True)
df_avg_delay_causes_1.drop(['late_aircraft_delay'],axis = 1, inplace = True)
print(df_avg_delay_causes_1.shape)
df_avg_delay_causes_1.head(2)
```

```
(289637, 21)
```

Out[27]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	ar
0	2004	1	DL	Delta Air Lines Inc.	PBI	West Palm Beach/Palm Beach, FL: Palm Beach Int...	650.0	126.0	21.06	6.44	51.58	1.0	
1	2004	1	DL	Delta Air Lines Inc.	PDX	Portland, OR: Portland International	314.0	61.0	14.09	2.61	34.25	0.0	

In [28]:

```
# display a static summary of the dataframe
df_avg_delay_causes_1.describe()
```

Out[28]:

	year	month	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	arr_
count	289637.000000	289637.000000	289193.000000	288968.000000	289193.000000	289193.000000	289193.000000	289193.000000	289
mean	2011.562369	6.516215	389.580585	75.004270	21.097864	2.675956	25.127567	0.173963	
std	5.161139	3.426717	1042.744693	203.225578	46.928801	10.173334	88.057001	0.829669	
min	2003.000000	1.000000	1.000000	0.000000	0.000000	0.000000	-0.010000	0.000000	
25%	2007.000000	4.000000	60.000000	10.000000	3.230000	0.000000	1.950000	0.000000	

50%	2011.000000	7.000000	124.000000	24.000000	8.530000	0.620000	5.920000	0.000000	
75%	2016.000000	9.000000	279.000000	58.000000	20.000000	2.080000	16.120000	0.000000	
max	2020.000000	12.000000	21977.000000	6377.000000	1792.070000	717.940000	4091.270000	80.560000	4

we can see that we get infinity for the mean value as we discover that we has zero values

- so i will remove zero values from data frame and re level up my data

In [29]:

```
#replace zero values  with nan
df_avg_delay_causes.copy()
df_avg_delay_causes_2 = df_avg_delay_causes.replace(0, np.nan)
df_avg_delay_causes_2
```

Out[29]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_
0	2004	1	DL	Delta Air Lines Inc.	PBI	West Palm Beach/Palm Beach, FL: Palm Beach Int...	650.0	126.0	21.06	6.44	51.58	
1	2004	1	DL	Delta Air Lines Inc.	PDX	Portland, OR: Portland International	314.0	61.0	14.09	2.61	34.25	Na
2	2004	1	DL	Delta Air Lines Inc.	PHL	Philadelphia, PA: Philadelphia International	513.0	97.0	27.60	0.42	51.86	Na
3	2004	1	DL	Delta Air Lines Inc.	PHX	Phoenix, AZ: Phoenix Sky Harbor International	334.0	78.0	20.14	2.02	39.39	Na
4	2004	1	DL	Delta Air Lines Inc.	PIT	Pittsburgh, PA: Pittsburgh International	217.0	47.0	8.08	0.44	21.89	Na
...	...	...	...	...	...	...	...	...	...	...	...	
289632	2019	1	MQ	Envoy Air	RIC	Richmond, VA: Richmond International	195.0	68.0	12.12	1.87	17.97	Na
289633	2019	1	MQ	Envoy Air	ROA	Roanoke, VA: Roanoke Blacksburg Regional Woodr...	52.0	14.0	2.74	0.69	2.46	Na
289634	2019	1	MQ	Envoy Air	ROC	Rochester, NY: Greater Rochester International	106.0	26.0	4.67	2.26	11.81	Na

289635	2019	1	MQ	Envoy Air	RST	Rochester, MN: Rochester International	116.0	35.0	6.83	6.92	11.50	Na
289636	2019	1	MQ	Envoy Air	SAT	San Antonio, TX: San Antonio International	26.0	4.0	1.16	0.64	1.92	Na

289637 rows × 22 columns

In [30]:

```
# display a sum of zero value in the dataframe for each columan
# we will drop zero value as we will level up report data in additional steps

print((df_avg_delay_causes_2 == 0).sum())
```

```
year                   0
month                  0
carrier                0
carrier_name           0
AIRPORT                0
airport_name           0
arr_flights            0
arr_del15              0
carrier_ct             0
weather_ct             0
nas_ct                 0
security_ct            0
late_aircraft_ct       0
arr_cancelled          0
arr_diverted           0
arr_delay              0
carrier_delay          0
weather_delay          0
nas_delay              0
security_delay         0
late_aircraft_delay    0
def_avg_Delay          0
dtype: int64
```

In [31]:

```
# display a summary of the dataframe info

df_avg_delay_causes_2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 289637 entries, 0 to 289636
Data columns (total 22 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   year                 289637 non-null  int64
 1   month                289637 non-null  int64
 2   carrier              289637 non-null  object
 3   carrier_name         289637 non-null  object
 4   AIRPORT              289637 non-null  object
 5   airport_name         289637 non-null  object
 6   arr_flights          289193 non-null  float64
 7   arr_del15            281373 non-null  float64
 8   carrier_ct           274385 non-null  float64
 9   weather_ct           176557 non-null  float64
```

```
 10  nas_ct                265593 non-null  float64
 11  security_ct           37898 non-null   float64
 12  late_aircraft_ct      256148 non-null  float64
 13  arr_cancelled         173288 non-null  float64
 14  arr_diverted          81511 non-null   float64
 15  arr_delay             281373 non-null  float64
 16  carrier_delay         274387 non-null  float64
 17  weather_delay         176579 non-null  float64
 18  nas_delay             265618 non-null  float64
 19  security_delay        37906 non-null   float64
 20  late_aircraft_delay   256153 non-null  float64
 21  def_avg_Delay         289193 non-null  float64
dtypes: float64(16), int64(2), object(4)
memory usage: 48.6+ MB
```

In [32]:

```
#drop NaN value
df_avg_delay_causes_3=df_avg_delay_causes_2.dropna()
delay_causes_df=df_avg_delay_causes_3.copy()
delay_causes_df.head(2)
```

Out[32]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	l
21	2004	1	DL	Delta Air Lines Inc.	SLC	Salt Lake City, UT: Salt Lake City International	3012.0	756.0	163.51	33.04	427.21	1.60	
60	2004	1	EV	Atlantic Southeast Airlines	DFW	Dallas/Fort Worth, TX: Dallas/Fort Worth Inter...	4322.0	582.0	152.53	91.84	270.74	4.89	

In [33]:

```
#create usa_airports csv file ,Read the csv file, and check its top 2 rows
delay_causes_df.to_csv (r'C:\Users\Abdelrazek\PycharmProjects\Flight\delay_causes.csv',
index = False, header=True)
df_delay_causes= pd.read_csv('delay_causes.csv')
df_delay_causes.head(2)
```

Out[33]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	la
0	2004	1	DL	Delta Air Lines Inc.	SLC	Salt Lake City, UT: Salt Lake City International	3012.0	756.0	163.51	33.04	427.21	1.60	
1	2004	1	EV	Atlantic Southeast Airlines	DFW	Dallas/Fort Worth, TX: Dallas/Fort Worth Inter...	4322.0	582.0	152.53	91.84	270.74	4.89	

In [34]:

```
# display a static summary of the dataframe
df_delay_causes.describe()
```

Out[34]:

	year	month	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	late_aircraft_ct
count	16903.000000	16903.000000	16903.000000	16903.000000	16903.000000	16903.000000	16903.000000	16903.000000	16903.000000
mean	2011.244217	6.414542	2609.453233	505.065965	119.759147	17.025426	177.586204	1.809004	188.88627
std	4.981090	3.435825	2844.515359	568.002107	127.641867	34.618548	264.992637	2.496775	218.25081
min	2003.000000	1.000000	21.000000	4.000000	0.270000	0.010000	0.170000	0.010000	0.02000
25%	2007.000000	3.000000	564.000000	127.000000	37.000000	2.790000	31.320000	0.570000	36.75000
50%	2011.000000	6.000000	1575.000000	304.000000	79.590000	6.890000	80.660000	1.000000	108.82000
75%	2016.000000	9.000000	3819.500000	690.000000	161.000000	17.565000	212.920000	2.090000	263.75000
max	2020.000000	12.000000	21977.000000	6377.000000	1792.070000	717.940000	4091.270000	80.560000	1885.47000

In [35]:

```
#level up and calculate the Delay in (min/flight) for each airport and carrier
#add a new column in the dataframe name 'aircraft_delay(min/flight)'
df_delay_causes
df_delay_causes['aircraft_delay(min/flight)'] = df_delay_causes['late_aircraft_delay']
/df_delay_causes['late_aircraft_ct']/df_delay_causes['arr_flights']
df_delay_causes.drop(['late_aircraft_ct'],axis = 1, inplace = True)
df_delay_causes.drop(['late_aircraft_delay'],axis = 1, inplace = True)
print(df_delay_causes.shape)
df_delay_causes.head(2)
```

```
(16903, 21)
```

Out[35]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	carrier_ct	weather_ct	nas_ct	security_ct	ar
0	2004	1	DL	Delta Air Lines Inc.	SLC	Salt Lake City, UT: Salt Lake City International	3012.0	756.0	163.51	33.04	427.21	1.60	
1	2004	1	EV	Atlantic Southeast Airlines	DFW	Dallas/Fort Worth, TX: Dallas/Fort Worth Inter...	4322.0	582.0	152.53	91.84	270.74	4.89	

In [36]:

```
#summary of the dataframe
df_delay_causes.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16903 entries, 0 to 16902
Data columns (total 21 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   year                        16903 non-null  int64
 1   month                       16903 non-null  int64
 2   carrier                     16903 non-null  object
 3   carrier name                16903 non-null  object
```

```
 4   AIRPORT                     16903 non-null  object
 5   airport_name                16903 non-null  object
 6   arr_flights                 16903 non-null  float64
 7   arr_del15                   16903 non-null  float64
 8   carrier_ct                  16903 non-null  float64
 9   weather_ct                  16903 non-null  float64
 10  nas_ct                      16903 non-null  float64
 11  security_ct                 16903 non-null  float64
 12  arr_cancelled               16903 non-null  float64
 13  arr_diverted                16903 non-null  float64
 14  arr_delay                   16903 non-null  float64
 15  carrier_delay               16903 non-null  float64
 16  weather_delay               16903 non-null  float64
 17  nas_delay                   16903 non-null  float64
 18  security_delay              16903 non-null  float64
 19  def_avg_Delay               16903 non-null  float64
 20  aircraft_delay(min/flight)  16903 non-null  float64
dtypes: float64(15), int64(2), object(4)
memory usage: 2.7+ MB
```

In [37]:

```
#level up and calculate the Delay in (min/flight) for each airport and carrier
#add a new column in the dataframe name 'nas_delay(min/flight)'
df_nas_delay_causes = df_delay_causes.copy()
df_nas_delay_causes
df_nas_delay_causes['nas_delay(min/flight)'] = df_nas_delay_causes['nas_delay']
/df_nas_delay_causes['nas_ct']/df_delay_causes['arr_flights']
df_nas_delay_causes.drop(['nas_ct'],axis = 1, inplace = True)
df_nas_delay_causes.drop(['nas_delay'],axis = 1, inplace = True)
print(df_nas_delay_causes.shape)

df_nas_delay_causes.head(2)
```

```
(16903, 20)
```

Out[37]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	carrier_ct	weather_ct	security_ct	arr_cancel
0	2004	1	DL	Delta Air Lines Inc.	SLC	Salt Lake City, UT: Salt Lake City International	3012.0	756.0	163.51	33.04	1.60	3
1	2004	1	EV	Atlantic Southeast Airlines	DFW	Dallas/Fort Worth, TX: Dallas/Fort Worth Inter...	4322.0	582.0	152.53	91.84	4.89	12

In [38]:

```
#level up and calculate the Delay in (min/flight) for each airport and carrier
#add a new column in the dataframe name 'weather_delay(min/flight)'
df_weather_delay_causes = df_nas_delay_causes.copy()
df_weather_delay_causes
df_weather_delay_causes['weather_delay(min/flight)'] =
df_weather_delay_causes['weather_delay']
/df_weather_delay_causes['weather_ct']/df_delay_causes['arr_flights']
df_weather_delay_causes.drop(['weather_ct'],axis = 1, inplace = True)
df_weather_delay_causes.drop(['weather_delay'],axis = 1, inplace = True)
print(df_weather_delay_causes.shape)
```

```
df_weather_delay_causes.head(2)
```

```
(16903, 19)
```

Out[38]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	carrier_ct	security_ct	arr_cancelled	arr_dive
0	2004	1	DL	Delta Air Lines Inc.	SLC	Salt Lake City, UT: Salt Lake City International	3012.0	756.0	163.51	1.60	38.0	
1	2004	1	EV	Atlantic Southeast Airlines	DFW	Dallas/Fort Worth, TX: Dallas/Fort Worth Inter...	4322.0	582.0	152.53	4.89	128.0	

In [39]:

```
#level up and calculate the Delay in (min/flight) for each airport and carrier
#add a new column in the dataframe name 'carrier_delay(min/flight)'
df_carrier_delay_causes = df_weather_delay_causes.copy()
df_carrier_delay_causes
df_carrier_delay_causes['carrier_delay(min/flight)'] =
df_carrier_delay_causes['carrier_delay']
/df_carrier_delay_causes['carrier_ct']/df_delay_causes['arr_flights']
df_carrier_delay_causes.drop(['carrier_ct'],axis = 1, inplace = True)
df_carrier_delay_causes.drop(['carrier_delay'],axis = 1, inplace = True)
print(df_carrier_delay_causes.shape)

df_carrier_delay_causes.head(2)
```

```
(16903, 18)
```

Out[39]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	security_ct	arr_cancelled	arr_diverted	arr_d
0	2004	1	DL	Delta Air Lines Inc.	SLC	Salt Lake City, UT: Salt Lake City International	3012.0	756.0	1.60	38.0	1.0	31
1	2004	1	EV	Atlantic Southeast Airlines	DFW	Dallas/Fort Worth, TX: Dallas/Fort Worth Inter...	4322.0	582.0	4.89	128.0	26.0	31

In [40]:

```
#level up and calculate the Delay in (min/flight) for each airport and carrier
#add a new column in the dataframe name 'security_delay(min/flight)'
df_security_delay_causes = df_carrier_delay_causes.copy()
df_security_delay_causes
df_security_delay_causes['security_delay(min/flight)'] =
df_carrier_delay_causes['security_delay']
/df_carrier_delay_causes['security_ct']/df_delay_causes['arr_flights']
df_security_delay_causes.drop(['security_ct'],axis = 1, inplace = True)
df_security_delay_causes.drop(['security_delay'],axis = 1, inplace = True)
print(df security delay causes.shape)
```

```
df_security_delay_causes.head(2)
```

```
(16903, 17)
```

Out[40]:

	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_av
0	2004	1	DL	Delta Air Lines Inc.	SLC	Salt Lake City, UT: Salt Lake City International	3012.0	756.0	38.0	1.0	31310.0	537
1	2004	1	EV	Atlantic Southeast Airlines	DFW	Dallas/Fort Worth, TX: Dallas/Fort Worth Inter...	4322.0	582.0	128.0	26.0	31531.0	145

In [41]:

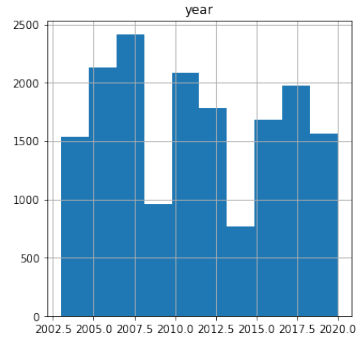
```
#create df_delay_main , and check its top 2 rows
df_delay_main =df_security_delay_causes.copy()
df_delay_main.head(2)
```

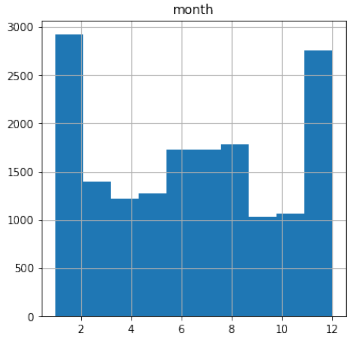
Out[41]:

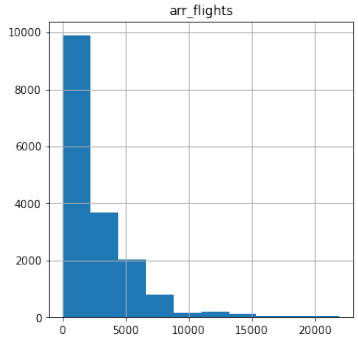
	year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_av
0	2004	1	DL	Delta Air Lines Inc.	SLC	Salt Lake City, UT: Salt Lake City International	3012.0	756.0	38.0	1.0	31310.0	537
1	2004	1	EV	Atlantic Southeast Airlines	DFW	Dallas/Fort Worth, TX: Dallas/Fort Worth Inter...	4322.0	582.0	128.0	26.0	31531.0	145

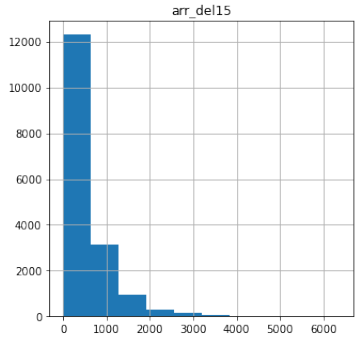
In [42]:

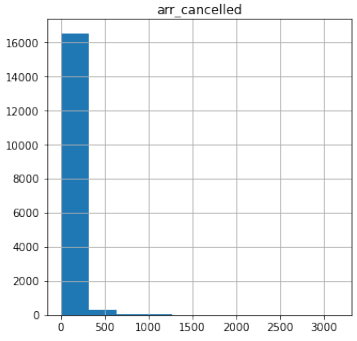
```
#Explore what the histogram of the data looks like after level up data
df_delay_main.hist(figsize=(25,25));
```

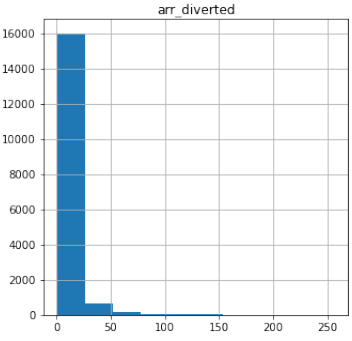


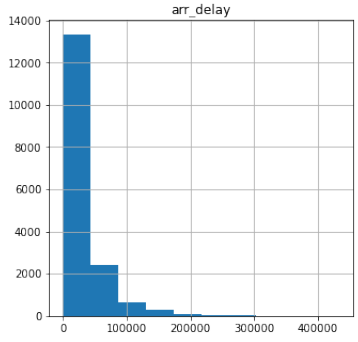


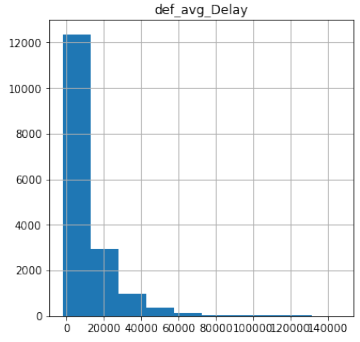


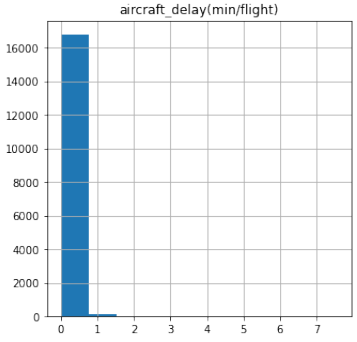


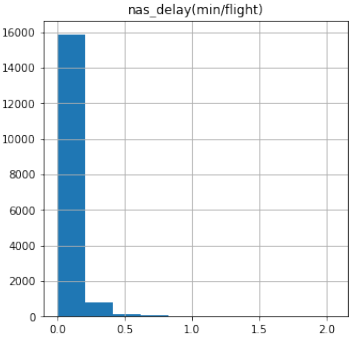


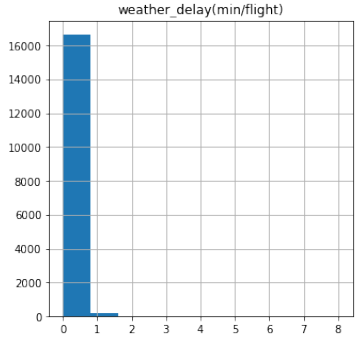


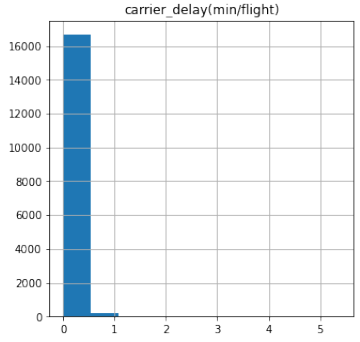


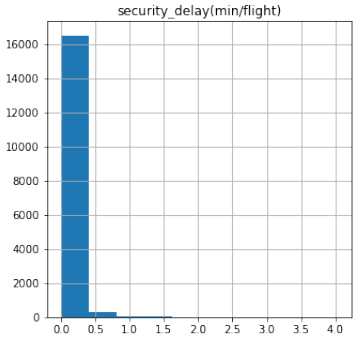












In [43]:

```
#create delay_main csv file ,Read the csv file, and check its top 2 rows

df_delay_main.to_csv (r'C:\Users\Abdelrazek\PycharmProjects\Flight\delay_main.csv', index
= False, header=True)
df_delay_main = pd.read_csv('delay_main.csv')
df_delay_main.head(2)
```

Out[43]:

year	month	carrier	carrier_name	AIRPORT	airport_name	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_av

0	2004	1	DL	Delta Air Lines Inc.	SLC	Salt Lake City, UT: Salt Lake City International	3012.0	756.0	38.0	1.0	31310.0	537
1	2004	1	EV	Atlantic Southeast Airlines	DFW	Dallas/Fort Worth, TX: Dallas/Fort Worth Inter...	4322.0	582.0	128.0	26.0	31531.0	145

In [44]:

```
# count carrier_name
df_delay_main.carrier_name.value_counts()
```

Out[44]:

```
Southwest Airlines Co.          4567
American Airlines Inc.          2168
SkyWest Airlines Inc.           1399
ExpressJet Airlines Inc.        1386
JetBlue Airways                 1065
US Airways Inc.                  978
Delta Air Lines Inc.             655
Alaska Airlines Inc.             645
Continental Air Lines Inc.       638
United Air Lines Inc.            410
Northwest Airlines Inc.          409
American Eagle Airlines Inc.     382
Mesa Airlines Inc.               323
Spirit Air Lines                 290
Comair Inc.                      240
Envoy Air                        238
Atlantic Southeast Airlines      223
PSA Airlines Inc.                152
Republic Airline                 139
Pinnacle Airlines Inc.           133
Virgin America                   108
America West Airlines Inc.        89
Hawaiian Airlines Inc.            72
Allegiant Air                     63
Atlantic Coast Airlines           37
Endeavor Air Inc.                 36
Independence Air                  22
ATA Airlines d/b/a ATA            19
Frontier Airlines Inc.            15
Aloha Airlines Inc.                2
Name: carrier name, dtype: int64
```

In [45]:

```
#group data by carrier,year and month for each carrier

df_delay=df_delay_main.copy()
df_delay=df_delay.groupby(['carrier_name','year','month']).sum()
df_delay
```

Out[45]:

			arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)	n
carrier_name	**year**	**month**								
ATA Airlines d/b/a ATA	**2003**	**6**	2139.0	262.0	20.0	1.0	12309.0	3443.021712	0.028037	
		7	2203.0	492.0	24.0	5.0	35539.0	11074.021712	0.038449	

		8	2215.0	369.0	19.0	2.0	24233.0	7866.021712	0.035180
		9	2245.0	193.0	32.0	1.0	10265.0	2488.021712	0.027951
		11	2627.0	589.0	52.0	13.0	46136.0	12092.043424	0.428060
...	**...**	**...**	...	...	...	...	...	...	...
Virgin America	**2017**	**11**	3317.0	785.0	51.0	34.0	41625.0	6216.065136	0.929567
		12	3317.0	541.0	13.0	26.0	26605.0	6072.043424	0.077100
	2018	**1**	3081.0	510.0	55.0	5.0	26490.0	2567.043424	0.085732
		2	1735.0	341.0	10.0	39.0	16835.0	2922.021712	0.038433
		3	3831.0	1019.0	183.0	13.0	62723.0	10966.065136	0.250225

2653 rows × 11 columns

In [46]:

```
#group data by carrier
df_carrier_delay=df_delay_main.copy()
carrier_delay=df_carrier_delay.groupby(['carrier_name']).sum()
print(carrier_delay.shape)
carrier_delay.head(2)
```

```
(30, 13)
```

Out[46]:

	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)
carrier_name									
ATA Airlines d/b/a ATA	38076	119	31878.0	6443.0	437.0	50.0	411255.0	1.374684e+05	2.002676
Alaska Airlines Inc.	1297217	4269	1197926.0	213502.0	15385.0	2446.0	10731444.0	3.618192e+06	91.927932

In [47]:

```
#create usa_carrier csv file
carrier_delay.to_csv (r'C:\Users\Abdelrazek\PycharmProjects\Flight\carrier_delay.csv',
index = True, header=True)
carrier_delay_1 = pd.read_csv('carrier_delay.csv')
carrier_delay_1.head(2)
```

Out[47]:

	carrier_name	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flig
0	ATA Airlines d/b/a ATA	38076	119	31878.0	6443.0	437.0	50.0	411255.0	1.374684e+05	2.002
1	Alaska Airlines Inc.	1297217	4269	1197926.0	213502.0	15385.0	2446.0	10731444.0	3.618192e+06	91.927

In [48]:

```
#reindex dataframe
carrier_delay_df=carrier_delay_1.reindex
carrier_delay_df=carrier_delay_1.set_index('carrier_name')
carrier_delay_df.head()
```

Out[48]:

carrier_name	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)
ATA Airlines d/b/a ATA	38076	119	31878.0	6443.0	437.0	50.0	411255.0	1.374684e+05	2.002676
Alaska Airlines Inc.	1297217	4269	1197926.0	213502.0	15385.0	2446.0	10731444.0	3.618192e+06	91.927932
Allegiant Air	127181	326	36854.0	9207.0	1239.0	142.0	725316.0	2.342964e+05	12.643597
Aloha Airlines Inc.	4015	3	3066.0	167.0	22.0	2.0	6207.0	-1.189566e+02	0.050396
America West Airlines Inc.	178371	611	245686.0	39735.0	3494.0	284.0	1883085.0	4.536789e+05	7.736412

In [49]:

```
#reduce arrival flight Dividing Values by 1000
carrier_delay_df2=carrier_delay_1.copy()
carrier_delay_df2['arr_flights'] = (carrier_delay_df2['arr_flights'] / 1000)
carrier_delay_df2.head()
```

Out[49]:

	carrier_name	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flig
0	ATA Airlines d/b/a ATA	38076	119	31.878	6443.0	437.0	50.0	411255.0	1.374684e+05	2.002
1	Alaska Airlines Inc.	1297217	4269	1197.926	213502.0	15385.0	2446.0	10731444.0	3.618192e+06	91.927
2	Allegiant Air	127181	326	36.854	9207.0	1239.0	142.0	725316.0	2.342964e+05	12.643
3	Aloha Airlines Inc.	4015	3	3.066	167.0	22.0	2.0	6207.0	-1.189566e+02	0.050
4	America West Airlines Inc.	178371	611	245.686	39735.0	3494.0	284.0	1883085.0	4.536789e+05	7.736

In [50]:

```
carrier_delay_df2.columns
```

Out[50]:

```
Index(['carrier_name', 'year', 'month', 'arr_flights', 'arr_del15',
       'arr_cancelled', 'arr_diverted', 'arr_delay', 'def_avg_Delay',
       'aircraft_delay(min/flight)', 'nas_delay(min/flight)',
```

```
       'weather_delay(min/flight)', 'carrier_delay(min/flight)',
       'security_delay(min/flight)'],
      dtype='object')
```

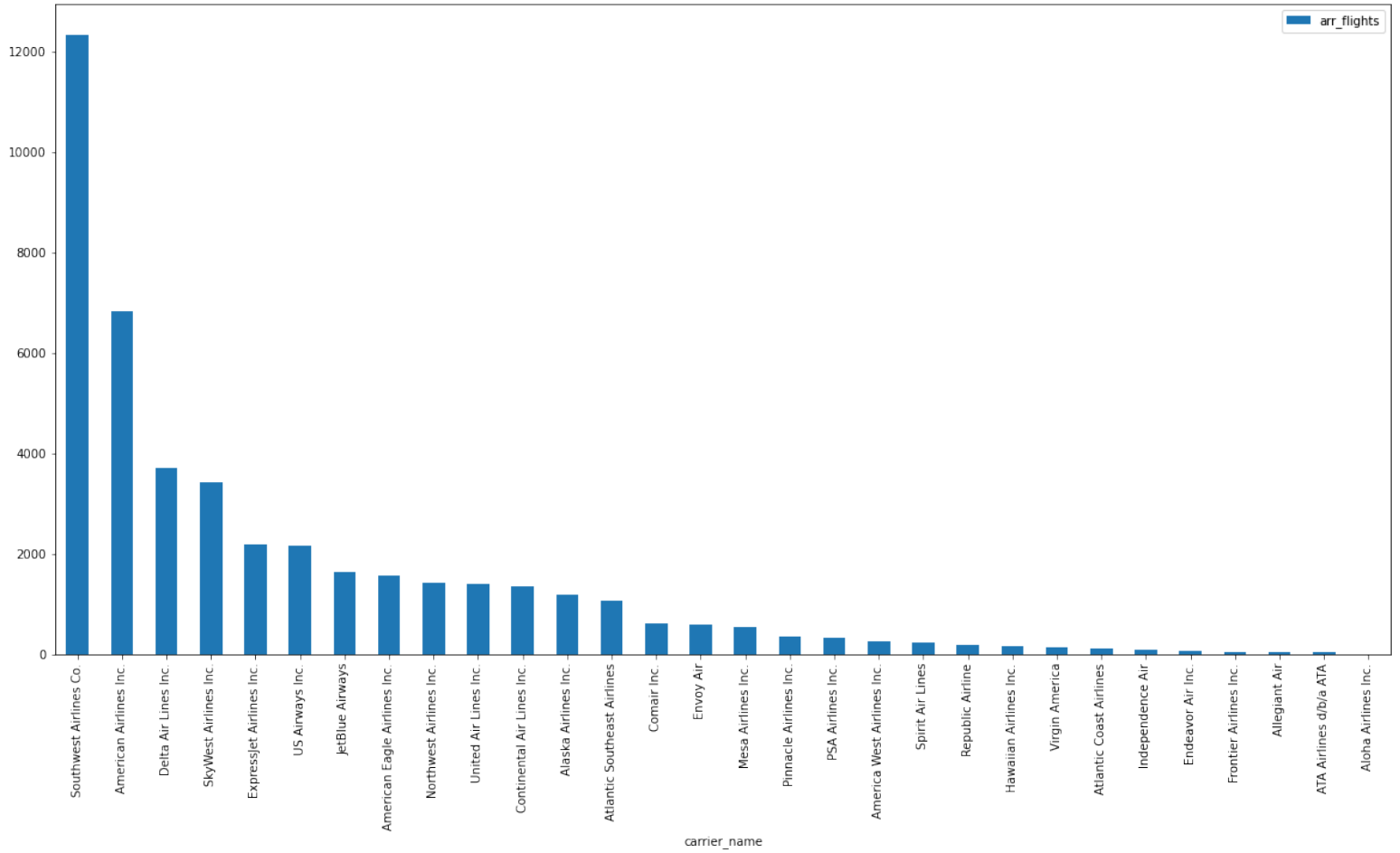
1.Analysis Airline Layer

What is the most carrier agent makeing flights?

In [51]:

```
#get the most carrier agent make flights
most_aircraft_flights=carrier_delay_df2.sort_values(by='arr_flights',ascending=False )
most_aircraft_flights.plot(kind="bar",x='carrier_name',y='arr_flights',figsize=(20,10)),sort

print('The Mean of Arrival Flight/1000')
print(most_aircraft_flights['arr_flights'].mean())
print('Southwest Airlines Co is the most carrier agent makeing flights')
plt.show()
```

```
The Mean of Arrival Flight/1000
1470.2529333333332
Southwest Airlines Co is the most carrier agent makeing flights
```

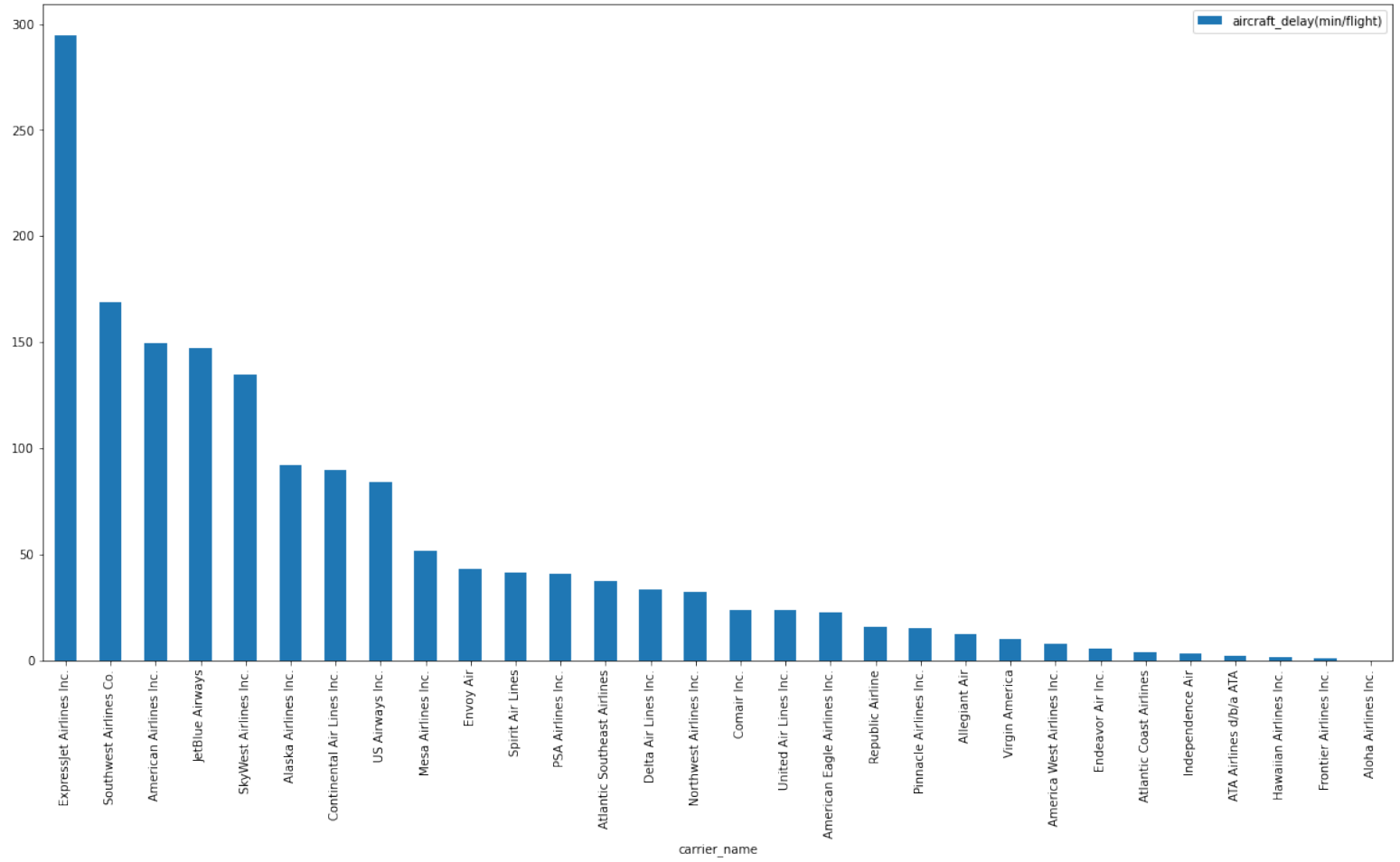


What is the most carrier agent makeing delay ?

In [52]:

```
#get the most carrier agent make delay
most_aircraft_delay=carrier_delay_df2.sort_values(by='aircraft_delay(min/flight)',ascending=
 )
most_aircraft_delay.plot(kind="bar",x='carrier_name',y='aircraft_delay(min/flight)',figsize=

print('The Mean of Aircraft Delay(min/flight)')
print(most_aircraft_delay['aircraft_delay(min/flight)'].mean())
print('ExpressJet Airlines Inc is the most carrier agent makeing Delay')

plt.show()
```

```
The Mean of Aircraft Delay(min/flight)
53.01640733771036
ExpressJet Airlines Inc is the most carrier agent makeing Delay
```

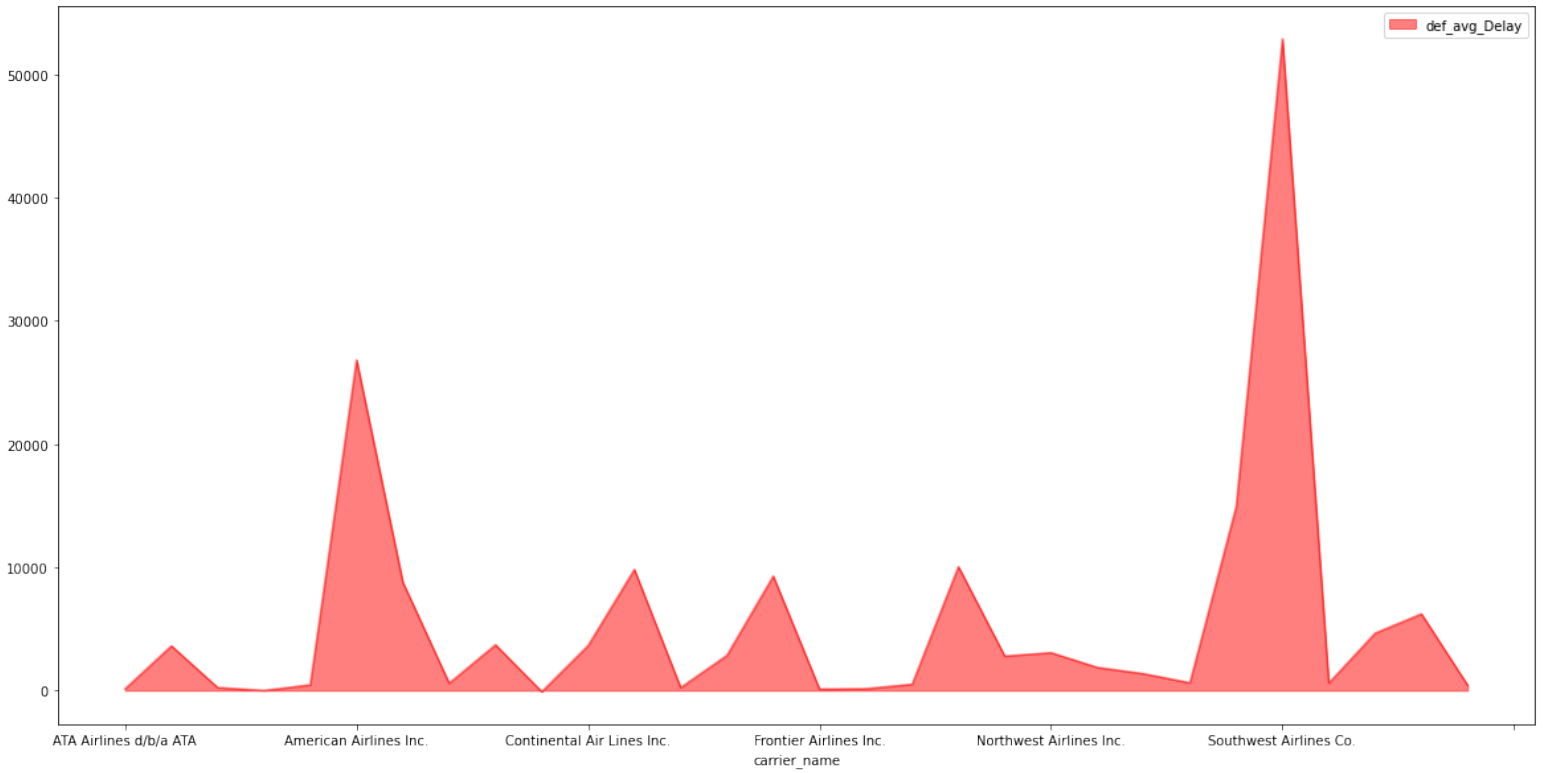


What is carrier agent make delay more than def avg Delay?

In [53]:

```
#get the most carrier agent make delay more than def_avg_Delay
carrier_delay_df2['def_avg_Delay'] = (carrier_delay_df2['def_avg_Delay'] / 1000)
carrier_delay_df2.plot(kind='area',x='carrier_name',y='def_avg_Delay',
color='red',stacked=False ,figsize=(20,10)),sorted
print('The Mean of def_avg_Delay/1000')
print(most_aircraft_delay['def_avg_Delay'].mean())

plt.show()
```

```
The Mean of def_avg_Delay/1000
5675511.9000122715
```



In [54]:

```
#group data by airports
df_airports_delay=df_delay_main.copy()
airports_delay=df_airports_delay.groupby(['AIRPORT']).sum()
print(airports_delay.shape)
airports_delay.head(2)
```

```
(254, 13)
```

Out[54]:

AIRPORT	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)	nas_
ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684	
ABI	10075	33	1026.0	224.0	20.0	5.0	11507.0	-3153.891439	1.430998	

In [55]:

```
#create usa_airports csv file
airports_delay.to_csv (r'C:\Users\Abdelrazek\PycharmProjects\Flight\airports_delay.csv',
index = True, header=True)
airports_delay = pd.read_csv('airports_delay.csv')
airports_delay.head(2)
```

Out[55]:

	AIRPORT	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)	n
0	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684	

1	ABI	10075	33	1026.0	224.0	20.0	5.0	11507.0	-3153.891439	1.430998

In [56]:

```
airports_delay.reindex
```

Out[56]:

```
<bound method DataFrame.reindex of     AIRPORT    year  month  arr_flights  arr_del15  arr_c
ancelled  \
0       ABE    6029     15        391.0      119.0           12.0
1       ABI   10075     33       1026.0      224.0           20.0
2       ABQ  221093    743     140850.0    26747.0          781.0
3       ACK    6028     24        251.0       84.0           24.0
4       ACT    2018      7        146.0       35.0           10.0
..      ...     ...    ...          ...        ...            ...
249     VLD    2007      7         87.0       33.0            3.0
250     VPS   22125     95       1721.0      513.0           56.0
251     WRG    6026     21        186.0       46.0           10.0
252     XNA   48256    148       6225.0     1528.0          225.0
253     XWA    2019     12         79.0       17.0            1.0

     arr_diverted  arr_delay  def_avg_Delay  aircraft_delay(min/flight)  \
0             3.0     8453.0   -2605.934864                    1.953684
1             5.0    11507.0   -3153.891439                    1.430998
2           347.0  1214923.0  530594.388336                    8.682357
3             6.0     5261.0   -3740.934864                    2.461677
4             1.0     1688.0    -906.978288                    0.314142
..            ...        ...            ...                         ...
249           1.0     2157.0   -1316.978288                    0.536398
250          18.0    29499.0   -8180.761166                    6.237859
251           6.0     2399.0   -3518.934864                    2.247519
252          43.0    87985.0   -6303.478909                    8.160817
253           5.0     1701.0    -606.978288                    1.777319

     nas_delay(min/flight)  weather_delay(min/flight)  \
0                 1.895971                   2.112518
1                 1.039528                   2.207824
2                 5.353606                  10.648411
3                 2.770015                   4.746400
4                 0.235193                   0.911444
..                     ...                        ...
249               0.731452                   0.752459
250               3.339884                   6.542708
251               2.037139                   2.370968
252               5.002513                  10.286498
253               0.449459                   1.338156

     carrier_delay(min/flight)  security_delay(min/flight)
0                     1.573425                    0.950482
1                     1.106756                    1.220141
2                     7.687052                    6.416964
3                     2.687697                    3.259593
4                     0.354659                    0.353513
..                         ...                         ...
249                   0.842673                    0.433390
250                   6.983359                    4.257582
251                   3.587611                    1.597255
252                   6.925848                    3.949613
253                   1.140559                    0.240346

[254 rows x 14 columns]>
```

In [57]:

```
#join  df_airports and df_airports to get airports delay with LATITUDE and LONGITUDE
df_geo=pd.merge(airports_delay ,usa_airports_df_1 ,how ='left',on='AIRPORT')
```

```
df_geo
```

Out[57]:

	AIRPORT	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)
0	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684
1	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684
2	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684
3	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684
4	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684
...	...	...	...	...	...	...	...	...	...	...
892	XNA	48256	148	6225.0	1528.0	225.0	43.0	87985.0	-6303.478909	8.160817
893	XNA	48256	148	6225.0	1528.0	225.0	43.0	87985.0	-6303.478909	8.160817
894	XNA	48256	148	6225.0	1528.0	225.0	43.0	87985.0	-6303.478909	8.160817
895	XWA	2019	12	79.0	17.0	1.0	5.0	1701.0	-606.978288	1.777319
896	XWA	2019	12	79.0	17.0	1.0	5.0	1701.0	-606.978288	1.777319

897 rows × 22 columns

In [58]:

```
df_geo_1=df_geo.drop_duplicates('AIRPORT')
print(df_geo_1.shape)
df_geo_1.head()
```

```
(254, 22)
```

Out[58]:

	AIRPORT	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)
0	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684
6	ABI	10075	33	1026.0	224.0	20.0	5.0	11507.0	-3153.891439	1.430998
9	ABQ	221093	743	140850.0	26747.0	781.0	347.0	1214923.0	530594.388336	8.682357
14	ACK	6028	24	251.0	84.0	24.0	6.0	5261.0	-3740.934864	2.461677
19	ACT	2018	7	146.0	35.0	10.0	1.0	1688.0	-906.978288	0.314142

In [59]:

```
#create usa_airports geo csv file
df_geo_1.to_csv (r'C:\Users\Abdelrazek\PycharmProjects\Flight\airports_geo.csv', index =
False, header=True)
df_airports_geo = pd.read_csv('airports_geo.csv')
df_airports_geo.head(2)
```

Out[59]:

	AIRPORT	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)	n
0	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684	
1	ABI	10075	33	1026.0	224.0	20.0	5.0	11507.0	-3153.891439	1.430998	

In [60]:

```
# Combining Lattitude and Longitude to create coordinates:
df_airports_geo_1=df_airports_geo.copy()
df_airports_geo_1['coordinates'] =
df_airports_geo_1[['LONGITUDE','LATITUDE']].values.tolist()
df_airports_geo_1.head()
```

Out[60]:

	AIRPORT	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)
0	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684
1	ABI	10075	33	1026.0	224.0	20.0	5.0	11507.0	-3153.891439	1.430998
2	ABQ	221093	743	140850.0	26747.0	781.0	347.0	1214923.0	530594.388336	8.682357
3	ACK	6028	24	251.0	84.0	24.0	6.0	5261.0	-3740.934864	2.461677
4	ACT	2018	7	146.0	35.0	10.0	1.0	1688.0	-906.978288	0.314142

In [61]:

```
# Change the coordinates to a geoPoint
from shapely.geometry import Point
df_airports_geo_1['coordinates'] = df_airports_geo_1['coordinates'].apply(Point)
df_airports_geo_1.head()
```

Out[61]:

	AIRPORT	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)
0	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684
1	ABI	10075	33	1026.0	224.0	20.0	5.0	11507.0	-3153.891439	1.430998
2	ABQ	221093	743	140850.0	26747.0	781.0	347.0	1214923.0	530594.388336	8.682357
3	ACK	6028	24	251.0	84.0	24.0	6.0	5261.0	-3740.934864	2.461677

4	ACT	2018	7	146.0	35.0	10.0	1.0	1688.0	-906.978288	0.314142

In [62]:

```
df_airports_geo_1.to_csv
(r'C:\Users\Abdelrazek\PycharmProjects\Flight\airports_coordinates.csv', index = False,
header=True)
geo= pd.read_csv('airports_coordinates.csv')
geo.head(2)
```

Out[62]:

	AIRPORT	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flight)	n
0	ABE	6029	15	391.0	119.0	12.0	3.0	8453.0	-2605.934864	1.953684	
1	ABI	10075	33	1026.0	224.0	20.0	5.0	11507.0	-3153.891439	1.430998	

2.Analysis Airports Layer

What is The Most US Airports had Weather Delay?

In [63]:

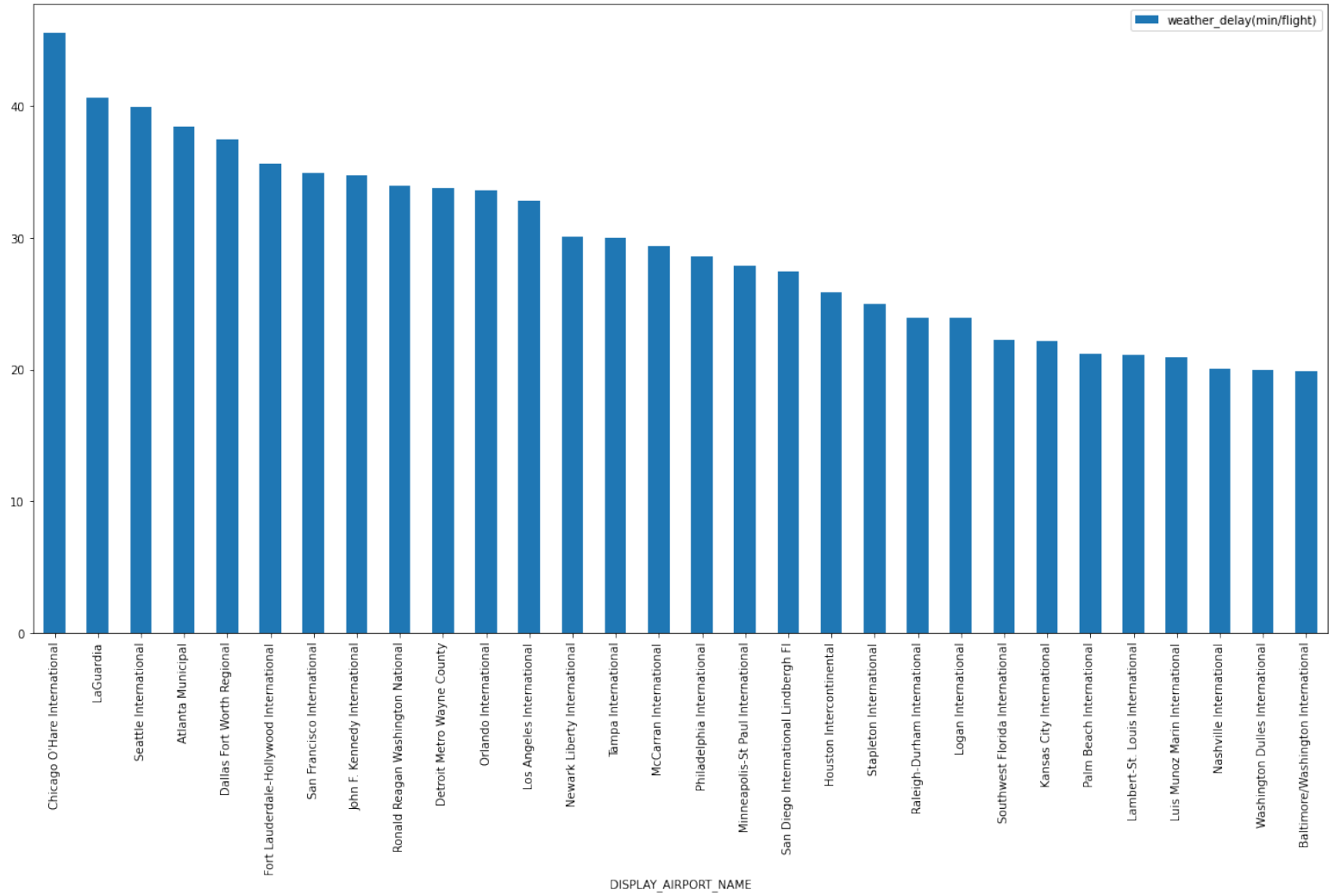
```
#us airport distribution by state and Weather Delay info
print ('US Airports distribution by state and Weather Delay info')
fig = px.scatter_mapbox(geo, lat="LATITUDE", lon="LONGITUDE",
hover_name="AIRPORT_STATE_NAME", hover_data=["AIRPORT_STATE_NAME",
"weather_delay(min/flight)"],
                        color_discrete_sequence=["fuchsia"], zoom=3, height=300)
fig.update_layout(mapbox_style="open-street-map")
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
fig.show()
```

```
US Airports distribution by state and Weather Delay info
```

In [64]:

```
#get the most airports makeing weather_delay
most_airports_delay=geo.sort_values(by='weather_delay(min/flight)',ascending=False )
most_airports_delay.head(30).plot(kind="bar",x='DISPLAY_AIRPORT_NAME',y='weather_delay(min/f

print('The Mean of Weather Delay(min/flight)')
print(geo['weather_delay(min/flight)'].mean())
print('Chicago O Hare International is the Most Airports with weather Delay')

plt.show()
```

```
The Mean of Weather Delay(min/flight)
7.83864917649772
Chicago O Hare International is the Most Airports with weather Delay
```



In [65]:

```
#using Google API Maps
import gmaps

gmaps.configure(api_key="AIzaSyAIX8emBZKPsXTbOxs_fqHfm0lIalYQ1_8")# Fill in with your API
```

key

In [66]:

```
#us airport distribution by state and Weather Delay info #using Google API Maps #map view
#print('Map of US Airports shown the most affected with Weather Delay')

locations = geo[['LATITUDE', 'LONGITUDE']]
weights = geo['weather_delay(min/flight)']
fig = gmaps.figure()
fig.add_layer(gmaps.heatmap_layer(locations, weights=weights))
fig
```

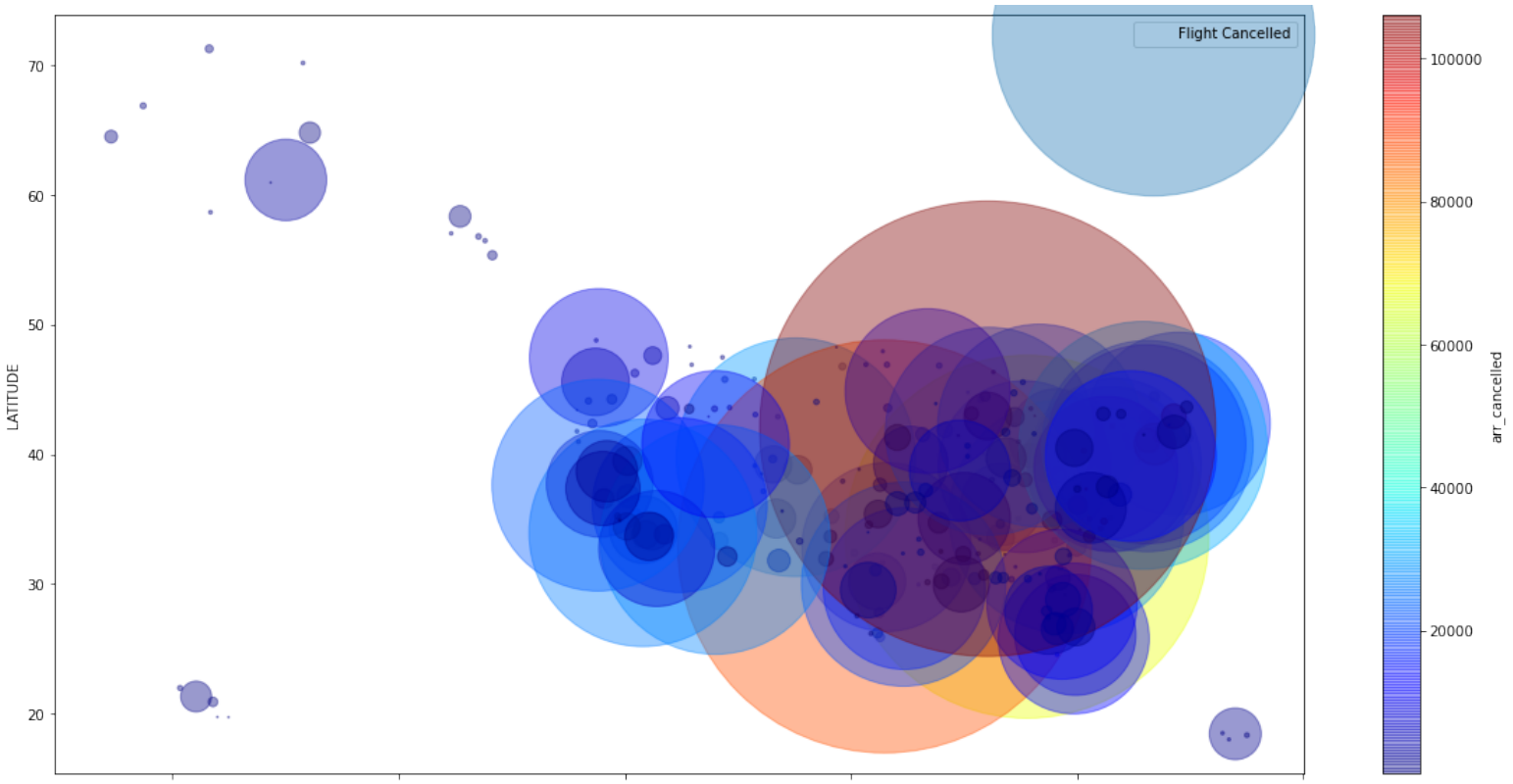
Map of US Airports shown the most affected with Weather Delay

What is Most US Airport by LONGITUDE,LATITUDE had Flight Cancelled

In [67]:

```
# What is Most US Airport by Longitude, Latitude had Flight Cancelled
print('Flight Cancelled by US Airport Longitude, Latitude')
geo.plot(kind="scatter", x="LONGITUDE", y="LATITUDE",
    s=geo['arr_cancelled'], label="Flight Cancelled",
    c="arr_cancelled", cmap=plt.get_cmap("jet"),
    colorbar=True, alpha=0.4, figsize=(20,10),
)

plt.legend()
plt.show()
```

```
Flight Cancelled by US Airport Longitude, Latitude
```

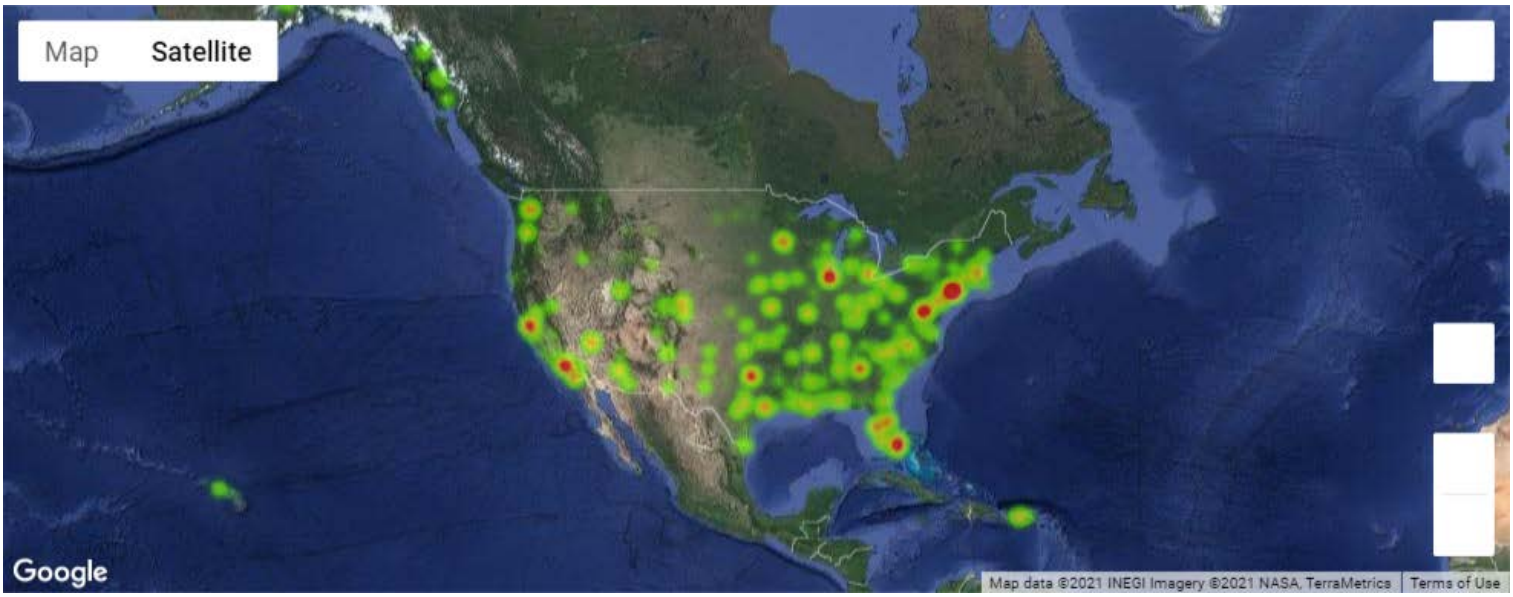


What is Most US Airport had Carrier Delay?

In [68]:

```
#us airport distribution by state and Carrier Delay info #using Google API Maps #satellite
view
#print('Map satellite view of US Airports shown the most affected with Carrier Delay')
locations = df_geo_1[['LATITUDE', 'LONGITUDE']]
weights = df_geo_1['carrier_delay(min/flight)']
fig = gmaps.figure()
fig.add_layer(gmaps.heatmap_layer(locations, weights=weights))
fig
```

Map satellite view of US Airports shown the most affected with Carrier Delay



In [69]:

```
#group data by state
df_state_delay=geo.copy()
df_state_delay=df_state_delay.groupby(['AIRPORT_STATE_NAME']).sum()
print(df_state_delay.shape)
df_state_delay.head(5)
```

```
(51, 16)
```

Out[69]:

AIRPORT_STATE_NAME	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay
Alabama	174894	533	31254.0	7324.0	477.0	195.0	392864.0	4.343589e+04	
Alaska	560842	1890	199511.0	38313.0	4092.0	763.0	1859119.0	3.999081e+05	
Arizona	1088309	3496	2170576.0	353602.0	27423.0	3200.0	17363210.0	6.942681e+06	
Arkansas	118594	322	17083.0	4308.0	457.0	110.0	238549.0	1.012628e+04	
California	4442767	14487	4451001.0	934650.0	72561.0	8304.0	50659675.0	1.911307e+07	

In [70]:

```
#create usa_state csv file
df_state_delay.to_csv (r'C:\Users\Abdelrazek\PycharmProjects\Flight\state_delay.csv',
index = True, header=True)
df_state_delay = pd.read_csv('state_delay.csv')
df_state_delay.head(2)
```

Out[70]:

	AIRPORT_STATE_NAME	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay
0	Alabama	174894	533	31254.0	7324.0	477.0	195.0	392864.0	43435.888957	
1	Alaska	560842	1890	199511.0	38313.0	4092.0	763.0	1859119.0	399908.057688	

In [71]:

```
#rename columns to apply merge

df_state_delay.rename(columns={'AIRPORT_STATE_NAME' : 'STATE_NAME'}, inplace=True)
df_state_delay.head(2)
```

Out[71]:

	STATE_NAME	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/flig
0	Alabama	174894	533	31254.0	7324.0	477.0	195.0	392864.0	43435.888957	23.1766
1	Alaska	560842	1890	199511.0	38313.0	4092.0	763.0	1859119.0	399908.057688	72.540(

In [72]:

```
#join  df_state_delay and df_airports to get states delay with LATITUDE and LONGITUDE
df_state = pd.read_csv('state.csv')

df_state_geo=pd.merge(df_state_delay ,df_state ,how ='outer',on='STATE_NAME')
df_state_geo.drop(['LATITUDE'],axis = 1, inplace = True)
df_state_geo.drop(['LONGITUDE'],axis = 1, inplace = True)
print(df_state_geo.shape)

df_state_geo.head(2)
```

```
(53, 18)
```

Out[72]:

	STATE_NAME	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(min/fli
0	Alabama	174894.0	533.0	31254.0	7324.0	477.0	195.0	392864.0	43435.888957	23.17(
1	Alaska	560842.0	1890.0	199511.0	38313.0	4092.0	763.0	1859119.0	399908.057688	72.54

In [73]:

```
#drop NaN value
most_state_delay=df_state_geo.dropna()
most_state_delay
```

Out[73]:

	STATE_NAME	year	month	arr_flights	arr_del15	arr_cancelled	arr_diverted	arr_delay	def_avg_Delay	aircraft_delay(m
0	Alabama	174894.0	533.0	31254.0	7324.0	477.0	195.0	392864.0	4.343589e+04	2
1	Alaska	560842.0	1890.0	199511.0	38313.0	4092.0	763.0	1859119.0	3.999081e+05	1
2	Arizona	1088309.0	3496.0	2170576.0	353602.0	27423.0	3200.0	17363210.0	6.942681e+06	2
3	Arkansas	118594.0	322.0	17083.0	4308.0	457.0	110.0	238549.0	1.012628e+04	1
4	California	4442767.0	14487.0	4451001.0	934650.0	72561.0	8304.0	50659675.0	1.911307e+07	14
5	Colorado	1164865.0	3768.0	1797350.0	319220.0	30323.0	5497.0	17936969.0	7.004795e+06	4
6	Connecticut	86539.0	258.0	16982.0	4127.0	410.0	61.0	223704.0	3.575893e+04	
7	Florida	3410419.0	10469.0	2570463.0	536859.0	37296.0	7301.0	31394528.0	1.068237e+07	1

8	Georgia	1140523.0	3722.0	4222231.0	798309.0	67653.0	9168.0	48669467.0	1.367817e+07	3
9	Hawaii	185285.0	543.0	169549.0	17815.0	548.0	184.0	828966.0	1.315830e+05	
10	Idaho	82401.0	256.0	19921.0	4528.0	335.0	179.0	250032.0	5.111589e+04	
11	Illinois	1916615.0	6212.0	4270081.0	917197.0	128270.0	12608.0	60090124.0	2.148794e+07	4
12	Indiana	171057.0	476.0	50398.0	10450.0	864.0	158.0	595416.0	1.345338e+05	1
13	Iowa	72438.0	210.0	5995.0	1524.0	218.0	48.0	90311.0	-2.201422e+04	
14	Kansas	42223.0	133.0	3564.0	936.0	101.0	34.0	55877.0	-1.354854e+04	
15	Kentucky	359495.0	1182.0	557606.0	92792.0	15359.0	934.0	4901023.0	2.736379e+05	2
16	Louisiana	398159.0	1301.0	157499.0	32529.0	1925.0	619.0	1610686.0	4.687433e+05	
17	Maine	22106.0	74.0	1524.0	401.0	76.0	12.0	23660.0	-1.033076e+04	
18	Maryland	541186.0	1777.0	1042827.0	172060.0	15324.0	2440.0	9078749.0	4.169426e+06	
19	Massachusetts	714082.0	2262.0	659739.0	155397.0	17145.0	982.0	10212525.0	3.614586e+06	
20	Michigan	705975.0	2192.0	1152559.0	202037.0	21296.0	1919.0	12222041.0	3.905214e+06	3
21	Minnesota	647794.0	2040.0	1117503.0	190427.0	13896.0	2421.0	11389860.0	3.090917e+06	2
22	Mississippi	98459.0	338.0	8400.0	1987.0	202.0	65.0	111258.0	-3.775294e+04	2
23	Missouri	886679.0	2821.0	712665.0	134684.0	8132.0	1824.0	6808912.0	2.833484e+06	
24	Montana	28144.0	110.0	3288.0	799.0	43.0	30.0	42324.0	-8.891696e+03	
25	Nebraska	102558.0	303.0	16729.0	4258.0	348.0	89.0	235498.0	1.698611e+04	
26	Nevada	1027822.0	3305.0	1609266.0	299088.0	16018.0	2436.0	14520866.0	6.599594e+06	2
27	New Hampshire	78317.0	287.0	23622.0	4826.0	309.0	70.0	256190.0	7.956685e+04	
28	New Jersey	745784.0	2421.0	1070273.0	321158.0	31446.0	4354.0	22563502.0	4.660431e+06	2
29	New Mexico	229153.0	757.0	141198.0	26830.0	804.0	353.0	1219452.0	5.251915e+05	
30	New York	2079743.0	6474.0	1580809.0	394541.0	44946.0	6771.0	26001815.0	6.655109e+06	10
31	North Carolina	971823.0	3109.0	1626602.0	261474.0	30926.0	3122.0	14421908.0	5.131910e+06	5
32	North Dakota	20135.0	60.0	1530.0	320.0	53.0	22.0	19386.0	-9.414783e+03	
33	Ohio	532446.0	1650.0	388744.0	75811.0	5777.0	860.0	4294131.0	1.730893e+06	3
34	Oklahoma	229080.0	747.0	43393.0	10266.0	709.0	251.0	518441.0	7.108248e+04	2
35	Oregon	354008.0	1162.0	167158.0	30683.0	2443.0	375.0	1531785.0	5.241928e+05	1
36	Pennsylvania	764536.0	2397.0	763387.0	158507.0	15639.0	2103.0	9448856.0	2.468642e+06	3
37	Puerto Rico	287616.0	913.0	115146.0	28097.0	1383.0	278.0	1687954.0	2.934551e+05	2
38	Rhode Island	114566.0	343.0	40535.0	8304.0	572.0	118.0	459189.0	1.424922e+05	
39	South Carolina	156770.0	526.0	19463.0	4651.0	543.0	100.0	287043.0	-2.569031e+04	2
40	South Dakota	14097.0	38.0	1508.0	282.0	48.0	10.0	19082.0	-5.740848e+03	
41	Tennessee	767830.0	2473.0	654056.0	114742.0	7843.0	1370.0	5928128.0	2.052750e+06	1

42	Texas	3610182.0	11508.0	7529551.0	1360285.0	140712.0	30186.0	79203635.0	3.161296e+07	1.
44	Utah	577063.0	1833.0	988687.0	139308.0	11083.0	1464.0	7390630.0	2.680912e+06	
45	Vermont	18084.0	67.0	1302.0	306.0	42.0	10.0	18658.0	-7.302805e+03	
46	Virginia	1166377.0	3736.0	836086.0	163144.0	24974.0	2363.0	10025312.0	3.544582e+06	6
47	Washington	828721.0	2733.0	1010843.0	182730.0	10002.0	1746.0	9159378.0	3.343476e+06	2
48	West Virginia	28172.0	89.0	2910.0	667.0	108.0	28.0	41359.0	-5.266696e+03	
49	Wisconsin	211199.0	583.0	63855.0	14228.0	1329.0	263.0	812700.0	2.151153e+05	
50	Wyoming	10058.0	16.0	821.0	197.0	31.0	8.0	11651.0	-4.358891e+03	

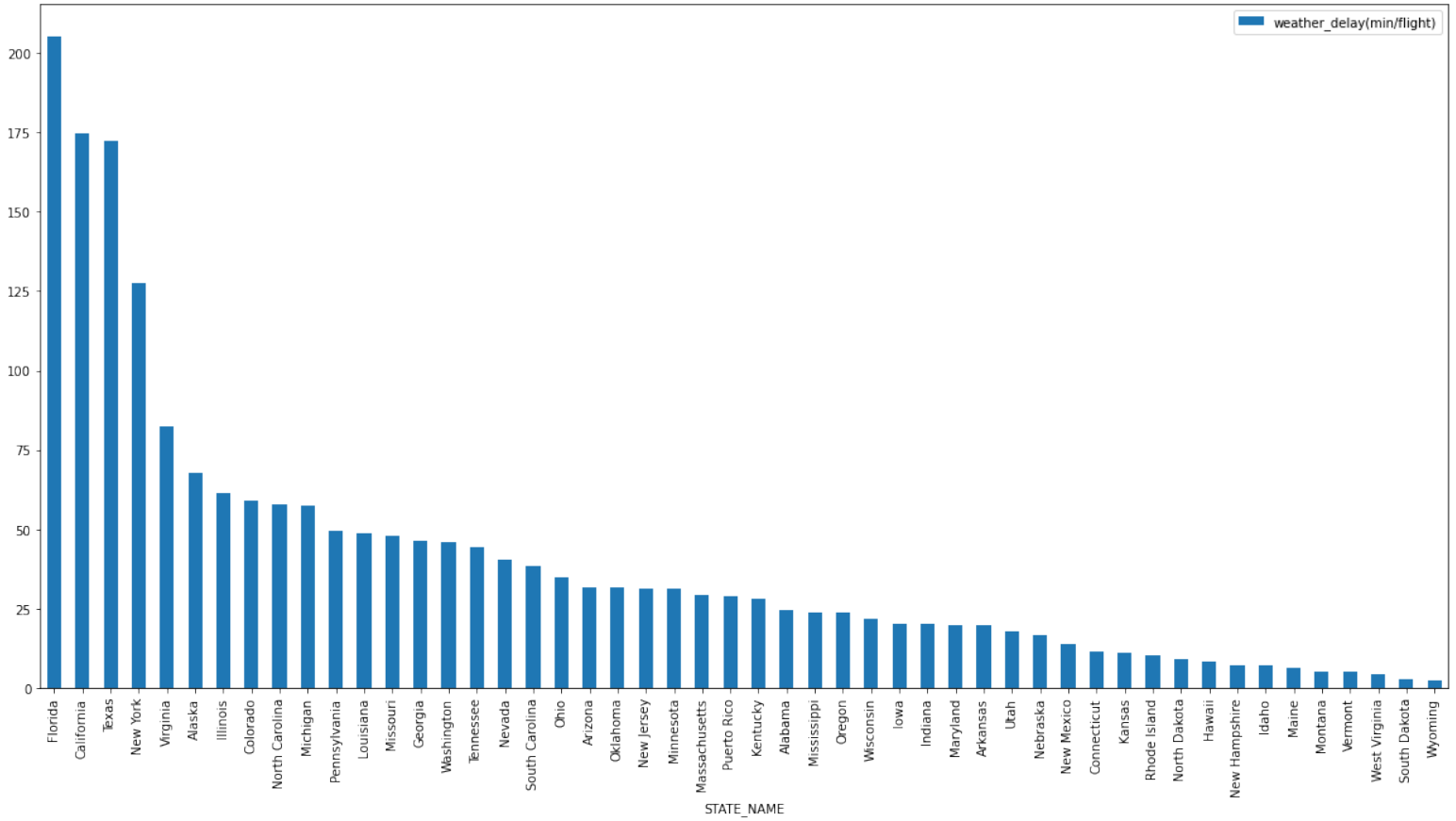
3.Analysis State Layer

What is The Most US State had Weather Delay?

In [74]:

```
#get the most state makeing weather_delay

most_state_delay=most_state_delay.sort_values(by='weather_delay(min/flight)',ascending=False
 )
most_state_delay.plot(kind="bar",x='STATE_NAME',y='weather_delay(min/flight)',figsize=(20,10

print('The Mean of Weather Delay(min/flight) by US State ')
print(most_state_delay['weather_delay(min/flight)'].mean())
print('Florida,California,Texas is the Most US State affected by Weather Delay')
plt.show()
```

```
The Mean of Weather Delay(min/flight) by US State
39.75564591352922
Florida,California,Texas is the Most US State affected by Weather Delay
```



In [75]:

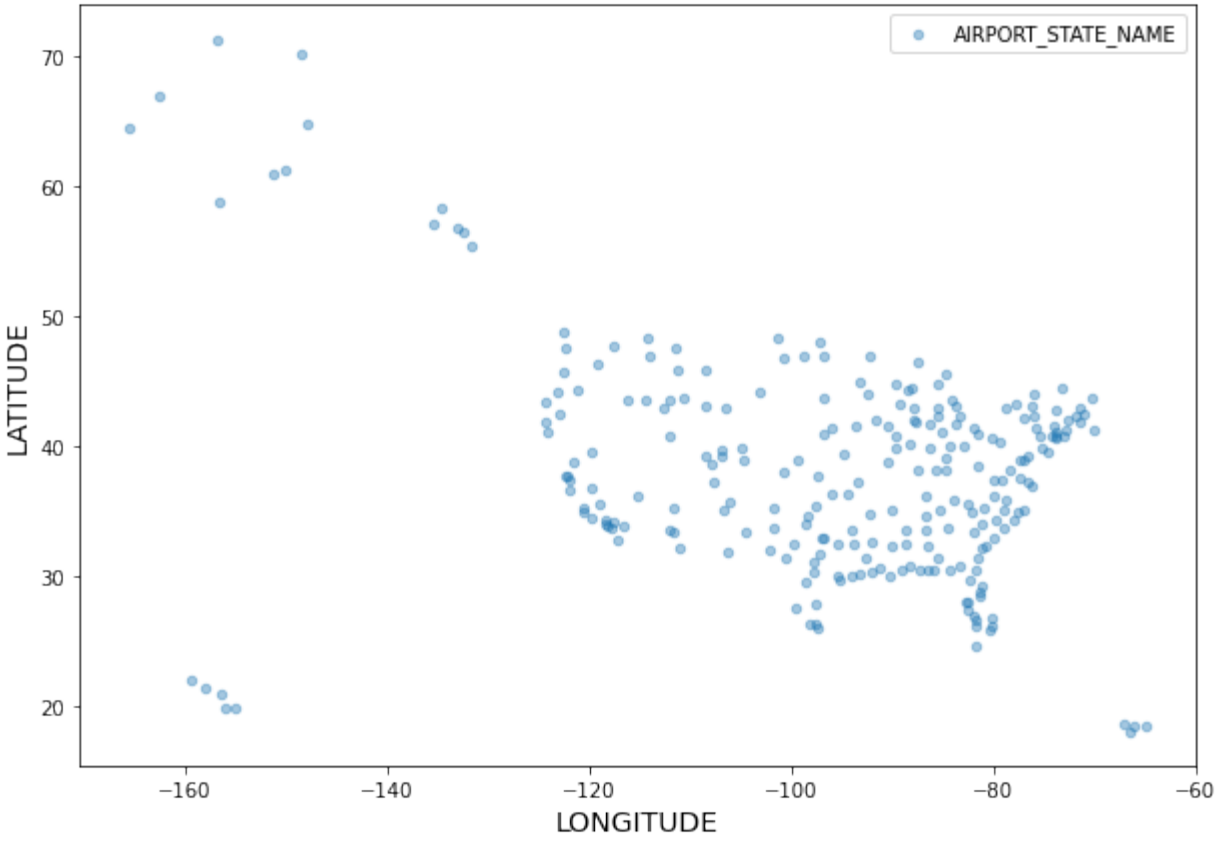
```
#us state and Weather Delay info #using Google API Maps #satellite view
#print('Map view of Most US State affected by Weather Delay')
#print('Florida,California,Texas is the Most US State affected by Weather Delay')
locations = most_state_delay[['latitude', 'longitude']]
weights = most_state_delay['weather_delay(min/flight)']
fig = gmaps.figure()
fig.add_layer(gmaps.heatmap_layer(locations, weights=weights))
fig
```

Map view of Most US State affected by Weather Delay

Florida,California,Texas is the Most US State affected by Weather Delay

In [76]:

```
ax = geo.plot(kind="scatter", x="LONGITUDE", y="LATITUDE", figsize=(10,7),colorbar=False,
alpha=0.4,)
plt.ylabel("LATITUDE", fontsize=14)
plt.xlabel("LONGITUDE", fontsize=14)

state = geo["AIRPORT_STATE_NAME"]


plt.legend(["AIRPORT_STATE_NAME"])
plt.show()
```



4. Data Visualization

Project 3.Communicate Data Findings

In [ ]:

```
!jupyter nbconvert airports.ipynb --to html
```

In [ ]:

```
!jupyter nbconvert airports.ipynb --to slides --post serve --no-input --no-prompt
```

In [ ]:

```
#!jupyter nbconvert airports.html --to pdf --template classic
```

In [ ]:

```
#!jupyter nbconvert airports.ipynb --to slides --reveal-prefix reveal.js --post serve
```