-- Step 1: Create the dataset

```
data = {
   "Song": ["Sunshine", "Moonlight", "Rainy Day", "Party All Night", "Love Story",
         "Hip Hop Beat", "Classical Calm", "Rock Anthem", "Pop Hit", "Summer Vibes"],
   "Artist": ["Alice", "Bob", "Charlie", "Alice", "Eve",
          "Frank", "Grace", "Henry", "Alice", "Bob"],
   "Genre": ["Pop", "Pop", "Jazz", "Pop", "Pop",
         "HipHop", "Classical", "Rock", "Pop", "Pop"],
   "StreamsMillions": [120, 250, np.nan, 180, 300,
                90, 60, 200, np.nan, 150],
   "DurationSeconds": [210, np.nan, 300, 180, 220,
                200, 360, 250, 230, np.nan],
   "ReleaseYear": [2021, 2020, 2019, 2021, 2018,
              2020, 2017, 2019, 2021, 2022]
}
df = pd.DataFrame(data)
```

-- Step 2: Show the First Rows

Display the first 5 rows of the dataset.

-- Step 3: Check for Missing Data

Check for missing values.

If any missing:

- ✓ StreamsMillions → fill with 0
- ✓ DurationSeconds → fill with average duration

-- Step 4: Add a New Column

Create a new column StreamCategory (apply to StreamsMillions column):

- Low → < 50
- Medium → 50–200
- High → > 200

-- Step 5: Filter Data

List all **Pop songs** with **High streams**.

-- Step 6: Sort Songs

- Sort all songs by StreamsMillions descending.
- Show the **top 3 most streamed songs**.

-- Step 7: Grouping & Aggregation

Group by Artist and calculate:

- Total streams
- Average song duration

-- Step 8: Genre Analysis

- Find the genre with the **highest average streams**.
- Show the **top 2 songs per genre** by streams.

-- Step 9: Conditional Column

Create a new column HitSong:

- True if StreamsMillions > 150 **and** DurationSeconds < 240
- False otherwise