22 May:

Documentation was read and various videos were watched to learn how to use the Spotify API. First, things like creating a token and obtaining authorization were learned. Then, commands that allow retrieving information about songs, artists, albums, and playlists using the features provided by Spotipy were learned, and work was done on how to access the information obtained from these commands.

After gaining sufficient knowledge about Spotipy, work was done on how to create a database in the most efficient way. It was decided to store the data with 4 different tables in SQL. Then, how to use SQL through Python was learned. For this, PostgreSQL documentation and videos were watched.

23 May:

A playlist containing 5370 rock/metal songs was created on Spotify to build the database. Work was done on appropriate SQL queries and a loop to fetch information from the playlist. Today, efforts were focused on how to precisely retrieve the desired information and resolve errors that occurred while saving the data. It was realized that the tables were not well-constructed, so corrections were made. As a result, information from 5370 songs was successfully saved into the artist, album, track, and analyze tables.

24 May:

Work was done on the audio analysis data of the songs. Using this data, different methods for creating a recommendation algorithm based on a given song were explored. Among the methods studied, an algorithm using cosine similarity to recommend songs was developed. The results were decent but need further improvement.

25 May:

To improve the algorithm, work was done on feature engineering. It was learned that better results could be achieved by transforming the data to make the distribution Gaussian. Outlier removal techniques were also studied. All the learned methods were applied to the data, and the recommendation system was tested to see if it yielded better results. After the changes, it was observed that the song recommendations were significantly more accurate.

26 May:

The created data was visualized in the most efficient way possible. Improvements were made to all the written code, and everything was prepared for presentation. The results were organized and transferred to a Jupyter Notebook in a structured manner.