Samantha Beard

Term Project Milestone 2 - Cleaning/Formatting Flat File Source

Milestone Objective

Perform at least 5 data transformation and/or cleansing steps to your flat file data. You can do the same transformation multiple times if needed to clean your data. The goal is a clean dataset at the end of the milestone.

Make sure you clearly label each transformation (Step #1, Step #2, etc.) in your code and describe what it is doing in 1-2 sentences.

Project / Milestone Explanation

The goal of my project is to look at if popularity of songs correlates to winning Grammys. We are starting with the flat file that shows 30,000 songs played on spotify. I will be transforming this database to be more usable for my future analysis.

source: https://www.kaggle.com/datasets/joebeachcapital/30000-spotify-songs?select=spotify_songs.csv

File Set Up

In [1]:
```
# import packages

import pandas as pd
import re
```

In [2]:
```
# import cvs file to dataframe
spotify_df = pd.read_csv('spotify_songs.csv')

#show top 5
spotify_df.head()
```

Out[2]:

	track_id	track_name	track_artist	track_popularity	track_album_id	track_album_n
0	6f807x0ima9a1j3VPbc7VN	I Don't Care (with Justin Bieber) - Loud Luxur...	Ed Sheeran	66	2oCs0DGTsRO98Gh5ZSl2Cx	I Don't Care (Justin Bieber) [L Lux
1	0r7CVbZTWZgbTCYdfa2P31	Memories - Dillon Francis Remix	Maroon 5	67	63rPSO264uRjW1X5E6cWv6	Memories (D Francis Re
2	1z1Hg7Vb0AhHDiEmnDE79l	All the Time - Don Diablo Remix	Zara Larsson	70	1HoSmj2eLcsrR0vE9gThr4	All the Time (Diablo Re
3	75FpbthrwQmzHlBJLuGdC7	Call You Mine - Keanu Silva Remix	The Chainsmokers	60	1nqYsOef1yKKuGOVchbsk6	Call You Mine - Ren
4	1e8PAfcKUYoKkxPhrHqw4x	Someone You Loved - Future Humans Remix	Lewis Capaldi	69	7m7vv9wlQ4i0LFuJiE2zsQ	Someone Loved (Fu Humans Re

5 rows × 23 columns

In [3]:
```
# row count
startingRowCount = len(spotify_df.index)
startingRowCount
```

Out[3]:
```
32833
```

Step 1

The first transformation will be to ensure consistency in naming. The first letter of each word in column track_name, track_artist, and track_album_name is capitalized. Remove words in parenteses and brackets and words after hyphen for track_name.

In [4]:
```
#Pandas str.title will update the string so that each word is capitalized in specified df column
spotify_df['track_name'] = spotify_df['track_name'].str.title()
spotify_df['track_artist'] = spotify_df['track_artist'].str.title()
spotify_df['track_album_name'] = spotify_df['track_album_name'].str.title()

spotify_df.head()
```

Out[4]:

	track_id	track_name	track_artist	track_popularity	track_album_id	track_album_n
0	6f807x0ima9a1j3VPbc7VN	I Don'T Care (With Justin Bieber) - Loud Luxur...	Ed Sheeran	66	2oCs0DGTsRO98Gh5ZSl2Cx	I Don'T Care (' Justin Bieber) [I Lux
1	0r7CVbZTWZgbTCYdfa2P31	Memories - Dillon Francis Remix	Maroon 5	67	63rPSO264uRjW1X5E6cWv6	Memories (D Francis Re
2	1z1Hg7Vb0AhHDiEmnDE79l	All The Time - Don Diablo Remix	Zara Larsson	70	1HoSmj2eLcsrR0vE9gThr4	All The Time (Diablo Re
3	75FpbthrwQmzHlBJLuGdC7	Call You Mine - Keanu Silva Remix	The Chainsmokers	60	1nqYsOef1yKKuGOVchbsk6	Call You Mine - Ren
4	1e8PAfcKUYoKkxPhrHqw4x	Someone You Loved - Future Humans Remix	Lewis Capaldi	69	7m7vv9wlQ4i0LFuJiE2zsQ	Someone Loved (Fu Humans Re

5 rows × 23 columns

In [5]:
```
# remove words after hyphen by using the hyphen as a delimiter with pandas.series.str.rsplit
spotify_df['track_name'] = spotify_df['track_name'].str.rsplit('-', 1).str[0]
spotify_df.head()
```

Out[5]:

	track_id	track_name	track_artist	track_popularity	track_album_id	track_album_n
0	6f807x0ima9a1j3VPbc7VN	I Don'T Care (With Justin Bieber)	Ed Sheeran	66	2oCs0DGTsRO98Gh5ZSl2Cx	I Don'T Care (' Justin Bieber) [I Lux
1	0r7CVbZTWZgbTCYdfa2P31	Memories	Maroon 5	67	63rPSO264uRjW1X5E6cWv6	Memories (D Francis Re
2	1z1Hg7Vb0AhHDiEmnDE79l	All The Time	Zara Larsson	70	1HoSmj2eLcsrR0vE9gThr4	All The Time (Diablo Re
3	75FpbthrwQmzHlBJLuGdC7	Call You Mine	The Chainsmokers	60	1nqYsOef1yKKuGOVchbsk6	Call You Mine - Ren
4	1e8PAfcKUYoKkxPhrHqw4x	Someone You Loved	Lewis Capaldi	69	7m7vv9wlQ4i0LFuJiE2zsQ	Someone Loved (Fu Humans Re

5 rows × 23 columns

In [6]:
```
# remove words within parentheses using pandas.series.str.replace to replace each occurrence of
spotify_df['track_name'] = spotify_df['track_name'].str.replace(r"\(.*\)","", regex=True)
```

```
spotify_df.head()
```

Out[6]:

	track_id	track_name	track_artist	track_popularity	track_album_id	track_album_n
0	6f807x0ima9a1j3VPbc7VN	I Don'T Care	Ed Sheeran	66	2oCs0DGTsRO98Gh5ZSl2Cx	I Don'T Care (' Justin Bieber) [I Lux
1	0r7CVbZTWZgbTCYdfa2P31	Memories	Maroon 5	67	63rPSO264uRjW1X5E6cWv6	Memories (D Francis Re
2	1z1Hg7Vb0AhHDiEmnDE79l	All The Time	Zara Larsson	70	1HoSmj2eLcsrR0vE9gThr4	All The Time (Diablo Re
3	75FpbthrwQmzHlBJLuGdC7	Call You Mine	The Chainsmokers	60	1nqYsOef1yKKuGOVchbsk6	Call You Mine - Ren
4	1e8PAfcKUYoKkxPhrHqw4x	Someone You Loved	Lewis Capaldi	69	7m7vv9wlQ4i0LFuJiE2zsQ	Someone Loved (Fu Humans Re

5 rows × 23 columns

Step 2

The second transformation will be to create a new column as a primary key. The rows may currently have unique track_id, track_album_id, or playlist_id, however, we are not looking at versions of the song and are more concerned with having unique track_name_artist.

In [7]:
```
# concatenate track_name and track_artist for primary key and new column track_name_artist
spotify_df["track_name_artist"] = spotify_df["track_name"] + " - " + spotify_df["track_artist"]
spotify_df.head()
```

Out[7]:

	track_id	track_name	track_artist	track_popularity	track_album_id	track_album_n
0	6f807x0ima9a1j3VPbc7VN	I Don'T Care	Ed Sheeran	66	2oCs0DGTsRO98Gh5ZSl2Cx	I Don'T Care (' Justin Bieber) [I Lux
1	0r7CVbZTWZgbTCYdfa2P31	Memories	Maroon 5	67	63rPSO264uRjW1X5E6cWv6	Memories (D Francis Re
2	1z1Hg7Vb0AhHDiEmnDE79l	All The Time	Zara Larsson	70	1HoSmj2eLcsrR0vE9gThr4	All The Time (Diablo Re
3	75FpbthrwQmzHlBJLuGdC7	Call You Mine	The Chainsmokers	60	1nqYsOef1yKKuGOVchbsk6	Call You Mine - Ren
4	1e8PAfcKUYoKkxPhrHqw4x	Someone You Loved	Lewis Capaldi	69	7m7vv9wlQ4i0LFuJiE2zsQ	Someone Loved (Fu Humans Re

5 rows × 24 columns

In [8]:
```
# new column was added at the end and I want it at the beginning. reordering columns by calling
# followed by remaining columns in same order.
spotify_df = spotify_df[["track_name_artist"] + [col for col in spotify_df.columns if col != "tr
spotify_df.head()
```

Out[8]:

	track_name_artist	track_id	track_name	track_artist	track_popularity	track_album
0	I Don'T Care - Ed Sheeran	6f807x0ima9a1j3VPbc7VN	I Don'T Care	Ed Sheeran	66	2oCs0DGTsRO98Gh5ZSl
1	Memories - Maroon 5	0r7CVbZTWZgbTCYdfa2P31	Memories	Maroon 5	67	63rPSO264uRjW1X5E6cV
2	All The Time - Zara Larsson	1z1Hg7Vb0AhHDiEmnDE79l	All The Time	Zara Larsson	70	1HoSmj2eLcsrR0vE9gT
3	Call You Mine - The Chainsmokers	75FpbthrwQmzHlBJLuGdC7	Call You Mine	The Chainsmokers	60	1nqYsOef1yKKuGOVchb
4	Someone You Loved - Lewis Capaldi	1e8PAfcKUYoKkxPhrHqw4x	Someone You Loved	Lewis Capaldi	69	7m7vv9wlQ4i0LFuJiE2

5 rows × 24 columns

Step 3

The third tranformation will be to drop any duplicates of the track_name_artist column that I just created.

In [9]:
```
# pandas.DataFrame.drop_duplicates allows me to drop duplicates and defaults to keeping the firs
# subset allows me to specify duplicates in a specific column

spotify_df = spotify_df.drop_duplicates(subset=['track_name_artist'])
droppedDupsCount = len(spotify_df.index)
droppedDupsCount
```

Out[9]:
```
25767
```

In [10]:
```
# This removed 6,979 rows
```

Step 4

The fourth transformation will drop unnecessary columns: track_id and track_album_id because these are specific IDs for Spotify and not for the song. Additionally, I will drop playlist_name, playlist_id, playlist_genre, and playlist_subgenre. These are being removed because they apply to the playlist the song was on and not the genre of the song.

In [11]:
```
# dropping columns using pandas drop
spotify_df = spotify_df.drop(['track_id', 'track_album_id', 'playlist_name', 'playlist_id', 'pla
spotify_df.head()
```

Out[11]:

	track_name_artist	track_name	track_artist	track_popularity	track_album_name	track_album_release_date	dan
0	I Don'T Care - Ed Sheeran	I Don'T Care	Ed Sheeran	66	I Don'T Care (With Justin Bieber) [Loud Luxury...	2019-06-14	
1	Memories - Maroon 5	Memories	Maroon 5	67	Memories (Dillon Francis Remix)	2019-12-13	
2	All The Time - Zara Larsson	All The Time	Zara Larsson	70	All The Time (Don Diablo Remix)	2019-07-05	
3	Call You Mine - The Chainsmokers	Call You Mine	The Chainsmokers	60	Call You Mine - The Remixes	2019-07-19	
4	Someone You Loved - Lewis Capaldi	Someone You Loved	Lewis Capaldi	69	Someone You Loved (Future Humans Remix)	2019-03-05	

Step 5

Lastly, I will drop songs with a popularity score of 0. I had originally thought about dropping those below 50, however, there are only 30,000 songs on the list out of 100 million on Spotify. Therefore, a song making the list could signify higher popularity than most.

In [12]:
```
# removing songs with a popularity score of 0 by saying the df is equal to the df where track po
spotify_df = spotify_df[spotify_df.track_popularity != 0]
endingRowCount = len(spotify_df.index)
endingRowCount
```

Out[12]:
```
23458
```

In [13]:
```
# this removed an additional 2,309 rows and 9,288 rows were removed from the initial file.
```

In [14]:
```
spotify_df
```

Out[14]:

	track_name_artist	track_name	track_artist	track_popularity	track_album_name	track_album_release_date
0	I Don'T Care - Ed Sheeran	I Don'T Care	Ed Sheeran	66	I Don'T Care (With Justin Bieber) [Loud Luxury...	2019-06-14
1	Memories - Maroon 5	Memories	Maroon 5	67	Memories (Dillon Francis Remix)	2019-12-13
2	All The Time - Zara Larsson	All The Time	Zara Larsson	70	All The Time (Don Diablo Remix)	2019-07-05
3	Call You Mine - The Chainsmokers	Call You Mine	The Chainsmokers	60	Call You Mine - The Remixes	2019-07-19
4	Someone You Loved - Lewis Capaldi	Someone You Loved	Lewis Capaldi	69	Someone You Loved (Future Humans Remix)	2019-03-05
...	...	...	...	...	...	...
32828	City Of Lights - Lush & Simon	City Of Lights	Lush & Simon	42	City Of Lights (Vocal Mix)	2014-04-28
32829	Closer - Tegan And Sara	Closer	Tegan And Sara	20	Closer Remixed	2013-03-08
32830	Sweet Surrender - Starkillers	Sweet Surrender	Starkillers	14	Sweet Surrender (Radio Edit)	2014-04-21
32831	Only For You - Mat Zo	Only For You	Mat Zo	15	Only For You (Remixes)	2014-01-01
32832	Typhoon - Julian Calor	Typhoon	Julian Calor	27	Typhoon/Storm	2014-03-03

23458 rows × 18 columns

Step 6

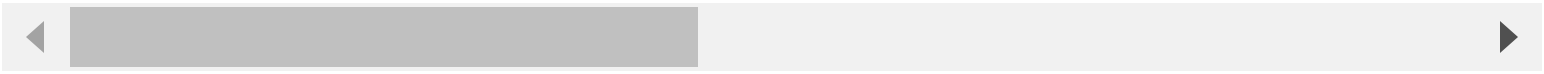
After looking at the df, I don't like that the duration is in ms as it is not easily read, I am going to add columns for seconds (s) and minutes (m).

In [15]:
```
spotify_df["duration_s"] = spotify_df["duration_ms"]/1000
spotify_df["duration_m"] = spotify_df["duration_s"]/60
spotify_df
```

Out[15]:

	track_name_artist	track_name	track_artist	track_popularity	track_album_name	track_album_release_date
0	I Don'T Care - Ed Sheeran	I Don'T Care	Ed Sheeran	66	I Don'T Care (With Justin Bieber) [Loud Luxury...	2019-06-14
1	Memories - Maroon 5	Memories	Maroon 5	67	Memories (Dillon Francis Remix)	2019-12-13
2	All The Time - Zara Larsson	All The Time	Zara Larsson	70	All The Time (Don Diablo Remix)	2019-07-05
3	Call You Mine - The Chainsmokers	Call You Mine	The Chainsmokers	60	Call You Mine - The Remixes	2019-07-19
4	Someone You Loved - Lewis Capaldi	Someone You Loved	Lewis Capaldi	69	Someone You Loved (Future Humans Remix)	2019-03-05
...	...	...	...	...	...	...
32828	City Of Lights - Lush & Simon	City Of Lights	Lush & Simon	42	City Of Lights (Vocal Mix)	2014-04-28
32829	Closer - Tegan And Sara	Closer	Tegan And Sara	20	Closer Remixed	2013-03-08
32830	Sweet Surrender - Starkillers	Sweet Surrender	Starkillers	14	Sweet Surrender (Radio Edit)	2014-04-21
32831	Only For You - Mat Zo	Only For You	Mat Zo	15	Only For You (Remixes)	2014-01-01
32832	Typhoon - Julian Calor	Typhoon	Julian Calor	27	Typhoon/Storm	2014-03-03

23458 rows × 20 columns



Paragraph of the ethical implications of data wrangling specific to your datasource and the steps you completed.

While some would argue that a version of a song could be more popular than another version, I don't think removing these duplicates is a huge ethical issue as we are going to be comparing this to lists of grammy winning songs and the song title and artist are still the same. Additionally, I don't think there are ethical implications of removing songs with a popularity score of 0. I think if I had gone with my initial plan of less than than 50 there could be some potential ethical issues. Data wrangling in general can have the ethical implication of adding bias when removing data, however, I believe I have navigated this as best I can while cleaning the data.