

```
from google.colab import files
files.upload()
```

[Choose Files](#) No file chosen

Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving airlines.csv to airlines.csv

Saving airports.csv to airports.csv

Saving flights.csv to flights.csv

Saving planes.csv to planes.csv

Saving weather.csv to weather.csv

{'airlines.csv': b'IATA_CODE,AIRLINE\nUA,United Air Lines Inc.\nAA,American Airlines

'airports.csv': b'IATA_CODE,AIRPORT,CITY,STATE,COUNTRY,LATITUDE, LONGITUDE\nABE,Lehigh Valley International Airport

'flights.csv': b'"year","month","day","dep_time","dep_delay","arr_time","arr_delay",

'planes.csv': b'tailnum,year,type,manufacturer,model,engines,seats,speed,engine\nN16819,Bombardier CRJ-900,Boeing,Boeing 737 MAX 8,3,160,440,CFM56-7B

'weather.csv': b'origin.vear.month.day.hour.temp.dewp.humid.wind_dir.wind_speed.wind_dir

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
airlines = pd.read_csv("airlines.csv")
airports = pd.read_csv("airports.csv")
flights = pd.read_csv("flights.csv")
planes = pd.read_csv("planes.csv")
weather = pd.read_csv("weather.csv")
```

how many total number of days does the flights table cover?

Answer = 365

```
flights.head(10)
```

	year	month	day	dep_time	dep_delay	arr_time	arr_delay	carrier	tailnum	flight
0	2013	1	1	517.0	2.0	830.0	11.0	UA	N14228	1545
1	2013	1	1	533.0	4.0	850.0	20.0	UA	N24211	1714
2	2013	1	1	542.0	2.0	923.0	33.0	AA	N619AA	1183
3	2013	1	1	544.0	-1.0	1004.0	-18.0	B6	N804JB	725
4	2013	1	1	554.0	-6.0	812.0	-25.0	DL	N668DN	437
5	2013	1	1	554.0	-4.0	740.0	12.0	UA	N39463	1627
6	2013	1	1	555.0	-5.0	913.0	19.0	B6	N516JB	542
7	2013	1	1	557.0	-3.0	709.0	-14.0	EV	N829AS	571
8	2013	1	1	557.0	-3.0	838.0	-8.0	B6	N593JB	542
9	2013	1	1	558.0	-2.0	753.0	8.0	AA	N3ALAA	3211

```
flights['Date'] = pd.to_datetime(flights[['year', 'month', 'day']])
flights
```

	year	month	day	dep_time	dep_delay	arr_time	arr_delay	carrier	tailnum
0	2013	1	1	517.0	2.0	830.0	11.0	UA	N14228
1	2013	1	1	533.0	4.0	850.0	20.0	UA	N24211
2	2013	1	1	542.0	2.0	923.0	33.0	AA	N619AA
3	2013	1	1	544.0	-1.0	1004.0	-18.0	B6	N804JB
4	2013	1	1	554.0	-6.0	812.0	-25.0	DL	N668DN
...
336771	2013	9	30	NaN	NaN	NaN	NaN	9E	NaN
336772	2013	9	30	NaN	NaN	NaN	NaN	9E	NaN

```
len(flights['Date'].unique())
```

365

how many departure cities (not airports) does the flights database cover?

Answer = 2 cities

```
flights['origin'].unique()
```

```
array(['EWR', 'LGA', 'JFK'], dtype=object)
```

```
airports[airports['IATA_CODE']=='JFK']
```

	IATA_CODE	AIRPORT	CITY	STATE	COUNTRY	LATITUDE	LONGITUDE
166	JFK	John F. Kennedy International Airport (New York)	New York	NY	USA	40.63975	-73.77893

```
City = []
for row in flights['origin']:
    if row == 'EWR':
        City.append('Newark')
    elif row == 'LGA':
        City.append('New York')
```

```

else:
    flights['departure cities'] = City
    flights.head(10)

```

	year	month	day	dep_time	dep_delay	arr_time	arr_delay	carrier	tailnum	flight
0	2013	1	1	517.0	2.0	830.0	11.0	UA	N14228	1545
1	2013	1	1	533.0	4.0	850.0	20.0	UA	N24211	1715
2	2013	1	1	542.0	2.0	923.0	33.0	AA	N619AA	1183
3	2013	1	1	544.0	-1.0	1004.0	-18.0	B6	N804JB	725
4	2013	1	1	554.0	-6.0	812.0	-25.0	DL	N668DN	437
5	2013	1	1	554.0	-4.0	740.0	12.0	UA	N39463	1629
6	2013	1	1	555.0	-5.0	913.0	19.0	B6	N516JB	547
7	2013	1	1	557.0	-3.0	709.0	-14.0	EV	N829AS	571
8	2013	1	1	557.0	-3.0	838.0	-8.0	B6	N593JB	547
9	2013	1	1	558.0	-2.0	753.0	8.0	AA	N3ALAA	3211



```
len(flights['departure cities'].unique())
```

2

what is the relationship between flights and planes tables?

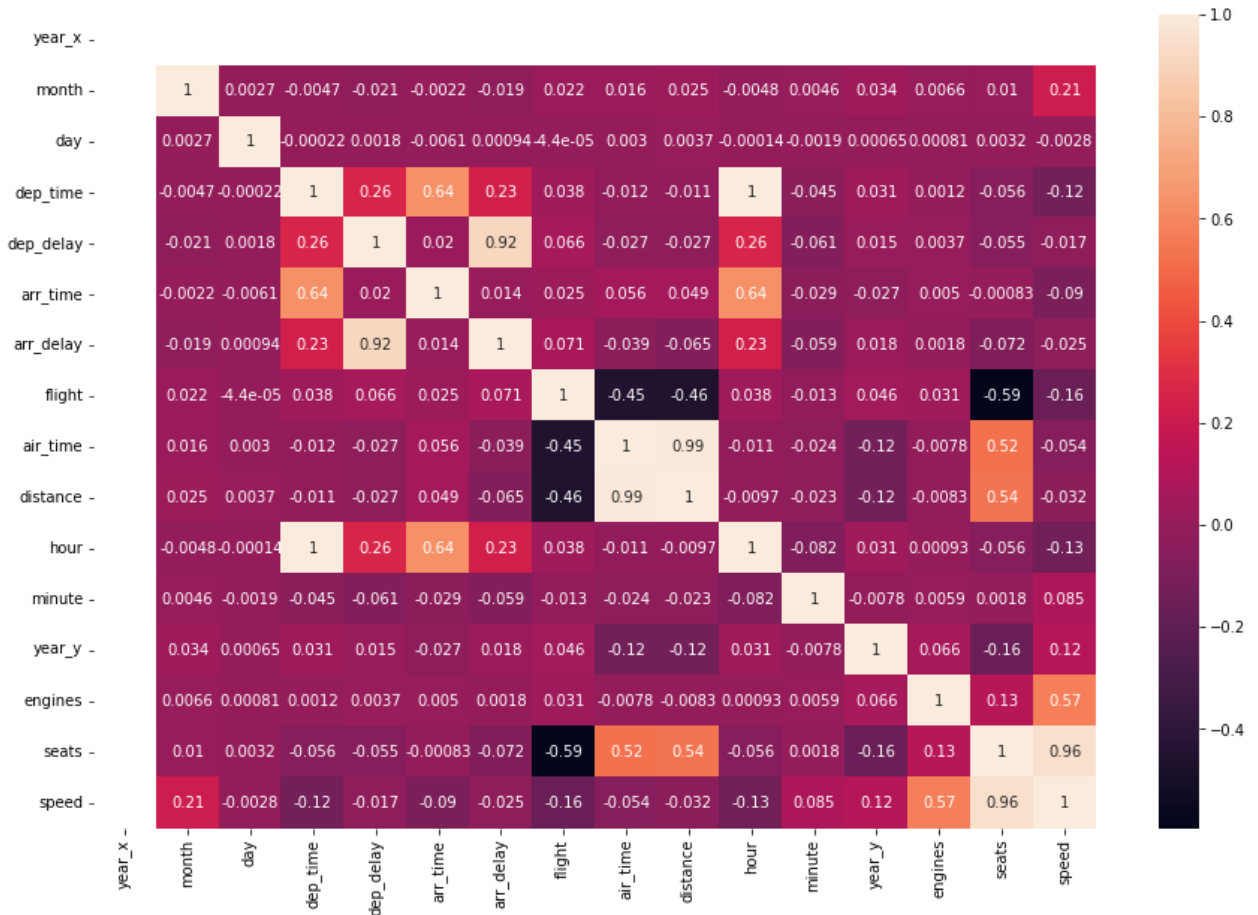
Answer:- The relationship between flights and planes tables is tailnum.

```
df1 = pd.merge(flights,planes,on='tailnum')
df1
```

	year_x	month	day	dep_time	dep_delay	arr_time	arr_delay	carrier	tail
0	2013	1	1	517.0	2.0	830.0	11.0	UA	N14
1	2013	1	8	1435.0	-5.0	1717.0	-29.0	UA	N14
2	2013	1	9	717.0	17.0	812.0	-3.0	UA	N14
3	2013	1	9	1143.0	-1.0	1425.0	-20.0	UA	N14
4	2013	1	13	835.0	11.0	1030.0	39.0	UA	N14
...
284165	2013	9	20	1758.0	-7.0	1929.0	-24.0	OO	N76
284166	2013	9	22	1759.0	-6.0	1945.0	-8.0	OO	N77
284167	2013	9	23	1759.0	-6.0	1935.0	-18.0	OO	N77
284168	2013	9	24	1751.0	-14.0	1937.0	-16.0	OO	N78
284169	2013	9	28	712.0	-8.0	955.0	-30.0	AS	N55

```
import seaborn as sns
plt.figure(figsize=(15,10))
sns.heatmap(df1.corr(), annot=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7fbb168a8390>
```



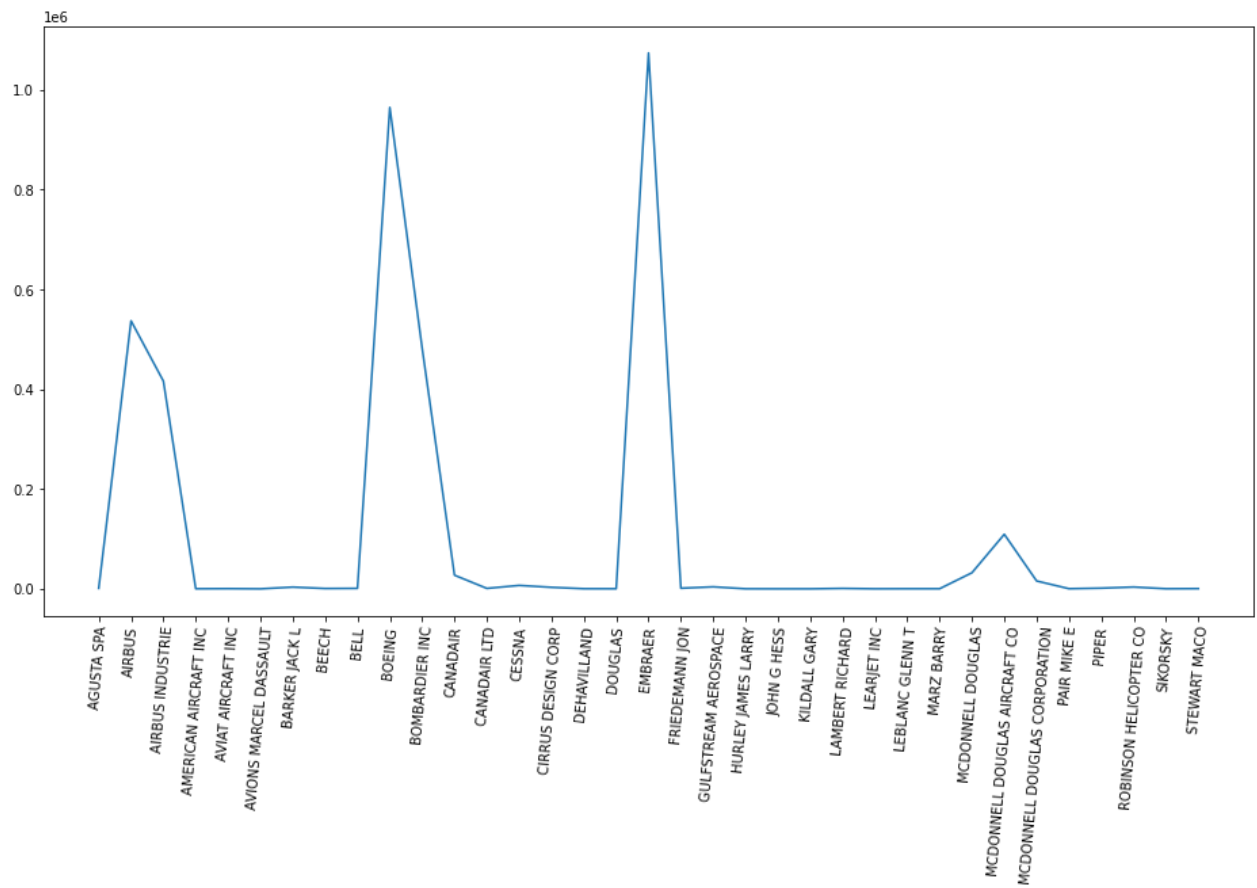
which airplane manufacturer incurred the most delays in the analysis period?

Answer:- EMBRAER

```
df1.columns
```

```
Index(['year_x', 'month', 'day', 'dep_time', 'dep_delay', 'arr_time',
      'arr_delay', 'carrier', 'tailnum', 'flight', 'origin', 'dest',
      'air_time', 'distance', 'hour', 'minute', 'Date', 'departure cities',
      'year_y', 'type', 'manufacturer', 'model', 'engines', 'seats', 'speed',
      'engine'],
      dtype='object')
```

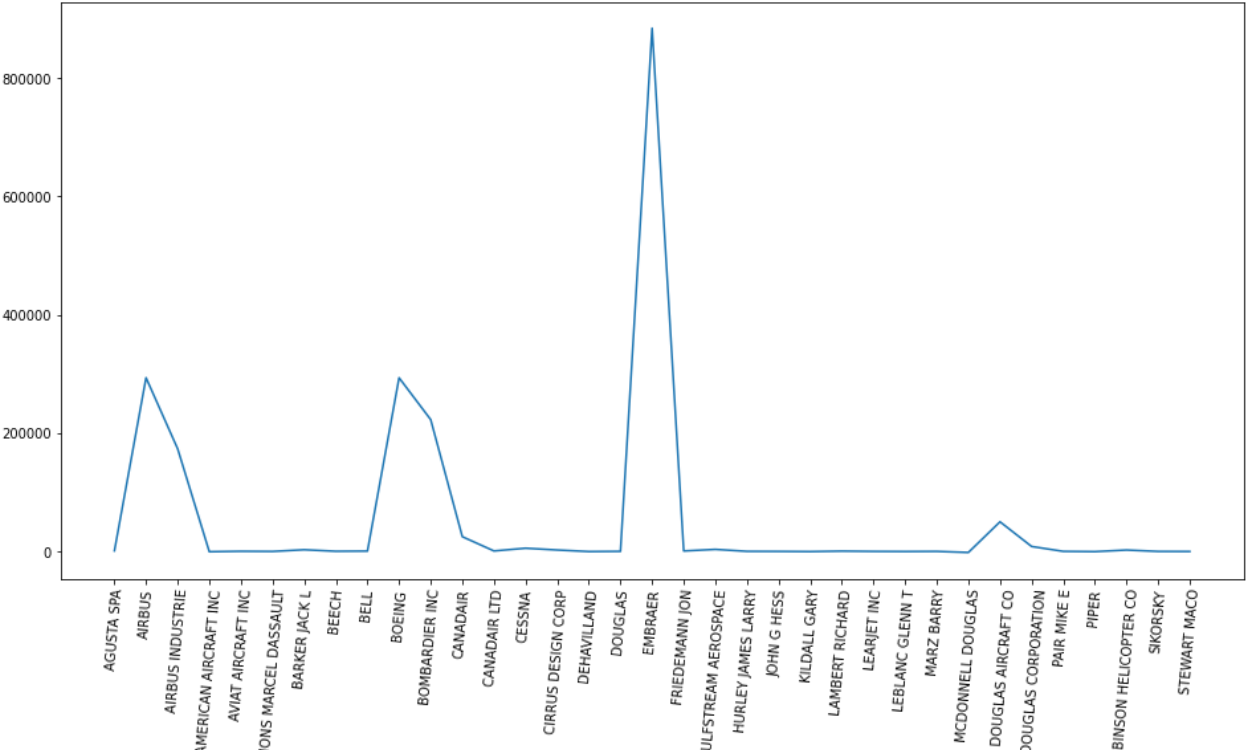
```
depdelay = df1.groupby('manufacturer')['dep_delay'].sum()
plt.figure(figsize=(16,8))
plt.plot(depdelay)
plt.xticks(rotation=85)
plt.show()
```



```

arrdelay = df1.groupby('manufacturer')['arr_delay'].sum()
plt.figure(figsize=(16,8))
plt.plot(arrdelay)
plt.xticks(rotation=85)
plt.show()

```



which are the two most connected cities? answer:- Sorry i am not able to answer this question