```python
import pandas as pd
import matplotlib.pyplot as plt
from tabulate import tabulate


# Load your CSV file
df = pd.read_csv('music_playlist.csv')
print("🎵 Loaded Playlist Successfully!")
```

```
🎵 Loaded Playlist Successfully!
```

Start coding or generate with AI.

```python
print("\n--- 🎧 Full Playlist ---")
print(tabulate(df, headers='keys', tablefmt='fancy_grid', showindex=False))
```

```
--- 🎧 Full Playlist ---
╒═══════════════════╤══════════════════╤═════════╤══════════╤════════╤═══════════╕
│ Song              │ Artist           │ Genre   │ Duration │ Rating │ PlayCount │
╞═══════════════════╪══════════════════╪═════════╪══════════╪════════╪═══════════╡
│ Blinding Lights   │ The Weeknd       │ Pop     │     3.2  │    4.9 │       350 │
├───────────────────┼──────────────────┼─────────┼──────────┼────────┼───────────┤
│ Shape of You      │ Ed Sheeran       │ Pop     │     4.24 │    4.7 │       500 │
├───────────────────┼──────────────────┼─────────┼──────────┼────────┼───────────┤
│ Bohemian Rhapsody │ Queen            │ Rock    │     5.55 │    5   │       420 │
├───────────────────┼──────────────────┼─────────┼──────────┼────────┼───────────┤
│ Believer          │ Imagine Dragons  │ Rock    │     3.24 │    4.8 │       460 │
├───────────────────┼──────────────────┼─────────┼──────────┼────────┼───────────┤
│ Closer            │ The Chainsmokers │ EDM     │     4.04 │    4.5 │       300 │
├───────────────────┼──────────────────┼─────────┼──────────┼────────┼───────────┤
│ Faded             │ Alan Walker      │ EDM     │     3.32 │    4.6 │       380 │
├───────────────────┼──────────────────┼─────────┼──────────┼────────┼───────────┤
│ Perfect           │ Ed Sheeran       │ Pop     │     4.4  │    4.8 │       510 │
├───────────────────┼──────────────────┼─────────┼──────────┼────────┼───────────┤
│ Lose Yourself     │ Eminem           │ Hip-Hop │     5.2  │    4.9 │       550 │
├───────────────────┼──────────────────┼─────────┼──────────┼────────┼───────────┤
│ God's Plan        │ Drake            │ Hip-Hop │     3.18 │    4.6 │       490 │
├───────────────────┼──────────────────┼─────────┼──────────┼────────┼───────────┤
│ Starboy           │ The Weeknd       │ Pop     │     3.5  │    4.4 │       270 │
╘═══════════════════╧══════════════════╧═════════╧══════════╧════════╧═══════════╛
```

```python
# Filtering
choice = input("Filter by Genre (g) or Artist (a)? ").lower()
if choice == 'g':
    genre = input("Enter genre (e.g. Pop, Rock, EDM): ")
    filtered = df[df['Genre'].str.lower() == genre.lower()]
elif choice == 'a':
    artist = input("Enter artist name: ")
    filtered = df[df['Artist'].str.lower() == artist.lower()]
else:
    filtered = pd.DataFrame()
    print("Invalid choice")

# Show filtered
if not filtered.empty:
    print("\n--- 🎼 Filtered Results ---")
    print(tabulate(filtered, headers='keys', tablefmt='fancy_grid', showindex=False))
    if input("Save filtered results? (y/n): ").lower() == 'y':
        filtered.to_csv('filtered_playlist.csv', index=False)
        print("✅ Saved as 'filtered_playlist.csv'")
else:
    print("⚠️ No results found.")
```

```
Filter by Genre (g) or Artist (a)?  g
Enter genre (e.g. Pop, Rock, EDM):  Pop

--- 🎼 Filtered Results ---
╒═════════════════╤════════════╤═══════╤══════════╤════════╤═══════════╕
│ Song            │ Artist     │ Genre │ Duration │ Rating │ PlayCount │
╞═════════════════╪════════════╪═══════╪══════════╪════════╪═══════════╡
│ Blinding Lights │ The Weeknd │ Pop   │     3.2  │    4.9 │       350 │
├─────────────────┼────────────┼───────┼──────────┼────────┼───────────┤
│ Shape of You    │ Ed Sheeran │ Pop   │     4.24 │    4.7 │       500 │
├─────────────────┼────────────┼───────┼──────────┼────────┼───────────┤
│ Perfect         │ Ed Sheeran │ Pop   │     4.4  │    4.8 │       510 │
├─────────────────┼────────────┼───────┼──────────┼────────┼───────────┤
│ Starboy         │ The Weeknd │ Pop   │     3.5  │    4.4 │       270 │
╘═════════════════╧════════════╧═══════╧══════════╧════════╧═══════════╛
Save filtered results? (y/n):  y
✅ Saved as 'filtered_playlist.csv'
```

```
sort_col = input("Sort by 'Rating' or 'PlayCount': ")
if sort_col in df.columns:
    sorted_df = df.sort_values(by=sort_col, ascending=False)
    print(f"\n--- Sorted by {sort_col} ---")
    print(tabulate(sorted_df[['Song', 'Artist', sort_col]], headers='keys', tablefmt='fancy_grid', showindex=False))
else:
    print("⚠️ Invalid column")
```

```
Sort by 'Rating' or 'PlayCount':  Rating

--- Sorted by Rating ---
╒═══════════════════╤══════════════════╤══════════╕
│ Song              │ Artist           │   Rating │
╞═══════════════════╪══════════════════╪══════════╡
│ Bohemian Rhapsody │ Queen            │      5   │
├───────────────────┼──────────────────┼──────────┤
│ Blinding Lights   │ The Weeknd       │      4.9 │
├───────────────────┼──────────────────┼──────────┤
│ Lose Yourself     │ Eminem           │      4.9 │
├───────────────────┼──────────────────┼──────────┤
│ Believer          │ Imagine Dragons  │      4.8 │
├───────────────────┼──────────────────┼──────────┤
│ Perfect           │ Ed Sheeran       │      4.8 │
├───────────────────┼──────────────────┼──────────┤
│ Shape of You      │ Ed Sheeran       │      4.7 │
├───────────────────┼──────────────────┼──────────┤
│ Faded             │ Alan Walker      │      4.6 │
├───────────────────┼──────────────────┼──────────┤
│ God's Plan        │ Drake            │      4.6 │
├───────────────────┼──────────────────┼──────────┤
│ Closer            │ The Chainsmokers │      4.5 │
├───────────────────┼──────────────────┼──────────┤
│ Starboy           │ The Weeknd       │      4.4 │
╘═══════════════════╧══════════════════╧══════════╛
```

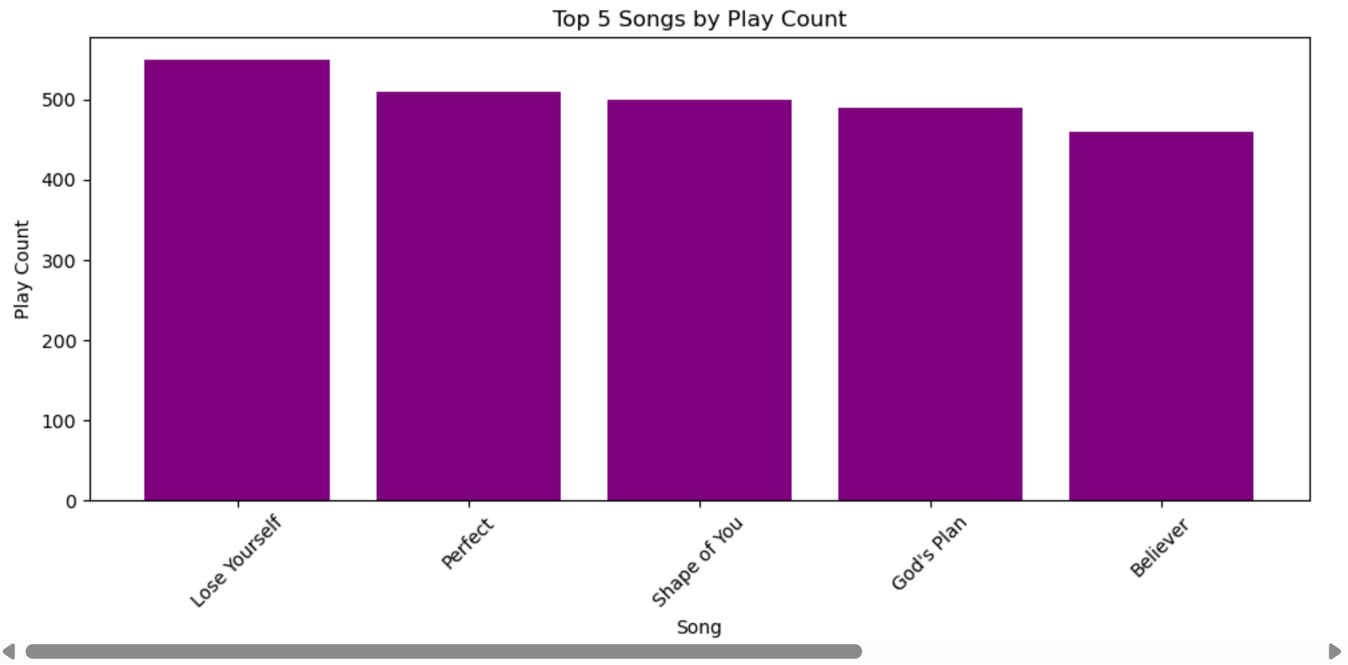
```
search = input("Search for a song keyword: ").lower()
search_df = df[df['Song'].str.lower().str.contains(search)]

if not search_df.empty:
    print("\n--- 🔎 Search Results ---")
    print(tabulate(search_df, headers='keys', tablefmt='fancy_grid', showindex=False))
    if input("Save search results? (y/n): ").lower() == 'y':
        search_df.to_csv('search_results.csv', index=False)
        print("✅ Saved as 'search_results.csv'")
else:
    print("⚠️ No matching songs found.")
```

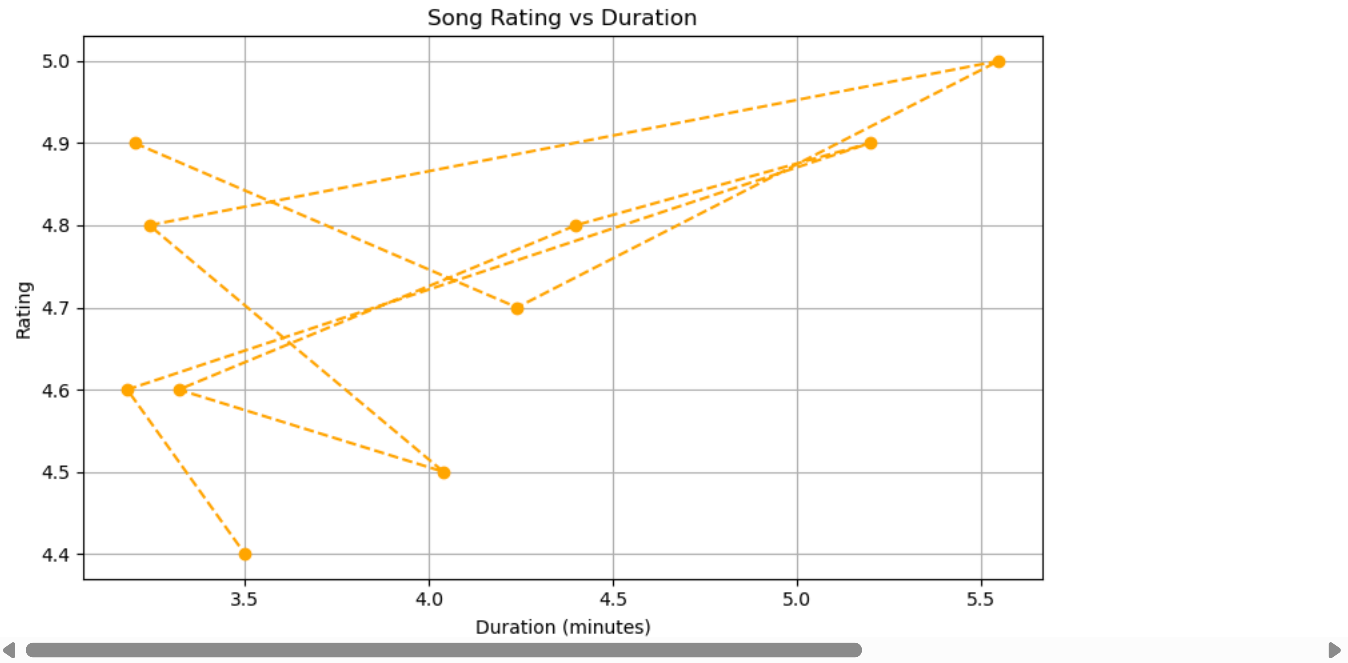
```
Search for a song keyword:  love
⚠️ No matching songs found.
```

```
top = df.sort_values(by='PlayCount', ascending=False).head(5)

plt.figure(figsize=(10,5))
plt.bar(top['Song'], top['PlayCount'], color='purple')
plt.title('Top 5 Songs by Play Count')
plt.xlabel('Song')
plt.ylabel('Play Count')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

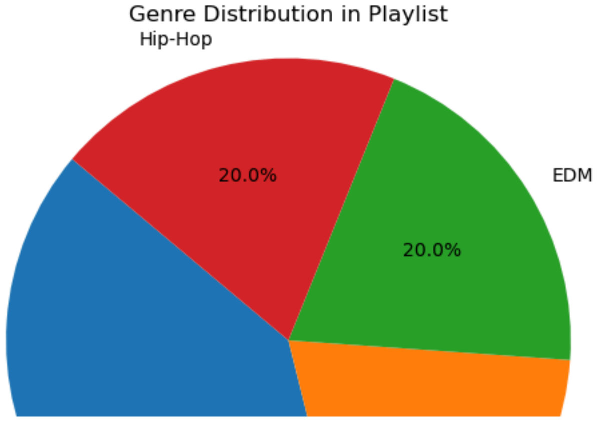


```
plt.figure(figsize=(8, 5))
plt.plot(df['Duration'], df['Rating'], marker='o', linestyle='--', color='orange')
plt.title('Song Rating vs Duration')
plt.xlabel('Duration (minutes)')
plt.ylabel('Rating')
plt.grid(True)
plt.tight_layout()
plt.show()
```



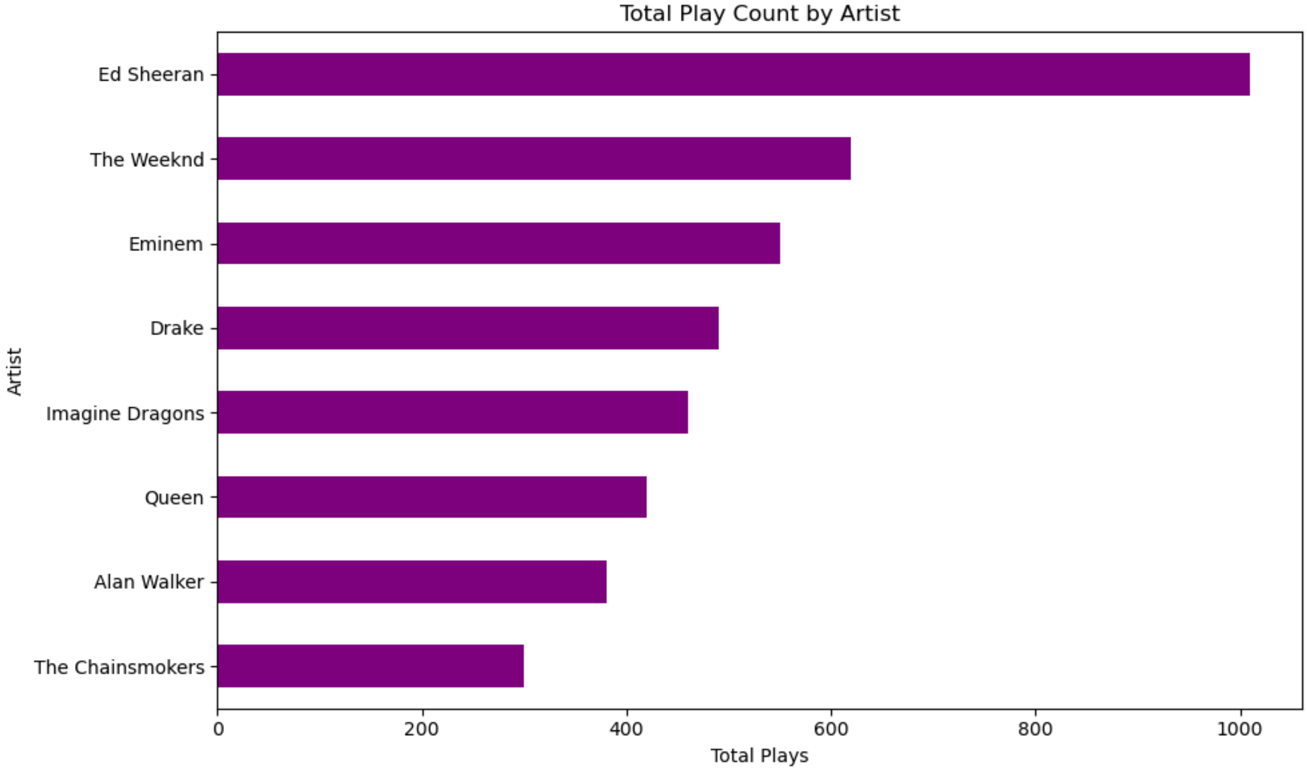
```
genre_counts = df['Genre'].value_counts()

plt.figure(figsize=(6, 6))
plt.pie(genre_counts, labels=genre_counts.index, autopct='%1.1f%%', startangle=140)
plt.title('Genre Distribution in Playlist')
plt.axis('equal')  # Equal aspect ratio = perfect circle
plt.show()
```



```
artist_plays = df.groupby('Artist')['PlayCount'].sum().sort_values()

plt.figure(figsize=(10, 6))
artist_plays.plot(kind='barh', color='purple')
plt.title('Total Play Count by Artist')
plt.xlabel('Total Plays')
plt.tight_layout()
plt.show()
```


Total Play Count by Artist
Ed Sheeran
The Weeknd
Eminem
Drake
Imagine Dragons
Queen
Alan Walker
The Chainsmokers
Artist
Total Plays