In [1]:
```
import pandas as pd
import seaborn as sns
import plotly.express as px

import matplotlib.pyplot as plt
```

In [2]:
```
import plotly.io as pio
pio.renderers.default = "plotly_mimetype+notebook"
```

Matplotlib

For this excercise, we have written the following code to load the stock dataset built into plotly express.

In [3]:
```
stocks = px.data.stocks()
stocks.head()
```

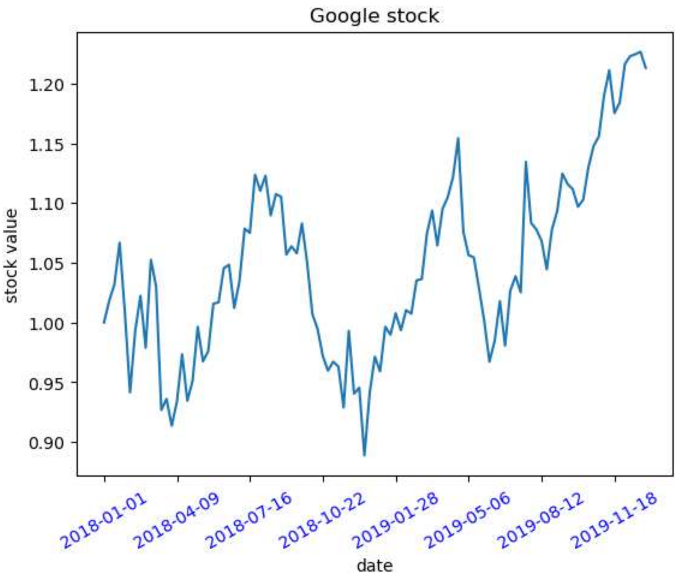
Out[3]:

	date	GOOG	AAPL	AMZN	FB	NFLX	MSFT
0	2018-01-01	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000
1	2018-01-08	1.018172	1.011943	1.061881	0.959968	1.053526	1.015988
2	2018-01-15	1.032008	1.019771	1.053240	0.970243	1.049860	1.020524
3	2018-01-22	1.066783	0.980057	1.140676	1.016858	1.307681	1.066561
4	2018-01-29	1.008773	0.917143	1.163374	1.018357	1.273537	1.040708

Question 1:

Select a stock and create a suitable plot for it. Make sure the plot is readable with relevant information, such as date, values.

In [4]:
```
# YOUR CODE HERE
x = stocks.date
y = stocks.GOOG
columns = stocks.iloc[:, 1].count()
fig, ax = plt.subplots()
plt.plot(x,y)
plt.xticks(range(0,columns,14),color='blue',rotation=30)
ax.set_title('Google stock')
ax.set_xlabel('date')
ax.set_ylabel('stock value')
plt.show()
```

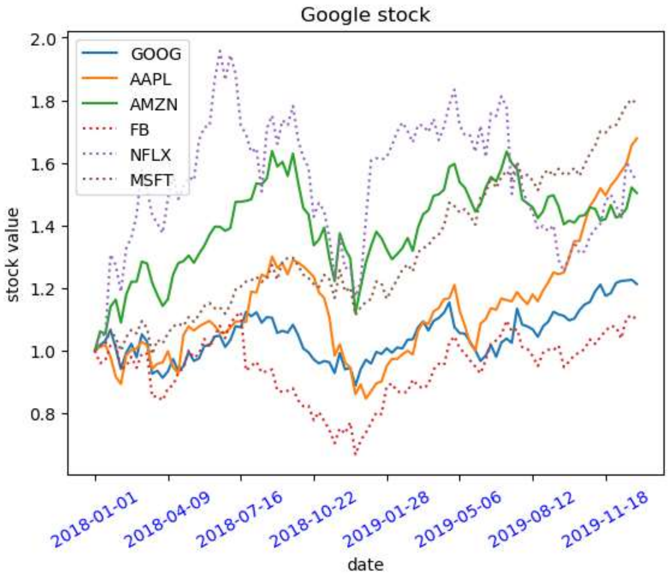


Question 2:

You've already plot data from one stock. It is possible to plot multiples of them to support comparison.
To highlight different lines, customise line styles, markers, colors and include a legend to the plot.

In [5]:

```
# YOUR CODE HERE
#create data
x = stocks.date
y = stocks.GOOG
z = stocks.AAPL
m = stocks.AMZN
n = stocks.FB
p = stocks.NFLX
q = stocks.MSFT

#set lins
columns = stocks.iloc[:, 1].count()
fig, ax = plt.subplots()
plt.plot(x, y, label='GOOG')
plt.plot(x, z, label='AAPL')
plt.plot(x, m, label='AMZN')
plt.plot(x, n, label='FB', linestyle=':')
plt.plot(x, p, label='NFLX', linestyle=':')
plt.plot(x, q, label='MSFT', linestyle=':')
plt.xticks(range(0,columns,14),color='blue',rotation=30)

#set labels
ax.set_title('Google stock')
ax.set_xlabel('date')
ax.set_ylabel('stock value')
plt.legend()
plt.show()
```



Seaborn

First, load the tips (https://github.com/mwaskom/seaborn-data/blob/master/tips.csv) dataset

In [6]:

```
tips = sns.load_dataset('tips')
tips.head()
```

Out[6]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

Question 3:

Let's explore this dataset. Pose a question and create a plot that support drawing answers for your question.

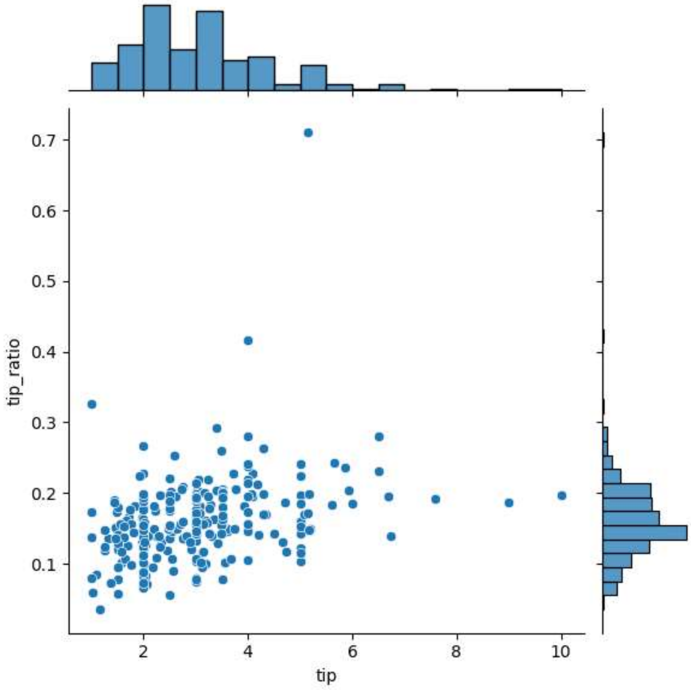
Some possible questions:

- Are there differences between male and female when it comes to giving tips?
- What attribute correlate the most with tip?

In [7]:

```
# YOUR CODE HERE
print('My question is: is there any relationship between the total bill and tip?')
tips['tip_ratio'] = tips['tip']/tips['total_bill']
sns.jointplot(x='tip', y='tip_ratio', data=tips)
plt.show()
```

```
My question is: is there any relationship between the total bill and tip?
```



Plotly Express

Question 4:

Redo the above exercises (challenges 2 & 3) with plotly express. Create diagrams which you can interact with.

The stocks dataset

Hints:

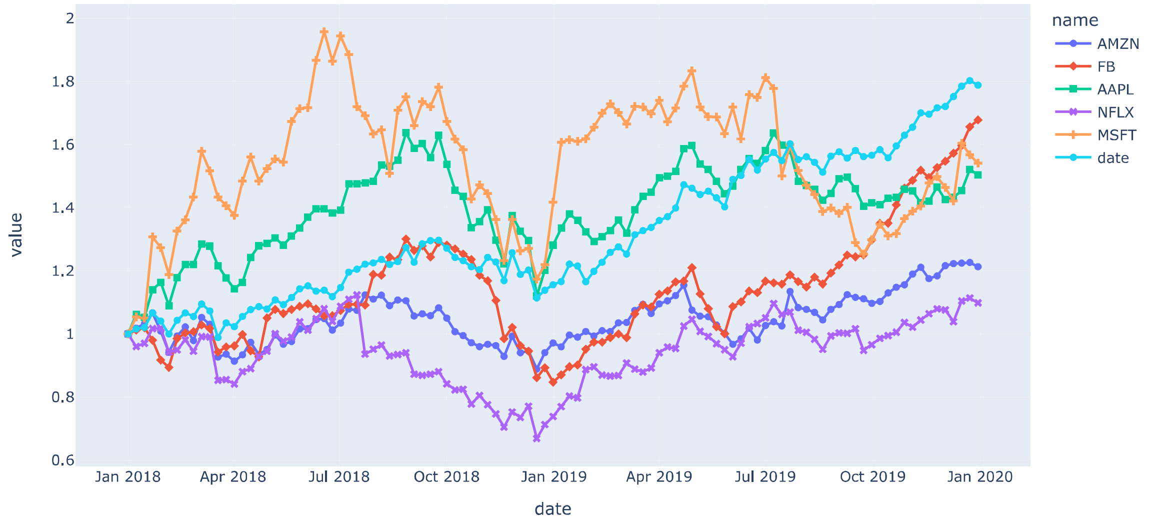
- Turn stocks dataframe into a structure that can be picked up easily with plotly express

In [8]:

```
# YOUR CODE HERE
#stocks = px.data.stocks()
#fig = px.line(stocks, x="date", y=stocks.columns, markers=True)
#fig.show()
```

In [9]:

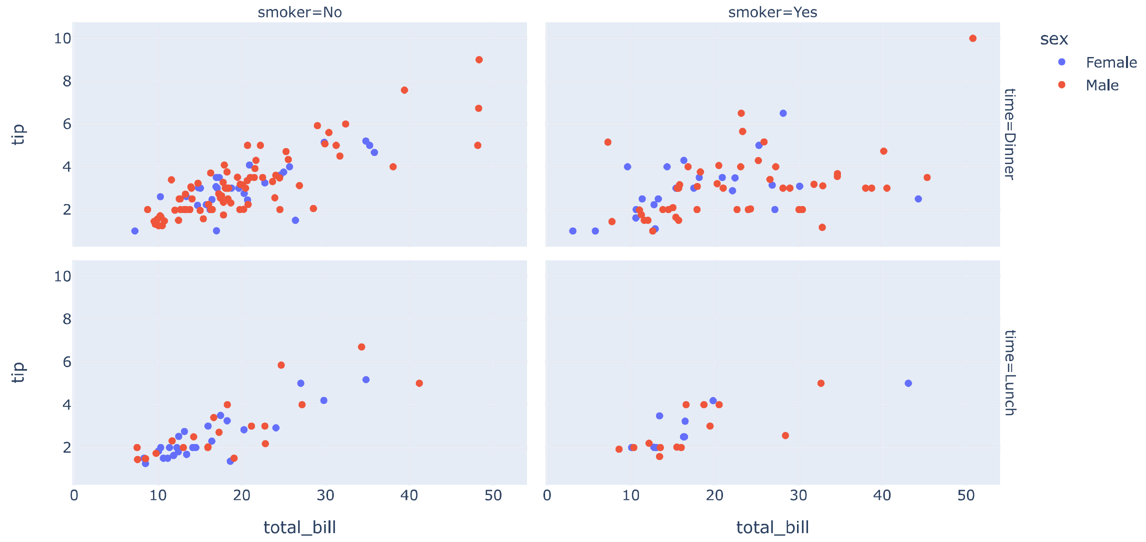
```
# YOUR CODE HERE
stocks = px.data.stocks()
names = list(set(stocks.columns))
names = names[1:]
col = ['date', 'value', 'name']

new_data = pd.DataFrame(columns = col)

t = 0
for i in range(len(stocks.index)):
    for j in range(6):
        new_data.loc[t] = [stocks.iloc[i, 0], stocks.iloc[i, j + 1], names[j]]
        t += 1

fig = px.line(new_data, x = 'date', y = 'value', color = 'name', symbol = 'name')
fig.show()
```



The tips dataset

In [10]:

```
# YOUR CODE HERE
df = px.data.tips()
fig = px.scatter(df, x="total_bill", y="tip", color="sex", facet_col="smoker", facet_row="time")
fig.show()
```



Question 5:

Recreate the barplot below that shows the population of different continents for the year 2007.

Hints:

- Extract the 2007 year data from the dataframe. You have to process the data accordingly
- use plotly bar (https://plotly.com/python-api-reference/generated/plotly.express.bar)
- Add different colors for different continents
- Sort the order of the continent for the visualisation. Use axis layout setting (https://plotly.com/python/reference/layout/xaxis/)

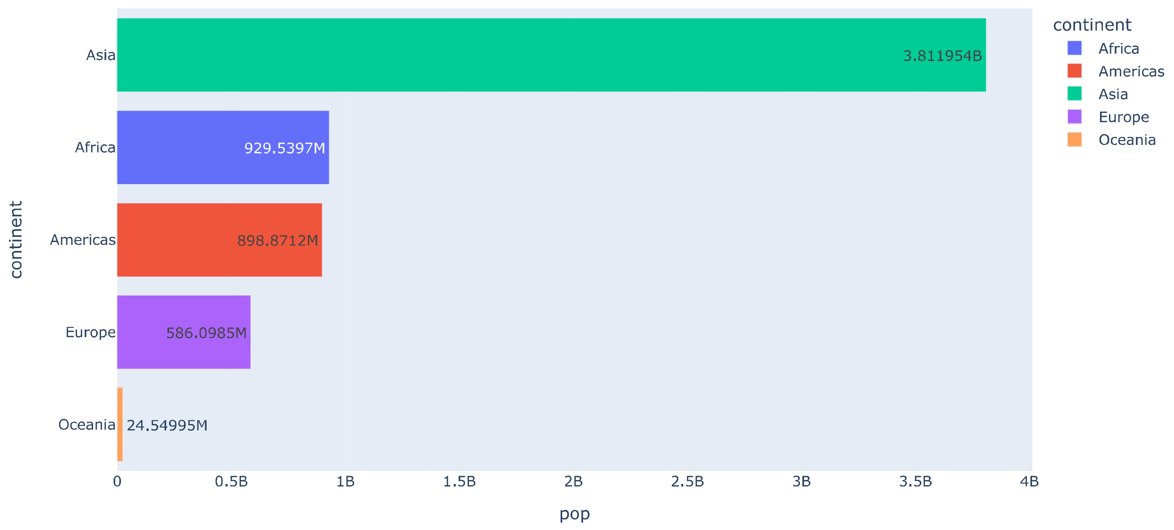
- Add text to each bar that represents the population

In [11]:
```
#load data
df = px.data.gapminder()
df.head()
```

Out[11]:

	country	continent	year	lifeExp	pop	gdpPercap	iso_alpha	iso_num
0	Afghanistan	Asia	1952	28.801	8425333	779.445314	AFG	4
1	Afghanistan	Asia	1957	30.332	9240934	820.853030	AFG	4
2	Afghanistan	Asia	1962	31.997	10267083	853.100710	AFG	4
3	Afghanistan	Asia	1967	34.020	11537966	836.197138	AFG	4
4	Afghanistan	Asia	1972	36.088	13079460	739.981106	AFG	4

In [12]:
```
# YOUR CODE HERE
df_2007 = df.query('year==2007')
df_2007_new = df_2007.groupby('continent').sum()
fig = px.bar(df_2007_new, x="pop", y=df_2007_new.index, color=df_2007_new.index, orientation='h', text_auto= True)
fig = fig.update_yaxes(categoryorder = 'total ascending')
fig.show()
```



In [ ]: