

Probability Software Assignment

Name -:Pranay Jain
Roll no -: AI22BTECH11020

1 INTRODUCTION

The goal of this project was to create a Python program that plays audio files from a specified folder in a random order. The program utilizes the pygame library for audio playback and the numpy library for shuffling the playlist. The code allows to move/skip to the next song by clicking a button everytime.

2 IMPLEMENTATION

The program is implemented in Python and consists of the following key components:

- **File Selection:** The user provides the path to the folder containing the audio files.
- **Setting up Pygame mixer:** The audio directory variable stores the directory path where the audio files are located. `pygame.mixer.init()` initializes the Pygame mixer module for audio playback.
- **Shuffling audio files:** This block of code retrieves all the MP3 files in the audio directory using `os.listdir()` and filters only the files ending with ".mp3". The file paths are then stored in the audio files list. `np.random.shuffle(audio_files)` shuffles the list randomly using NumPy's `random.shuffle()` function.
- **Playing shuffled audio files:** `pygame.mixer.music.load(audio_file)` loads the current audio file. The base name of the audio file (without the directory path) is stored in the song name variable. `pygame.mixer.music.play()` starts playing the audio file.
- **Handling user input:** After starting the audio playback, it prints the name of the currently playing song. The loop waits for user input using `input()` and prompts for a command. If the command is "n" (for "next"), it stops the current song using `pygame.mixer.music.stop()` and breaks out of the loop, moving on to the next audio file.

3 USAGE

To use the program, follow these steps:

- 1) Run the program in a Python environment (Python 3 or above).
- 2) Provide the path to the folder containing the audio files.
- 3) The program will play the audio files in a random order each time it is run.

4 DEPENDENCIES

The program relies on the following external libraries:

- **pygame:** Used for audio playback and volume control.
- **numpy:** Used for shuffling the playlist.

Ensure that these libraries are installed in the Python environment before running the program.

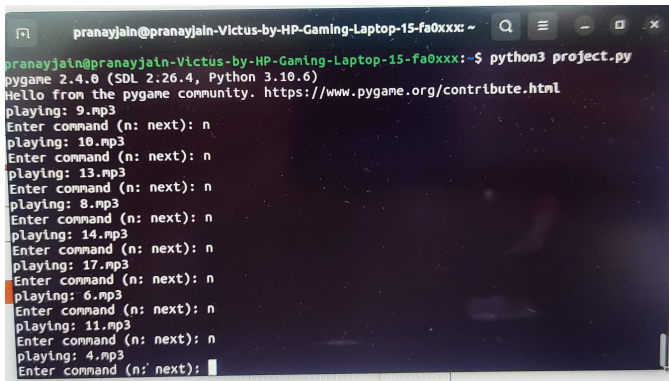
5 CONCLUSION

The project successfully achieves its objective of creating a random audio player. The program allows users to enjoy their audio collection in a randomized order. The combination of pygame and numpy libraries provides a seamless user experience.

In summary, the random audio player project demonstrates the effective utilization of Python libraries to create an enjoyable audio playback experience.

OUTPUT

Random numbers are generated on the display.



```
pranayjain@pranayjain-Victus-by-HP-Gaming-Laptop-15-fa0xxx: ~  
pranayjain@pranayjain-Victus-by-HP-Gaming-Laptop-15-fa0xxx:~$ python3 project.py  
pygame 2.4.0 (SDL 2.26.4, Python 3.10.6)  
Hello from the pygame community. https://www.pygame.org/contribute.html  
playing: 9.mp3  
Enter command (n: next): n  
playing: 10.mp3  
Enter command (n: next): n  
playing: 13.mp3  
Enter command (n: next): n  
playing: 8.mp3  
Enter command (n: next): n  
playing: 14.mp3  
Enter command (n: next): n  
playing: 17.mp3  
Enter command (n: next): n  
playing: 6.mp3  
Enter command (n: next): n  
playing: 11.mp3  
Enter command (n: next): n  
playing: 4.mp3  
Enter command (n: next):
```

Fig. 1. output