In this report we set out to find the following.

Objectives:

1. Study the correlation between musical attributes and song popularity
2. Build machine learning models to predict song popularity based on various features

First objective: There's many theories discussed by musicians as to what makes a song good or bad. Why some songs get popular. Almost all the time, these discussions are personal opinions with no data to back up their assertions. Using data, we can find quantitative to show whether some musical attributes really do have an impact.

Regarding musical attributes, a particular curiosity is whether the key and mode of a song have a statistically significant impact upon song popularity. Its commonly thought in the music industry that major modes predominantly are more popular than minor modes, but there's usually no data to back up this assertion. I'm also curious whether the key of a song impacts the song success.

Second objective: Is it possible to derive an optimal formula for creating songs that are popular with listeners? We can use machine learning to try to identify the optimal musical attributes for a song. Beyond the scope of this paper, I intend to test out the results of the machine learning finding by creating some music that utilize the optimal music attributes and see if they generate songs that are more popular.

We obtained a dataset on Kaggle that contains comprehensive information on some of the most streamed songs on Spotify, enriched with additional insights from other popular streaming platforms like Apple Music, Deezer, and Shazam. It is ideal for music analysts, data scientists, and machine learning enthusiasts who are interested in exploring trends and characteristics of popular music tracks.

Data Preparation:

<u>Data Source and Quality</u>

Initial quality checks show missing values in the 'keys' column for 95 rows, necessitating data cleaning and feature selection. To represent song popularity, we selected streams as the most direct metric, considering its straightforward link to audience engagement. As playlist data (in_spotify_playlists, in_apple_playlists, etc.) indicate a song's exposure and chart rankings (in_spotify_charts, in_apple_charts) provide insight into relative popularity at a given time.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 953 entries, 0 to 952
Data columns (total 25 columns):
 #   Column                Non-Null Count  Dtype 
---  ------                --------------  ----- 
 0   track_name            953 non-null    object
 1   artist(s)_name        953 non-null    object
 2   artist_count          953 non-null    int64 
 3   released_year         953 non-null    int64 
 4   released_month        953 non-null    int64 
 5   released_day          953 non-null    int64 
 6   in_spotify_playlists  953 non-null    int64 
 7   in_spotify_charts     953 non-null    int64 
 8   streams               953 non-null    object
 9   in_apple_playlists    953 non-null    int64 
 10  in_apple_charts       953 non-null    int64 
 11  in_deezer_playlists   953 non-null    object
 12  in_deezer_charts      953 non-null    int64 
 13  in_shazam_charts      903 non-null    object
 14  bpm                   953 non-null    int64 
 15  key                   858 non-null    object
 16  mode                  953 non-null    object
 17  danceability_%        953 non-null    int64 
 18  valence_%             953 non-null    int64 
 19  energy_%              953 non-null    int64 
 20  acousticness_%        953 non-null    int64 
 21  instrumentalness_%    953 non-null    int64 
 22  liveness_%            953 non-null    int64 
 23  speechiness_%         953 non-null    int64 
 24  cover_url             953 non-null    object
dtypes: int64(17), object(8)
```

<u>Data Cleaning Steps</u>

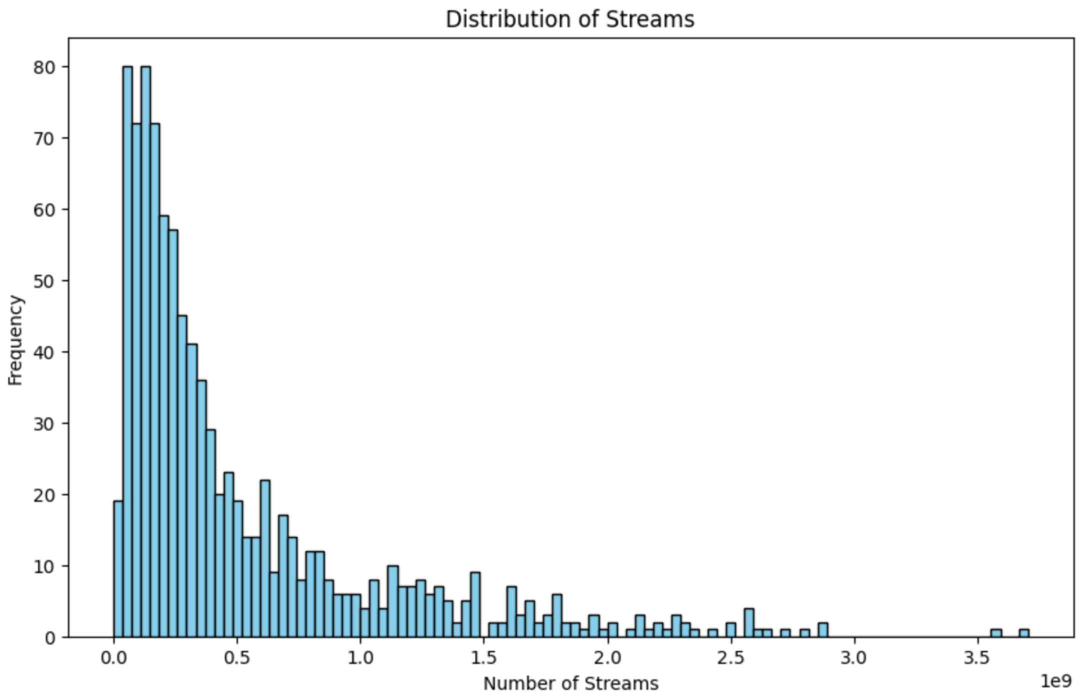
1. Converted streams to a numeric format and applied log transformation to normalize its wide range.
2. Dropped unnecessary columns.

```
Index: 952 entries, 0 to 952
Data columns (total 17 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   track_name          952 non-null    object 
 1   artist(s)_name      952 non-null    object 
 2   artist_count        952 non-null    int64  
 3   released_year       952 non-null    int64  
 4   released_month      952 non-null    int64  
 5   released_day        952 non-null    int64  
 6   streams             952 non-null    float64
 7   bpm                 952 non-null    int64  
 8   key                 952 non-null    object 
 9   mode                952 non-null    object 
 10  danceability_%      952 non-null    float64
 11  valence_%           952 non-null    float64
 12  energy_%            952 non-null    float64
 13  acousticness_%      952 non-null    float64
 14  instrumentalness_%  952 non-null    float64
 15  liveness_%          952 non-null    float64
 16  speechiness_%       952 non-null    float64
dtypes: float64(8), int64(5), object(4)
```

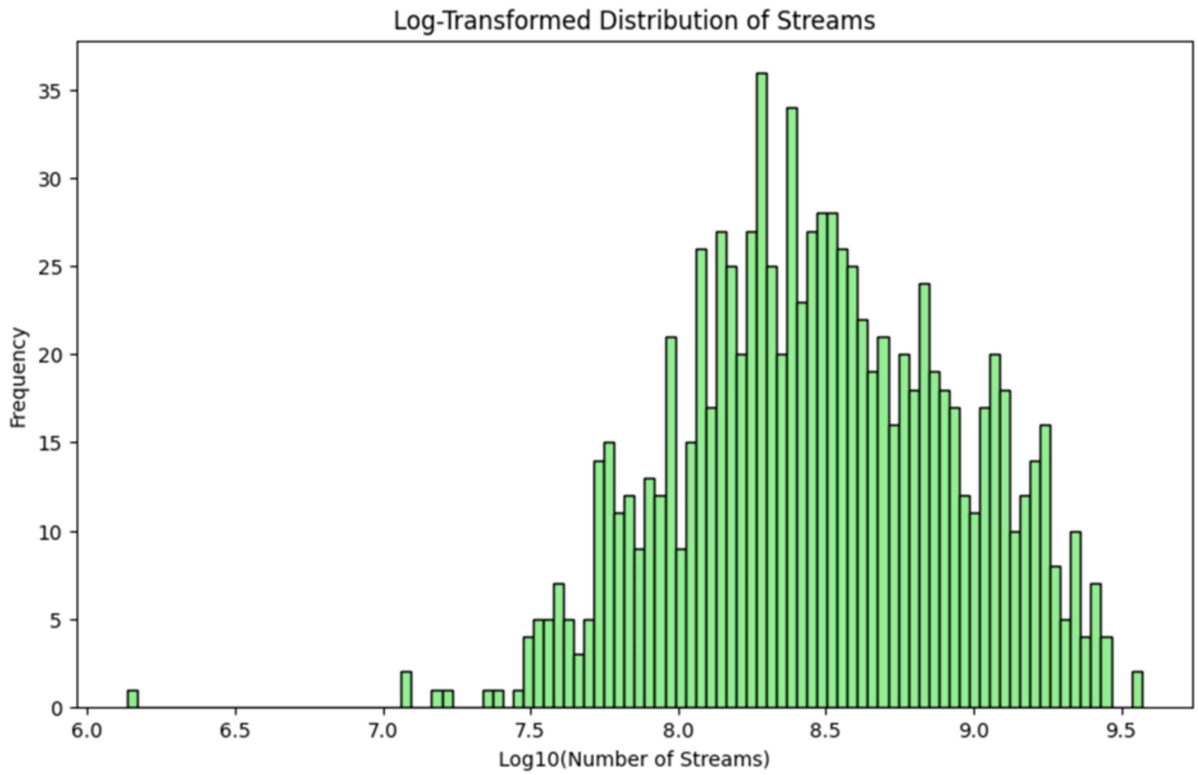
<u>Challenges and Observations</u>

- Distribution Analysis: The raw streams data exhibited a high variance with a right-skewed distribution.

```
count          952.00
mean     514137424.94
std      566856949.04
min           2762.00
25%      141636175.00
50%      290530915.00
75%      673869022.00
max     3703895074.00
Name: streams, dtype: float64
Range: 3703892312.0
Interquartile Range (IQR): 532232847.0
```



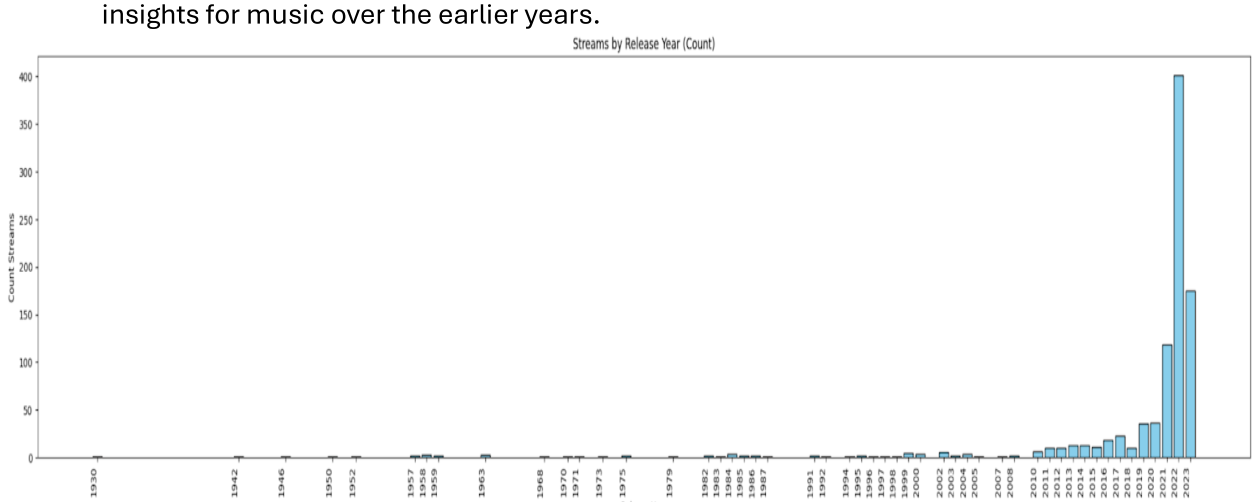
- For ease of viewing the dataset, we applied log transformation, the resultant chart shows improved interpretability, creating a bell-shaped curve while retaining original trends. We also excluded one outlier with an exceptionally low stream count (log10 value of 3.44 or 2762 streams). As this value contributes little to the understanding of overall trends. For this study, we'll drop that data point.



Values around log 10(streams)=8.5 represent songs with ~300,000,000 streams

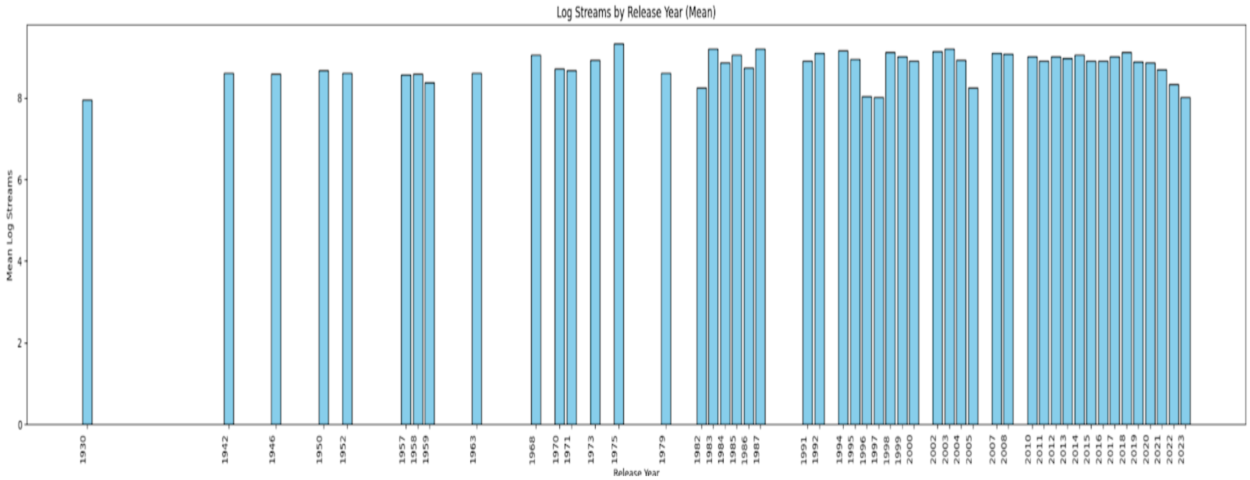
Above shows that while the log-transformation makes it easier to see where the majority of the song's streaming numbers lie, with few songs that take the top spot in streaming. But the data is still not normally distributed due to outliers and heavy tails.

- Trends: The dataset heavily focuses on post-2011 songs, especially 2022 releases, limiting insights for music over the earlier years.



To mitigate this, we explored bootstrapping techniques, however for years with only 1 song included, bootstrapped results repeatedly drawing the same songs. As such we ultimately decided to move ahead with the original dataset which emphasized post-2011 data. The results of the study will be focusing on the attributes of the music from recent years, specifically those released after 2011, with a heavy focus on the year of 2022.

- The dataset contains only highly streamed songs with a mean stream number over ~500MM.



Additional insights on musical attributes:

	instrumentalness_%	speechiness_%	liveness_%
count	951.00	951.00	951.00
mean	0.02	0.10	0.18
std	0.08	0.10	0.14
min	0.00	0.02	0.03
25%	0.00	0.04	0.10
50%	0.00	0.06	0.12
75%	0.00	0.11	0.24
max	0.91	0.64	0.97

The mean instrumentalness of the songs is low even accounting for the standard deviation, meaning that most songs are not instrumentals.

The song speechiness mean is low too even accounting for the standard deviation, indicating that most songs have very little speech elements in them. Most of the songs are mostly musically vocally driven.

The mean of the liveness is also low accounting for the standard deviation, indicating that most songs appear to be recorded in studio rather than being live performances.

key	mode	count
A	Major	41
A	Minor	33
A#	Major	26
A#	Minor	30
B	Major	35
B	Minor	46
C#	Major	73
C#	Minor	47
D	Major	66
D	Minor	15
D#	Major	12
D#	Minor	21
E	Major	17
E	Minor	45
F	Major	44
F	Minor	45
F#	Major	30
F#	Minor	43
G	Major	66
G	Minor	30
G#	Major	63
G#	Minor	28
Unknown	Major	75
Unknown	Minor	20

It appears that of the 951 songs, a larger proportion of songs fall upon the keys of C#. It is interesting to note that no songs appear to be in the key of C.

mode	count
Major	548
Minor	403

Focusing on just the mode aspect, we see there are a few more songs in the major key than the minor key.

Further checking if the key and mode of a song have a statistical influence on the stream popularity.

Analysis:

Study the correlation between musical attributes and song popularity

First, we conducted a two-way ANOVA test to see if there is an effect with key and mode on log transformed streams.

Null hypothesis: There is no effect of key, mode, or their interaction on the log-transformed stream counts.
Alternative hypothesis: At least one of the following effects is significant: The key of the song affects the log-transformed stream counts. The mode of the song affects the log-transformed stream counts. There is an interaction effect between key and mode on log-transformed stream counts.

After running a two-way ANOVA test, with the following results:

	sum_sq	df	F	PR(>F)
C(key)	1.54	11.00	0.63	0.80
C(mode)	0.62	1.00	2.79	0.10
C(key):C(mode)	2.93	11.00	1.20	0.28
Residual	205.33	927.00	NaN	NaN

The key has a very high p-value: 0.8. indicating that the key has no significant effect on the log-transformed streams.

The mode has a p-value of 0.10 The mode shows a marginally lower p-value but is still not significant at the 0.05 threshold. This suggests that mode may have some influence, but the evidence is insufficient to confirm a significant effect.

The interaction of key and mode has a p-value of 0.28, which indicate that the combination of key and mode has no significant influence on streams.

Based on the analysis results we cannot reject the null hypothesis. Hence key, mode, and their interaction do not have statistically significant effects on log-transformed stream counts.
These results are surprising as it is widely held belief amongst musicians that major keys (the major mode) dominates the majority of popular music.

We did a deep dive on broader view to incorporate all the musical attributes and examine their impact on song streams.

<u>Null hypothesis:</u> None of the musical attributes (danceability, speechiness, liveness, etc.), the song key, or the mode have a significant effect on the log-transformed number of streams.
<u>Alternative hypothesis:</u> At least one musical attribute, key, or mode has a significant effect on the log-transformed number of streams.

	sum_sq	df	F	PR(>F)
C(key)	1.84	11.00	0.77	0.67
C(mode)	0.28	1.00	1.28	0.26
bpm	0.07	1.00	0.33	0.56
danceability	1.24	1.00	5.70	0.02
valence	0.12	1.00	0.57	0.45
energy	0.67	1.00	3.11	0.08
acousticness	1.31	1.00	6.02	0.01
instrumentalness	0.25	1.00	1.15	0.28
liveness	0.81	1.00	3.72	0.05
speechiness	2.31	1.00	10.65	0.00
Residual	202.07	930.00	NaN	NaN

Further details of the statistics:

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                streams   R-squared:                       0.039
Model:                            OLS   Adj. R-squared:                  0.019
Method:                 Least Squares   F-statistic:                     1.908
Date:                Fri, 06 Dec 2024   Prob (F-statistic):            0.00947
Time:                        15:07:36   Log-Likelihood:                -612.90
No. Observations:                 951   AIC:                             1268.
Df Residuals:                     930   BIC:                             1370.
Df Model:                          20
Covariance Type:            nonrobust
=====================================================================================
                        coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------------------------------
Intercept             8.8089      0.149     58.925      0.000       8.515       9.102
C(key)[T.A#]          0.1546      0.083      1.861      0.063      -0.008       0.318
C(key)[T.B]           0.0787      0.076      1.041      0.298      -0.070       0.227
C(key)[T.C#]          0.1365      0.070      1.964      0.050       0.000       0.273
C(key)[T.D]           0.1164      0.076      1.538      0.124      -0.032       0.265
C(key)[T.D#]          0.1558      0.098      1.592      0.112      -0.036       0.348
C(key)[T.E]           0.1441      0.081      1.775      0.076      -0.015       0.303
C(key)[T.F]           0.0647      0.074      0.878      0.380      -0.080       0.209
C(key)[T.F#]          0.1176      0.078      1.513      0.131      -0.035       0.270
C(key)[T.G]           0.0360      0.073      0.497      0.620      -0.106       0.178
C(key)[T.G#]          0.0857      0.073      1.166      0.244      -0.059       0.230
C(key)[T.Unknown]     0.1026      0.073      1.403      0.161      -0.041       0.246
C(mode)[T.Minor]     -0.0371      0.033     -1.133      0.258      -0.101       0.027
bpm                   0.0003      0.001      0.577      0.564      -0.001       0.001
danceability         -0.2937      0.123     -2.387      0.017      -0.535      -0.052
valence               0.0596      0.079      0.756      0.450      -0.095       0.214
energy               -0.2168      0.123     -1.762      0.078      -0.458       0.025
acousticness         -0.1841      0.075     -2.453      0.014      -0.331      -0.037
instrumentalness     -0.1963      0.183     -1.074      0.283      -0.555       0.162
liveness             -0.2160      0.112     -1.930      0.054      -0.436       0.004
speechiness          -0.5156      0.158     -3.263      0.001      -0.826      -0.206
==============================================================================
Omnibus:                        5.654   Durbin-Watson:                   1.562
Prob(Omnibus):                  0.059   Jarque-Bera (JB):                5.598
Skew:                          -0.187   Prob(JB):                       0.0609
Kurtosis:                       3.035   Cond. No.                     1.73e+03
==============================================================================
```

<u>Key Observations from the Results:</u>

R-squared = 0.039: This means that only 3.9% of the variance in log-transformed streams is explained by the included musical attributes, suggesting an overall weak relationship. Adjusted R-squared = 0.019, reinforcing the weak predictive power of the model. Most of the variability in streaming numbers are unexplained, this is likely driven by non-musical factors (e.g., artist popularity, playlist inclusion, or marketing).

The F-statistic of 1.908 is low and indicates that the model has weak explanatory power. It doesn't do a good job of predicting the song streams. The Prob (F-statistic) is low enough to be significant though. So even though the model of the whole doesn't predict well, there is evidence that at least one of the musical attributes is related to the song streams.

A skew of –0.187 close to 0 suggests that the residuals are approximately symmetric, which is desirable in linear regression. Kurtosis (3.0353) is very close to 3, implying the residuals follow a normal distribution.

<u>Significant predictors supported by lower P-Value:</u>

Danceability's coefficient is -0.2937, indicating a negative relationship between danceability and streams. For a 1-unit increase in danceability, the log-streams decrease by approximately 0.2937 units. This is very surprising, I would have expected the opposite to be true in that the more danceable, the more streams.

Speechiness coefficient is -0.5156, indicating a negative relationship. Songs with higher speechiness tend to have lower log-streams. **Acousticness** coefficient is -0.1841, suggesting a negative relationship. Songs with higher acoustic qualities are less likely to have high log-streams.

Other musical attributes have high p-values above statistically significant levels, so we won't discuss those. They are showing no effect on song streams.

We reject the null hypothesis some of these attributes have significantly influence on streams.

<u>Build machine learning models to predict song popularity based on various features</u>

Let's take a look our second objective and use machine learning to try to identify the optimal musical attributes for a song.

Let's run a machine learning model on the data to predict the streams. Our model of choice is going to be the Random Forest Regressor. RandomForestRegressor is a machine learning algorithm used for regression tasks. It is part of the ensemble methods family and is implemented in libraries such as scikit-learn. Random forests operate by building multiple decision trees during training and combining their outputs to make predictions.

How It Works

1. Ensemble of Decision Trees:
    a. A Random Forest is essentially a collection (ensemble) of decision trees.
    b. Each tree is trained on a random subset of the data (with replacement, called bootstrapping).
2. Averaging Predictions:
    a. For regression tasks, predictions from each decision tree are averaged to provide the final output. This reduces variance and improves robustness.
3. Feature Randomness:

a. During training, each tree is allowed to consider only a random subset of features when splitting nodes. This prevents overfitting and ensures diversity among the trees.

RandomForestRegressor is a powerful, flexible algorithm that works well for many regression tasks. Its ability to handle non-linear relationships, robustness to overfitting, and feature importance analysis make it a popular choice among data scientists and machine learning practitioners, so this should be a good choice for our data.

We're going to focus on only a few of the columns that are logically independent from the streams.

'artist_count', 'released_month', 'released_day', 'streams', 'bpm', 'key', 'mode', 'danceability_%', 'valence_%', 'energy_%', 'acousticness_%', 'instrumentalness_%', 'liveness_%', 'speechiness_%'
We're not going to look at 'track_name', 'artist(s)_name',
'cover_url' because these are likely to provide confounding influences if any.
Track name and cover_url would provide nonmeaningful results.
Artist name might result in high streams, but it won't be useful for predicting a song without the exact same artist name. If it caused any influence, it would be overfitting the data.

We're not going to look at 'released_year', 'in_spotify_playlists', 'in_spotify_charts', 'streams', 'in_apple_playlists', 'in_apple_charts', 'in_deezer_playlists', 'in_deezer_charts', 'in_shazam_charts', because these are backwards looking results rather than potentially causal variable factors

After running a RandomForestRegressor model on the data, we obtained the following results:

Mean Squared Error: 2.7101427811146464e+17
R-squared: -0.10711700022669324

We have a very high Mean Squared error, which indicates that the predictions are far from the actual value. So this model did not do a good job of predicting. This is expected though as found from our earlier multi-regression testing suggested that these variables would not be very predictive.

Our R-squared value is negative which means the model performs worse than predicting the mean of the target variable for all observations. This suggests that our model is underfitting or that the relationship between the features the predictive streams is weak. This once again is expected from our earlier tests.

We can improve our model's performance with Hyperparameter Tuning with Grid Search.
Grid Search is a systematic method to perform hyperparameter tuning. It exhaustively searches through a specified subset of hyperparameter combinations to find the best-performing set based on a scoring metric like accuracy, mean squared error, or R-squared.

How Grid Search Works

1. Define the Hyperparameters to Tune: Specify a set of hyperparameters and their possible values.
2. Create a Grid of Hyperparameters: Grid Search will test every possible combination of the hyperparameters within the specified ranges.
3. Evaluate Each Combination: For each combination, the model is trained and validated using cross-validation to ensure robustness.
4. Select the Best Combination: The combination with the best score on the validation set is chosen.

After running our model with hyperparameters Tuning with Grid Search, we were able to obtain the following result:

Fitting 3 folds for each of 20 candidates, totalling 60 fits
Best parameters found: {'n_estimators': 100, 'min_samples_split': 2, 'min_samples_leaf': 4, 'max_depth': 10, 'bootstrap': True}

Optimized Mean Squared Error: 2.5225785342976077e+17
Optimized R-squared: -0.030495366955971992
Elapsed time for hyperparameter tuning: 27.90 seconds

After optimizing our model, we were able to reduce our Mean Squared Error slightly. But it's still a huge number. So our model still isn't very predictive.

Our optimized R-squared value improved a little too, now its closer to 0. But an R-Squared value of 0 just means that the model predicts none of the variance. So its still a ineffective model for predicting streams.

We performed some calculations and were able to get the model's feature importance for each feature:

Feature	**Importance**
released_day	0.182018
bpm	0.108816
danceability_%	0.103542
released_month	0.101770
acousticness_%	0.098744
valence_%	0.091530
liveness_%	0.087118
energy_%	0.071582
key	0.060537
speechiness_%	0.050265
artist_count	0.021124

mode	0.020221
instrumentalness_%	0.002732

Importance values indicate the relative contribution of each feature. A higher value means the feature is more influential in making predictions. The sum of all importance scores equals 1 (or 100% if expressed as percentages).

Ranked Importance:
Features with higher importance scores have a greater impact on the model's performance. For instance:
released_day (0.182018) is the most important feature, meaning variations in the day a song is released have the greatest effect on the target variable (e.g., streams).
instrumentalness_% (0.002732) is the least important feature, suggesting it has minimal influence on the predictions.

What we really want to know is the optimal combination of features to produce the highest streams. That's really the goal of the model we built.

We can take a subset of the feature combinations, predict the streams for the sample combinations and find the combination with the highest predicted streams. Here's the results of the findings.

Feature	**Value**
artist_count	3.0
released_month	7
released_day	31
bpm	168.89
key	11
mode	1
danceability_%	0.1111
valence_%	0.7778
energy_%	0.6667
acousticness_%	0
instrumentalness_%	0.8889
liveness_%	0.5556
speechiness_%	0.3333
predicted_streams	1,338,916,000

Interpreting the results:

artist_count: 3

The combination with the highest predicted streams corresponds to tracks with 3 artists. This suggests that, based on the model, songs with three artists tend to have higher streaming numbers.

released_month: 7 (July)
The highest predicted streams occur for songs released in July. This could indicate that songs released in the middle of the year may tend to perform better in terms of streams, possibly due to seasonal trends or events like summer releases.

released_day: 31
The predicted highest streams occur for songs released on the 31st day of the month. This may reflect a preference for the end of the month in terms of song release strategies or an anomaly in the data.
bpm: 168.89
Songs with a bpm (beats per minute) of approximately 169 bpm are predicted to have the highest streams. This bpm suggests a relatively fast tempo, potentially indicating that high-energy songs (e.g., dance, electronic, or upbeat genres) may perform better.

key: 11 (likely B major or related scale)
The key value of 11 corresponds to a musical key, potentially B major or related to the B scale, based on how keys are encoded numerically. This suggests that songs in this particular key tend to perform better.

mode: 1
Mode = 1 likely corresponds to major scale (the mode could be major or minor). A major mode generally denotes happier, more positive music, which is often more popular for streaming.

danceability_%: 11.11%
Songs with a danceability value of 11.11% are predicted to have high streams. This suggests a preference for songs that are less danceable (relatively low in terms of rhythm or groove), which may imply that other musical qualities (e.g., emotional appeal or lyrical content) are more important.

valence_%: 77.78%
The valence (a measure of positivity or happiness) is predicted to be around 77.78%, indicating that the highest predicted streams occur for more positive, happy songs.

energy_%: 66.67%
The energy level of the song is predicted to be 66.67%, indicating songs that are somewhat energetic but not extremely high-energy. This could reflect a preference for songs that are lively but not overly intense.

acousticness_%: 0.00%

The predicted highest streams occur for songs with 0% acousticness, meaning that fully electronic or non-acoustic tracks (with no acoustic instruments) are favored in terms of streaming performance.

instrumentalness_%: 88.89%
Songs with a high instrumentalness value of 88.89% (likely instrumental or low vocal content) are predicted to perform well in terms of streams, indicating that non-vocal, instrumental tracks might attract more listeners in this case.

liveness_%: 55.56%
The liveness value of 55.56% suggests that the best-performing tracks have a moderate level of live performance characteristics, possibly including crowd noise, audience participation, or a sense of being recorded in front of a live audience.

speechiness_%: 33.33%
The speechiness value of 33.33% indicates that the best-performing tracks are likely to have moderate spoken word content (e.g., rap, podcasts, or songs with significant lyrics spoken or rapped).
In summary, this combination of features suggests that the highest-performing songs in terms of streams tend to have:

- Three artists
- Released in July on the 31st
- A relatively fast tempo (around 169 bpm)
- A key of **B major** or similar
- A **major** scale mode
- Moderate energy, positivity (valence), and instrumental content
- Minimal acousticness
- Moderate danceability and speech content

Conclusion

Objective 1: Study the correlation between musical attributes and song popularity

R-squared = 0.039: This means that only 3.9% of the variance in log-transformed streams is explained by the included musical attributes, suggesting an overall weak relationship. Adjusted R-squared = 0.019, reinforcing the weak predictive power of the model. Most of the variability in streaming numbers are unexplained, this is likely driven by non-musical factors (e.g., artist popularity, playlist inclusion, or marketing).

The F-statistic of 1.908 is low and indicates that the model has weak explanatory power. It doesn't do a good job of predicting the song streams. The Prob (F-statistic) is low enough to be significant though. So even though the model of the whole doesn't predict well, there is evidence that at least one of the musical attributes is related to the song streams.

A skew of –0.187 close to 0 suggests that the residuals are approximately symmetric, which is desirable in linear regression. Kurtosis (3.0353) is very close to 3, implying the residuals follow a normal distribution.

Significant predictors supported by lower P-Value:

Danceability's coefficient is -0.2937, indicating a negative relationship between danceability and streams. For a 1-unit increase in danceability, the log-streams decrease by approximately 0.2937 units. This is very surprising, I would have expected the opposite to be true in that the more danceable, the more streams.

Speechiness coefficient is -0.5156, indicating a negative relationship. Songs with higher speechiness tend to have lower log-streams. Acousticness coefficient is -0.1841, suggesting a negative relationship. Songs with higher acoustic qualities are less likely to have high log-streams.

Other musical attributes have high p-values above statistically significant levels, so we won't discuss those. They are showing no effect on song streams.

We reject the null hypothesis some of these attributes have significantly influence on streams.

Objective 2: Build a machine learning model to predict song popularity based on various features.

We built a machine learning model using a random forest regressor model to predict song streams based on the song's musical attributes.
We optimized the model by Hyperparameter Tuning with Grid Search.
Optimized Mean Squared Error: 2.5225785342976077e+17
Optimized R-squared: -0.030495366955971992
After optimizing our model, we were able to reduce our Mean Squared Error slightly. But its still a huge number. So our model still isn't very predictive.

Our optimized R-squared value improved a little too, now its closer to 0. But an R-Squared value of 0 just means that the model predicts none of the variance. So its still a ineffective model for predicting streams.

Appendix 1.

Features of the dataset:

track_name: Name of the song. artist(s)_name: Name of the artist(s) performing the song.
artist_count: Number of artists contributing to the song.
released_year, released_month, released_day: Release date details.
Streaming Metrics:
in_spotify_playlists: Number of Spotify playlists the song is featured in.
in_spotify_charts: Rank of the song on Spotify charts. streams: Total number of streams on **Spotify.**
in_apple_playlists, in_apple_charts: Presence in Apple Music playlists and charts.
in_deezer_playlists, in_deezer_charts: Presence in Deezer playlists and charts.
in_shazam_charts: Rank on Shazam charts. Musical Attributes:
bpm: Beats per minute, representing the tempo of the song.
key: Key of the song. mode: Indicates whether the song is in a major or minor mode.
danceability_%: Suitability of the song for dancing.
valence_%: Positivity of the song's musical content.
energy_%: Perceived energy level of the song.
acousticness_%: Acoustic sound presence in the song.
instrumentalness_%: Proportion of instrumental content in the track.
liveness_%: Presence of live performance elements. speechiness_%: Amount of spoken words in the song.