In [9]:
```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

IMPORT AIRLINES, FLIGHTS, AND AIRPORTS CSV FILES

In [10]:
```python
airlines_raw = pd.read_csv('airlines.csv')
flights_raw = pd.read_csv('flights.csv')
airports_raw = pd.read_csv('airports.csv')
print("AIRPORTS")
print(airports_raw.info())
print("FLIGHTS")
print(flights_raw.info())
print("AIRLINES")
print(airlines_raw.info())
```

```
/Users/dallas/opt/anaconda3/lib/python3.7/site-packages/IPython/core/interactiveshell.py:3058: DtypeW
arning: Columns (7,8) have mixed types. Specify dtype option on import or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

```
AIRPORTS
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 322 entries, 0 to 321
Data columns (total 7 columns):
IATA_CODE    322 non-null object
AIRPORT      322 non-null object
CITY         322 non-null object
STATE        322 non-null object
COUNTRY      322 non-null object
LATITUDE     319 non-null float64
LONGITUDE    319 non-null float64
dtypes: float64(2), object(5)
memory usage: 17.7+ KB
None
FLIGHTS
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5819079 entries, 0 to 5819078
Data columns (total 31 columns):
YEAR                   int64
MONTH                  int64
DAY                    int64
DAY_OF_WEEK            int64
AIRLINE                object
FLIGHT_NUMBER          int64
TAIL_NUMBER            object
ORIGIN_AIRPORT         object
DESTINATION_AIRPORT    object
SCHEDULED_DEPARTURE    int64
DEPARTURE_TIME         float64
DEPARTURE_DELAY        float64
TAXI_OUT               float64
WHEELS_OFF             float64
SCHEDULED_TIME         float64
ELAPSED_TIME           float64
AIR_TIME               float64
DISTANCE               int64
WHEELS_ON              float64
TAXI_IN                float64
SCHEDULED_ARRIVAL      int64
ARRIVAL_TIME           float64
ARRIVAL_DELAY          float64
DIVERTED               int64
CANCELLED              int64
CANCELLATION_REASON    object
AIR_SYSTEM_DELAY       float64
SECURITY_DELAY         float64
AIRLINE_DELAY          float64
LATE_AIRCRAFT_DELAY    float64
WEATHER_DELAY          float64
dtypes: float64(16), int64(10), object(5)
memory usage: 1.3+ GB
None
AIRLINES
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14 entries, 0 to 13
Data columns (total 2 columns):
IATA_CODE    14 non-null object
AIRLINE      14 non-null object
dtypes: object(2)
memory usage: 352.0+ bytes
None
```

TAKING A LOOK AT THE FIRST FEW ROWS OF EACH CSV FILE

In [11]:
```python
airports_raw.head()
```

Out[11]:

	IATA_CODE	AIRPORT	CITY	STATE	COUNTRY	LATITUDE	LONGITUDE
0	ABE	Lehigh Valley International Airport	Allentown	PA	USA	40.65236	-75.44040
1	ABI	Abilene Regional Airport	Abilene	TX	USA	32.41132	-99.68190
2	ABQ	Albuquerque International Sunport	Albuquerque	NM	USA	35.04022	-106.60919
3	ABR	Aberdeen Regional Airport	Aberdeen	SD	USA	45.44906	-98.42183
4	ABY	Southwest Georgia Regional Airport	Albany	GA	USA	31.53552	-84.19447

In [12]:
```python
airlines_raw.head()
```

Out[12]:

	IATA_CODE	AIRLINE
0	UA	United Air Lines Inc.
1	AA	American Airlines Inc.
2	US	US Airways Inc.
3	F9	Frontier Airlines Inc.
4	B6	JetBlue Airways

In [13]:
```python
flights_raw.head()
```

Out[13]:

	YEAR	MONTH	DAY	DAY_OF_WEEK	AIRLINE	FLIGHT_NUMBER	TAIL_NUMBER	ORIGIN_AIRPORT	DESTINATION_AIRPORT	SCHEDU
0	2015	1	1	4	AS	98	N407AS	ANC	SEA	
1	2015	1	1	4	AA	2336	N3KUAA	LAX	PBI	
2	2015	1	1	4	US	840	N171US	SFO	CLT	
3	2015	1	1	4	AA	258	N3HYAA	LAX	MIA	
4	2015	1	1	4	AS	135	N527AS	SEA	ANC	

5 rows × 31 columns

REMOVING ALL FLIGHTS THAT ARE NOT IN JANUARY

In [16]:
```python
flights_raw = flights_raw[flights_raw['MONTH'] == 1]
flights_raw.tail()
```

Out[16]:

	YEAR	MONTH	DAY	DAY_OF_WEEK	AIRLINE	FLIGHT_NUMBER	TAIL_NUMBER	ORIGIN_AIRPORT	DESTINATION_AIRPORT	SCI
469963	2015	1	31	6	B6	839	N658JB	JFK	BQN	
469964	2015	1	31	6	DL	1887	N855NW	SEA	DTW	
469965	2015	1	31	6	F9	300	N218FR	DEN	TPA	
469966	2015	1	31	6	F9	422	N954FR	DEN	ATL	
469967	2015	1	31	6	UA	1104	N73251	ANC	DEN	

5 rows × 31 columns

SWAPPING AIRLINE CODE IN FLIGHTS DATA FOR THE ACTUAL AIRLINE NAME

In [17]:
```python
airline_code_map = {}
for index, row in airlines_raw.iterrows():
    airline_code_map[row['IATA_CODE']] = row['AIRLINE']

print(airline_code_map)
```

```
{'UA': 'United Air Lines Inc.', 'AA': 'American Airlines Inc.', 'US': 'US Airways Inc.', 'F9': 'Front
ier Airlines Inc.', 'B6': 'JetBlue Airways', 'OO': 'Skywest Airlines Inc.', 'AS': 'Alaska Airlines In
c.', 'NK': 'Spirit Air Lines', 'WN': 'Southwest Airlines Co.', 'DL': 'Delta Air Lines Inc.', 'EV': 'A
tlantic Southeast Airlines', 'HA': 'Hawaiian Airlines Inc.', 'MQ': 'American Eagle Airlines Inc.', 'V
X': 'Virgin America'}
```

In [24]:
```python
flights_raw['AIRLINE_FULL'] = flights_raw.apply(lambda x: airline_code_map[x['AIRLINE']], axis=1)
flights_raw[['AIRLINE', 'AIRLINE_FULL']].head()
```

```
/Users/dallas/opt/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:1: SettingWithCopyWarni
ng:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
g.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
```

Out[24]:

	AIRLINE	AIRLINE_FULL
0	AS	Alaska Airlines Inc.
1	AA	American Airlines Inc.
2	US	US Airways Inc.
3	AA	American Airlines Inc.
4	AS	Alaska Airlines Inc.

REMOVING COLUMNS THAT WE DO NOT NEED FROM FLIGHTS DATAFRAME

In [25]:
```python
flights_raw.columns
```

Out[25]:
```
Index(['YEAR', 'MONTH', 'DAY', 'DAY_OF_WEEK', 'AIRLINE', 'FLIGHT_NUMBER',
       'TAIL_NUMBER', 'ORIGIN_AIRPORT', 'DESTINATION_AIRPORT',
       'SCHEDULED_DEPARTURE', 'DEPARTURE_TIME', 'DEPARTURE_DELAY', 'TAXI_OUT',
       'WHEELS_OFF', 'SCHEDULED_TIME', 'ELAPSED_TIME', 'AIR_TIME', 'DISTANCE',
       'WHEELS_ON', 'TAXI_IN', 'SCHEDULED_ARRIVAL', 'ARRIVAL_TIME',
       'ARRIVAL_DELAY', 'DIVERTED', 'CANCELLED', 'CANCELLATION_REASON',
       'AIR_SYSTEM_DELAY', 'SECURITY_DELAY', 'AIRLINE_DELAY',
       'LATE_AIRCRAFT_DELAY', 'WEATHER_DELAY', 'AIRLINE_FULL'],
      dtype='object')
```

In [32]:
```python
cols_to_use = ['AIRLINE_FULL', 'DAY_OF_WEEK', 'ORIGIN_AIRPORT', 'DESTINATION_AIRPORT', 'DEPARTURE_TIME'
, 'DEPARTURE_DELAY']
flights = flights_raw[cols_to_use]
flights.head()
```

Out[32]:

	AIRLINE_FULL	DAY_OF_WEEK	ORIGIN_AIRPORT	DESTINATION_AIRPORT	DEPARTURE_TIME	DEPARTURE_DELAY
0	Alaska Airlines Inc.	4	ANC	SEA	2354.0	-11.0
1	American Airlines Inc.	4	LAX	PBI	2.0	-8.0
2	US Airways Inc.	4	SFO	CLT	18.0	-2.0
3	American Airlines Inc.	4	LAX	MIA	15.0	-5.0
4	Alaska Airlines Inc.	4	SEA	ANC	24.0	-1.0

ADDING A COLUMN WITH VALUE 1 IF THE FLIGHT WAS DELAYED OR 0 IF NOT

In [33]:
```python
flights['DEPARTURE_DELAY'].value_counts()
```

Out[33]:
```
-5.0       34507
-4.0       34040
-3.0       33740
-2.0       31979
-1.0       28222
           ...
 618.0         1
 468.0         1
 605.0         1
 694.0         1
 1023.0        1
Name: DEPARTURE_DELAY, Length: 653, dtype: int64
```

In [34]:
```python
flights['HAS_DELAY'] = flights.apply(lambda x: 1 if x['DEPARTURE_DELAY'] > 0 else 0, axis=1)
flights.head()
```

```
/Users/dallas/opt/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:1: SettingWithCopyWarni
ng:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
g.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
```

Out[34]:

	AIRLINE_FULL	DAY_OF_WEEK	ORIGIN_AIRPORT	DESTINATION_AIRPORT	DEPARTURE_TIME	DEPARTURE_DELAY	HAS_DELAY
0	Alaska Airlines Inc.	4	ANC	SEA	2354.0	-11.0	0
1	American Airlines Inc.	4	LAX	PBI	2.0	-8.0	0
2	US Airways Inc.	4	SFO	CLT	18.0	-2.0	0
3	American Airlines Inc.	4	LAX	MIA	15.0	-5.0	0
4	Alaska Airlines Inc.	4	SEA	ANC	24.0	-1.0	0

SOME FLIGHTS LEFT AHEAD OF SCHEDULE, ADDING COLUMN TO WITH VALUE 1 IF FLIGHT LEFT EARLY 0 OTHERWISE

In [35]:
```python
flights['EARLY'] = flights.apply(lambda x: 1 if x['DEPARTURE_DELAY'] < 0 else 0, axis=1)
flights.head()
```

```
/Users/dallas/opt/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:1: SettingWithCopyWarni
ng:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
g.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
```

Out[35]:

	AIRLINE_FULL	DAY_OF_WEEK	ORIGIN_AIRPORT	DESTINATION_AIRPORT	DEPARTURE_TIME	DEPARTURE_DELAY	HAS_DELAY	EARLY
0	Alaska Airlines Inc.	4	ANC	SEA	2354.0	-11.0	0	
1	American Airlines Inc.	4	LAX	PBI	2.0	-8.0	0	
2	US Airways Inc.	4	SFO	CLT	18.0	-2.0	0	
3	American Airlines Inc.	4	LAX	MIA	15.0	-5.0	0	
4	Alaska Airlines Inc.	4	SEA	ANC	24.0	-1.0	0	

VISUALIZING DATA

DELAY COUNT BY AIRLINE

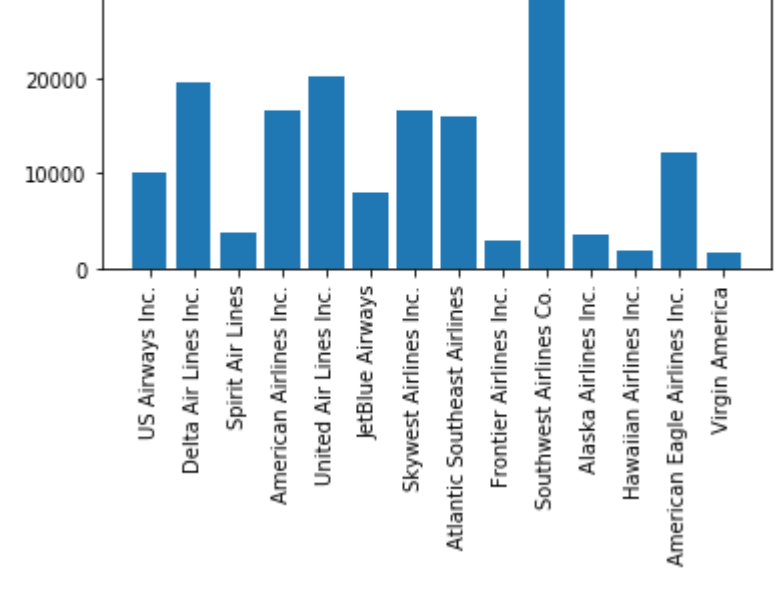
In [72]:
```python
delay_counts = {}
for index, row in flights.iterrows():
    if row['HAS_DELAY'] == 1 and row['AIRLINE_FULL'] not in delay_counts.keys():
        delay_counts[row['AIRLINE_FULL']] = 1
    elif row['HAS_DELAY'] == 1 and row['AIRLINE_FULL'] in delay_counts.keys():
        delay_counts[row['AIRLINE_FULL']] += 1

delay_counts
```

Out[72]:
```
{'US Airways Inc.': 10036,
 'Delta Air Lines Inc.': 19477,
 'Spirit Air Lines': 3710,
 'American Airlines Inc.': 16558,
 'United Air Lines Inc.': 20273,
 'JetBlue Airways': 7858,
 'Skywest Airlines Inc.': 16594,
 'Atlantic Southeast Airlines': 16090,
 'Frontier Airlines Inc.': 2979,
 'Southwest Airlines Co.': 43742,
 'Alaska Airlines Inc.': 3595,
 'Hawaiian Airlines Inc.': 1872,
 'American Eagle Airlines Inc.': 12269,
 'Virgin America': 1574}
```

In [73]:
```python
plt.xticks(rotation=90)
plt.bar(delay_counts.keys(), delay_counts.values())
```

Out[73]: <BarContainer object of 14 artists>



ROUND DEPARTURE TIME TO HOUR IN WHICH IT OCCURS

In [79]:
```python
def hourRound(time):
    return int(time / 100)
```

In [85]:
```python
flights.dropna(inplace=True)
flights['HOUR'] = flights.apply(lambda x: hourRound(x['DEPARTURE_TIME']), axis=1)
flights.head()
```

```
/Users/dallas/opt/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:1: SettingWithCopyWarni
ng:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
g.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
/Users/dallas/opt/anaconda3/lib/python3.7/site-packages/ipykernel_launcher.py:2: SettingWithCopyWarni
ng:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexin
g.html#returning-a-view-versus-a-copy
```

Out[85]:

	AIRLINE_FULL	DAY_OF_WEEK	ORIGIN_AIRPORT	DESTINATION_AIRPORT	DEPARTURE_TIME	DEPARTURE_DELAY	HAS_DELAY	EARLY
0	Alaska Airlines Inc.	4	ANC	SEA	2354.0	-11.0	0	
1	American Airlines Inc.	4	LAX	PBI	2.0	-8.0	0	
2	US Airways Inc.	4	SFO	CLT	18.0	-2.0	0	
3	American Airlines Inc.	4	LAX	MIA	15.0	-5.0	0	
4	Alaska Airlines Inc.	4	SEA	ANC	24.0	-1.0	0	

In [86]:
```python
flights['HOUR'].value_counts()
```

Out[86]:
```
17    30521
13    29864
11    29580
8     29524
15    28785
10    28507
6     28389
16    28195
12    27828
9     27789
14    27780
7     26946
19    26203
18    26139
20    20068
21    14331
5     13507
22     8132
23     3525
0      1324
1       651
4       437
2       190
3        62
24       34
Name: HOUR, dtype: int64
```

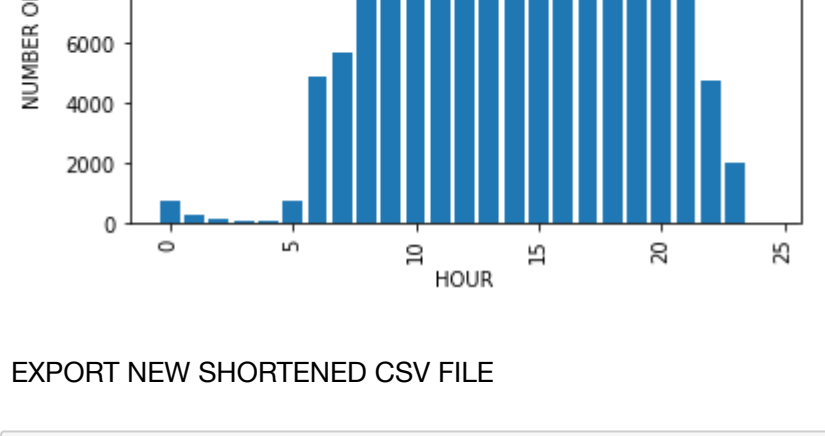
VISUALIZING DELAYS PER HOUR

In [88]:
```python
hours = {}
for index, row in flights.iterrows():
    if row['HAS_DELAY'] == 1 and row['HOUR'] not in hours.keys():
        hours[row['HOUR']] = 1
    elif row['HAS_DELAY'] == 1:
        hours[row['HOUR']] += 1
hours
```

Out[88]:
```
{0: 773,
 1: 303,
 2: 126,
 5: 740,
 4: 42,
 7: 5707,
 6: 4911,
 8: 7845,
 9: 8648,
 12: 10791,
 10: 10409,
 11: 11190,
 14: 12319,
 17: 13874,
 15: 12906,
 13: 12415,
 16: 12976,
 18: 12717,
 21: 7671,
 19: 12858,
 20: 10577,
 22: 4766,
 23: 1983,
 3: 51,
 24: 29}
```

In [90]:
```python
plt.xticks(rotation=90)
plt.xlabel("HOUR")
plt.ylabel("NUMBER OF DELAYS")
plt.bar(hours.keys(), hours.values())
```

Out[90]: <BarContainer object of 25 artists>



EXPORT NEW SHORTENED CSV FILE

In [91]:
```python
flights.to_csv("shortened.csv")
```

In [ ]: