

Report for My Music Player code

Muppalla Bhavesh Chowdary
CS22Btech11041

1 INTRODUCTION

This document provides an explanation of a simple music player implemented using the Python programming language and the Tkinter and Pygame libraries.

2 CODE EXPLANATION

2.1 Importing Libraries

The code begins by importing the necessary libraries: Tkinter and mixer from pygame, prog (presumably a user-defined module), random, and time.

2.2 Initializing the Mixer

The mixer.init() function is called to initialize the audio mixer from the pygame library.

2.3 Print Statement

The line print(prog.finalarr) outputs the value of the finalarr variable from the prog module.

2.4 Play Function

The play function takes a song parameter, which is used to construct the file path of the corresponding MP3 file. The mixer.music.load() function loads the MP3 file, and mixer.music.play() starts playing the music.

2.5 Play All Function

The playall function plays all the songs in the finalarr list of the prog module. It calls the play function for each song, incrementing the songindex variable in the prog module after each song is played.

2.6 Pause and Resume Functions

The pausesong function pauses the currently playing music using mixer.music.pause(), and the resumesong function resumes the paused music using mixer.music.unpause().

2.7 Next Song Function

The nextsong function stops the currently playing music with mixer.music.stop() and proceeds to play the next song in the finalarr list, similar to the playall function.

2.8 Randomizer algorithm

2.8.1 Importing Libraries: The code snippet begins by importing the necessary libraries: numpy as num, playsound, and math.

2.8.2 Array Initialization: Two arrays are initialized with 20 elements each:

- arr: An array filled with zeros.
- finalarr: An array filled with zeros.

2.8.3 Variable Initialization: Two variables are initialized:

- index: A counter variable starting from 0.
- songindex: Another counter variable starting from 0.

2.8.4 Randomization Loop: The code enters a while loop that continues until index reaches 20. Within each iteration of the loop, the following steps are performed:

- 1) Generate a random number between 0 and 20 using num.random.uniform(0, 20).
- 2) Floor the random number to the nearest integer using math.floor(dummy).
- 3) Check if the element at the randomly generated index (dummy) in the arr array is 0.
- 4) If the condition is true, assign the value of dummy to the corresponding index in the finalarr array.
- 5) Set the element at the dummy index in the arr array to 1.
- 6) Increment the index variable by 1.

2.9 GUI Initialization

The code creates a Tkinter window using Tk(), sets its geometry and title, and makes it non-resizable.

2.10 Button Widgets

Five buttons are created using the `Button` class, with labels "Play", "Pause", "Resume", "Next", and "Randomize". Each button is associated with a specific function (`playall`, `pausesong`, `resumesong`, `nextsong`, and `randomizer`) using the `command` parameter.

2.11 Song Information

The information of the currently playing song is displayed below the "Next" button.

2.12 Window Update and Main Loop

The `root.update()` function updates the window, and `root.mainloop()` starts the Tkinter event loop, which keeps the window visible and processes user interactions until the program is terminated.

2.13 output



The screenshot shows a terminal window on the left and a Tkinter window titled "My music player" on the right. The terminal window displays the output of running a Python script, including the pygame version (2.4.0), a welcome message, a list of song indices, and a list of song files. The Tkinter window displays a simple music player interface with five buttons: "play", "pause", "resume", "next", and a text label showing the current song file path: `./mp3/15.mp3`.

```

bhavesh@bhavesh-HP-Pavilion-Laptop-14-dv2xxx:~/Desktop/Probability$ /bin/python3 /home/bhavesh/Desktop/Probability/userinterface.py
pygame 2.4.0 (SDL 2.26.4, Python 3.10.6)
Hello from the pygame community. https://www.pygame.org/contribute.html
[17, 3, 1, 12, 4, 19, 9, 6, 15, 0, 11, 7, 18, 13, 14, 8, 5, 10, 16, 2]
./mp3/17.mp3
./mp3/3.mp3
./mp3/1.mp3
./mp3/12.mp3
./mp3/4.mp3
./mp3/19.mp3
./mp3/9.mp3
./mp3/6.mp3
./mp3/15.mp3
  
```

My music player

play

pause

resume

next

./mp3/15.mp3