Billboard is a music industry organization that produces weekly rankings of the most popular songs and albums in the United States based on data from various sources. The Billboard charts are widely regarded as a standard measure of a song or album's success

and have been used by the industry for over 80 years.

Billboard gathers its data from various sources, including:

- Sales
- Streaming
- radio
- airplay

using a proprietary formula to calculate rankings based on this data.

This data is then compiled and published in the form of charts that are updated on a weekly monthly and yearly basis.

Spotify is a popular music streaming platform that provides users with access to a vast library of songs, albums, and playlists. In addition to its streaming service, Spotify also offers a range of **developer tools and APIs** that allow developers to access and utilize Spotify's vast catalog of audio content.

One such tool is the **Spotify Audio Features API**, which provides developers with detailed information about the **audio features of a particular song**, including **tempo**, **key**, **mode**, **energy**, and **danceability**.

Both Billboard and Spotify are considered reliable sources of music data due to the vast amount of data they collect and the rigorous methodologies they use to calculate rankings and provide information. **Billboard's data** is widely regarded as the **industry standard** and is used by music professionals to inform business decisions and measure the success of their releases. **Spotify's data**, on the other hand, is especially useful because it offers **in-depth details on a song's audio qualities**. As a result, we were able to **provide the Billboards tracks the musical and sound quality context we needed** to analyze each songs.

In conclusion, both **Billboard** and **Spotify** provide valuable data and information about the music industry and are widely regarded as reliable sources of information. Whether you're a music industry professional looking to measure the success of your releases, or a developer looking to create custom applications and tools that enhance the user experience on the platform, both Billboard and Spotify offer a wealth of data and resources to help you achieve your goals.

During the project, we utilized Python as our primary Integrated Development Environment (IDE). Using the **Selenium** library, we extracted data from **Billboard's yearly charts**. For each song, we extracted crucial information such as the song's name, artist name, year of the chart, and location in the chart.

Once we had collected all the necessary data from **Billboard**, we made **API requests** to **Spotify's APIs** to retrieve **additional information on each song**. Specifically, we utilized the **Track Audio Features API** to gather information on each song's audio features.

Overall, our process involved combining the power of two different tools to gather and analyze data. By using **Python** and **Selenium** to **extract data from Billboard's** yearly charts and Spotify's APIs to obtain audio features for each song, we were able to gain a comprehensive understanding of the songs' characteristics. This project highlights the importance of utilizing different techniques to achieve the best results.

In this project, we started by extracting audio features from Billboard songs using Spotify's API.

After acquiring all data from billboard, we utilized Spotify's API built in method to create data on songs that similar to the songs on Billboard but didn't make it there. for the same years, limiting the search to the US market. They removed any duplicate tracks that they had already extracted from the Billboard charts and then made another request to the Spotify API to get the audio features for these non-Billboard tracks.

However, we found that the dataset was not diverse enough and decided to extract audio features **for two additional years of Billboard tracks**, resulting in an additional 200 tracks. By doing this, they aimed to **increase the diversity** of their dataset and create a better representation of popular music in the US.

Overall, this project used a combination of **Billboard** and **non-Billboard tracks** to create a **comprehensive dataset of audio features**, that contain various music-related information such as musical analysis to popularity.

The project involved analyzing a dataset with the goal of **predicting hit songs**. The first step involved **visualizing** the data using histograms and boxplots to gain a better understanding of its distribution and identifying any potential outliers.

The next step involved cleaning the data using the **IQR** method, which involves removing any data points that fall outside the lower and upper bounds of the interquartile range. This was done to remove any potential **outliers** that could skew the data and negatively impact the accuracy of any machine learning models that were subsequently trained on the data.

After **cleaning** the data, we visualized it **again** using **bar plots** and **pie charts** to get a sense of the overall composition of the data.

A **correlation heatmap** was generated to identify any high-correlated columns in the data. This information was then used to create scatter plots to further explore the relationship between these variables.

Various **supervised machine learning models** were then generated, including **KNN**, **decision tree, logistic regression, and naive Bayes**. These models were trained on the dataset to predict hit songs, and their accuracy, precision scores, and confusion matrixes were calculated to determine which model performed the best.

We evaluated the success of each machine learning model and ultimately chose KNN as the best model to predict hit songs. The KNN model was then trained with a k value of 3, and its accuracy and precision scores were calculated.

However, after evaluating the machine learning results, we discovered that a **significant and important part of the data had been removed during the cleaning data section**. As a result, we decided to go back to the beginning of the machine learning part and train **a new KNN model using the original data without cleaning the outliers.**

We chose the **best k value** from the previous iteration of the model, which was 3, and trained the new KNN model on the original dataset. The model's accuracy and precision scores were then calculated and compared to the scores from the previous iteration of the model.

Ultimately, we **found** that the model **trained** on the **original dataset** without removing outliers **performed better** than the model that had been trained on the cleaned dataset. This suggests that the outliers that were removed during the cleaning process **contained important information that was necessary for more accurate predictions.**

In summary, the project involved analyzing a dataset to predict hit songs. The data was visualized using **histograms** and **boxplots**, and outliers were removed using the IQR method. The data was **visualized again** using bar plots and pie charts, and a correlation heatmap was generated to identify high-correlated columns in the data.

Various machine learning models were generated and trained on the data, and the KNN model was ultimately chosen as the best model to predict hit songs. However, after **evaluating the machine learning results**, we discovered that an **important part** of the data had been removed during the cleaning process. We eventually trained a new KNN model on the original dataset without removing outliers, and found that it **performed better than the previous iteration of the model.**

Through this project, we have learned the **importance** of data **visualization**, **data cleaning**, and **machine learning** in predicting hit songs. We found that visualizing data using histograms, boxplots, bar plots, and pie charts allowed us to gain a better understanding of the data and identify potential outliers.

We also learned about the IQR method for cleaning data, which involved removing outliers that could negatively impact the accuracy of machine learning models. We generated various supervised machine learning models, including KNN, decision tree, logistic regression, and naive Bayes, and evaluated their performance using accuracy and precision scores.

Additionally, we made the **conclusion** that **hit songs** should not be radical in **any parameter** such as speechness, instrumentality, tempo, and more.

the prime conclusion are, when you try to compose a hit, you need to use these guide line:

- Time signature is mostly 4
- Songs that has “danceability”>0.6 , has more chance to be **popular**.
- **Writing song** in the following **Keys**: C, C#, G, G# and 11 seems more effective then the other keys. And if a song is in the key: C , C# or G# they have the biggest chance to be popular.
- **Volume of track (**loudness) should be no more than > -10db

This **highlights** the importance of considering various factors when predicting the success of a song, rather than relying on a single parameter. Overall, this project has provided valuable insights into the process of predicting hit songs and the importance of data analysis and machine learning in this task.