

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

```
In [2]: songs = pd.read_csv(r"D:\Data Science\Data Science Career Readiness Program\Datasets\Dataset 1.csv")
songs.head()
```

Out[2]:

	<b>Unnamed:</b> 0	<b>name</b>	<b>album</b>	<b>release_date</b>	<b>track_number</b>	<b>id</b>
0	0	Concert Intro Music - Live	Licked Live In NYC	2022-06-10	1	2IEkywLJ4ykbhi1yRQvmsT
1	1	Street Fighting Man - Live	Licked Live In NYC	2022-06-10	2	6GVgVJBKkGJoRfarYRvGTU
2	2	Start Me Up - Live	Licked Live In NYC	2022-06-10	3	1Lu761pZ0dBTGpzxaQoZNW
3	3	If You Can't Rock Me - Live	Licked Live In NYC	2022-06-10	4	1agTQzOTUnGNggycxEqiDH
4	4	Don't Stop - Live	Licked Live In NYC	2022-06-10	5	7piGJR8YndQBQWVXv6KtQw

◀ ▶

```
In [3]: songs.shape
```

Out[3]: (1610, 18)

```
In [4]: songs.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1610 entries, 0 to 1609
Data columns (total 18 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Unnamed: 0        1610 non-null   int64  
 1   name              1610 non-null   object  
 2   album              1610 non-null   object  
 3   release_date      1610 non-null   object  
 4   track_number      1610 non-null   int64  
 5   id                1610 non-null   object  
 6   uri               1610 non-null   object  
 7   acousticness      1610 non-null   float64 
 8   danceability      1610 non-null   float64 
 9   energy             1610 non-null   float64 
 10  instrumentalness  1610 non-null   float64 
 11  liveness          1610 non-null   float64 
 12  loudness          1610 non-null   float64 
 13  speechiness       1610 non-null   float64 
 14  tempo              1610 non-null   float64 
 15  valence            1610 non-null   float64 
 16  popularity         1610 non-null   int64  
 17  duration_ms       1610 non-null   int64  
dtypes: float64(9), int64(4), object(5)
memory usage: 226.5+ KB
```

In [5]: `songs.describe()`

	Unnamed: 0	track_number	acousticness	danceability	energy	instrumentalness	livenes
<b>count</b>	1610.000000	1610.000000	1610.000000	1610.000000	1610.000000	1610.000000	1610.000
<b>mean</b>	804.500000	8.613665	0.250475	0.468860	0.792352	0.164170	0.491
<b>std</b>	464.911282	6.560220	0.227397	0.141775	0.179886	0.276249	0.349
<b>min</b>	0.000000	1.000000	0.000009	0.104000	0.141000	0.000000	0.021
<b>25%</b>	402.250000	4.000000	0.058350	0.362250	0.674000	0.000219	0.153
<b>50%</b>	804.500000	7.000000	0.183000	0.458000	0.848500	0.013750	0.379
<b>75%</b>	1206.750000	11.000000	0.403750	0.578000	0.945000	0.179000	0.893
<b>max</b>	1609.000000	47.000000	0.994000	0.887000	0.999000	0.996000	0.998

In [6]: `songs.columns`

```
Out[6]: Index(['Unnamed: 0', 'name', 'album', 'release_date', 'track_number', 'id',
               'uri', 'acousticness', 'danceability', 'energy', 'instrumentalness',
               'liveness', 'loudness', 'speechiness', 'tempo', 'valence', 'popularity',
               'duration_ms'],
              dtype='object')
```

## Clustering songs based on popularity

In [7]: `# We can divide the songs into categories based on popularity. The popularity of the songs is skewed right.`  
`# So, Let's divide the cohort into 4 parts.`

In [8]: `# Part 1`  
`part_1 = songs[songs['popularity'] <= 25]`  
`part_1.head()`

Out[8]:

	Unnamed: 0	name	album	release_date	track_number		
18	18	It's Only Rock 'N' Roll - Live	Licked Live In NYC	2022-06-10	19	1Wkbdv5UDv3sLS34WWJOAN	spotify:t
19	19	When The Whip Comes Down - Live	Licked Live In NYC	2022-06-10	20	4hTRileyEpgtFpZ64mYXkp	spot
20	20	Brown Sugar - Live	Licked Live In NYC	2022-06-10	21	2JMQKVRanp4auj0hGkw3GE	spotify
22	22	Jumpin Jack Flash - Live	Licked Live In NYC	2022-06-10	23	7p0sTERcMXc1j2Ff4QlcT	spoc
38	38	Little Red Rooster - Live At The El Mocambo 1977	Live At The El Mocambo	2022-05-13	16	1ZVvk6O2dxWKDyQIKIB2eOa	spotify

In [9]: `# Part 2`  
`part_2 = songs[(songs['popularity'] > 25) & (songs['popularity'] <= 50)]`  
`part_2.head()`

Out[9]:

	<b>Unnamed:</b> 0	<b>name</b>	<b>album</b>	<b>release_date</b>	<b>track_number</b>		<b>id</b>
0	0	Concert Intro Music - Live	Licked Live In NYC	2022-06-10	1	2IEkywLJ4ykbhi1yRQvmsT	spotify:track:
1	1	Street Fighting Man - Live	Licked Live In NYC	2022-06-10	2	6GVgVJBKkGJoRfarYRvGTU	spotify:track:6
2	2	Start Me Up - Live	Licked Live In NYC	2022-06-10	3	1Lu761pZ0dBTGpzxaQoZNW	spotify:track:1Lu
3	3	If You Can't Rock Me - Live	Licked Live In NYC	2022-06-10	4	1agTQzOTUnGNggycEqiDH	spotify:track:1ag
4	4	Don't Stop - Live	Licked Live In NYC	2022-06-10	5	7piGJR8YndQBQWVXv6KtQw	spotify:track:7pi

In [10]: # Part 3  
part\_3 = songs[(songs['popularity'] > 50) & (songs['popularity'] <= 75)]  
part\_3.head()

Out[10]:

	<b>Unnamed:</b> 0	<b>name</b>	<b>album</b>	<b>release_date</b>	<b>track_number</b>		<b>id</b>
684	684	Anybody Seen My Baby? - Remastered	Bridges To Babylon (Remastered)	1997-09-29		2	4tXr9v3K7nWW1aebWqR1B7 spc
724	724	Love Is Strong	Voodoo Lounge (Remastered 2009)	1994-07-11	1	3ls1ugfPG3hdFWTq15nNBV	sp
870	870	Heaven - Remastered	Tattoo You (2009 Re- Mastered)	1981-08-24	9	0QVJsWtmgj9SWZaLYiX2i3	s
872	872	Waiting On A Friend - Remastered 2009	Tattoo You (2009 Re- Mastered)	1981-08-24	11	4Wd9pEtEnZNDjgiswGOpJb	sp
880	880	Emotional Rescue - Remastered 2009	Emotional Rescue (2009 Re- Mastered)	1980-06-20	8	5IATbFZds3cbOx8YxuMuko	si

```
In [11]: # Part 4
part_4 = songs[(songs['popularity'] > 75) & (songs['popularity'] <= 100)]
part_4.head()
```

Out[11]:

	Unnamed: 0	name	album	release_date	track_number		id
862	862	Start Me Up - Remastered 2009	Tattoo You (2009 Re-Mastered)	1981-08-24	1	7HKez549fwJQDzx3zLjHkC	
1248	1248	Gimme Shelter	Let It Bleed	1969-12-05	1	6H3kDe7CGoWYBabAeVWGID	spo
1403	1403	Paint It, Black	Aftermath	1966-04-15	1	63T7DJ1AFDD6Bn8VzG6JE8	s
1472	1472	(I Can't Get No) Satisfaction - Mono Version	Out Of Our Heads	1965-07-30	7	2PzU4IB8Dr6mxV3IHuaG34	:

In [12]: sizes = [len(part\_1['popularity']), len(part\_2['popularity']), len(part\_3['popularity'])]

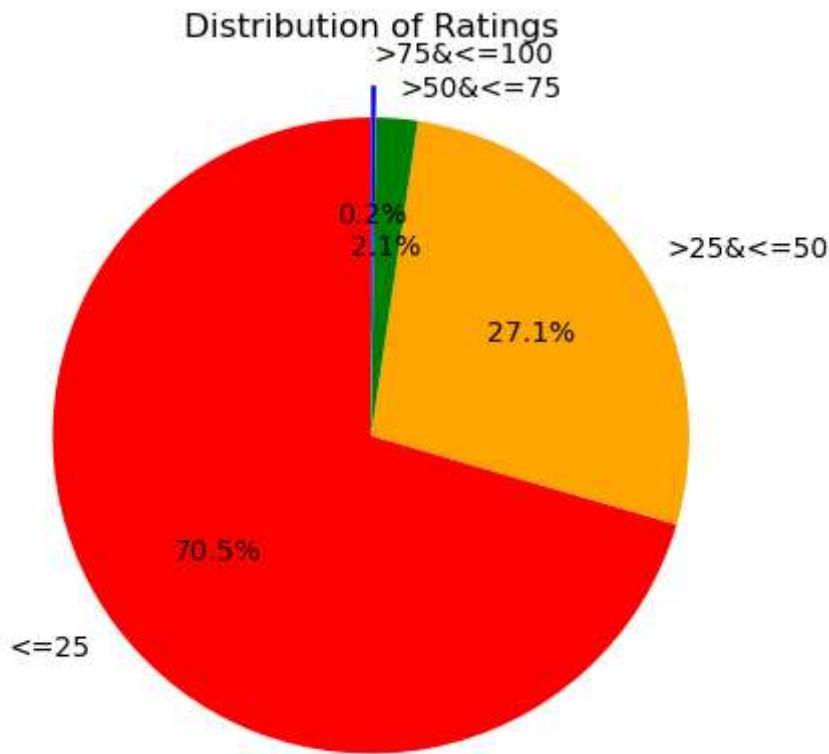
In [13]: colors = ['red', 'orange', 'green', 'blue']

In [14]: labels = ['<=25', '>25&<=50', '>50&<=75', '>75&<=100']

In [15]: x\_pos = np.arange(len(labels))

In [16]: explode = (0, 0, 0, 0.1)

In [17]: plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', startangle=90)
plt.axis('equal')
plt.title('Distribution of Ratings')
plt.show()



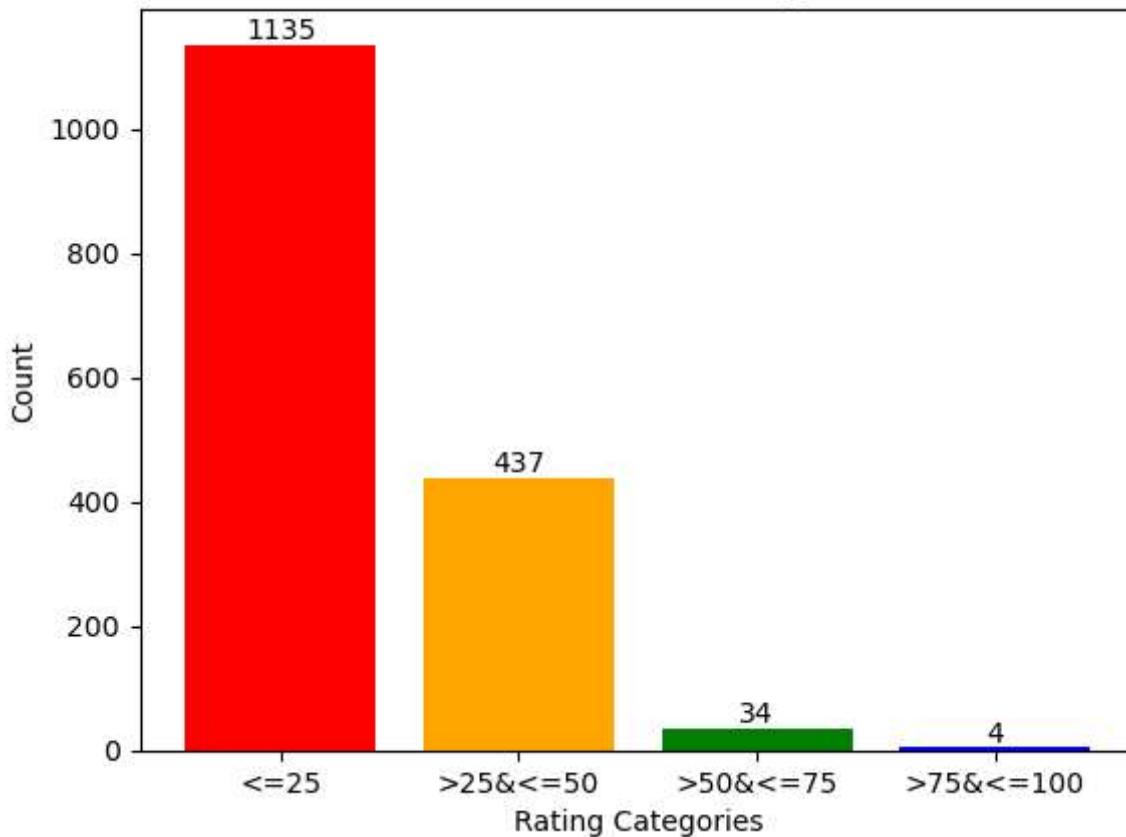
```
In [18]: plt.bar(x_pos, sizes, color=colors)
plt.xlabel('Rating Categories')
plt.ylabel('Count')
plt.title('Distribution of Ratings')

plt.xticks(x_pos, labels)

for i, size in enumerate(sizes):
    plt.text(i, size, str(size), ha='center', va='bottom')

plt.show()
```

### Distribution of Ratings



## Clustering based on Valence

```
In [19]: valence1 = songs[songs['valence'] <= 0.5]
valence1.head()
```

Out[19]:

	<b>Unnamed:</b> 0	<b>name</b>	<b>album</b>	<b>release_date</b>	<b>track_number</b>		<b>id</b>
<b>0</b>	0	Concert Intro Music - Live	Licked Live In NYC	2022-06-10		1	2IEkywLJ4ykbhi1yRQvmsT spotify:track:1IEkywLJ4ykbhi1yRQvmsT
<b>1</b>	1	Street Fighting Man - Live	Licked Live In NYC	2022-06-10		2	6GVgVJBKkGJoRfarYRvGTU spotify:track:6GVgVJBKkGJoRfarYRvGTU
<b>2</b>	2	Start Me Up - Live	Licked Live In NYC	2022-06-10		3	1Lu761pZ0dBTGpzxaQoZNW spotify:track:1Lu761pZ0dBTGpzxaQoZNW
<b>3</b>	3	If You Can't Rock Me - Live	Licked Live In NYC	2022-06-10		4	1agTQzOTUnGNggycxEqiDH spotify:track:1agTQzOTUnGNggycxEqiDH
<b>4</b>	4	Don't Stop - Live	Licked Live In NYC	2022-06-10		5	7piGJR8YndQBQWVXv6KtQw spotify:track:7piGJR8YndQBQWVXv6KtQw

In [20]: `valence1.shape`

Out[20]: (598, 18)

In [21]: `valence2 = songs[songs['valence'] > 0.5]  
valence2.head()`

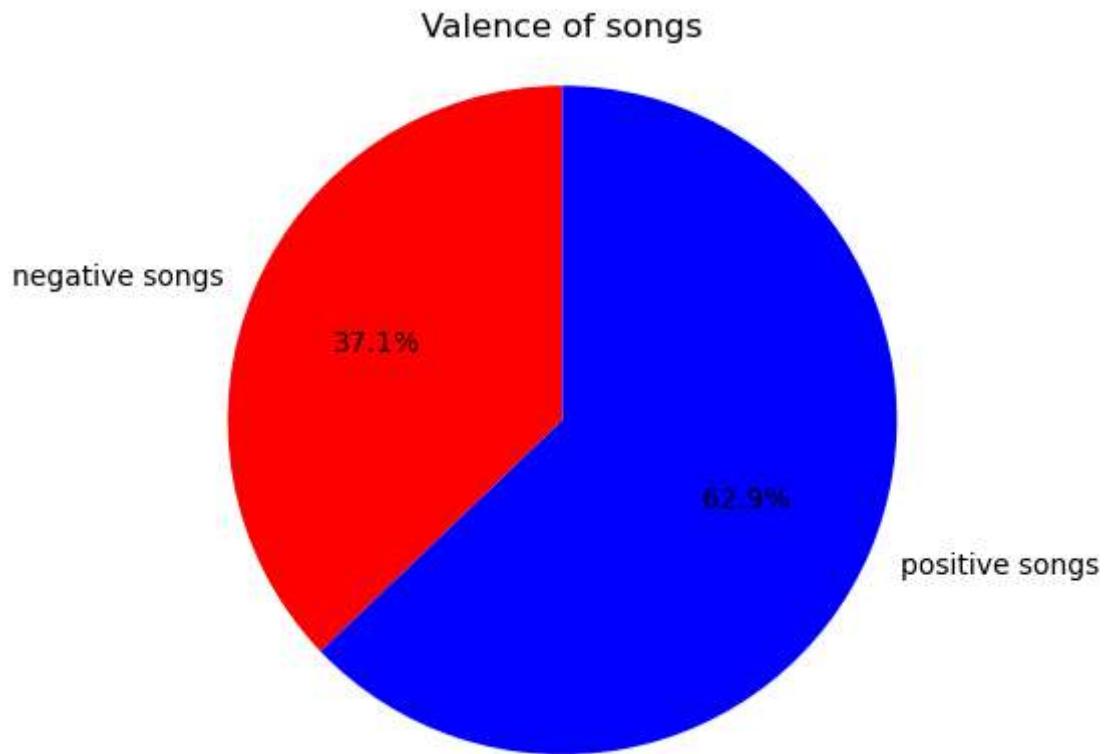
Out[21]:		Unnamed: 0	name	album	release_date	track_number		id
	15	15	Can't You Hear Me Knocking - Live	Licked Live In NYC	2022-06-10	16	2hyGhUTLHq3M5xquWrkN8s	spotify:tr
	23	23	Honky Tonk Women - Live At The El Mocambo 1977	Live At The El Mocambo	2022-05-13	1	7BPKw4y2CX3waC8IQnQDXB	spotify:tr
	24	24	All Down The Line - Live At The El Mocambo 1977	Live At The El Mocambo	2022-05-13	2	7nQvrUDwdrsksdM1Qeqf438	spotify:t
	25	25	Hand Of Fate - Live At The El Mocambo 1977	Live At The El Mocambo	2022-05-13	3	5L0g7jbPjo97KtsrYM3ep0	spoti
	26	26	Route 66 - Live At The El Mocambo 1977	Live At The El Mocambo	2022-05-13	4	0IJqBJWvI7k5I5eFLbbF8O	spoti

In [22]: `valence2.shape`

Out[22]: (1012, 18)

In [23]: `sizes = [len(valence1['valence']), len(valence2['valence'])]  
colors = ['red', 'blue']  
labels = ['negative songs', 'positive songs']  
explode = (0, 0)  
x_pos = np.arange(len(labels))`

In [24]: `plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', startangle=90)  
plt.axis('equal')  
plt.title('Valence of songs')  
plt.show()`

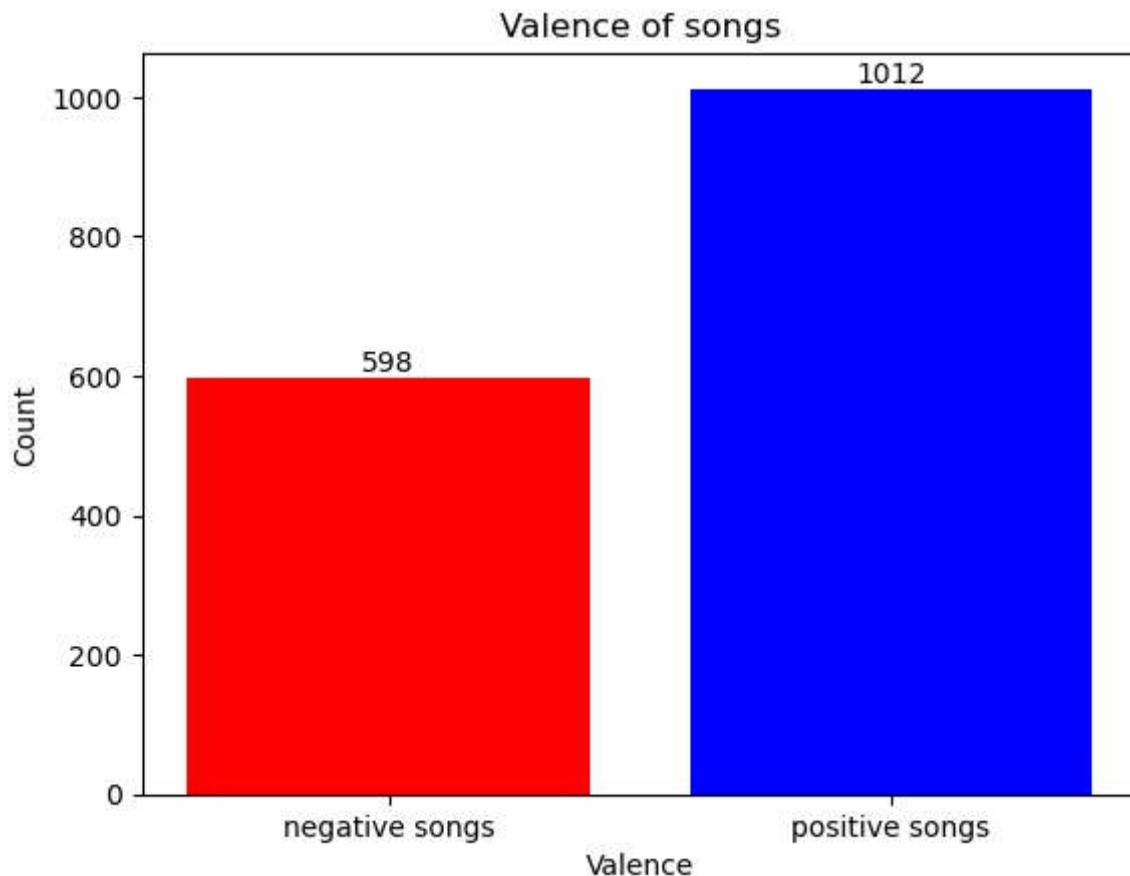


```
In [25]: plt.bar(x_pos, sizes, color=colors)
plt.xlabel('Valence')
plt.ylabel('Count')
plt.title('Valence of songs')

plt.xticks(x_pos, labels)

for i, size in enumerate(sizes):
    plt.text(i, size, str(size), ha='center', va='bottom')

plt.show()
```



## Clustering songs based on Speechiness

```
In [26]: less_music = songs[songs['speechiness'] > 0.66]
less_music.head()
```

```
Out[26]: Unnamed:
0    name  album  release_date  track_number  id  uri  acousticness  danceability  energy  i
```

```
In [27]: less_music.shape
```

```
Out[27]: (0, 18)
```

```
In [28]: medium_music = songs[(songs['speechiness'] >= 0.33) & (songs['speechiness'] <= 0.66)]
medium_music.head()
```

Out[28]:

		name	album	release_date	track_number		id
		Band Introductions - Live at Wembley Stadium 1982	Tattoo You (Super Deluxe)				
81	81			2021-10-22		16	79WmJqppHiyEeJ6S4EY4iG spotify:
280	280	Whoopi Goldberg Intro - Live	Voodoo Lounge Uncut (Live)	2018-11-16		1	4Q1mpLGI94xstgXlhoaCh5 spotify
308	308	Whoopi Goldberg Intro - Live	Voodoo Lounge Uncut (Live)	2018-11-16		1	53ccq2vLulz4HT3vAyBQDD spotify:
553	553	Charlie's Intro To Little Red Rooster - Live	Live 1965: Music From Charlie Is My Darling (L...	2012-11-06		8	4HvV7Xp4Qwg0pYQXFILFrC spotify:t
558	558	Everybody Needs Somebody To Love (Finale) - Live	Live 1965: Music From Charlie Is My Darling (L...	2012-11-06		13	0gBs7Dk1Xp7TTltv5U4BWspotify:

◀ ▶

In [29]: `medium_music.shape`Out[29]: `(5, 18)`In [30]: `more_music = songs[songs['speechiness'] < 0.33]  
more_music.head()`

Out[30]:

	<b>Unnamed:</b> 0	<b>name</b>	<b>album</b>	<b>release_date</b>	<b>track_number</b>	<b>id</b>
<b>0</b>	0	Concert Intro Music - Live	Licked Live In NYC	2022-06-10	1	2IEkywLJ4ykbhi1yRQvmsT spotify:track:1IEkywLJ4ykbhi1yRQvmsT
<b>1</b>	1	Street Fighting Man - Live	Licked Live In NYC	2022-06-10	2	6GVgVJBKkGJoRfarYRvGTU spotify:track:6GVgVJBKkGJoRfarYRvGTU
<b>2</b>	2	Start Me Up - Live	Licked Live In NYC	2022-06-10	3	1Lu761pZ0dBTGpzxaQoZNW spotify:track:1Lu761pZ0dBTGpzxaQoZNW
<b>3</b>	3	If You Can't Rock Me - Live	Licked Live In NYC	2022-06-10	4	1agTQzOTUnGNggycxEqiDH spotify:track:1agTQzOTUnGNggycxEqiDH
<b>4</b>	4	Don't Stop - Live	Licked Live In NYC	2022-06-10	5	7piGJR8YndQBQWVXv6KtQw spotify:track:7piGJR8YndQBQWVXv6KtQw

In [31]: `more_music.shape`

Out[31]: (1605, 18)

In [32]:

```
sizes = [len(less_music['speechiness']), len(medium_music['speechiness']), len(more_music['speechiness'])]
colors = ['red', 'blue', 'green']
labels = ['less music', 'medium music', 'more music']
explode = (0.1, 0.1, 0)
x_pos = np.arange(len(labels))
```

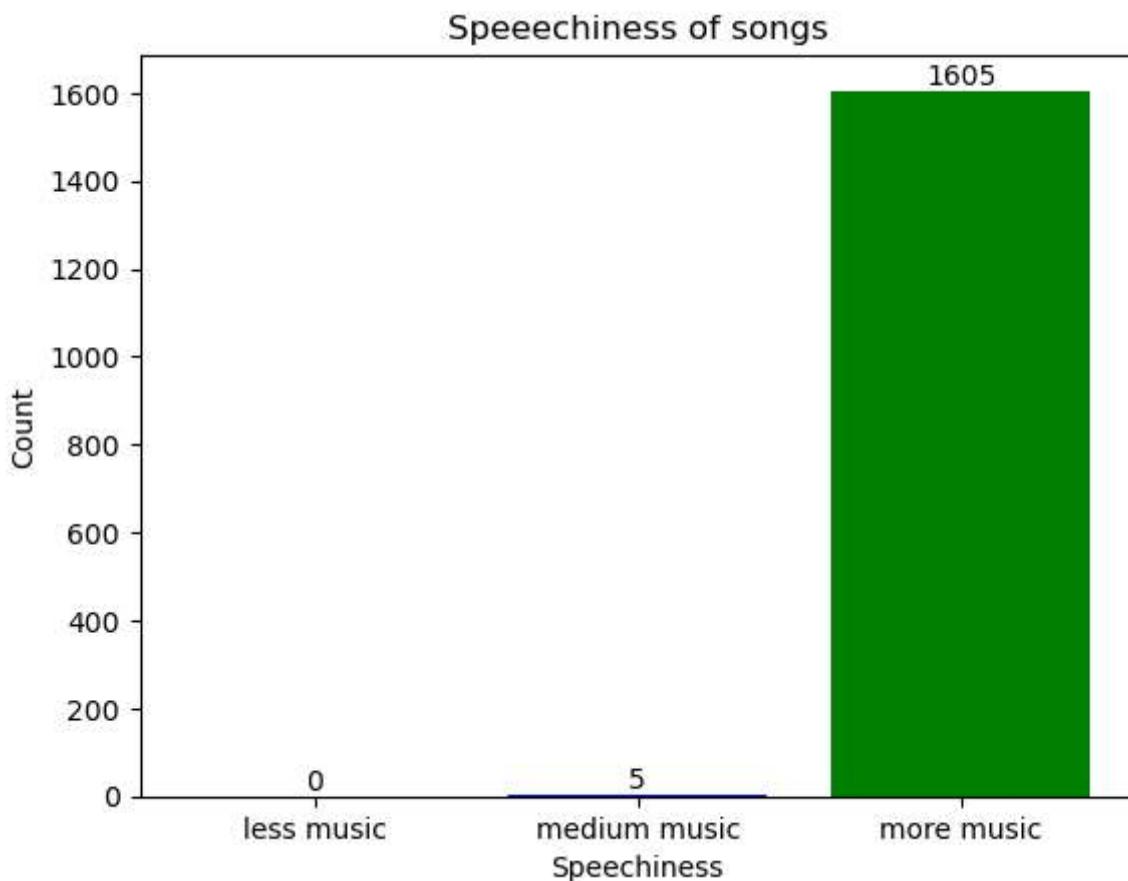
In [33]:

```
plt.bar(x_pos, sizes, color=colors)
plt.xlabel('Speechiness')
plt.ylabel('Count')
plt.title('Speechiness of songs')

plt.xticks(x_pos, labels)

for i, size in enumerate(sizes):
    plt.text(i, size, str(size), ha='center', va='bottom')

plt.show()
```



## Clustering songs based on Loudness

```
In [34]: lowDB = songs[songs['loudness'] < -40]
lowDB.head()
```

```
Out[34]: Unnamed:
0    name  album  release_date  track_number  id  uri  acousticness  danceability  energy  i
```

```
In [35]: mediumDB = songs[(songs['loudness'] >= -40) & (songs['loudness'] <= -20)]
mediumDB.head()
```

Out[35]:

	Unnamed: 0	name	album	release_date	track_number	
739	739	Continental Drift - Live	Flashpoint	1991-04-02	1	6fmxr9Wui3uYW7Gtpse
807	807	Key To The Highway - Piano Instrumental	Dirty Work	1986-03-24	11	5vaIQW6IAjNo5tUoZC
818	818	Key To The Highway - Piano Instrumental/Remast...	Dirty Work (Remastered 2009)	1986-03-24	11	3As7nxoMY2gOMxVdk:
1118	1118	Interlude A La El Hopo / Loveliest Night Of Th...	Jamming With Edward	1972-01-07	5	7hnehY2vACxvGRAarPr

In [36]: mediumDB.shape

Out[36]: (4, 18)

In [37]: highDB = songs[songs['loudness'] > -20]  
highDB.head()

Out[37]:

	Unnamed: 0	name	album	release_date	track_number		id
0	0	Concert Intro Music - Live	Licked Live In NYC	2022-06-10	1	2IEkywLJ4ykbhi1yRQvmsT	spotify:track:
1	1	Street Fighting Man - Live	Licked Live In NYC	2022-06-10	2	6GVgVJBKkGJoRfarYRvGTU	spotify:track:6
2	2	Start Me Up - Live	Licked Live In NYC	2022-06-10	3	1Lu761pZ0dBTGpzxaQoZNW	spotify:track:1Lu
3	3	If You Can't Rock Me - Live	Licked Live In NYC	2022-06-10	4	1agTQzOTUnGNggycxEqiDH	spotify:track:1ag
4	4	Don't Stop - Live	Licked Live In NYC	2022-06-10	5	7piGJR8YndQBQWVXv6KtQw	spotify:track:7pi

In [38]: highDB.shape

Out[38]: (1606, 18)

In [39]: sizes = [len(lowDB['loudness']), len(mediumDB['loudness']), len(highDB['loudness'])]  
colors = ['red', 'blue', 'green']

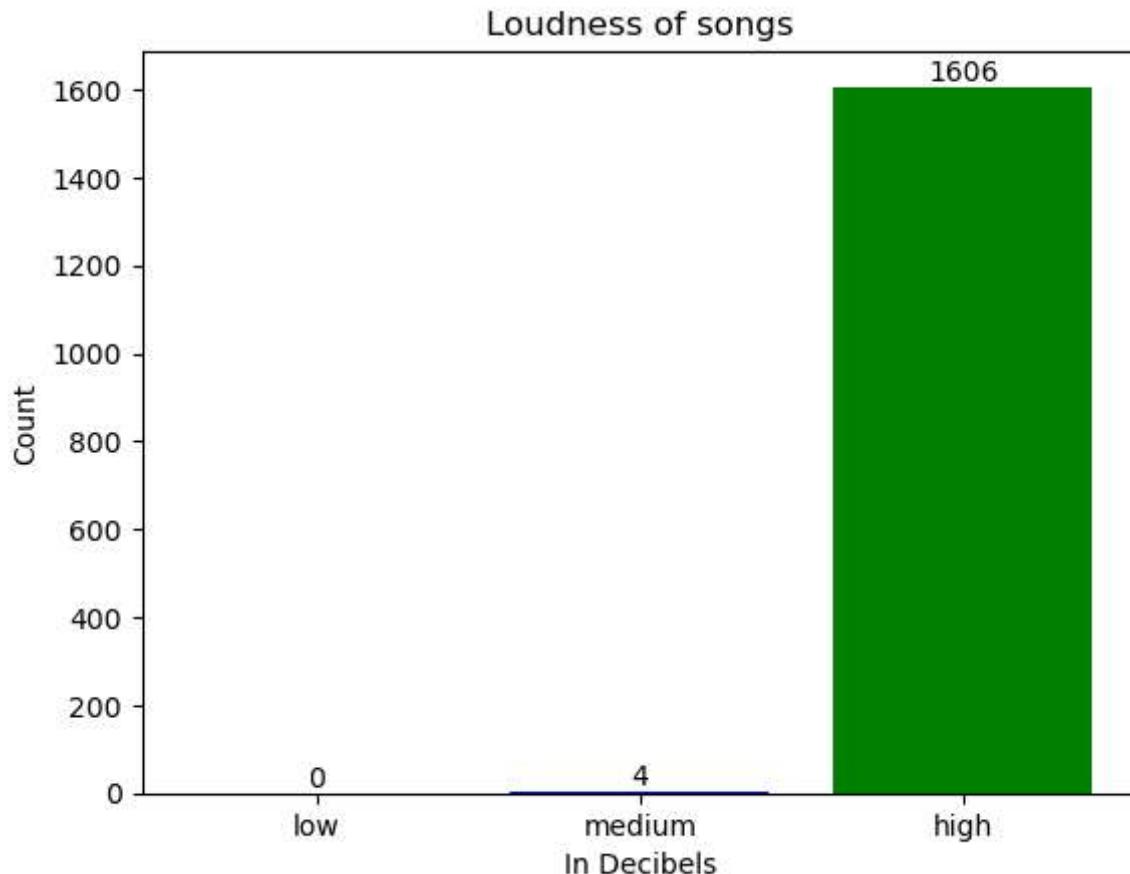
```
labels = ['low', 'medium', 'high']
explode = (0.1, 0.1, 0)
x_pos = np.arange(len(labels))
```

```
In [40]: plt.bar(x_pos, sizes, color=colors)
plt.xlabel('In Decibels')
plt.ylabel('Count')
plt.title('Loudness of songs')

plt.xticks(x_pos, labels)

for i, size in enumerate(sizes):
    plt.text(i, size, str(size), ha='center', va='bottom')

plt.show()
```



## Clustering songs based on Liveness

```
In [41]: strong = songs[songs['liveness'] > 0.8]
strong.head()
```

Out[41]:

	<b>Unnamed:</b> 0	<b>name</b>	<b>album</b>	<b>release_date</b>	<b>track_number</b>	<b>id</b>
<b>0</b>	0	Concert Intro Music - Live	Licked Live In NYC	2022-06-10	1	2IEkywLJ4ykbhi1yRQvmsT spotify:track:1IEkywLJ4ykbhi1yRQvmsT
<b>1</b>	1	Street Fighting Man - Live	Licked Live In NYC	2022-06-10	2	6GVgVJBKkGJoRfarYRvGTU spotify:track:6GVgVJBKkGJoRfarYRvGTU
<b>2</b>	2	Start Me Up - Live	Licked Live In NYC	2022-06-10	3	1Lu761pZ0dBTGpzxaQoZNW spotify:track:1Lu761pZ0dBTGpzxaQoZNW
<b>3</b>	3	If You Can't Rock Me - Live	Licked Live In NYC	2022-06-10	4	1agTQzOTUnGNggycxEqiDH spotify:track:1agTQzOTUnGNggycxEqiDH
<b>4</b>	4	Don't Stop - Live	Licked Live In NYC	2022-06-10	5	7piGJR8YndQBQWVXv6KtQw spotify:track:7piGJR8YndQBQWVXv6KtQw

In [42]: `strong.shape`

Out[42]: (497, 18)

In [43]: `weak = songs[songs['liveness'] <= 0.8]  
weak.head()`

Out[43]:		Unnamed: 0	name	album	release_date	track_number		id
	5	Monkey Man - Live	Licked Live In NYC		2022-06-10	6	3ixZ1NQk5sqvBZgWeldiqr	spotify
	23	Honky Tonk Women - Live At The El Mocambo 1977	Live At The El Mocambo		2022-05-13	1	7BPKw4y2CX3waC8IQnQDXB	spotify:tr
	25	Hand Of Fate - Live At The El Mocambo 1977	Live At The El Mocambo		2022-05-13	3	5L0g7jbPjo97KtsrYM3ep0	spotil
	26	Route 66 - Live At The El Mocambo 1977	Live At The El Mocambo		2022-05-13	4	0IjqBJWvI7k5I5eFLbbF8O	spoti
	28	Crazy Mama - Live At The El Mocambo 1977	Live At The El Mocambo		2022-05-13	6	0eumG7lbsUdmmIDr5VOSSC	spotify:tr

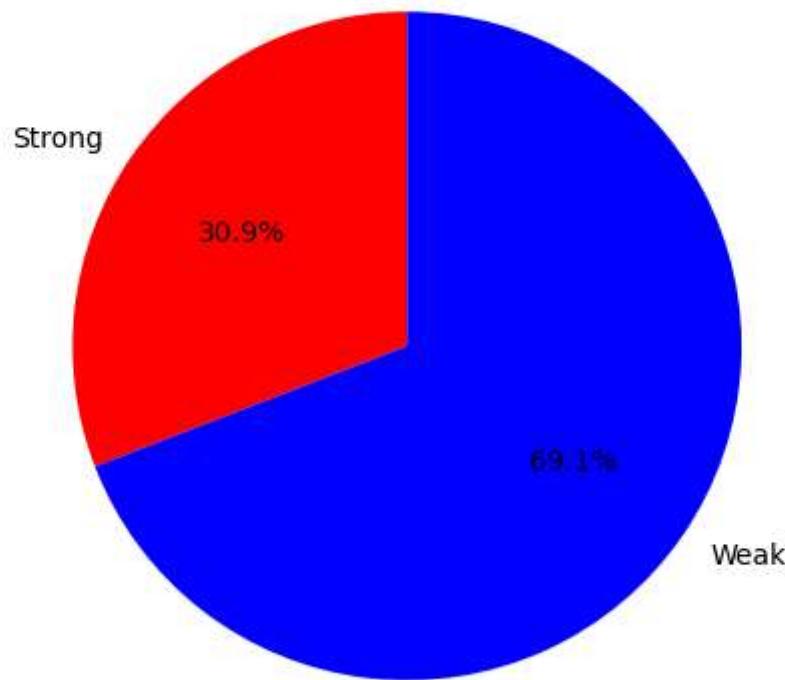
In [44]: `weak.shape`

Out[44]: `(1113, 18)`

In [45]: `sizes = [len(strong['liveness']), len(weak['liveness'])]  
colors = ['red', 'blue']  
labels = ['Strong', 'Weak']  
explode = (0, 0)  
x_pos = np.arange(len(labels))`

In [46]: `plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', startangle=90)  
plt.axis('equal')  
plt.title('Liveness of the songs')  
plt.show()`

### Liveness of the songs

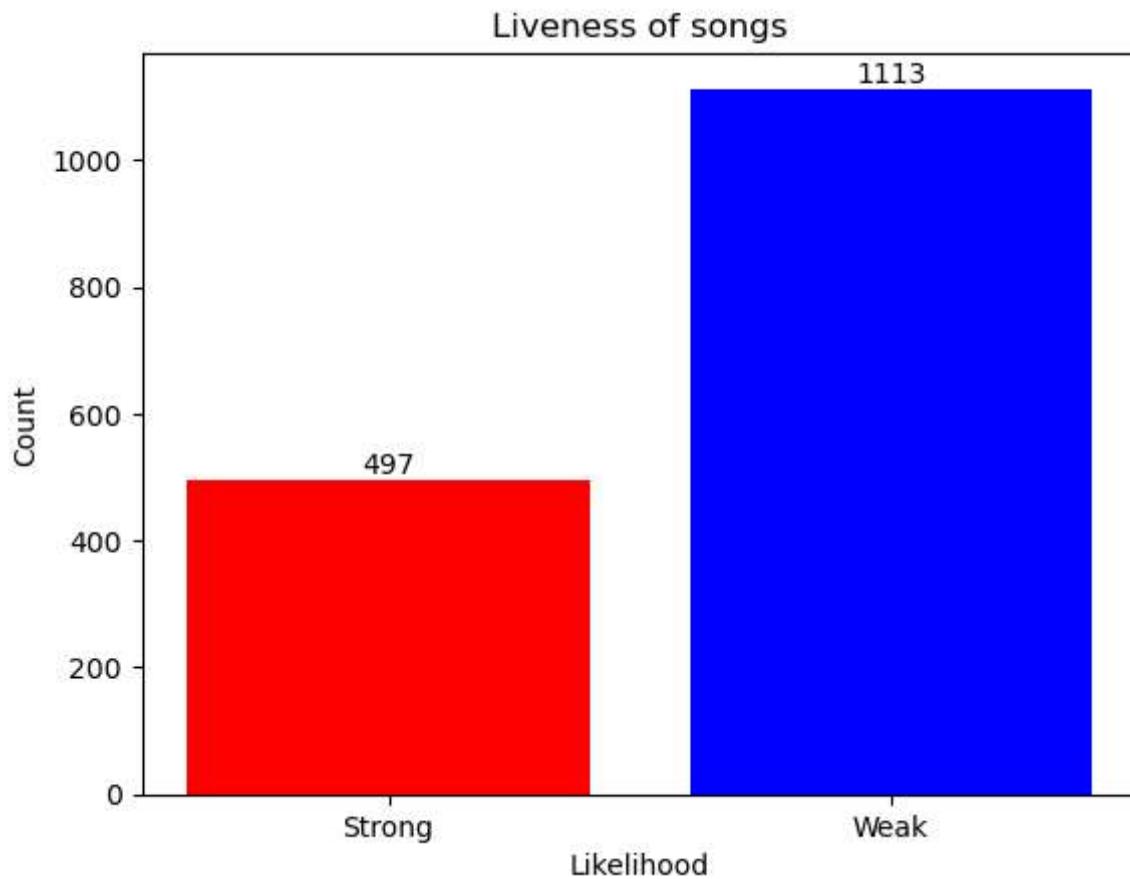


```
In [47]: plt.bar(x_pos, sizes, color=colors)
plt.xlabel('Likelihood')
plt.ylabel('Count')
plt.title('Liveness of songs')

plt.xticks(x_pos, labels)

for i, size in enumerate(sizes):
    plt.text(i, size, str(size), ha='center', va='bottom')

plt.show()
```



## Clustering songs based on Instrumentalness

```
In [48]: high = songs[songs['instrumentalness'] > 0.5]
high.head()
```

Out[48]:

		Unnamed: 0	name	album	release_date	track_number		id
0	0	Concert Intro Music - Live	Licked Live In NYC	2022-06-10	1	2IEkywLJ4ykbbhi1yRQvmsT	spot	
18	18	It's Only Rock 'N' Roll - Live	Licked Live In NYC	2022-06-10	19	1Wkbdv5UDv3sIS34WWJOAN	spotify:t	
20	20	Brown Sugar - Live	Licked Live In NYC	2022-06-10	21	2JMQKVRanp4auj0hGkw3GE	spotify	
22	22	Jumpin Jack Flash - Live	Licked Live In NYC	2022-06-10	23	7p0sTERcMXc1j2Ff4QlcT	spc	
40	40	Rip This Joint - Live At The El Mocambo 1977	Live At The El Mocambo	2022-05-13	18	6QtxrobAYG2mMfM9LcW6Y4	spotify:i	

In [49]: `high.shape`

Out[49]: (257, 18)

In [50]: `low = songs[songs['instrumentalness'] <= 0.5]  
low.head()`

Out[50]:

	<b>Unnamed:</b> 0	<b>name</b>	<b>album</b>	<b>release_date</b>	<b>track_number</b>	<b>id</b>
<b>1</b>	1	Street Fighting Man - Live	Licked Live In NYC	2022-06-10	2	6GVgVJBKkGJoRfarYRvGTU spotify:track:6
<b>2</b>	2	Start Me Up - Live	Licked Live In NYC	2022-06-10	3	1Lu761pZ0dBTGpzaQoZNW spotify:track:1Lu
<b>3</b>	3	If You Can't Rock Me - Live	Licked Live In NYC	2022-06-10	4	1agTQzOTUnGNggycxEqiDH spotify:track:1ag
<b>4</b>	4	Don't Stop - Live	Licked Live In NYC	2022-06-10	5	7piGJR8YndQBQWVXv6KtQw spotify:track:7pi
<b>5</b>	5	Monkey Man - Live	Licked Live In NYC	2022-06-10	6	3ixZ1NQk5sqvBZgWeDiqr spotify:track::

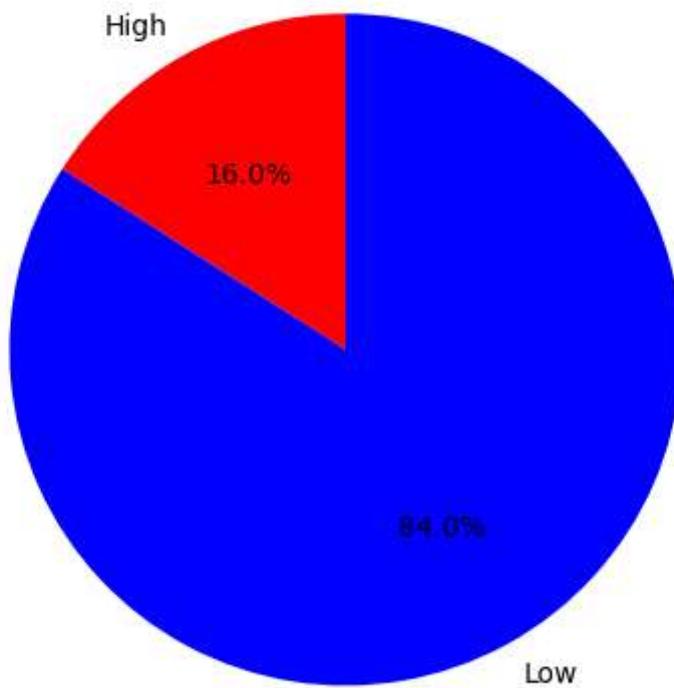
In [51]: `low.shape`

Out[51]:

```
In [52]: sizes = [len(high['instrumentalness']), len(low['instrumentalness'])]
colors = ['red', 'blue']
labels = ['High', 'Low']
explode = (0, 0)
x_pos = np.arange(len(labels))
```

```
In [53]: plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', startangle=90)
plt.axis('equal')
plt.title('Instrumentalness of the songs')
plt.show()
```

### Instrumentalness of the songs

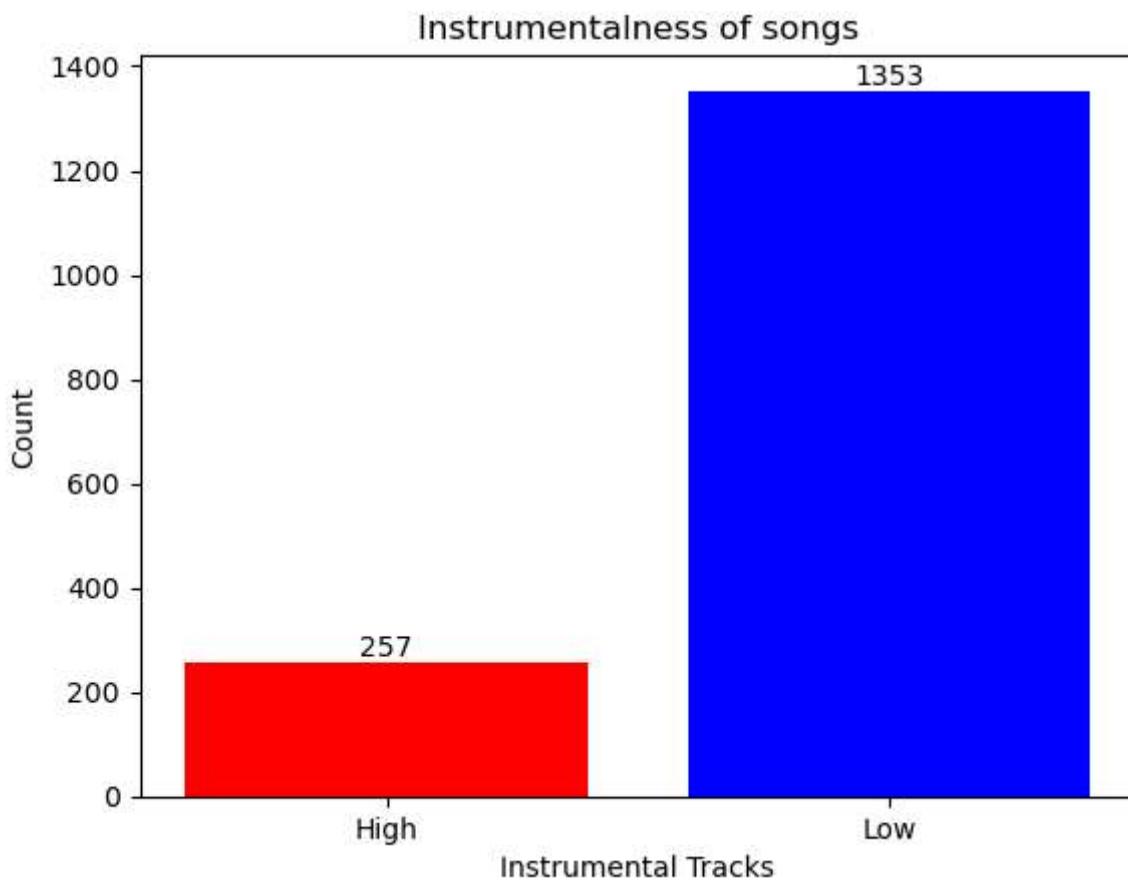


```
In [54]: plt.bar(x_pos, sizes, color=colors)
plt.xlabel('Instrumental Tracks')
plt.ylabel('Count')
plt.title('Instrumentalness of songs')

plt.xticks(x_pos, labels)

for i, size in enumerate(sizes):
    plt.text(i, size, str(size), ha='center', va='bottom')

plt.show()
```



## Clustering songs based on Energy

```
In [55]: very_high = songs[songs['energy'] >= 0.75]
very_high.shape
```

```
Out[55]: (1053, 18)
```

```
In [56]: high = songs[(songs['energy'] < 0.75) & (songs['energy'] > 0.5)]
high.shape
```

```
Out[56]: (415, 18)
```

```
In [57]: low = songs[(songs['energy'] <= 0.5) & (songs['energy'] > 0.25)]
low.shape
```

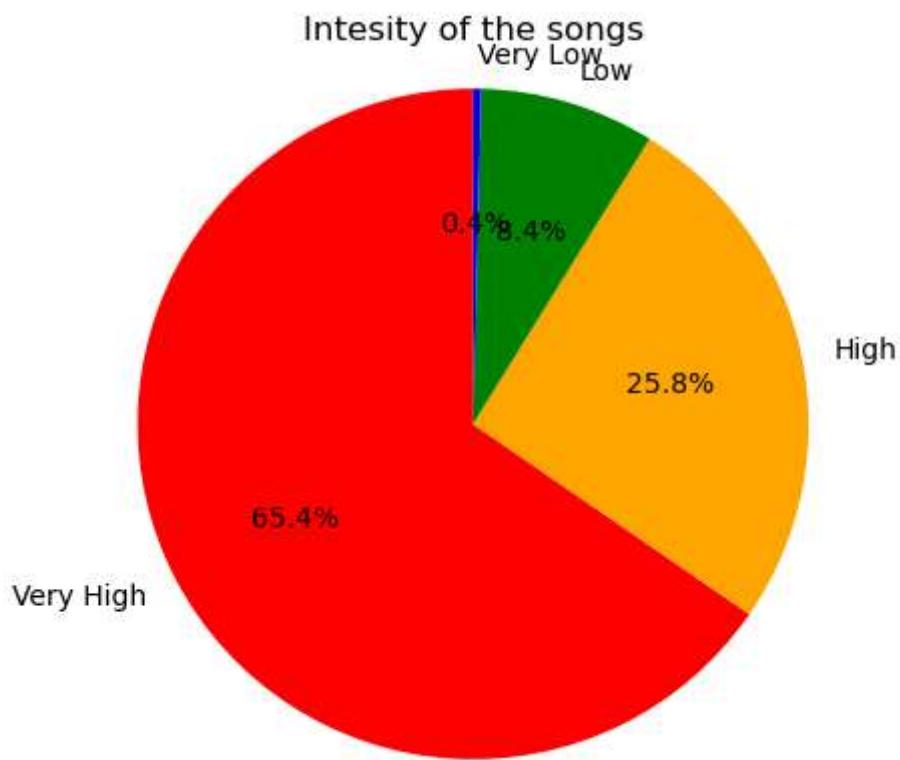
```
Out[57]: (136, 18)
```

```
In [58]: very_low = songs[songs['energy'] <= 0.25]
very_low.shape
```

```
Out[58]: (6, 18)
```

```
In [59]: sizes = [len(very_high['energy']), len(high['energy']), len(low['energy']), len(very_low['energy'])]
colors = ['red', 'orange', 'green', 'blue']
labels = ['Very High', 'High', 'Low', 'Very Low']
explode = (0, 0, 0, 0)
x_pos = np.arange(len(labels))
```

```
In [60]: plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', startangle=90)
plt.axis('equal')
plt.title('Intensity of the songs')
plt.show()
```



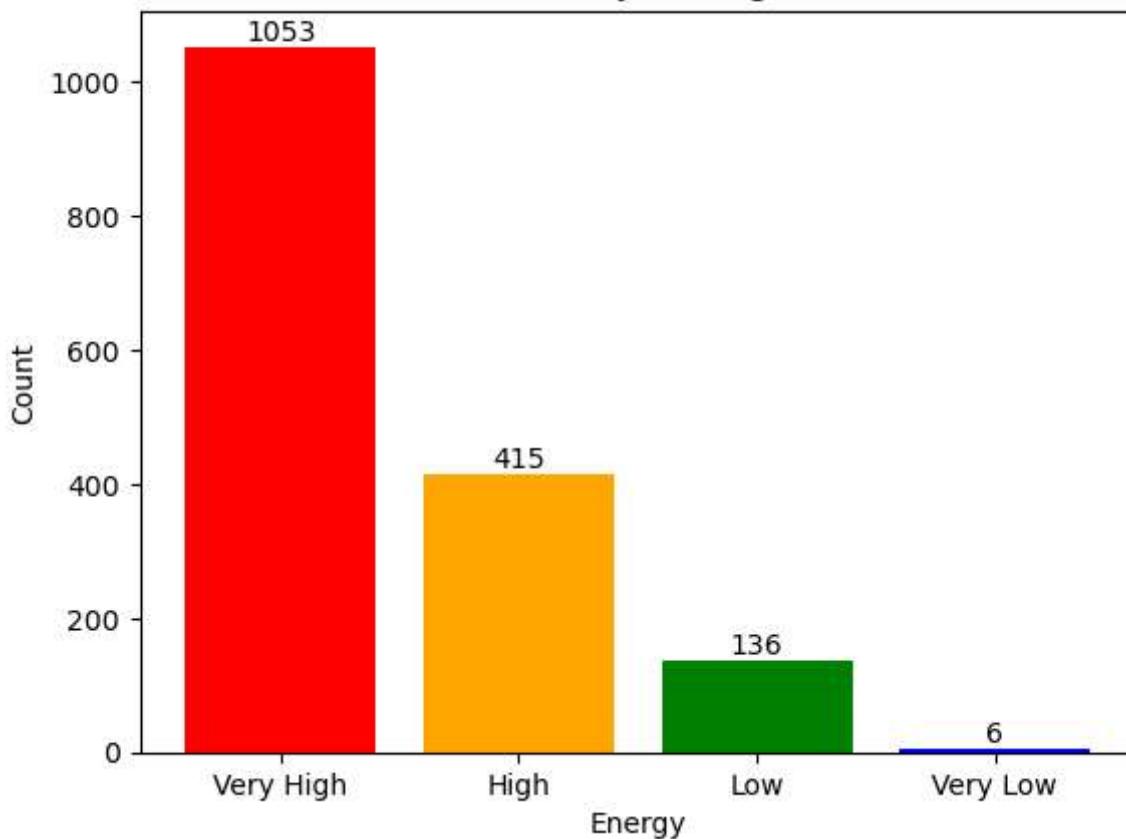
```
In [61]: plt.bar(x_pos, sizes, color=colors)
plt.xlabel('Energy')
plt.ylabel('Count')
plt.title('Intensity of songs')

plt.xticks(x_pos, labels)

for i, size in enumerate(sizes):
    plt.text(i, size, str(size), ha='center', va='bottom')

plt.show()
```

### Intensity of songs



```
In [62]: very_high = songs[songs['danceability'] >= 0.75]
very_high.shape
```

```
Out[62]: (33, 18)
```

```
In [63]: high = songs[(songs['danceability'] < 0.75) & (songs['danceability'] > 0.5)]
high.shape
```

```
Out[63]: (604, 18)
```

```
In [64]: low = songs[(songs['danceability'] <= 0.5) & (songs['danceability'] > 0.25)]
low.shape
```

```
Out[64]: (890, 18)
```

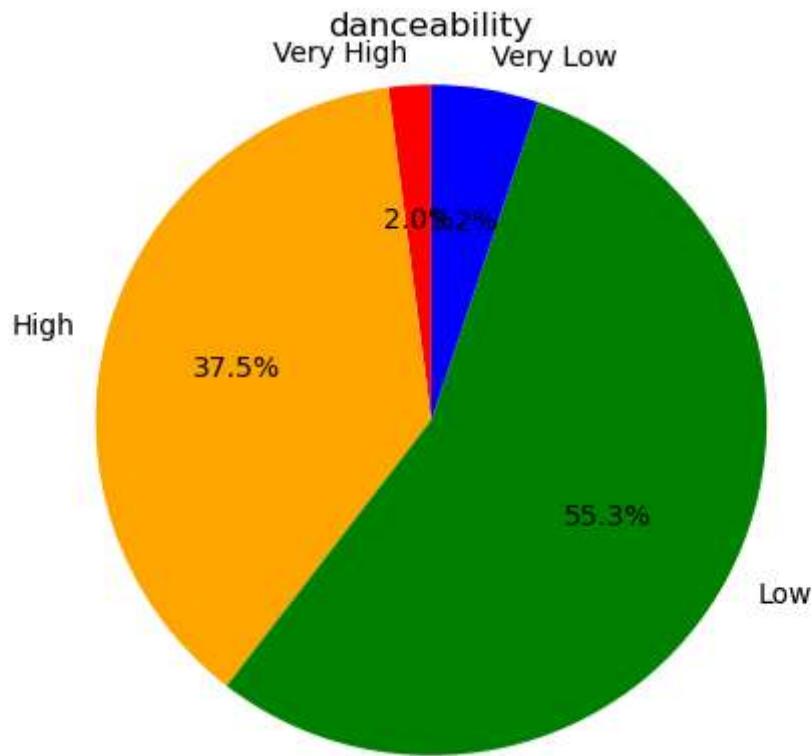
```
In [65]: very_low = songs[songs['danceability'] <= 0.25]
very_low.shape
```

```
Out[65]: (83, 18)
```

```
In [66]: sizes = [len(very_high['danceability']), len(high['danceability']), len(low['danceability']), len(very_low['danceability'])]
colors = ['red', 'orange', 'green', 'blue']
labels = ['Very High', 'High', 'Low', 'Very Low']
explode = (0, 0, 0, 0)
x_pos = np.arange(len(labels))
```

```
In [67]: plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', startangle=90)
```

```
plt.title('danceability')
plt.show()
```

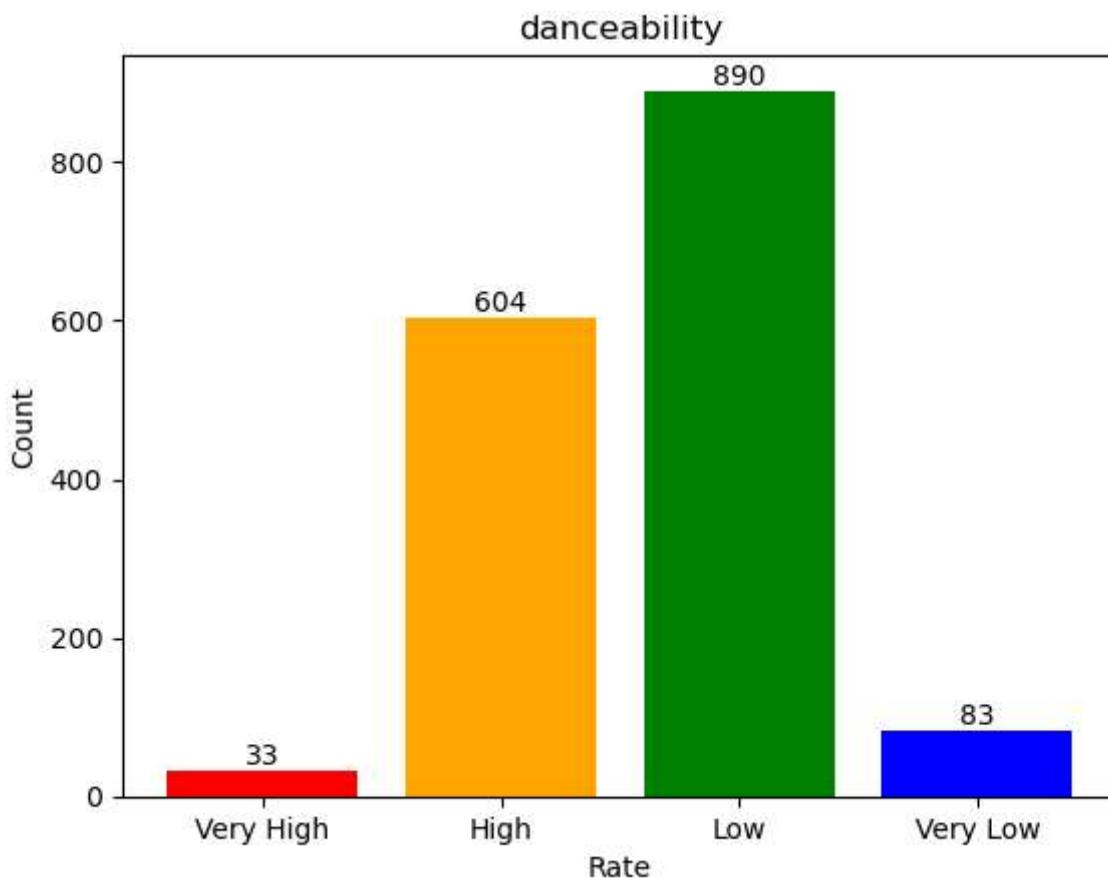


```
In [68]: plt.bar(x_pos, sizes, color=colors)
plt.xlabel('Rate')
plt.ylabel('Count')
plt.title('danceability')

plt.xticks(x_pos, labels)

for i, size in enumerate(sizes):
    plt.text(i, size, str(size), ha='center', va='bottom')

plt.show()
```



## Clustering songs based on acousticness

```
In [69]: very_high = songs[songs['acousticness'] >= 0.75]
very_high.shape
```

```
Out[69]: (43, 18)
```

```
In [70]: high = songs[(songs['acousticness'] < 0.75) & (songs['danceability'] > 0.5)]
high.shape
```

```
Out[70]: (622, 18)
```

```
In [71]: low = songs[(songs['acousticness'] <= 0.5) & (songs['danceability'] > 0.25)]
low.shape
```

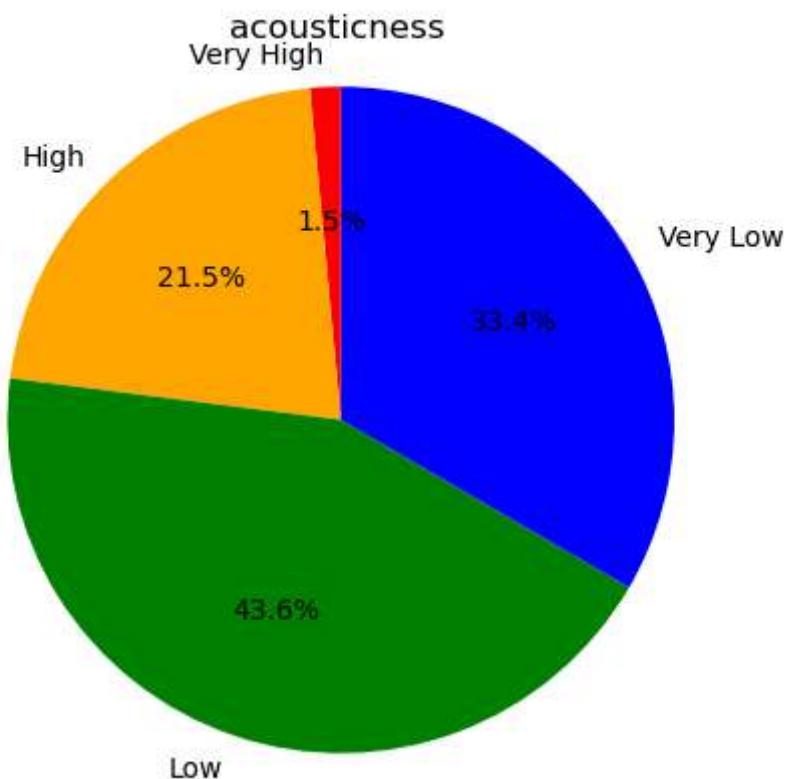
```
Out[71]: (1260, 18)
```

```
In [72]: very_low = songs[songs['acousticness'] <= 0.25]
very_low.shape
```

```
Out[72]: (965, 18)
```

```
In [73]: sizes = [len(very_high['acousticness']), len(high['acousticness']), len(low['acousticness']), len(very_low['acousticness'])]
colors = ['red', 'orange', 'green', 'blue']
labels = ['Very High', 'High', 'Low', 'Very Low']
explode = (0, 0, 0, 0)
x_pos = np.arange(len(labels))
```

```
In [74]: plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', startangle=90)
plt.axis('equal')
plt.title('acousticness')
plt.show()
```

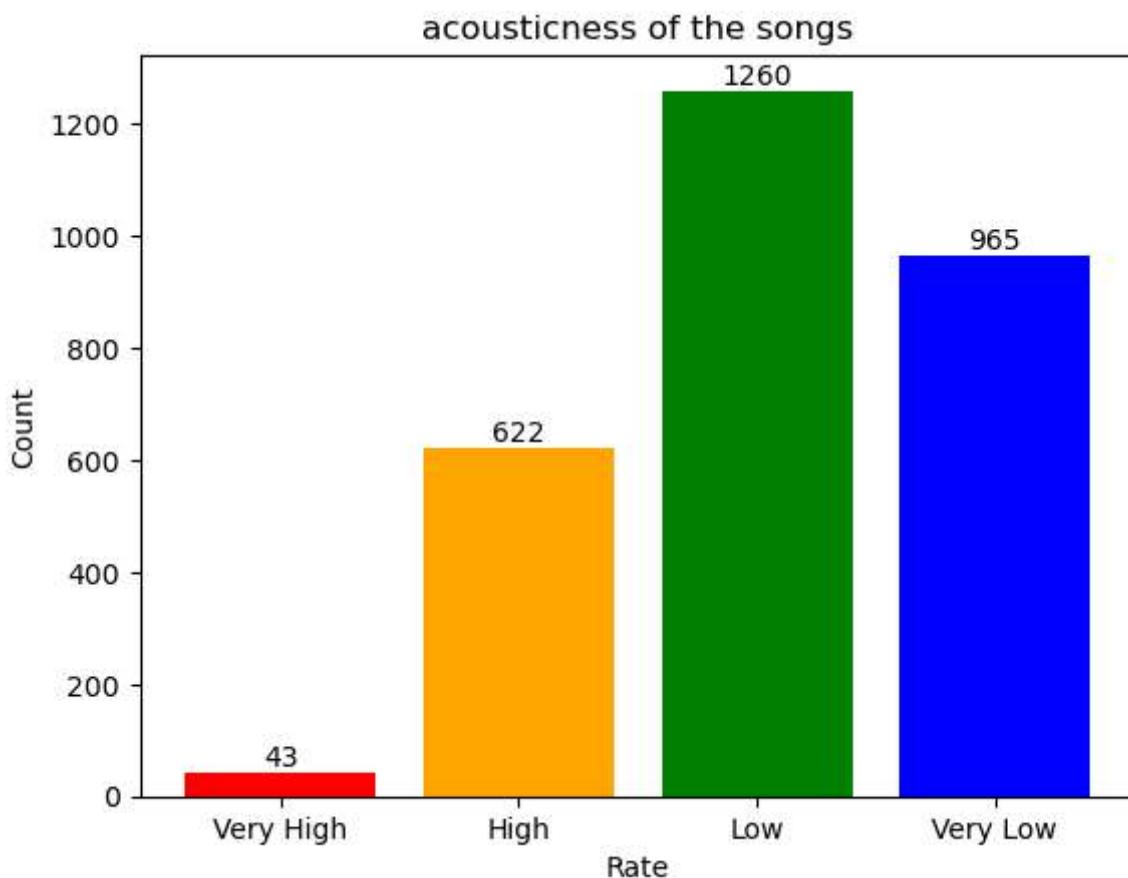


```
In [75]: plt.bar(x_pos, sizes, color=colors)
plt.xlabel('Rate')
plt.ylabel('Count')
plt.title('acousticness of the songs')

plt.xticks(x_pos, labels)

for i, size in enumerate(sizes):
    plt.text(i, size, str(size), ha='center', va='bottom')

plt.show()
```



Like this, we can divide the songs based on the various attributes that have been given.

and we can create number of clusters of songs depending on the features.

## CASE STUDY

Let's create a list of songs where the conditions for attributes of the songs are given below

Popularity - Medium

Valence - Positive

Speechiness - Less Speech and more Music

Loudness - High

Liveness - Strong

## Instrumentalness - Moderate

### Energy - High

### Danceability - Medium

### Acousticness - Medium

```
In [76]: songs[(songs['popularity'] <= 50) &
             (songs['valence'] >= 0.4) &
             (songs['speechiness'] <= 0.33) &
             (songs['loudness'] >= -30) &
             (songs['liveness'] >= 0.6) &
             (songs['instrumentalness'] >= 0.3) & (songs['instrumentalness'] <= 0.7) &
             (songs['energy'] >= 0.4) &
             (songs['danceability'] >= 0.3) & (songs['danceability'] <= 0.7) &
             (songs['acousticness'] >= 0.3) & (songs['acousticness'] <= 0.7)]
```

		Unnamed: 0	name	album	release_date	track_number		id
18	18	It's Only Rock 'N' Roll - Live	Licked Live In NYC	2022-06-10		19	1Wkbdv5UDv3sIS34WWJOAN	s
89	89	Start Me Up - Live at Wembley Stadium 1982	Tattoo You (Super Deluxe)	2021-10-22		24	4FbMzbRKaa8hTuDUHtLv6A	
281	281	Not Fade Away - Live	Voodoo Lounge Uncut (Live)	2018-11-16		2	1EdD9H50BIPO8eHD7oEXFC	
309	309	Not Fade Away - Live	Voodoo Lounge Uncut (Live)	2018-11-16		2	6YCOFkL6CC9gVXcqeRK9wQ	
839	839	Intro (Take The A Train) - Live / Remastered 2009	Still Life	1982-06-01		1	2cSjyaVx2RRLj1zn5tdB0C	
960	960	Around And Around - Live	Love You Live (Remastered)	1977-09-23		4	1wbrc6umQQ3StMmSF1uKEY	s
1523	1523	Oh Baby (We Got A Good Thing Goin') - Remaster...	Now!	1965-02-13		10	2cErrLqmekBoFhXXFNHTLQ	

In [ ]: