One potential use of the data pipeline is curated flight information for travelers. The following is a sample output showing the yearly departure count and average total delay (departure and arrival) categorized by airline at each airport.

In [1]:
```
#import necessary libraries
import pymysql as mysql
import pandas as pd
import numpy as numpy
import matplotlib.pyplot as plt
import warnings
import getpass
warnings.filterwarnings('ignore')
```

In [3]:
```
#connect to offline database
conn=mysql.connect(host='localhost',
                   port=int(3306),
                   user='root',
                   passwd=getpass.getpass('Enter password:'),
                   db='ada307airlines')
```

```
Enter password:·····
```

In [9]:
```
#check connection
check = pd.read_sql ("SELECT * from airlines_all2018""",conn)
check
```

Out[9]:

	id	carrier	name
0	1	9E	Endeavor Air Inc.
1	2	AA	American Airlines Inc.
2	3	AS	Alaska Airlines Inc.
3	4	B6	JetBlue Airways
4	5	DL	Delta Air Lines Inc.
5	6	EV	ExpressJet Airlines LLC d/b/a aha!
6	7	F9	Frontier Airlines Inc.
7	8	G4	Allegiant Air
8	9	HA	Hawaiian Airlines Inc.
9	10	MQ	Envoy Air
10	11	NK	Spirit Air Lines
11	12	OH	PSA Airlines Inc.
12	13	OO	SkyWest Airlines Inc.
13	14	QX	Horizon Air
14	15	UA	United Air Lines Inc.
15	16	VX	Virgin America
16	17	WN	Southwest Airlines Co.
17	18	YV	Mesa Airlines Inc.
18	19	YX	Republic Airline

In [35]:
```
#partition yearly flight count and average total delays by each airport and save as pandas dataframe
yearlyIAH= pd.read_sql ("""SELECT * from yearly_delay2018 WHERE origin='IAH'""", conn)
yearlyJFK= pd.read_sql ("""SELECT * from yearly_delay2018 WHERE origin='JFK'""", conn)
yearlySEA= pd.read_sql ("""SELECT * from yearly_delay2018 WHERE origin='SEA'""", conn)
yearlySAN= pd.read_sql ("""SELECT * from yearly_delay2018 WHERE origin='SAN'""", conn)
yearlyIAH.head(5)
```

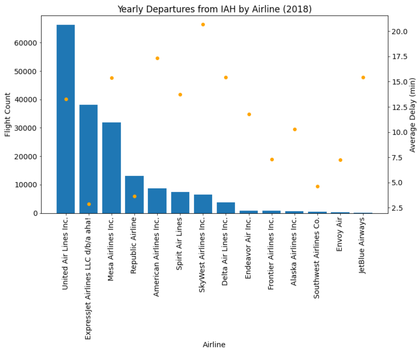
Out[35]:

	name	carrier	origin	yearly_flight_count	total_delay
0	United Air Lines Inc.	UA	IAH	66215	13.2871
1	ExpressJet Airlines LLC d/b/a aha!	EV	IAH	38142	2.8621
2	Mesa Airlines Inc.	YV	IAH	31979	15.3794
3	Republic Airline	YX	IAH	13132	3.6284
4	American Airlines Inc.	AA	IAH	8627	17.3182

Yearly Departures by Airline

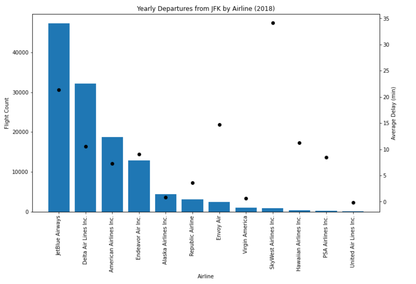
IAH (2018)

In [76]:
```
x=yearlyIAH['name']
y=yearlyIAH['yearly_flight_count']

fig = plt.figure(figsize =(12, 7))
plt.rcParams.update({'font.size': 14})

plt.xticks(rotation=90)
plt.bar(x,y)
plt.title("Yearly Departures from IAH by Airline (2018)")
plt.xlabel("Airline")
plt.ylabel("Flight Count")

plt.twinx()
y_delay=yearlyIAH['total_delay']
plt.scatter(x,y_delay,color='orange')
plt.ylabel("Average Delay (min)")
plt.show()
```



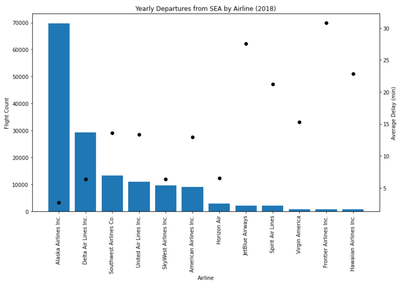
JFK (2018)

In [60]:
```
x1=yearlyJFK['name']
y1=yearlyJFK['yearly_flight_count']
fig = plt.figure(figsize =(12, 7))
plt.xticks(rotation=90)
plt.bar(x1,y1)
plt.title("Yearly Departures from JFK by Airline (2018)")
plt.xlabel("Airline")
plt.ylabel("Flight Count")

plt.twinx()
y1_delay=yearlyJFK['total_delay']
plt.scatter(x1,y1_delay,color='black')
plt.ylabel("Average Delay (min)")
plt.show()
```



SEA (2018)

In [62]:
```
x2=yearlySEA['name']
y2=yearlySEA['yearly_flight_count']
fig = plt.figure(figsize =(12, 7))
plt.xticks(rotation=90)
plt.bar(x2,y2)
plt.title("Yearly Departures from SEA by Airline (2018)")
plt.xlabel("Airline")
plt.ylabel("Flight Count")

plt.twinx()
y2_delay=yearlySEA['total_delay']
plt.scatter(x2,y2_delay,color='black')
plt.ylabel("Average Delay (min)")
plt.show()
```



SAN (2018)

In [63]:
```
x3=yearlySAN['name']
y3=yearlySAN['yearly_flight_count']
fig = plt.figure(figsize =(12, 7))
plt.xticks(rotation=90)
plt.bar(x3,y3)
plt.title("Yearly Departures from SAN by Airline (2018)")
plt.xlabel("Airline")
plt.ylabel("Flight Count")

plt.twinx()
y3_delay=yearlySAN['total_delay']
plt.scatter(x3,y3_delay,color='black')
plt.ylabel("Average Delay (min)")
plt.show()
```