1. Introduction: Explain why you chose the topic, the questions you are interested in studying. List team members and a description of how each contributed to the project.

The reason this topic and dataset was chosen by our group is two-fold. First, this dataset contained enough data variables that gives the dataset a high potential of analysis for interesting correlations. Second, the topic (music, specifically that streamed from Spotify) was found to be very relatable to our group's members, and thus intriguing.

Our group selected eight questions that we would like to answer using data analysis techniques. They include the following:
1.Is there an even distribution of the number of occurrences per Artist?
2.Is there a relationship between an Artist's number of occurrences and the popularity of their most popular song?
3.Does Genre correlate to Popularity?
4.Does Artist correlate to Popularity?
5.Do more popular song tend to be shorter?
6.Does BPM and Energy have any correlation?
7.Which artist shows up the most in the data set?
8.Do certain genres tend to have a higher energy or bpm?
9.Is there any correlation between genre and length?
10.Is there an even distribution of the number of occurrences per Genre?

Group Team Members (and contributions):
- Adam Scott (Data questions 1, 2, 6, 8. Repo management. Technical questions 1, 4, 6)
- Baron Schitka (Blog Post, Data Questions 3, 9, 10. Technical questions 2, 3, 4, 5, 6)
- Ashvin Kuruparan (Blog Post, Technical question 4, 6)
- Jimmy Kesikiadis (Blog Post, Data Questions 4, 5, 7. Technical questions 4, 6)

2. Description of data: Describe the dataset, how was it collect, how you accessed it, references/credit to source.

Top 50 Spotify Songs - 2019 https://www.kaggle.com/leonardopena/top50spotify2019

It is information on data points about the top 50 Spotify songs of 2019. The way it was collected was that it is the 50 most listened Songs on Spotify and Spotify itself collected the data with the data that they collected from their own software. We got the dataset from Kaggle.com.

3. Analysis of the data: Provide a detailed, well-organized description of data quality, including the features, any data that should be cleaned or pre-processed before you EDA.

1.Track.Name-Name of the Track
2.Artist.Name-Name of the Artist
3.Genre-the genre of the track
4.Beats.Per.Minute-The tempo of the song.
5.Energy-The energy of a song - the higher the value, the more energetic. song

6.Danceability-The higher the value, the easier it is to dance to this song.
7.Loudness..dB..-The higher the value, the louder the song.
8.Liveness-The higher the value, the more likely the song is a live recording.
9.Valence.-The higher the value, the more positive the mood for the song.
10.Length.-The duration of the song.
11.Acousticness..-The higher the value the more acoustic the song is.
12.Speechiness.-The higher the value the more spoken word the song contains.
13.Popularity-The higher the value the more popular the song is.
Track name, Artist name, genre, bpm, and length were all high quality and useful. The rest were not as useful as they seem more subjective and less useful for Data Science. There was no cleaning or prepositioning that had to be done. No data was removed as it would be better to have and show off than not. There was no data missing that would need to be removed as part of the clean up.

5. Potential Data Science: Based on your data analysis and findings. Describe any potential ideas if you were to pursue a data science or machine learning project using this dataset. If you don't find any potential, explain your rationale.

The conclusions of this data may have potential in recommendations for artists, by incorporating into their songs any variables that correlate with popularity and avoiding variables that are negatively correlated to it. Recommendations can be made to suggest the best elements to include in a song and a graphical representation could be made for ease of use. For machine learning, you could have the machine predict how popular a song will be before it comes out by using this data. With enough iterations the algorithm could figure out what weight to give each element in its prediction. The machine could also guess if a song was popular on Spotify, I predict that artist name would be weighted as very important in that prediction.

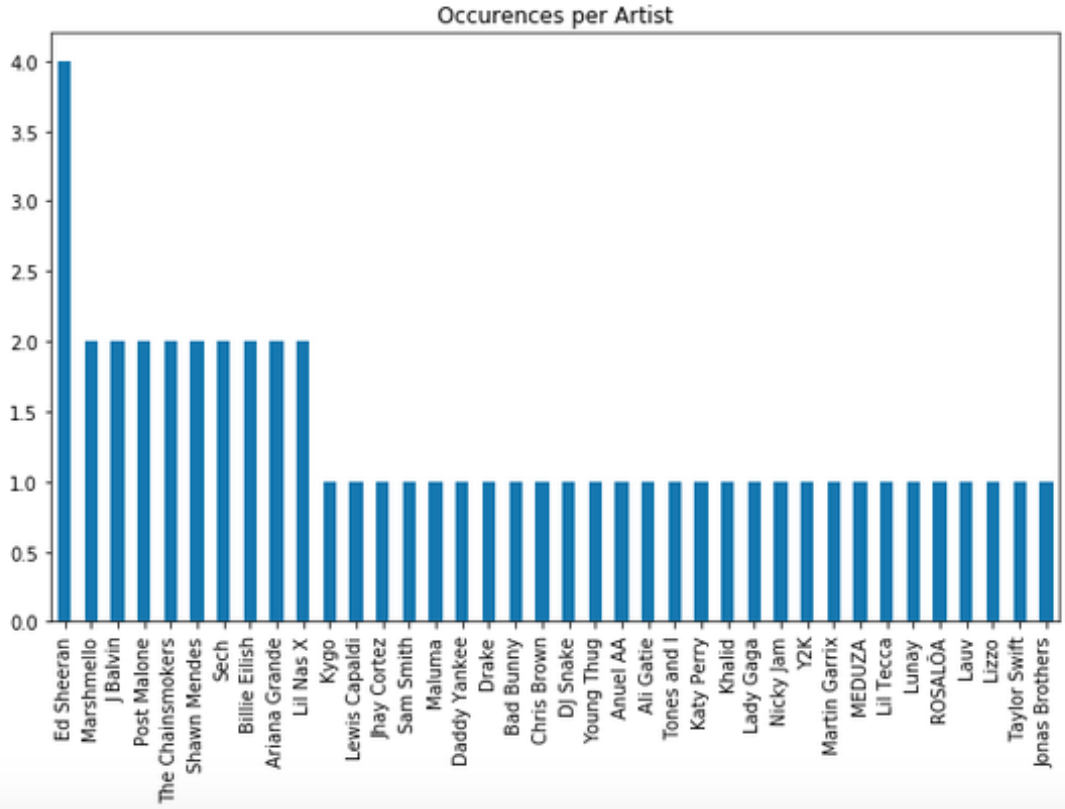
4. Exploratory Data Analysis: Provide a detailed, well-organized description of your findings, including textual description, graphs, and code. Your focus should be on both the results and the process. Include, as reasonable and relevant, approaches that didn't work, challenges, the data cleaning process, etc.

Question 1: Is there an even distribution of the number of occurrences per Artist?

The distribution of number of occurrences per artist was relatively even. The artist with the most occurrences was Ed Sheeran, who only had 4 songs. Meanwhile 9 artists had two songs, and the remaining 40 had 1 song each.

In [6]:
```
artist_counts = data['Artist.Name'].value_counts()
artist_counts.plot.bar(title='Occurences per Artist', figsize=(10,6))
```

Out[6]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f810c42e390>
```



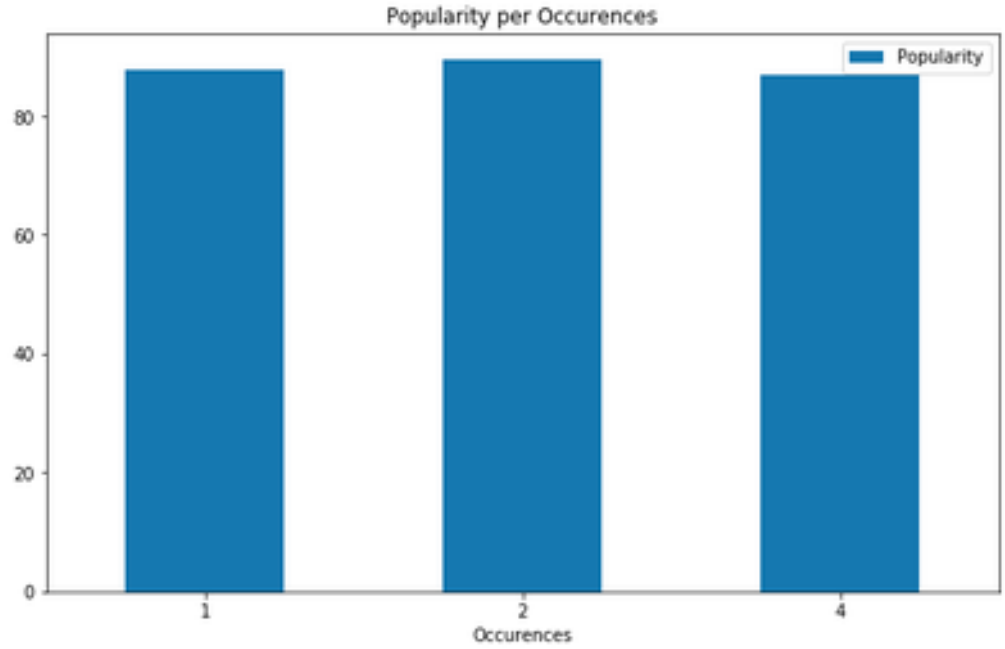
Question 2: Is there a relationship between an Artist's number of occurrences and the popularity of their most popular song?

The data shows that artists with the most occurences (4), actually had the lowest average most popular song with 87, which is less than those with least occurrences (1) who had average most popular song of 87.89. The group of artists with the highest average of most popular songs was those with 2 occurrences, who had an average of 89.44.

In [7]:
```
#Question 2
df = data
df['Occurences'] = df.groupby('Artist.Name')['Popularity'].transform('size')
df = df[['Artist.Name', 'Popularity', 'Occurences']]
df = df.groupby(['Artist.Name', 'Occurences'])['Popularity'].max().reset_index()

means = df.groupby('Occurences')['Popularity'].mean()
means = means.reset_index()
means = means.groupby('Occurences').first()
print(means)
means.plot.bar(title='Popularity per Occurences', rot=0, figsize=(10,6))
```

```
            Popularity
Occurences
1            87.892857
2            89.444444
4            87.000000
```

Out[7]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f810cd53bd0>
```

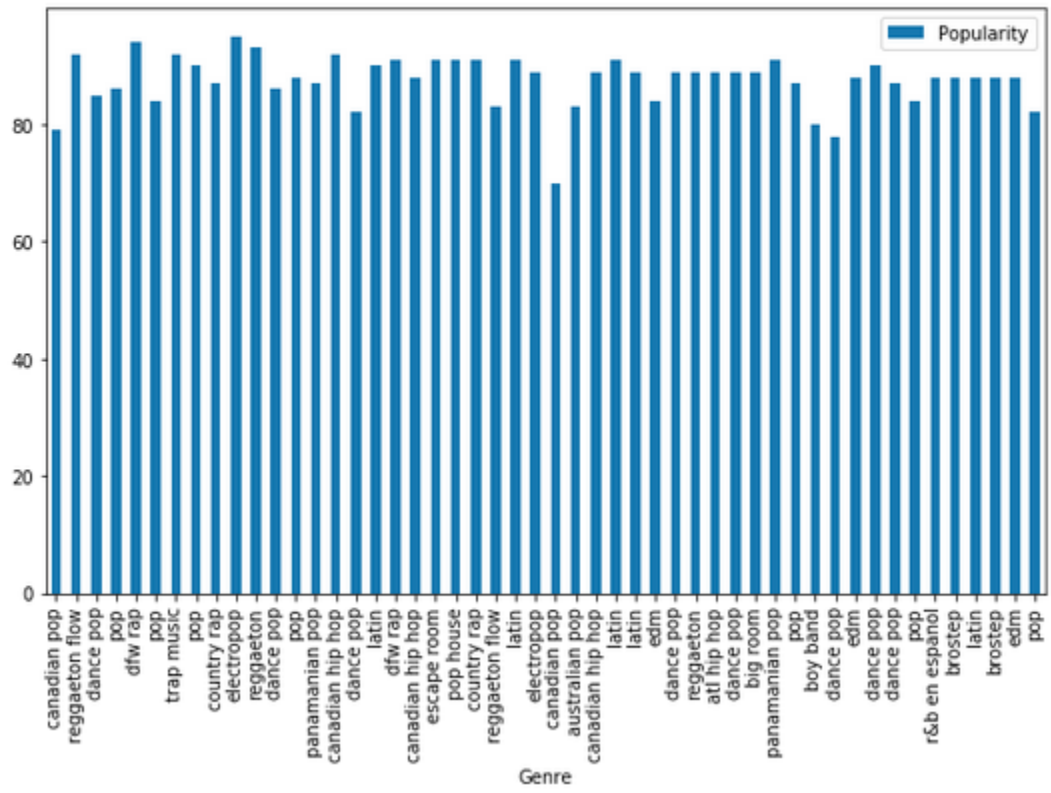


Question 3: Does Genre correlate to Popularity?

From what this dataset shows there doesn't seem to be a correlation between popularity and genre due to the most popular songs all being from different genres. The solution above also gave the correlation between popularity and every other element. I can specifically use this to answer the question "Does Artist correlate to Popularity?" along with any new questions involving popularity.

```
data.plot.bar(x='Genre', y='Popularity', figsize=(10,6))
```

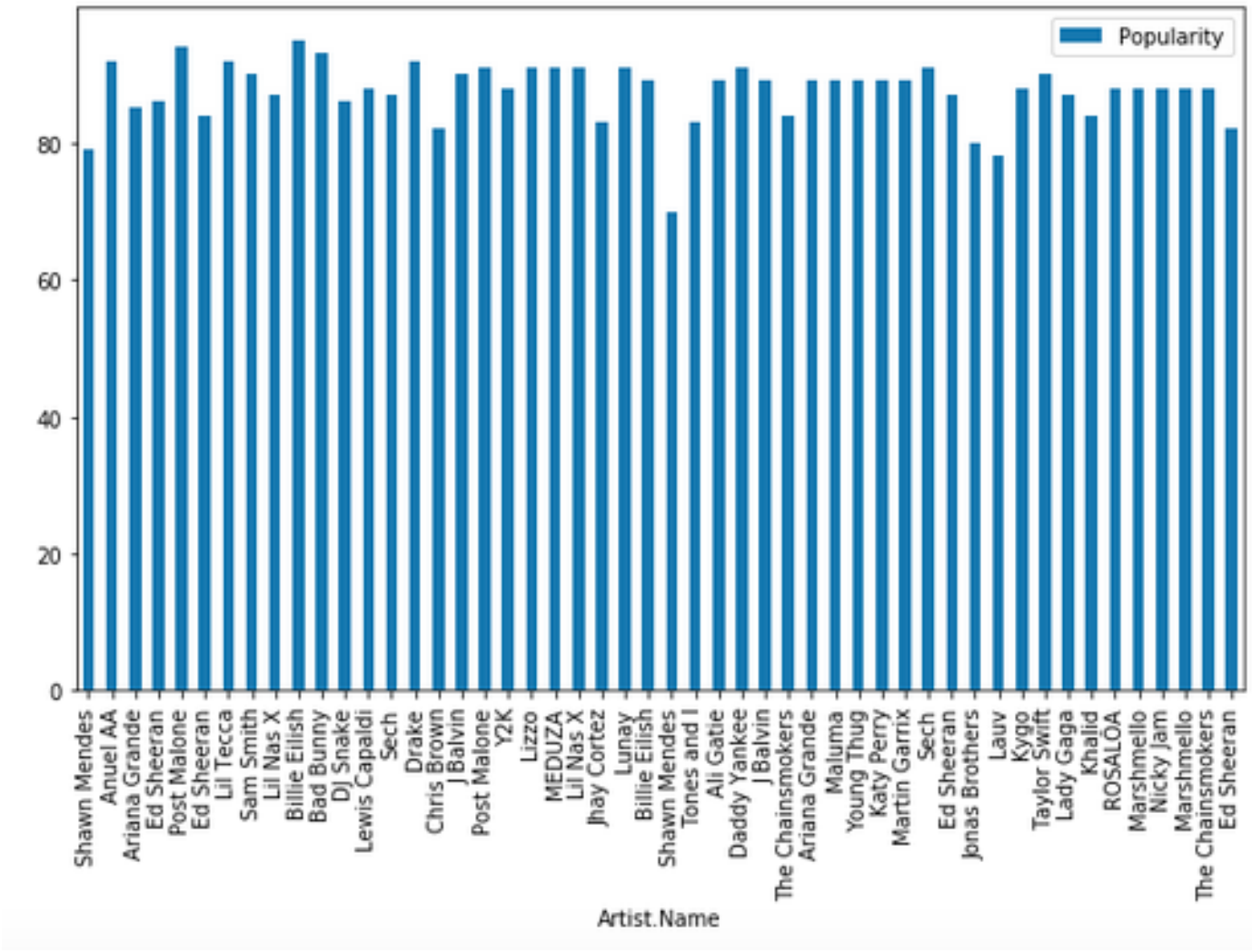
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f810e3f8990>
```



Question 4: Does Artist correlate to Popularity?
From the data show it can be concluded that certain artists are much more popular than others and that there is a correlation between certain artists and being more popular than others.

```
data.plot.bar(x='Artist.Name', y='Popularity', figsize=(10,6))
```

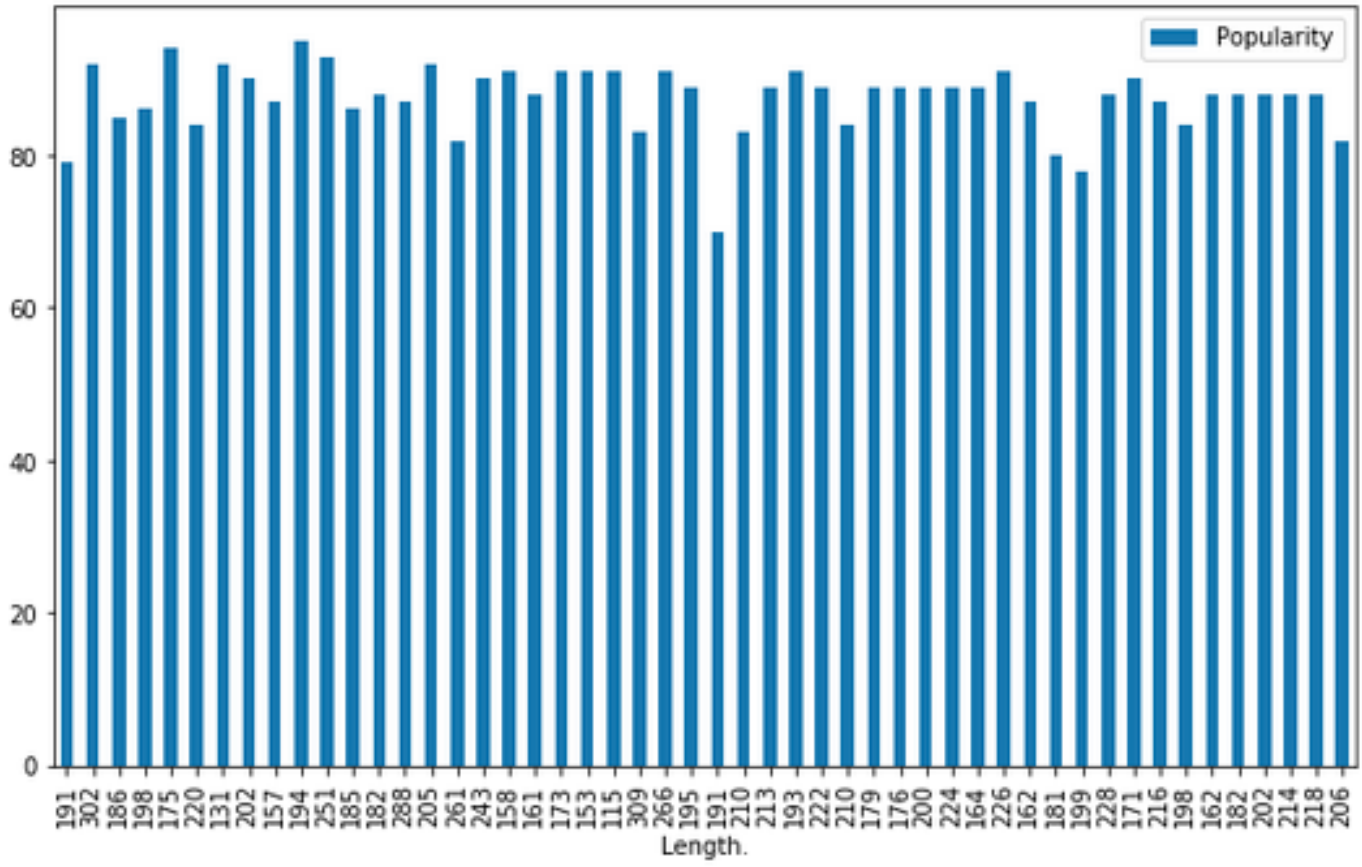
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f810eb50650>
```



Question 5: Do more popular songs tend to be shorter?
As seen in the data, the length of the song does not contribute to the popularity of the song. The more popular songs do not follow any pattern of being shorter or longer therefore the popularity of the song does not hinge on the length of the song.

```
data.plot.bar(x='Length.', y='Popularity', figsize=(10,6))
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f810ef62fd0>
```



Question 6: Does BPM and Energy have any correlation?Since the correlation coefficient between bpm and energy is approx. 0.04, this means there is no linear relationship between bpm and energy.

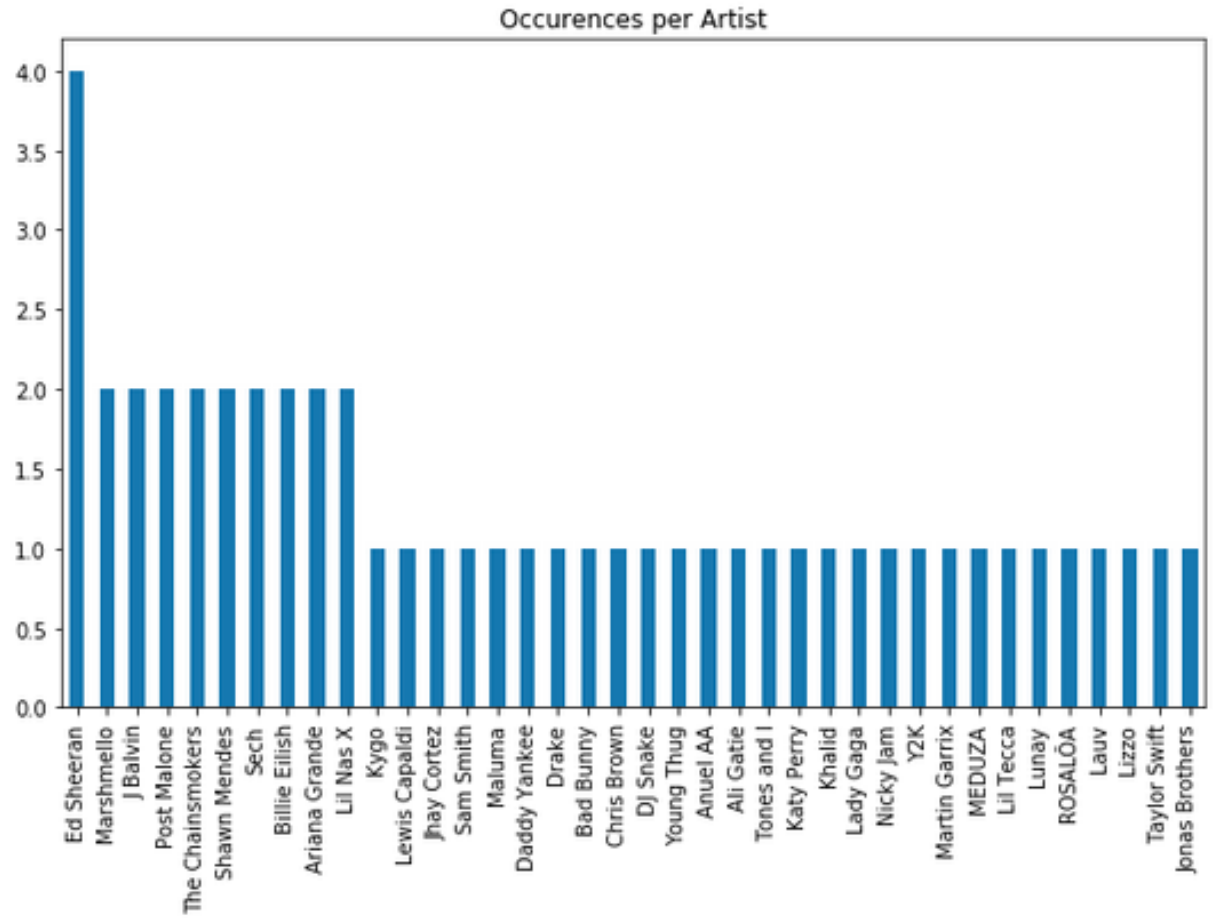
In [13]:
```
#Gets a list of all of the different BPM
def get_data_csv():
    collection = []
    with open('top50.csv', 'r') as f:
        for line in csv.DictReader(f):
            collection.append(line)
        return collection

# the data
data2 = get_data_csv()

def bpm():
    bpm = []
    for element in range(len(data2)):
        bpm.append(int(data2[element]['Beats.Per.Minute']))
    return bpm

#Gets a list of all of the different energy levels
def energy():
    energy = []
    for element in range(len(data2)):
        energy.append(int(data2[element]['Energy']))
    return energy

bpm_array = np.array(bpm())
energy_array = np.array(energy())

corr = np.corrcoef(bpm_array, energy_array)
corr[0, 1]
```

Out[13]:
```
0.04375559078974287
```

Question 7: Which artist shows up the most in the data set?
As seen in the data, the artists that show up the most are Ed Sheeran with 4 songs, seconded by Lil Nas X, Shawn Mendez, Marshmello, Post Malone, Sech, J Balvin, The Chainsmokers, and Billie Eilish all with two songs. The rest of the artists in the dataset have 1 song.

```
artist_counts.plot.bar(title='Occurences per Artist', figsize=(10,6))
```

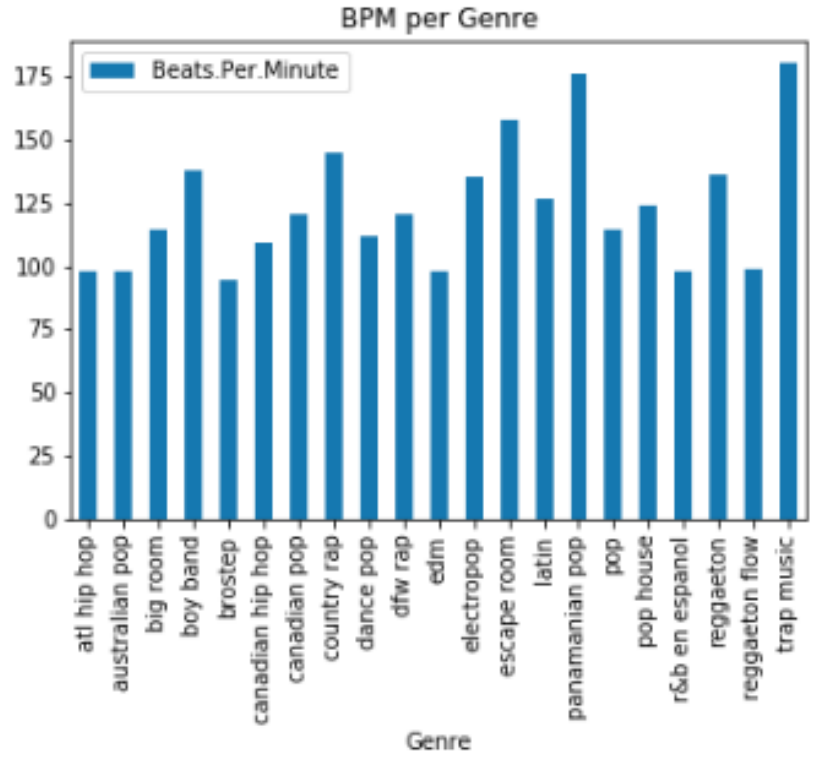
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f810f94f350>
```



Question 8: Do certain genres tend to have a higher energy or bpm?
It appears that both BPM and Energy differ significantly across genres. It also appears that the two do not necessarily correlate (for example reggaeton flow has very low BPM but very high Energy)

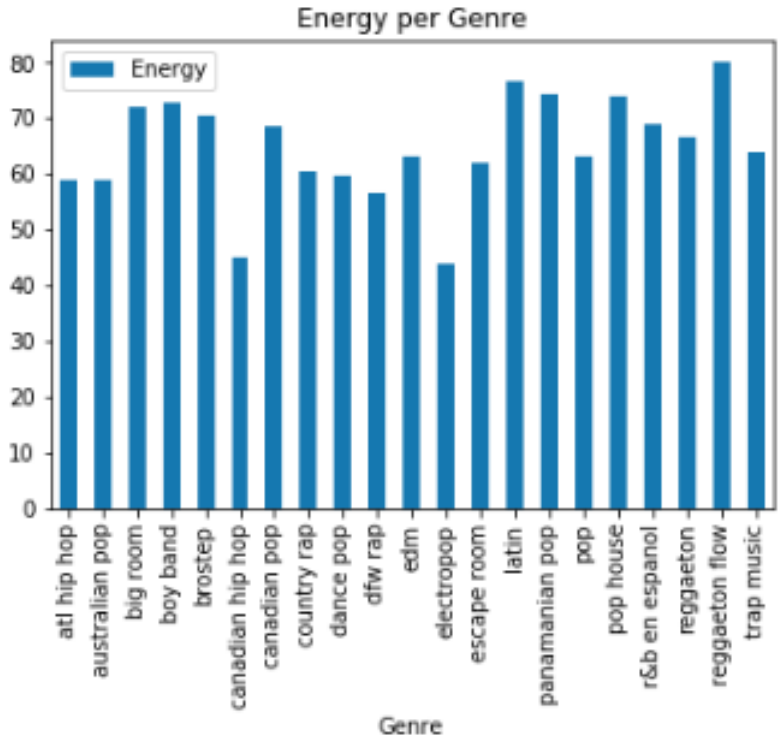
```
genre_bpm = data.groupby('Genre')['Beats.Per.Minute'].mean().reset_index()
genre_bpm = genre_bpm.groupby('Genre').first()
genre_bpm.plot.bar(title='BPM per Genre')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f810d34a490>
```



In [16]:
```
genre_energy = data.groupby('Genre')['Energy'].mean().reset_index()
genre_energy = genre_energy.groupby('Genre').first()
genre_energy.plot.bar(title='Energy per Genre')
```

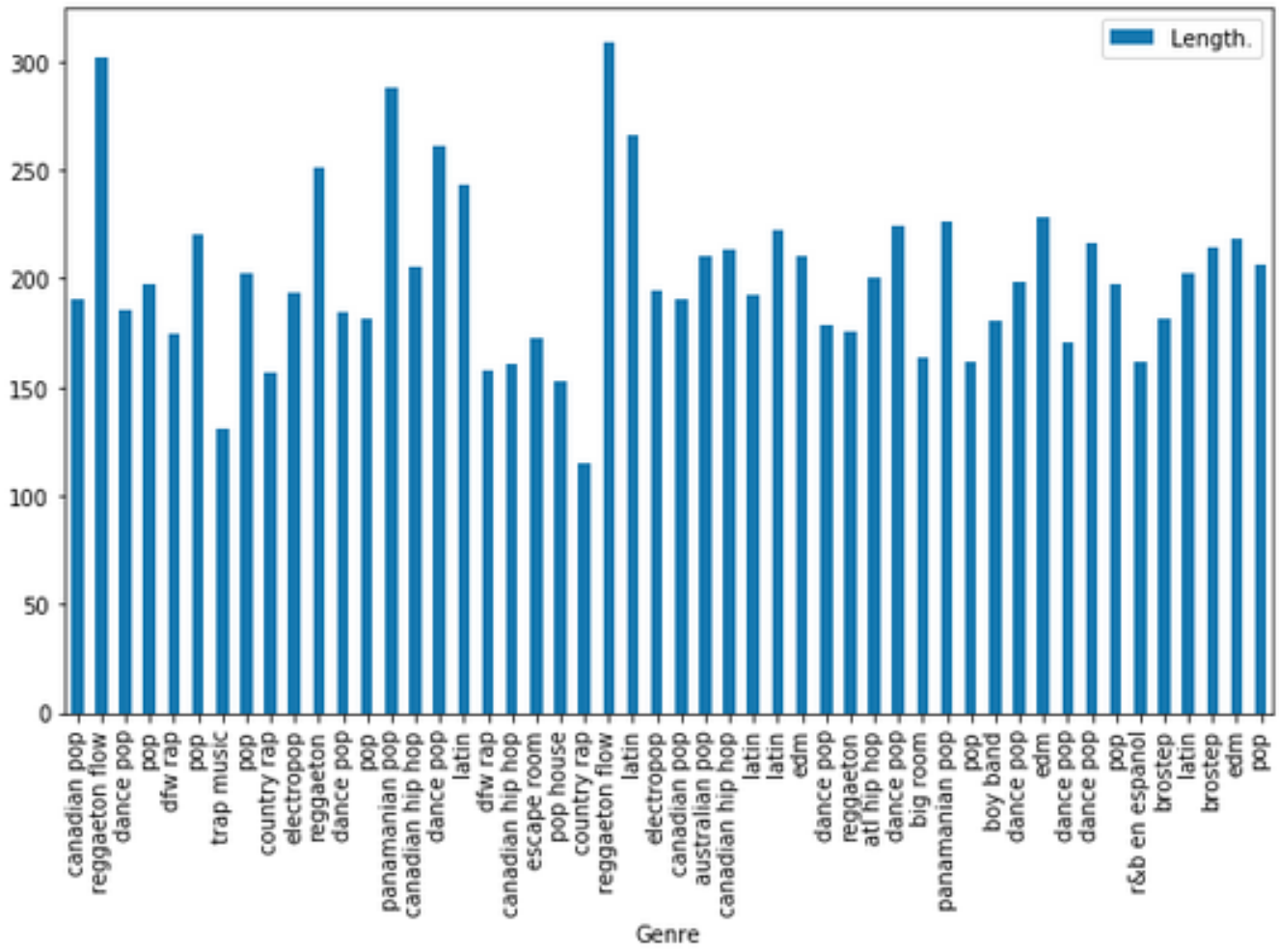
Out[16]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f810d72e110>
```



Question 9: Is there any correlation between genre and length?
From what this dataset shows, it seems that Reggaeton Flow, Latin and Pop music all have the highest length of songs.

```
data.plot.bar(x='Genre', y='Length.', figsize=(10,6))
```

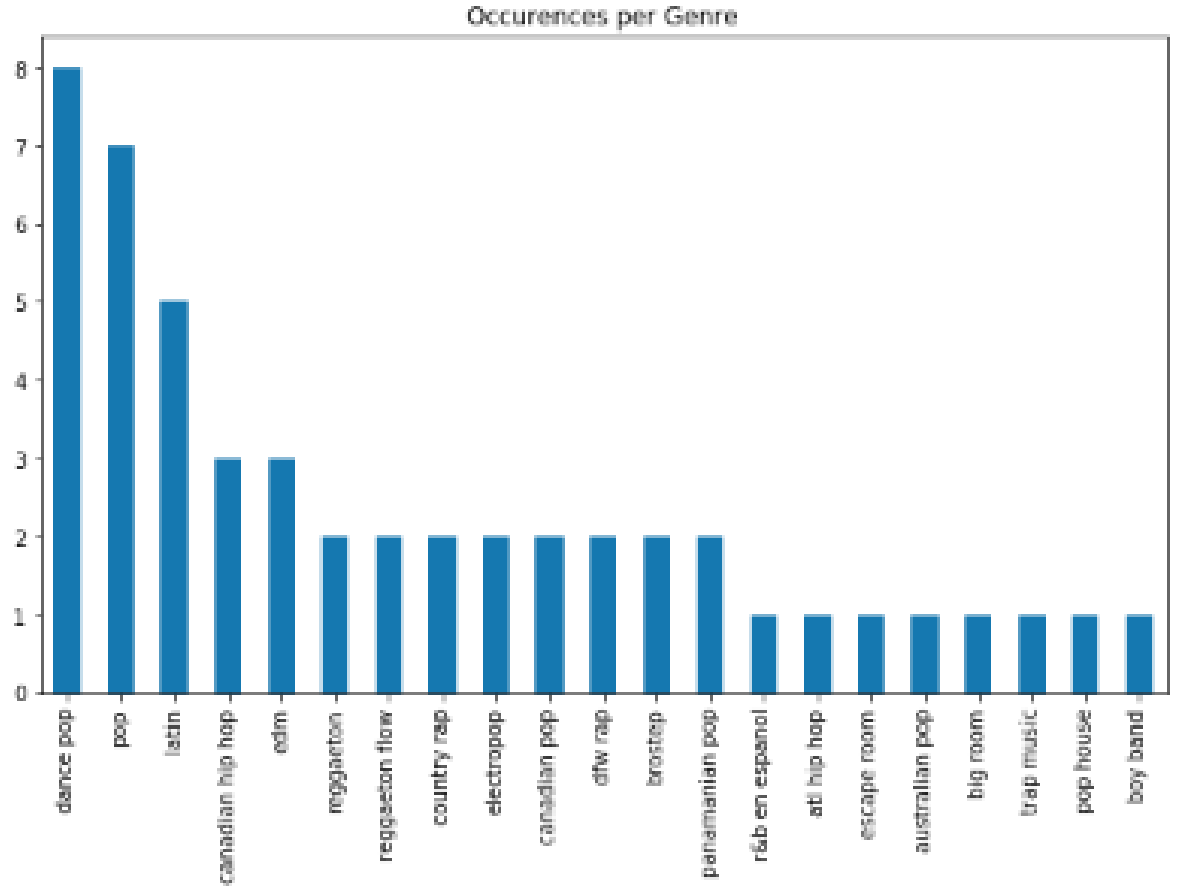
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f81101c7b50>
```



Question 10: Is there an even distribution of occurrences per genre?
Dance pop, Pop and Latin are the top 3. They had 8, 7 and 5 occurrences respectively. The rest had only 3 or less. 2 genres had 3 occurrences, 8 genres had 2 occurrences and another 8 had 1 occurrence.

In [18]:
```
genre_counts = data['Genre'].value_counts()
genre_counts.plot.bar(title='Occurences per Genre', figsize=(10,6))
```

Out[18]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f810d98f8d0>
```



6. Conclusion: Discuss limitations and future directions, lessons learned, maybe things you did not predict to find out or things you learned as you performed the analysis.

Baron Schitka has learned how better to create graphs using Seaborn and Matplot. Something that surprised him is the lack of correlation between Genre and popularity. Similarly Adam was surprised to find little correlation between BPM and energy. Some of the elements were not as useful as others, a common complaint was that for the Popularity element it was not a ranking but a rating out of 100. For example, there were multiple songs with the same popularity rating from 86-92. Elements like Danceability, Speechiness and Liveness were not as useful for data analysis as elements like Song name and Artist name.