

Music's importance spans over 30,000 years in human history. The link between music and mental health is extensively discussed due to its profound impact on well-being.



1. Music therapy (MT) utilizes music to enhance stress reduction, mood, and mental well-being. It's evidence-based, promoting "happy" hormones like oxytocin.
2. Research highlights music therapy's advantages for mental health issues like depression, trauma, schizophrenia. Music aids emotional processing, acting as a calming tool for anxiety and dysregulation.

About the data set...

1. This dataset is taken from Kaggle.
2. Survey results on music taste and self-reported mental health

Problem statement :

1. The dataset seeks to uncover potential correlations between an individual's musical preferences and their self-reported mental health status. The goal is to identify if certain genres of music have any influence on mental well-being. These findings could lead to a better understanding of how music interacts with mental health, possibly leading to more informed applications of Music Therapy (MT) or offering intriguing insights into the complexities of the human mind..

Attribute Datasets:

1. Timestamp : Date and time when form was submitted
2. Age : Respondent's age
3. Primary streaming service : Respondent's primary streaming service
4. Hours per day : Number of hours the respondent listens to music per day while working
5. While working : Does the respondent listen to music while studying/working?
6. Instrumentalist: Does the respondent play an instrument regularly?
7. Composer : Does the respondent compose music?
8. Fav genre : Respondent's favorite or top genre
9. Exploratory : Does the respondent actively explore new artists/genres?
10. Foreign languages : Does the respondent regularly listen to music with lyrics in a language they are not fluent in?
11. BPM : Beats per minute of favorite genre
12. Frequency [Classical] : How frequently the respondent listens to classical music
13. Frequency [Country] : How frequently the respondent listens to country music
14. Frequency [EDM] : How frequently the respondent listens to EDM music
15. Frequency [Folk] : How frequently the respondent listens to folks music
16. Frequency [Gospel] : How frequently the respondent listens to Gospel music
17. Frequency [Hip hop] : How frequently the respondent listens to hip hop music

18. Frequency [Jazz] : How frequently the respondent listens to jazz music
19. Frequency [K pop] : How frequently the respondent listens to K pop music
20. Frequency [Latin] : How frequently the respondent listens to Latin music
21. Frequency [Lofi] : How frequently the respondent listens to lofi music
22. Frequency [Metal] : How frequently the respondent listens to metal music
23. Frequency [Pop] : How frequently the respondent listens to pop music
24. Frequency [R&B] : How frequently the respondent listens to R&B music
25. Frequency [Rap] : How frequently the respondent listens to rap music
26. Frequency [Rock] : How frequently the respondent listens to rock music
27. Frequency [Video game music] : How frequently the respondent listens to video game music
28. Anxiety : Self-reported anxiety, on a scale of 0-10
29. Depression : Self-reported depression, on a scale of 0-10
30. Insomnia : Self-reported insomnia, on a scale of 0-10
31. OCD : Self-reported OCD, on a scale of 0-10
32. Music effects : Does music improve/worsen respondent's mental health conditions?
33. Permissions : Permissions to publicize data

Import necessary libraries

In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

import warnings

warnings.filterwarnings("ignore")
#sns.set(palette="magma")
sns.set(color_codes = True)
sns.set_palette(palette = 'magma', n_colors = 8)
```

In [2]:

```
color1 = sns.color_palette("magma",3)
color2 = sns.color_palette("magma")
```

Load Dataset

In [3]:

```
df = pd.read_csv(r"C:\Users\Dell\Downloads\archive (4)\mxmh_survey_results.csv")
```

In [4]:

```
df.head()
```

Out[4]:

	Timestamp	Age	Primary streaming service	Hours per day	While working	Instrumentalist	Composer	Fav genre	Exploratory	Foreign languages	...	Frequency [R&B]
0	8/27/2022 19:29:02	18.0	Spotify	3.0	Yes	Yes	Yes	Latin	Yes	Yes	...	Sometimes
1	8/27/2022 19:57:31	63.0	Pandora	1.5	Yes	No	No	Rock	Yes	No	...	Sometimes
2	8/27/2022 21:28:18	18.0	Spotify	4.0	No	No	No	Video game music	No	Yes	...	Never
3	8/27/2022 21:40:40	61.0	YouTube Music	2.5	Yes	No	Yes	Jazz	Yes	Yes	...	Sometimes
4	8/27/2022 21:54:47	18.0	Spotify	4.0	Yes	No	No	R&B	Yes	No	...	Very frequently

5 rows × 33 columns

In [5]:

```
# data set size check
df.shape
```

Out[5]: (736, 33)

```
In [6]: df.duplicated().sum()
```

```
Out[6]: 0
```

```
In [7]: # check unqiues values as per dataset
```

```
for i in df.columns:
    print(i)
    print(df[i].unique())
Frequency [Rap]
['Sometimes' 'Never' 'Very frequently' 'Rarely']
Frequency [Rap]
['Very frequently' 'Rarely' 'Never' 'Sometimes']
Frequency [Rock]
['Never' 'Very frequently' 'Rarely' 'Sometimes']
Frequency [Video game music]
['Sometimes' 'Rarely' 'Very frequently' 'Never']
Anxiety
[ 3.  7.  9.  8.  4.  5.  2.  1.  6. 10.  0.  7.5]
Depression
[ 0.  2.  7.  8.  3.  1.  4.  5.  9. 10.  6.  3.5]
Insomnia
[ 1.  2. 10.  3.  5.  7.  6.  0.  4.  8.  9.  3.5]
OCD
[ 0.  1.  2.  3.  9.  7.  4.  6. 10.  5.  8.  8.5  5.5]
Music effects
[nan 'No effect' 'Improve' 'Worsen']
Permissions
['I understand.']
```

```
In [8]: # check null values
```

```
df.isna().sum()
```

```
Out[8]: Timestamp      0
Age                  1
Primary streaming service  1
Hours per day        0
While working        3
Instrumentalist       4
Composer             1
Fav genre            0
Exploratory          0
Foreign languages    4
BPM                 107
Frequency [Classical]  0
Frequency [Country]   0
Frequency [EDM]       0
Frequency [Folk]      0
Frequency [Gospel]    0
Frequency [Hip hop]   0
Frequency [Jazz]      0
Frequency [K pop]     0
Frequency [Latin]     0
Frequency [Lofi]      0
Frequency [Metal]     0
Frequency [Pop]       0
Frequency [R&B]       0
Frequency [Rap]       0
Frequency [Rock]      0
Frequency [Video game music]  0
Anxiety              0
Depression           0
Insomnia             0
OCD                  0
Music effects        8
Permissions          0
dtype: int64
```

Missing values:

To handle missing values: delete if insignificant, impute with mean/median/mode, regression, or KNN; use forward/backward-fill for time-series, interpolate, apply domain knowledge, or create indicator variable. Choose based on data and analysis goals.

```
In [9]: df["Age"] = df["Age"].fillna(df["Age"].median())
df["Primary streaming service"] = df["Primary streaming service"].fillna(df["Primary streaming service"].mode()[0])
df["While working"] = df["While working"].fillna(df["While working"].mode()[0])
df["Instrumentalist"] = df["Instrumentalist"].fillna(df["Instrumentalist"].mode()[0])
df["Composer"] = df["Composer"].fillna(df["Composer"].mode()[0])
df["Foreign languages"] = df["Foreign languages"].fillna(df["Foreign languages"].mode()[0])
#df["BPM"] = df["BPM"].fillna(df["BPM"].median())
df["Music effects"] = df["Music effects"].fillna(df["Music effects"].mode()[0])
```

```
In [10]: df.isna().sum()
```

```
Out[10]: Timestamp                0
Age                                0
Primary streaming service          0
Hours per day                      0
While working                     0
Instrumentalist                   0
Composer                         0
Fav genre                        0
Exploratory                      0
Foreign languages                 0
BPM                             107
Frequency [Classical]             0
Frequency [Country]               0
Frequency [EDM]                   0
Frequency [Folk]                  0
Frequency [Gospel]                0
Frequency [Hip hop]               0
Frequency [Jazz]                  0
Frequency [K pop]                 0
Frequency [Latin]                 0
Frequency [Lofi]                  0
Frequency [Metal]                 0
Frequency [Pop]                   0
Frequency [R&B]                   0
Frequency [Rap]                   0
Frequency [Rock]                  0
Frequency [Video game music]      0
Anxiety                          0
Depression                       0
Insomnia                         0
OCD                              0
Music effects                     0
Permissions                       0
dtype: int64
```

```
In [11]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 736 entries, 0 to 735
Data columns (total 33 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Timestamp                             736 non-null    object
1   Age                                    736 non-null    float64
2   Primary streaming service             736 non-null    object
3   Hours per day                         736 non-null    float64
4   While working                         736 non-null    object
5   Instrumentalist                       736 non-null    object
6   Composer                             736 non-null    object
7   Fav genre                             736 non-null    object
8   Exploratory                           736 non-null    object
9   Foreign languages                     736 non-null    object
10  BPM                                   629 non-null    float64
11  Frequency [Classical]                 736 non-null    object
12  Frequency [Country]                  736 non-null    object
13  Frequency [EDM]                       736 non-null    object
14  Frequency [Folk]                      736 non-null    object
15  Frequency [Gospel]                   736 non-null    object
16  Frequency [Hip hop]                   736 non-null    object
17  Frequency [Jazz]                      736 non-null    object
18  Frequency [K pop]                     736 non-null    object
19  Frequency [Latin]                     736 non-null    object
20  Frequency [Lofi]                      736 non-null    object
21  Frequency [Metal]                     736 non-null    object
22  Frequency [Pop]                       736 non-null    object
23  Frequency [R&B]                       736 non-null    object
24  Frequency [Rap]                       736 non-null    object
25  Frequency [Rock]                      736 non-null    object
26  Frequency [Video game music]          736 non-null    object
27  Anxiety                               736 non-null    float64
28  Depression                            736 non-null    float64
29  Insomnia                              736 non-null    float64
30  OCD                                   736 non-null    float64
31  Music effects                         736 non-null    object
32  Permissions                           736 non-null    object
dtypes: float64(7), object(26)
memory usage: 189.9+ KB
```

```
In [12]: # change datatypes paricaler column
df["Age"] = df["Age"].astype(int)
```

```
In [13]: df.head(2)
```

Out[13]:

	Timestamp	Age	Primary streaming service	Hours per day	While working	Instrumentalist	Composer	Fav genre	Exploratory	Foreign languages	...	Frequency [R&B]
0	8/27/2022 19:29:02	18	Spotify	3.0	Yes	Yes	Yes	Latin	Yes	Yes	...	Sometimes
1	8/27/2022 19:57:31	63	Pandora	1.5	Yes	No	No	Rock	Yes	No	...	Sometimes

2 rows × 33 columns

Describe:

Handling missing values: addressing empty data slots by deleting, imputing with estimations, or using domain knowledge for accuracy.

In [14]:

df.describe().T

Out[14]:

	count	mean	std	min	25%	50%	75%	max
Age	736.0	2.520109e+01	1.204776e+01	10.0	18.0	21.0	28.0	89.0
Hours per day	736.0	3.572758e+00	3.028199e+00	0.0	2.0	3.0	5.0	24.0
BPM	629.0	1.589948e+06	3.987261e+07	0.0	100.0	120.0	144.0	999999999.0
Anxiety	736.0	5.837636e+00	2.793054e+00	0.0	4.0	6.0	8.0	10.0
Depression	736.0	4.796196e+00	3.028870e+00	0.0	2.0	5.0	7.0	10.0
Insomnia	736.0	3.738451e+00	3.088689e+00	0.0	1.0	3.0	6.0	10.0
OCD	736.0	2.637228e+00	2.842017e+00	0.0	0.0	2.0	5.0	10.0

In [15]:

df.describe(include="object").T

Out[15]:

	count	unique	top	freq
Timestamp	736	735	8/28/2022 16:15:08	2
Primary streaming service	736	6	Spotify	459
While working	736	2	Yes	582
Instrumentalist	736	2	No	501
Composer	736	2	No	610
Fav genre	736	16	Rock	188
Exploratory	736	2	Yes	525
Foreign languages	736	2	Yes	408
Frequency [Classical]	736	4	Rarely	259
Frequency [Country]	736	4	Never	343
Frequency [EDM]	736	4	Never	307
Frequency [Folk]	736	4	Never	292
Frequency [Gospel]	736	4	Never	535
Frequency [Hip hop]	736	4	Sometimes	218
Frequency [Jazz]	736	4	Never	261
Frequency [K pop]	736	4	Never	416
Frequency [Latin]	736	4	Never	443
Frequency [Lofi]	736	4	Never	280
Frequency [Metal]	736	4	Never	264
Frequency [Pop]	736	4	Very frequently	277
Frequency [R&B]	736	4	Never	225
Frequency [Rap]	736	4	Rarely	215
Frequency [Rock]	736	4	Very frequently	330
Frequency [Video game music]	736	4	Never	236
Music effects	736	3	Improve	550
Permissions	736	1	I understand.	736

Remove outliers

In [16]:

age_index = df[(df['Age'] > 70)].index
df = df.drop(age_index, axis=0)

In [17]:

bpm_index = df[df['BPM'] > 200].index
df = df.drop(bpm_index, axis=0)

```
In [18]: hpd_index = df[df['Hours per day'] >= 15].index
df = df.drop(hpd_index, axis=0)
```

```
In [19]: miss_bpm_genre = ['Country', 'Hip hop', 'Pop', 'Rock', 'Metal', 'Classical',
                          'Video game music', 'R&B', 'Folk', 'Jazz', 'EDM', 'Gospel',
                          'K pop', 'Rap']
```

```
In [20]: for i in miss_bpm_genre:
df['BPM'] = df['BPM'].fillna(round(df[df['Fav genre']== i]['BPM'].mode(), 0))
df['BPM']
```

Out[20]:

0	156.0
1	119.0
2	132.0
3	84.0
4	107.0
	...
731	120.0
732	160.0
733	120.0
734	170.0
735	98.0

Name: BPM, Length: 714, dtype: float64

```
In [21]: df.head(2)
```

Out[21]:

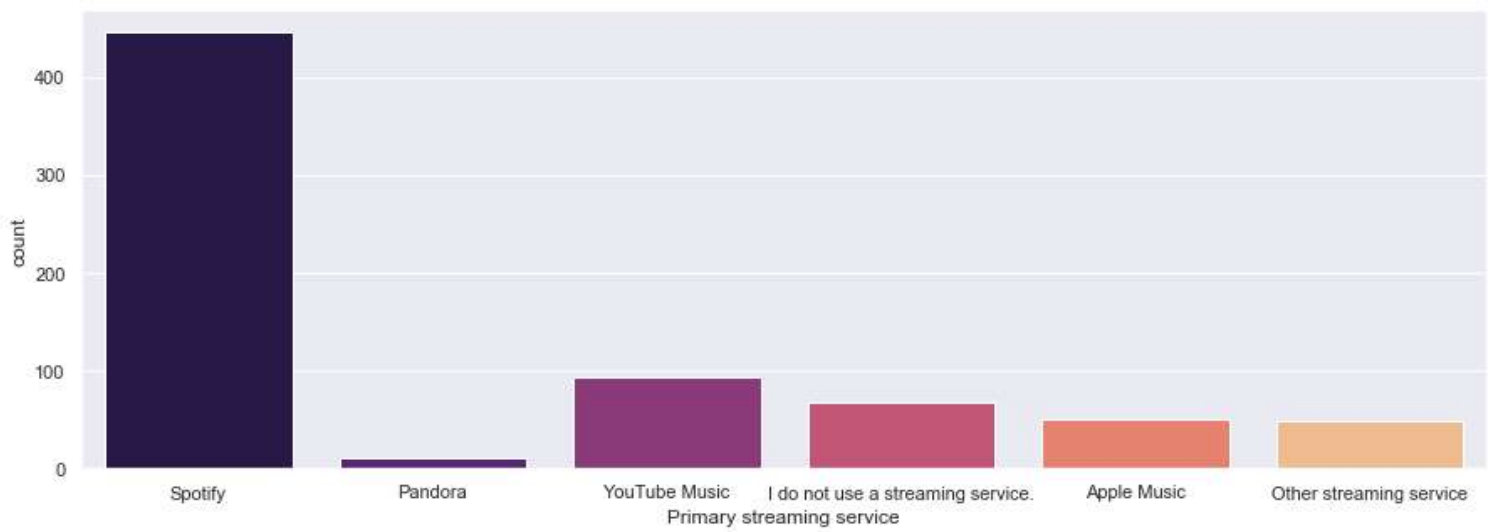
	Timestamp	Age	Primary streaming service	Hours per day	While working	Instrumentalist	Composer	Fav genre	Exploratory	Foreign languages	...	Frequency [R&B]
0	8/27/2022 19:29:02	18	Spotify	3.0	Yes	Yes	Yes	Latin	Yes	Yes	...	Sometimes
1	8/27/2022 19:57:31	63	Pandora	1.5	Yes	No	No	Rock	Yes	No	...	Sometimes

2 rows × 33 columns

Data Visualization

```
In [22]: plt.figure(figsize=(15,5))
sns.countplot(data=df,x="Primary streaming service",palette=color2)
```

Out[22]: <AxesSubplot:xlabel='Primary streaming service', ylabel='count'>

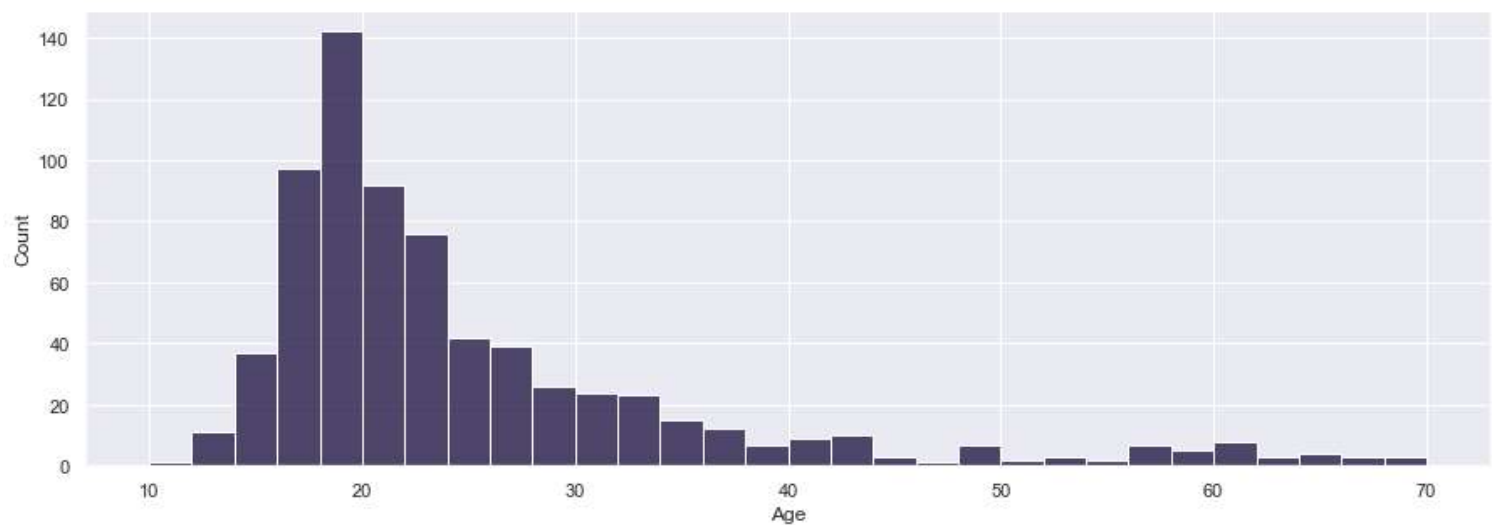


Observation:

1. Most people tend to use streaming services like Spotify and YouTube.

```
In [23]: plt.figure(figsize=(15,5))
sns.histplot(data=df,x="Age")
```

```
Out[23]: <AxesSubplot:xlabel='Age', ylabel='Count'>
```

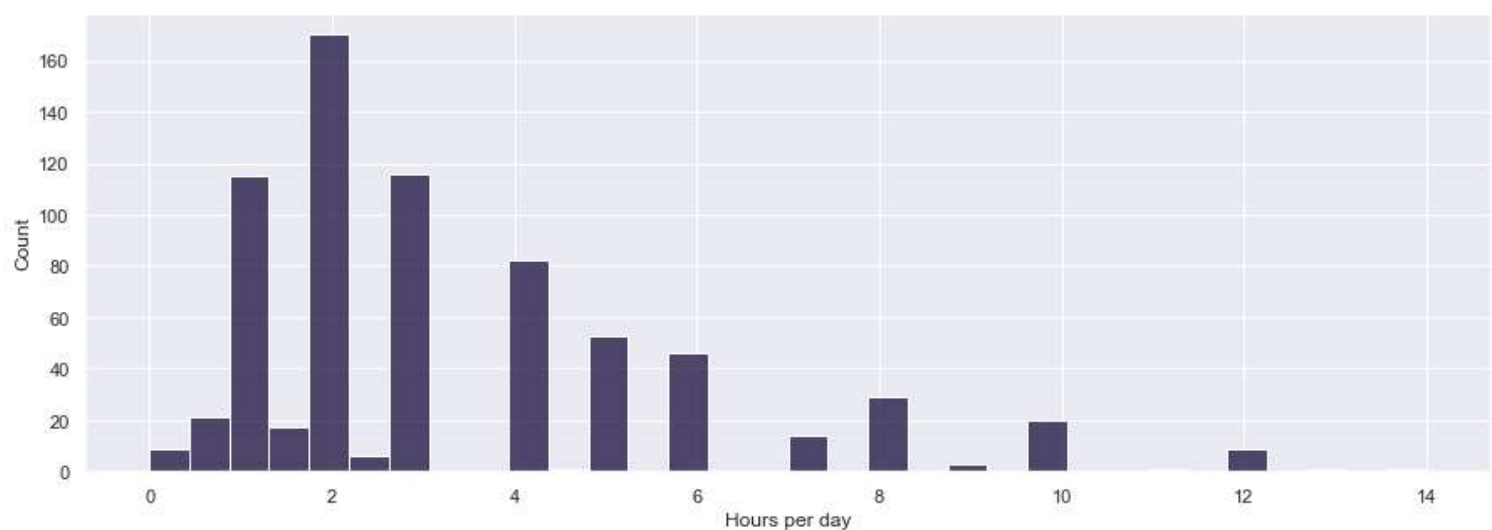


Observation:

1. Exclusively, the age group that predominantly engages with streaming music and songs ranges from 18 to 27 years old.

```
In [24]: plt.figure(figsize=(15,5))
sns.histplot(data=df,x="Hours per day")
```

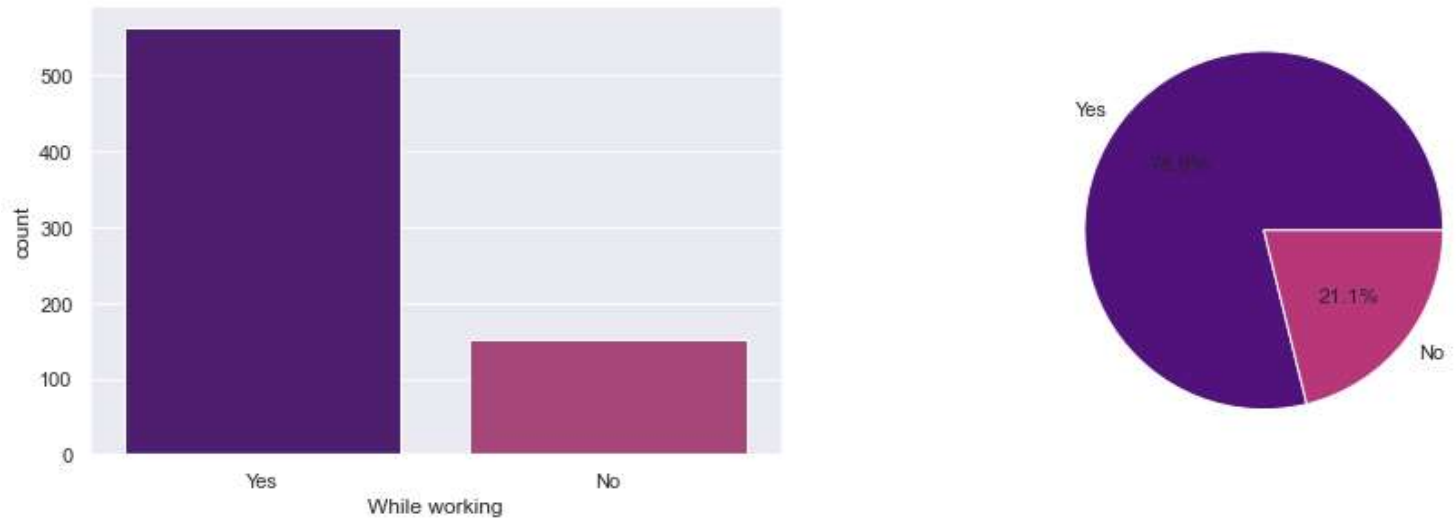
```
Out[24]: <AxesSubplot:xlabel='Hours per day', ylabel='Count'>
```



obsevation:

1. Most people listen to music for 1 to 4 hours per day.

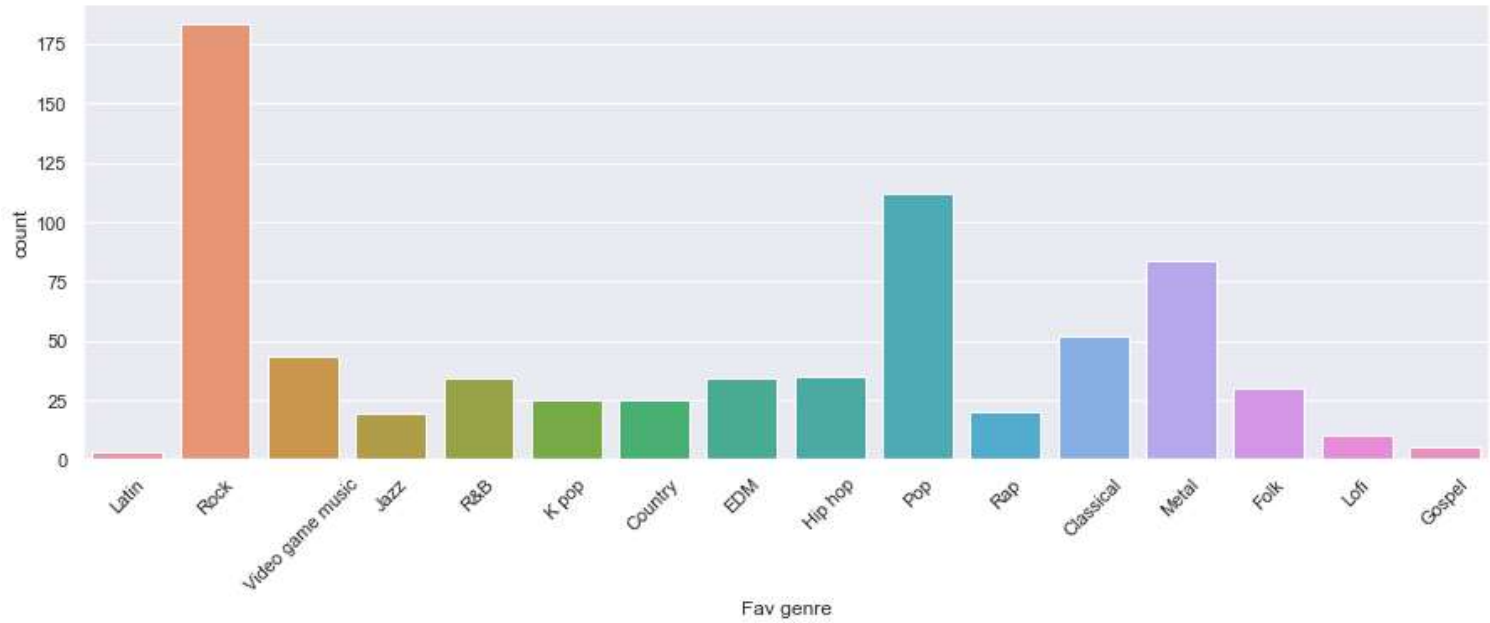

```
In [25]: plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
sns.countplot(data=df,x="While working",palette=color1)
plt.subplot(2,2,2)
plt.pie(df["While working"].value_counts(),colors=color1,labels=["Yes", "No"],autopct="%0.1f%%")
plt.show()
```



Observation:

- 1. Mostly working professionals. Most people listen to music, about 79%.

```
In [26]: plt.figure(figsize=(15,5))
sns.countplot(data=df,x="Fav genre")
plt.xticks(rotation=45)
plt.show()
```

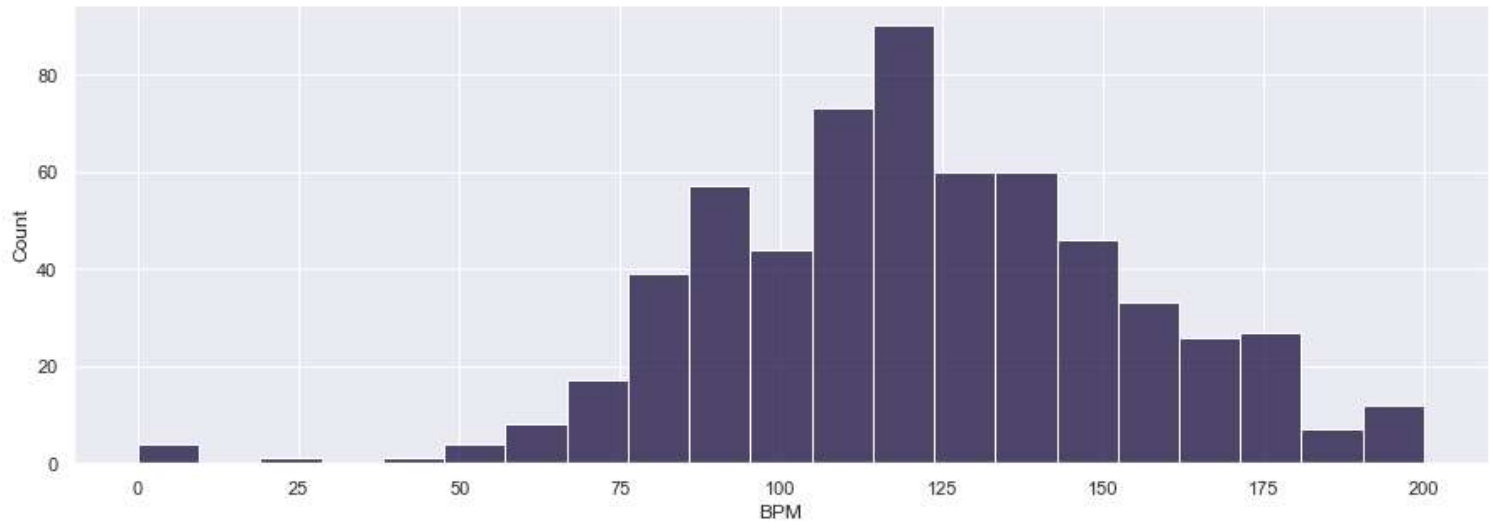


Observation:

- 1. Most people listen to rock, pop, and metal music.

```
In [27]: plt.figure(figsize=(15,5))
sns.histplot(df["BPM"])
```

```
Out[27]: <AxesSubplot:xlabel='BPM', ylabel='Count'>
```



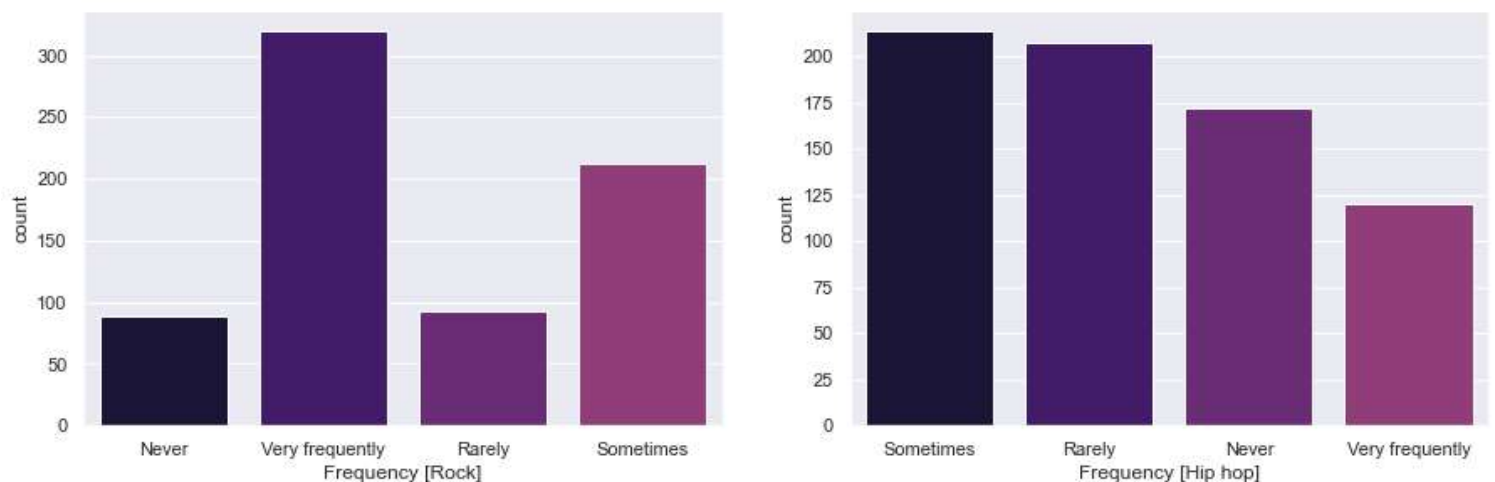
Observation:

1. Most people listen to songs with a beat in the range of 80 to 150.

```
In [28]: df.columns
```

```
Out[28]: Index(['Timestamp', 'Age', 'Primary streaming service', 'Hours per day',
               'While working', 'Instrumentalist', 'Composer', 'Fav genre',
               'Exploratory', 'Foreign languages', 'BPM', 'Frequency [Classical]',
               'Frequency [Country]', 'Frequency [EDM]', 'Frequency [Folk]',
               'Frequency [Gospel]', 'Frequency [Hip hop]', 'Frequency [Jazz]',
               'Frequency [K pop]', 'Frequency [Latin]', 'Frequency [Lofi]',
               'Frequency [Metal]', 'Frequency [Pop]', 'Frequency [R&B]',
               'Frequency [Rap]', 'Frequency [Rock]', 'Frequency [Video game music]',
               'Anxiety', 'Depression', 'Insomnia', 'OCD', 'Music effects',
               'Permissions'],
              dtype='object')
```

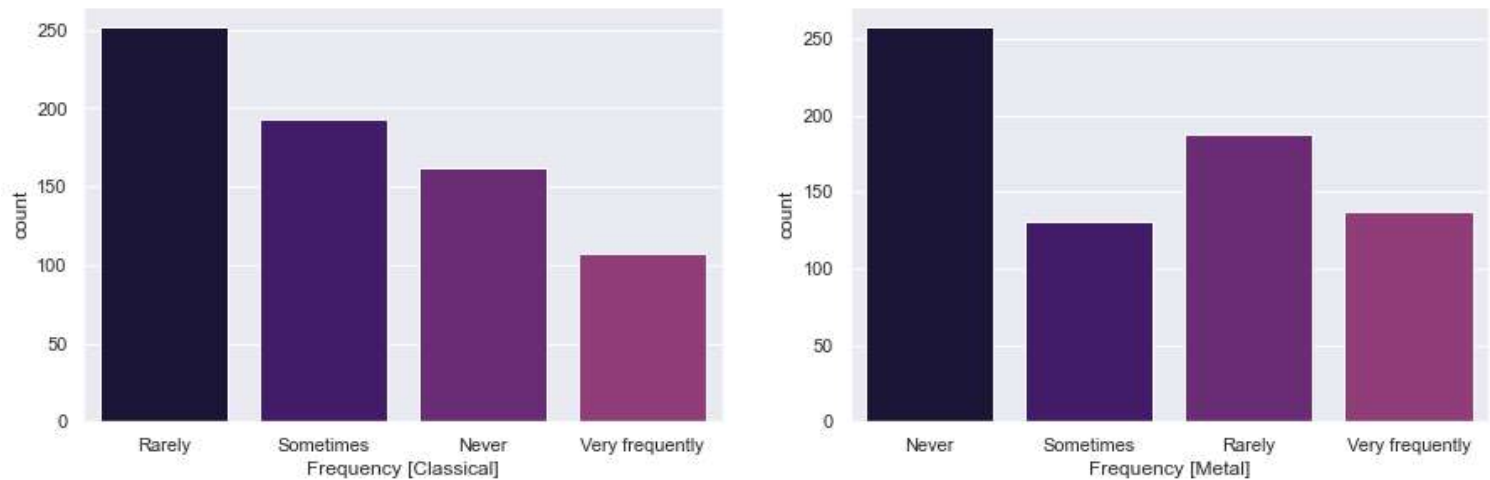
```
In [29]: plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
sns.countplot(data=df,x="Frequency [Rock]")
plt.subplot(2,2,2)
sns.countplot(data=df,x="Frequency [Hip hop]")
plt.show()
```



observation:

1. Most people listen to Rock songs with very frequently and sometime.
2. Most people listen to Hip Hop songs sometime or rarely.

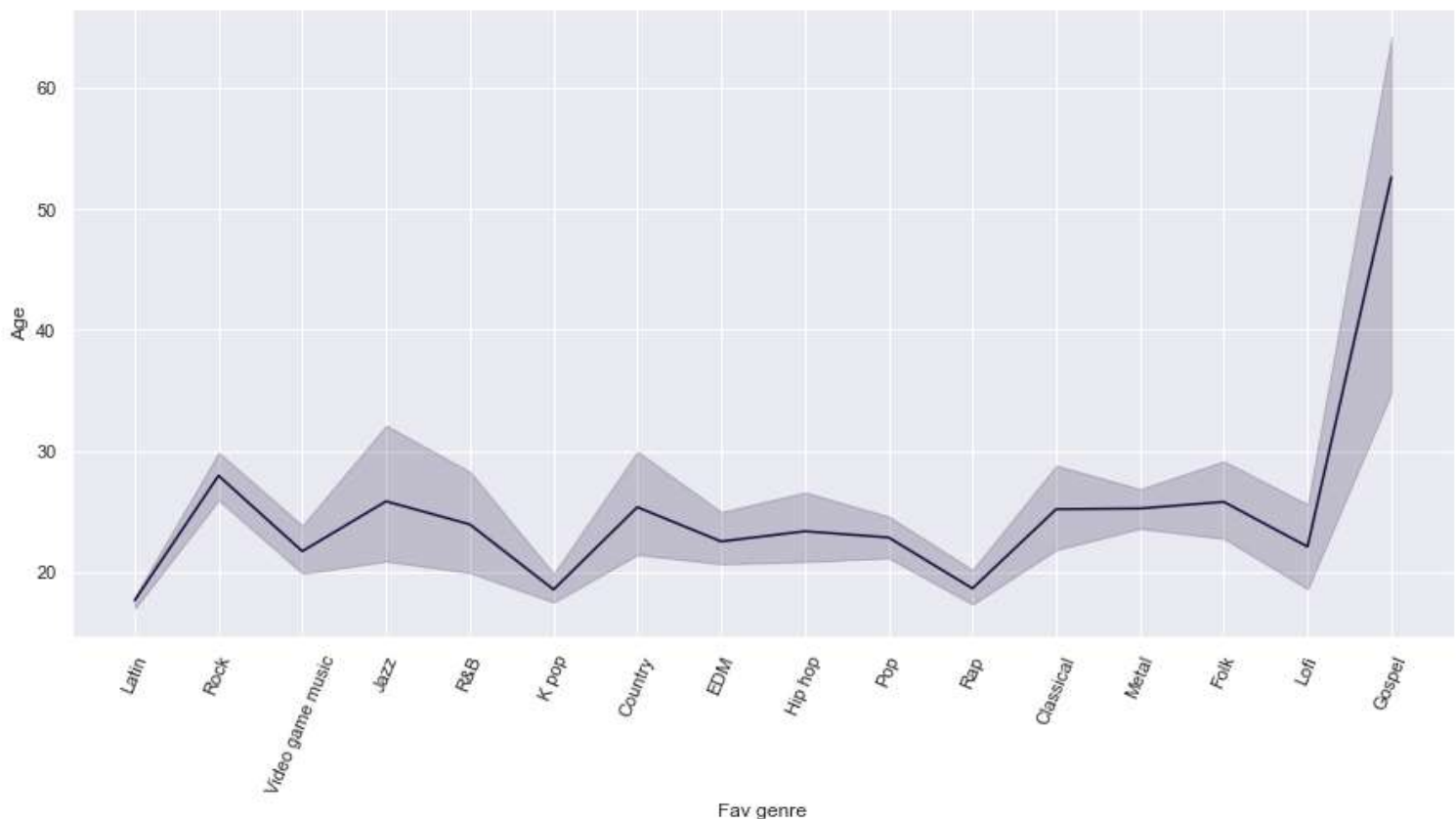
```
In [30]: plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
sns.countplot(data=df,x="Frequency [Classical]")
plt.subplot(2,2,2)
sns.countplot(data=df,x="Frequency [Metal]")
plt.show()
```



observation:

1. Most people listen to classical songs rarely and sometime.
2. Most people listen to Metal songs never or rarely.

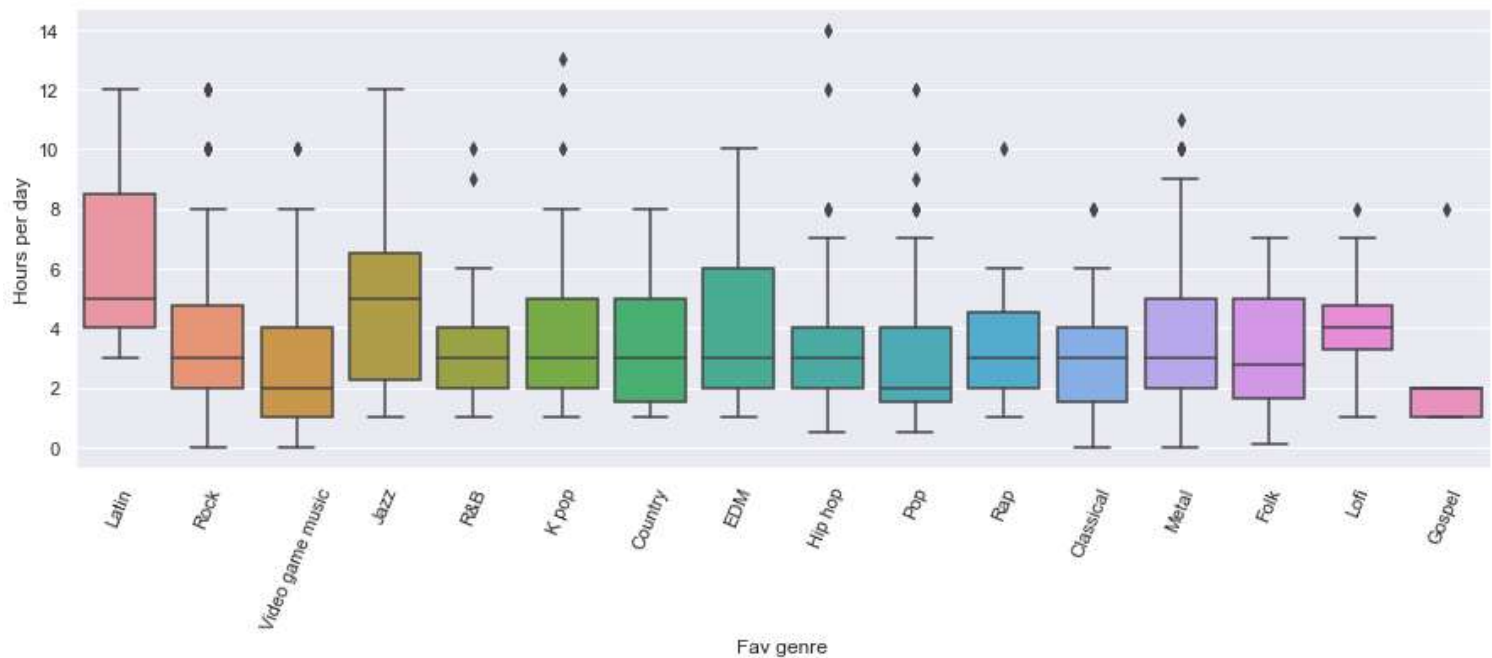
```
In [31]: plt.figure(figsize=(15,7))
sns.lineplot(data=df,x="Fav genre",y="Age")
plt.xticks(rotation=67)
plt.show()
```



Observation:

1. The average age of individuals within the age range of 20 to 30 in the specified area who listen to all types of songs is around 25 years old. The maximum age group that predominantly enjoys listening to Lofi songs falls between 30 to 50 years old..

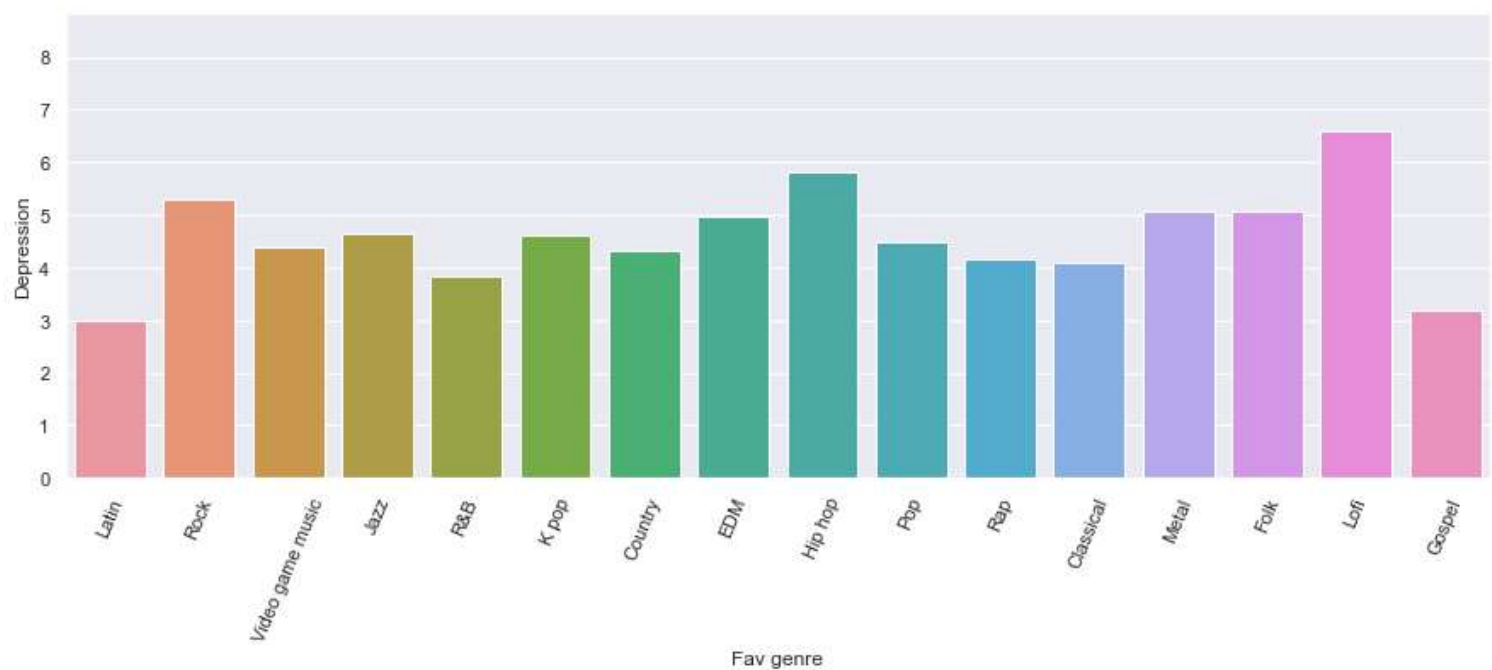
```
In [32]: plt.figure(figsize=(15,5))
sns.boxplot(data=df,x="Fav genre",y="Hours per day")
plt.xticks(rotation=67)
plt.show()
```



Observation:

1. Most people in the area listen to Latin and Jazz songs for an average of one hour per day.

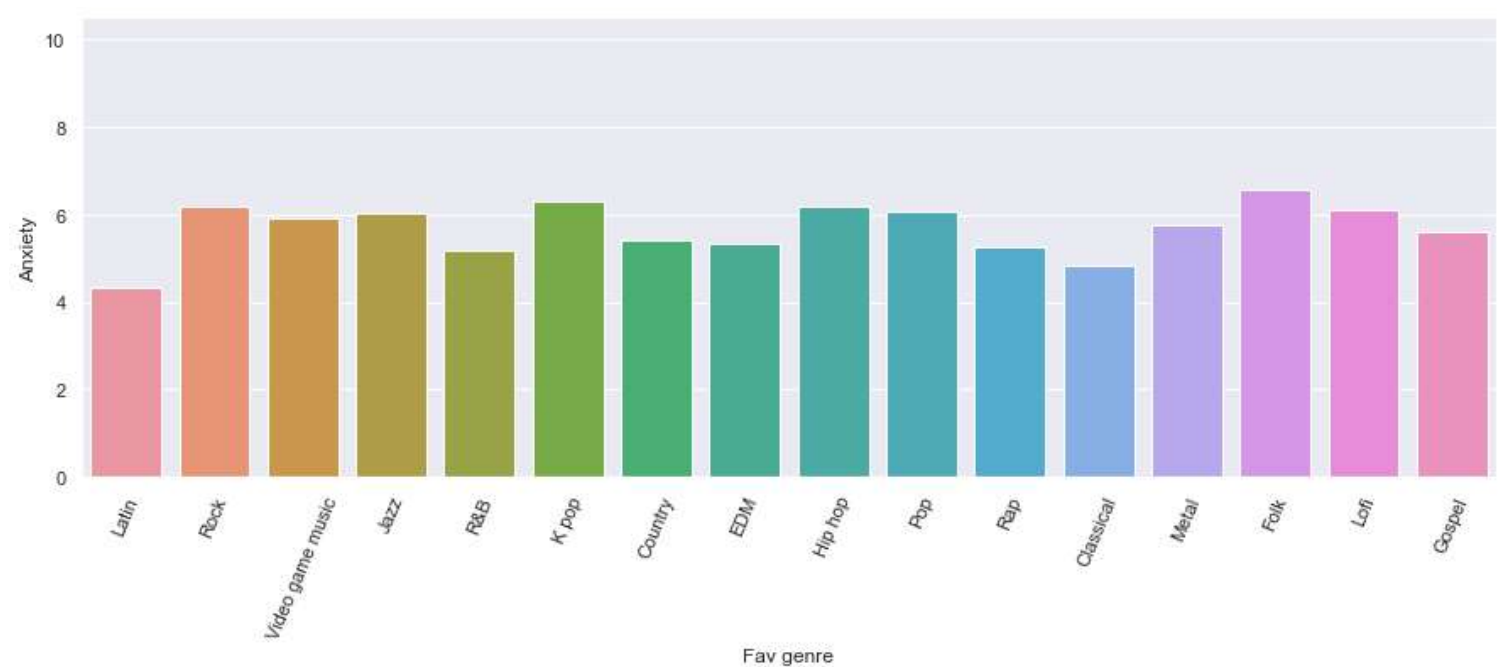
```
In [33]: plt.figure(figsize=(15,5))
sns.barplot(data=df,x="Fav genre",y="Depression",errwidth=0)
plt.xticks(rotation=67)
plt.show()
```



Observation:

1. Listening to Lofi, rock, and hip-hop doesn't directly indicate depression. Mental health is complex; music is one of many factors. Seek professional help if needed.

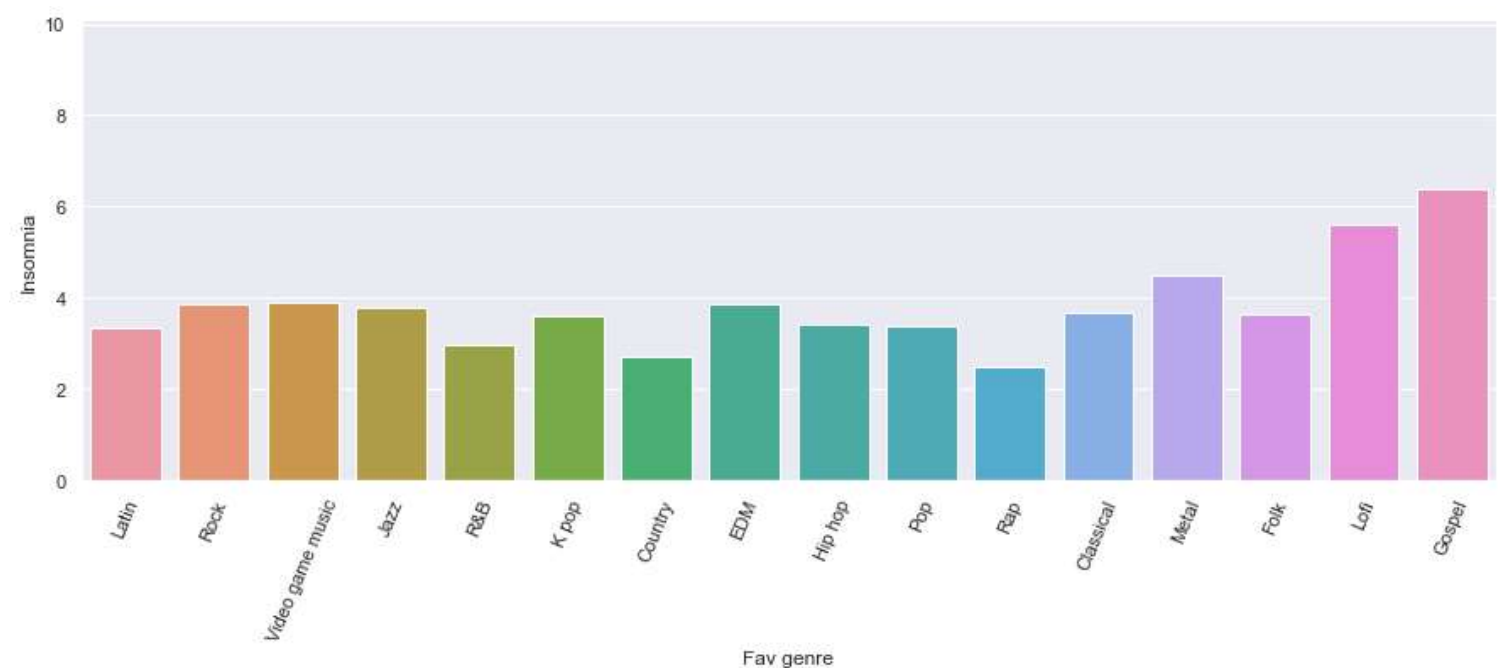
```
In [34]: plt.figure(figsize=(15,5))
sns.barplot(data=df,x="Fav genre",y="Anxiety",errwidth=0)
plt.xticks(rotation=67)
plt.show()
```



Observation:

- 1. Many individuals experiencing anxiety are drawn to genres like rock, K-pop, and folk. However, musical preferences are diverse and not solely indicative of mental health issues. Professional support is crucial for managing anxiety.

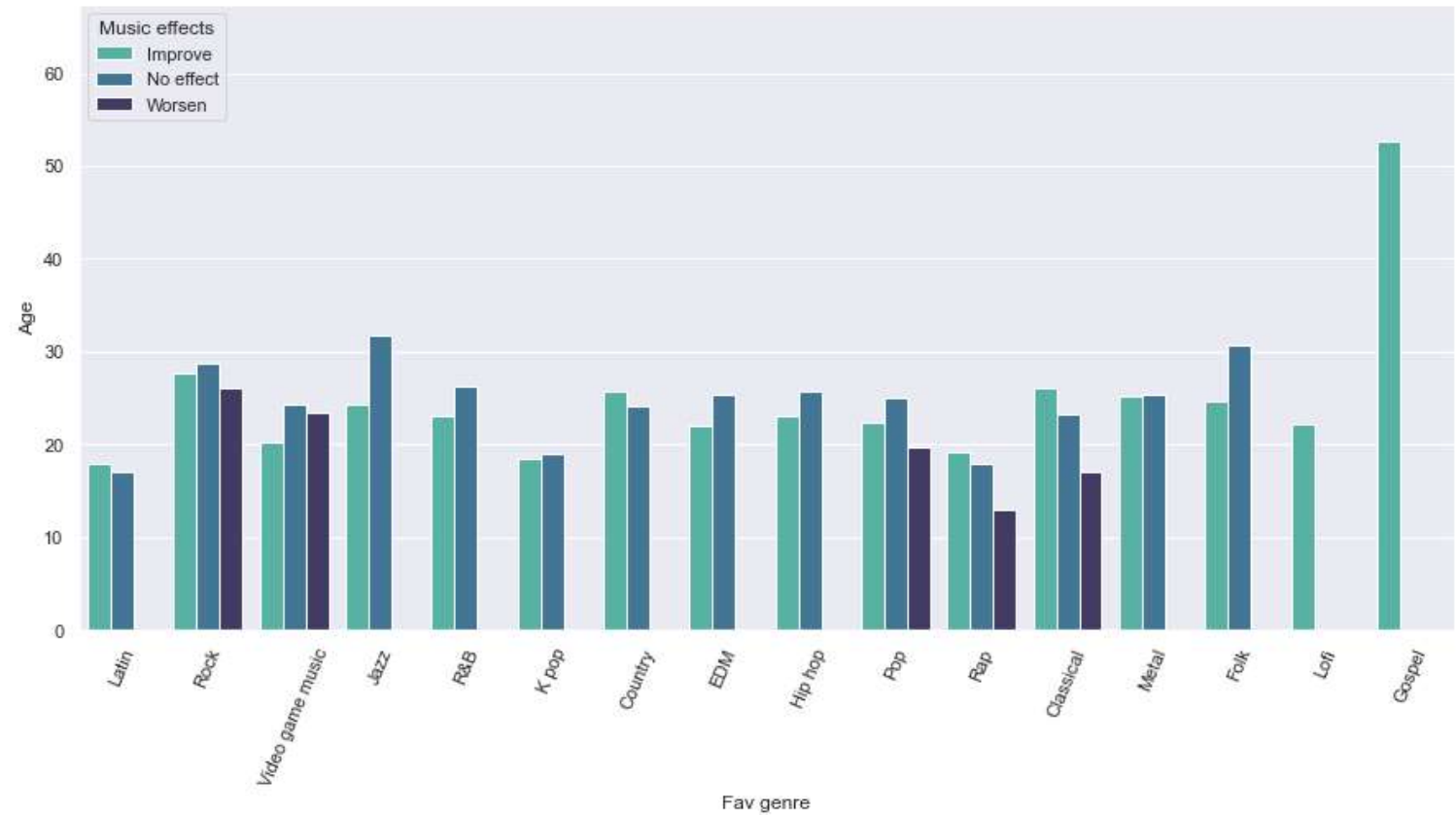
```
In [35]: plt.figure(figsize=(15,5))
sns.barplot(data=df,x="Fav genre",y="Insomnia",errwidth=0)
plt.xticks(rotation=67)
plt.show()
```



Observation:

- 1. While many people who listen to Gospel and Lofi songs might experience insomnia, it's crucial to understand that musical choices alone don't directly cause insomnia. Insomnia has multiple causes and requires proper evaluation and treatment.

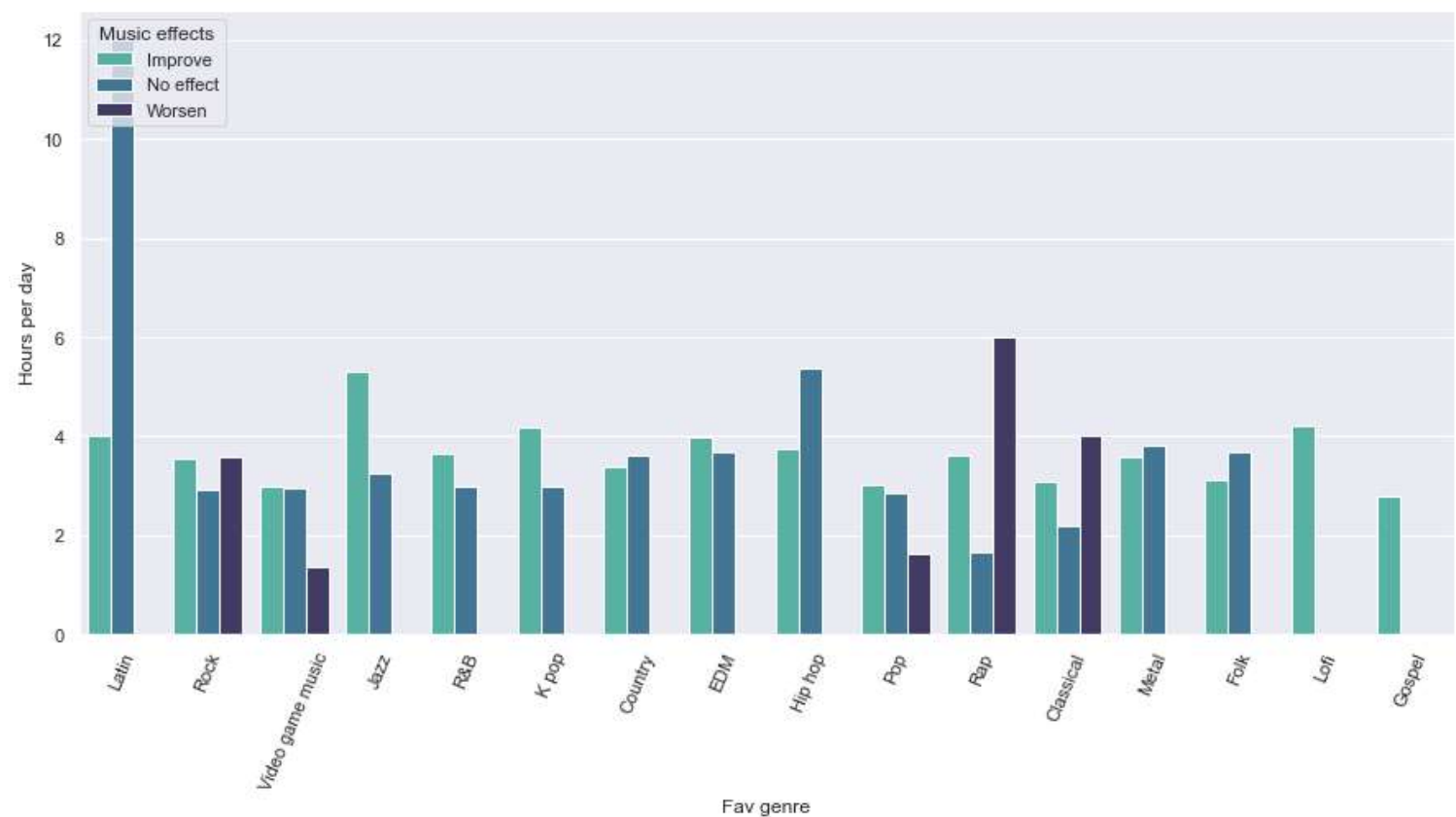
```
In [36]: plt.figure(figsize=(15,7))
sns.barplot(data=df,x="Fav genre",y="Age",hue="Music effects",errwidth=0,palette="mako_r")
plt.xticks(rotation=67)
plt.show()
```



Observation:

- 1. Listeners of all ages generally experience some improvement from music, but genres like Rock, Video Game Music, Pop, Rap, and Classical seem to worsen conditions for some individuals.

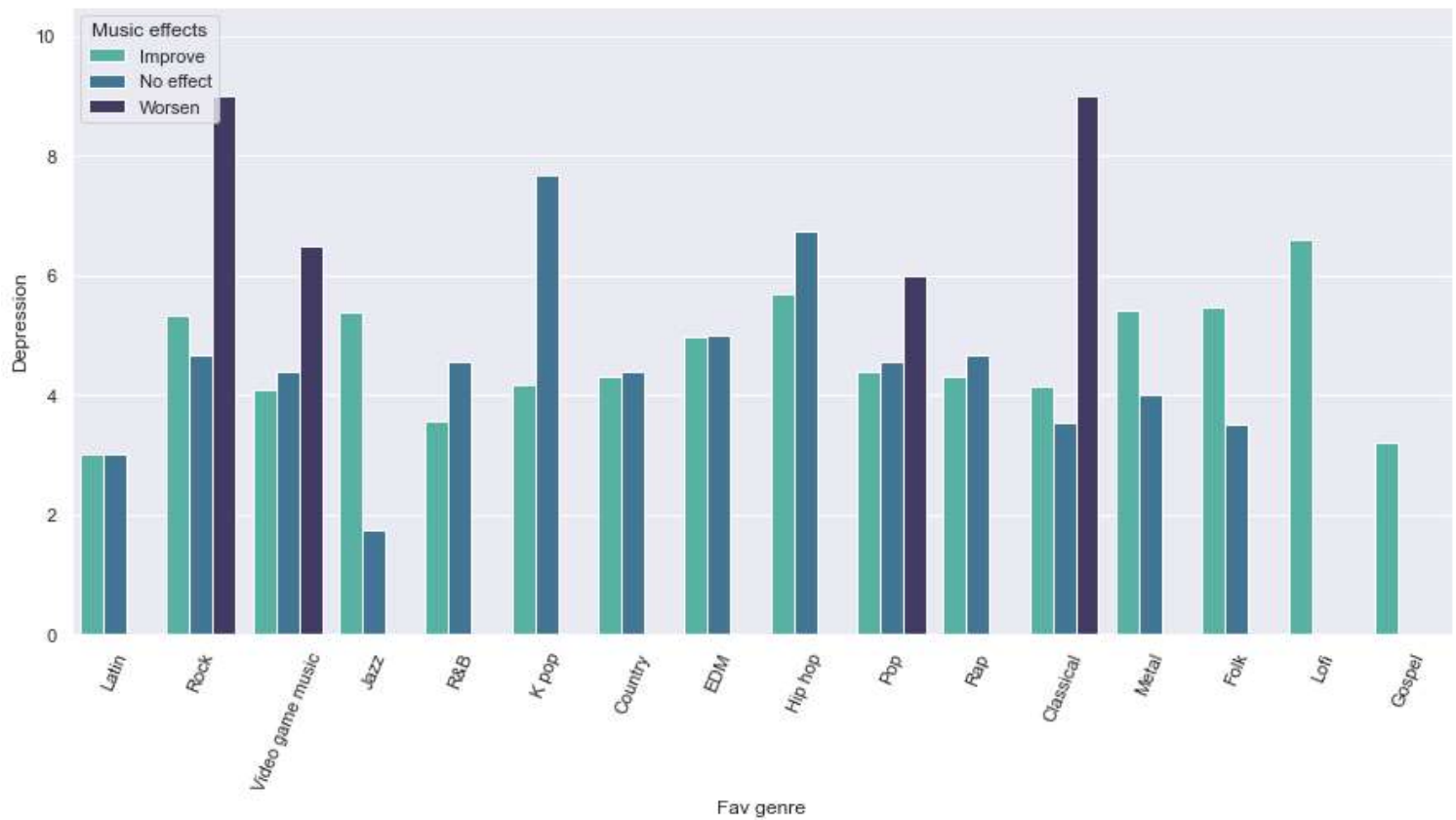
```
In [37]: plt.figure(figsize=(15,7))
sns.barplot(data=df,x="Fav genre",y="Hours per day",hue="Music effects",errwidth=0,palette="mako_r")
plt.xticks(rotation=67)
plt.show()
```



Observation:

1. Listeners of Jazz, Lofi, and K-pop songs tend to experience improvements in their mood and overall well-being as they engage with music for several hours per day.

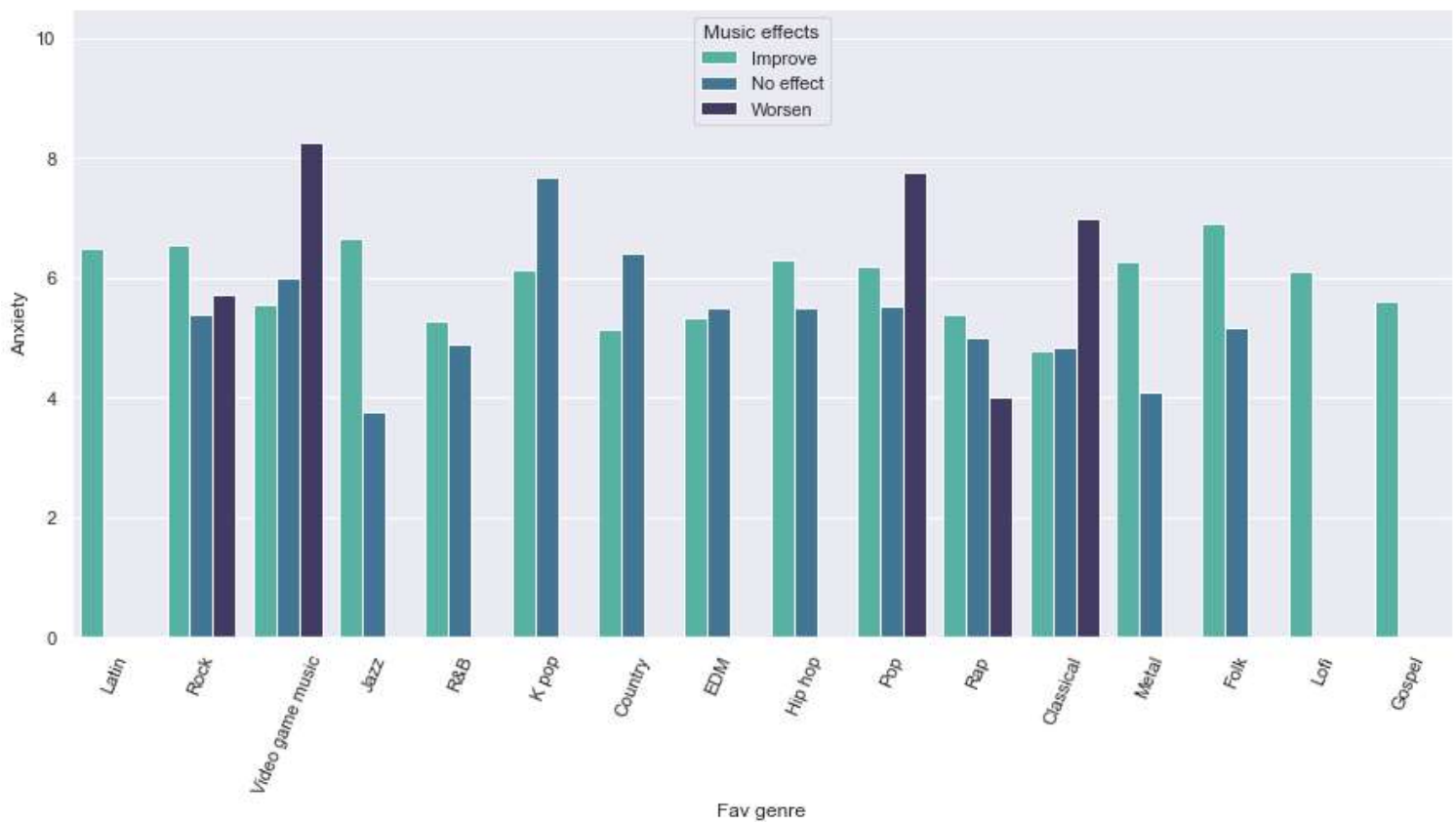
```
In [38]: plt.figure(figsize=(15,7))
sns.barplot(data=df,x="Fav genre",y="Depression",hue="Music effects",errwidth=0,palette="mako_r")
plt.xticks(rotation=67)
plt.show()
```



Observation:

1. Listeners of Rock, Video Game Music, Pop, and Classical genres tend to exhibit a higher level of depression.

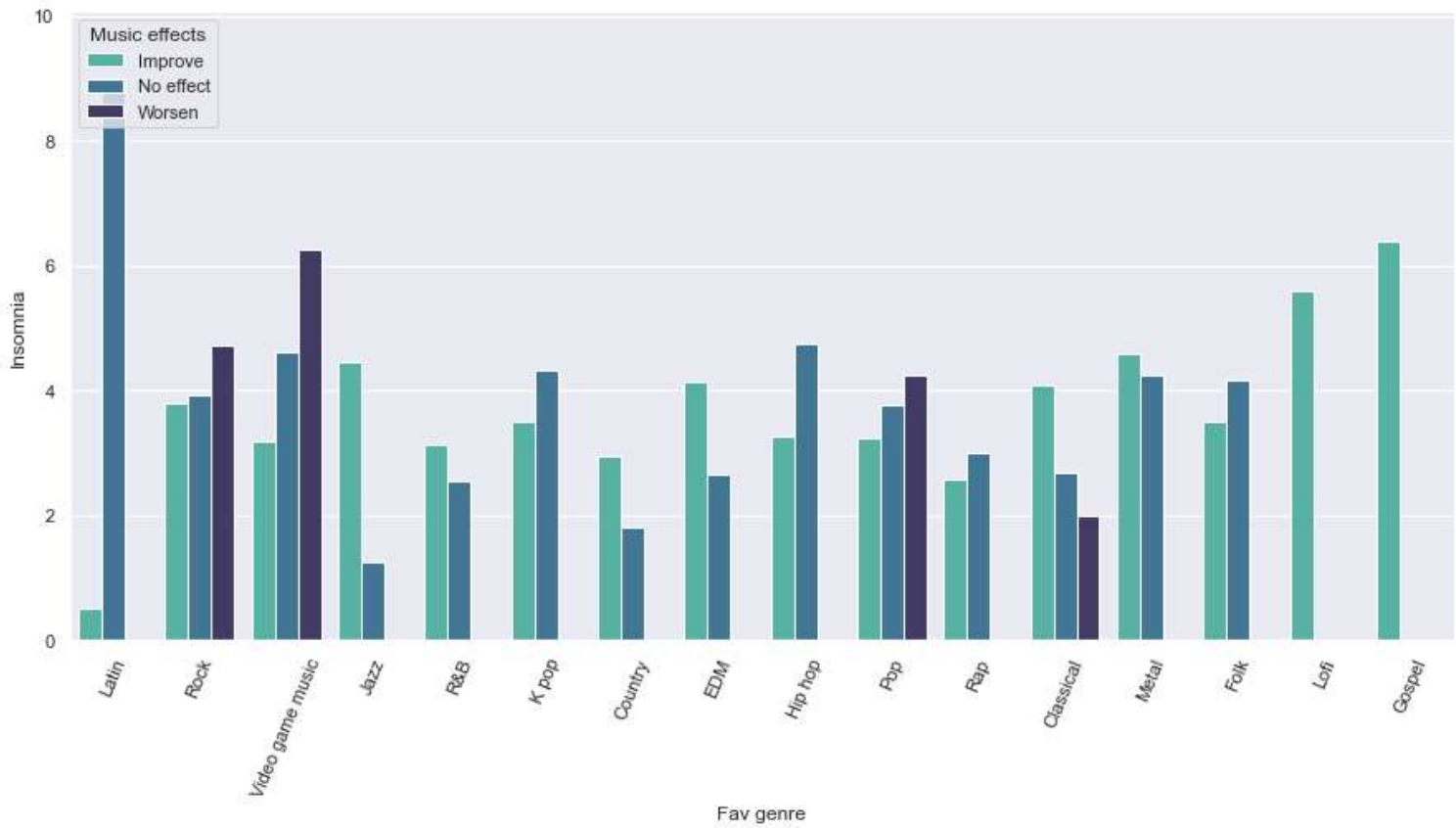
```
In [39]: plt.figure(figsize=(15,7))
sns.barplot(data=df,x="Fav genre",y="Anxiety",hue="Music effects",errwidth=0,palette="mako_r")
plt.xticks(rotation=67)
plt.show()
```



Observation:

1. Listeners of Rock, Video Game Music, Pop, Rap, and Classical exhibit varying levels of anxiety. Among these, Video Game Music listeners tend to experience the highest level of anxiety.

```
In [40]: plt.figure(figsize=(15,7))
sns.barplot(data=df,x="Fav genre",y="Insomnia",hue="Music effects",errwidth=0,palette="mako_r")
plt.xticks(rotation=67)
plt.show()
```



Observation:

1. Listeners of Rock, Video Game Music, Pop, and Classical genres show a propensity for experiencing insomnia. Among these, Video Game Music listeners tend to have the highest level of insomnia.

conclusion:

Exactly, you've captured it well. Music can have a dual impact on mental health – it can be beneficial for some individuals, improving their mood and overall well-being, while for others, certain genres like Rock, Video Game Music, Pop, and Classical might not have a positive effect and could potentially worsen mental health conditions. It's crucial to recognize that music's impact is highly individual and depends on various factors including personal preferences, experiences, and existing mental health conditions. If someone is experiencing difficulties with their mental health, seeking professional guidance and support is recommended.