

Creating a Music Playlist playing random songs

Ladva Umanshiva
AI22BTECH11016

INTRODUCTION

In this report, the procedure of making music playlist in which random songs are played by the user, scripted in Python, is mentioned. Also the song should not repeat until the entire playlist is played. The program uses libraries such as '**random**', '**tkinter**', '**os**', '**pydub**', '**pygame**' etc. I have made a GUI for the same.

PROCEDURE

- 1) Importing the required libraries:
 - In the beginning, import the required libraries such as '**os**', '**pydub**', '**pygame**', '**tkinter**', '**random**'
- 2) Creating dashboard for GUI:
 - Using **tkinter** library, defining the dimensions, background colour, title etc. for the GUI.
- 3) Creating a Playlist:
 - A function named **create _ playlist** is defined, that takes a file location or file path as parameter.
 - In the function, use **os** library to fetch the music files from the specified location.
 - Shuffle the list of fetched music files using **random.shuffle** command.
 - In the last step, return the shuffled music playlist.
- 4) Converting '**.m4a**' files to '**.wav**':
 - A function named **convert _ to _ wav** is defined to convert the **.m4a** files to **.wav** type files, as pygame do not recognize **.m4a** files
- 5) Defining functions to play song and stop song:
 - **play_song** and **stop_song** are defined for playing and stopping song when the play button is pressed. Pressing the '**play**' button will play the next song stopping the current song playing.
- 6) Function to play random songs:
 - A function named **play_random_song** is defined which takes parameters such as file path, playlist, songs played, and label as arguments.
 - While loop ensures that music plays until and unless the user terminates the program.
 - Select any random song from playlist using method **random.choice** of random module.
 - Remove the selected song from playlist and add to played song, which ensures that the song do not repeat unless the entire playlist is played.
 - Construct the full path to selected song with methods of **os** library
 - Check if the selected song is converted to **.wav** file and play the song.
- 7) Play button, label showing current song and tkinter mainloop:
 - Play button is used for playing the song. It also works as next button. Methods of **tkinter** is used for creating a play button.
 - Label shows the name of current song playing on the GUI dashboard.
 - The command **canvas.mainloop** is the tkinter loop for the GUI.

CONCLUSION

In this way, a music playlist can be generated using Python playing random songs with a interactive GUI.