Software Assignment Report

Pallala Rishitha (BT22BTECH11011)

Assignment :

Code is written to take to take audio playlist and shuffle them using random variables. User interface has been created which has the options to 'shuffle and play' and 'Exit' from the user interface.

Analysis of code:

Libraries imported :

1. tkinter is used for creating the graphical user interface.
2. numpy is used for shuffling the playlist.
3. pygame is used for audio playback.

Defining the 'AudioPlaylistUI' class::

1. AudioPlaylistUI class is responsible for creating and managing the user interface.
2. init method is the class constructor and sets up the initial state of the interface.

Initializing the interface:

1. tk.Listbox widget is used to display and manage the playlist of audio files.
2. widget is packed into the interface window with a vertical padding of 10 pixels.

Adding the "Shuffle and Play" button:

1. tkButton widget is created with the text Shuffle and Play and assigned the command selfshuffleandplay .
2. When the button is clicked, it will call the shuffleandplay method of the class.
3. button is packed into the interface window with a vertical padding of 5 pixels.

Adding the "Exit" button:

1. The tkButton widget is created with the text "Exit" and assigned the command selfexit interface.
2. When the button is clicked, it will call the exit interface method of the class.
3. The button is packed into the interface window with a vertical padding of 5 pixels.

Defining the shuffle and play method:

1. The shuffle and play method is called when the ”Shuffle and Play” button is clicked.
2. It shuffles the items in the playlist by converting them to a list, shuffling the list, and updating the playlist accordingly.
3. The playlist index is set to 0 to start playing from the beginning.
4. The playing flag is set to True to indicate that the playlist is being played.
5. The play next method is called to start playing the shuffled playlist.

Defining the play next method:

1. The play next method is responsible for playing the next audio file in the shuffled playlist.
2. It checks if the playback is still ongoing (selfplaying) and if there are more items in the playlist to play.
3. It gets the audio file path from the current playlist index, loads the file into pygamemixermusic, and plays it.
4. It waits for the playback to finish using a while loop and then increments the playlist index to move to the next item.
5. It calls itself recursively to play the next audio file until the end of the playlist is reached.
6. When the playlist is finished, the playing flag is set to False.

Defining the exit interface method:

1. The exit interface method is called when the ”Exit” button is clicked.
2. It stops the current playback by setting the playing flag to False and stopping the music using pygamemixermusicstop().
3. It destroys the interface window by calling selfmasterdestroy().

Creating the interface window and running the event loop:

1. The tkTk() function creates the main window for the interface.
2. An instance of the AudioPlaylistUI class is created with the window as the master parameter.
3. The rootmainloop() call starts the event loop, which listens for user interactions and updates the interface accordingly.

How the code works:

By running this code, you will have a graphical user interface with the ”Shuffle and Play” and ”Exit” buttons Clicking the ”Shuffle and Play” button will shuffle the playlist and start playing the audio files. Clicking the ”Exit” button will stop the playback and close the interface window.

codes:

click here for Github link of the python code of the playlist

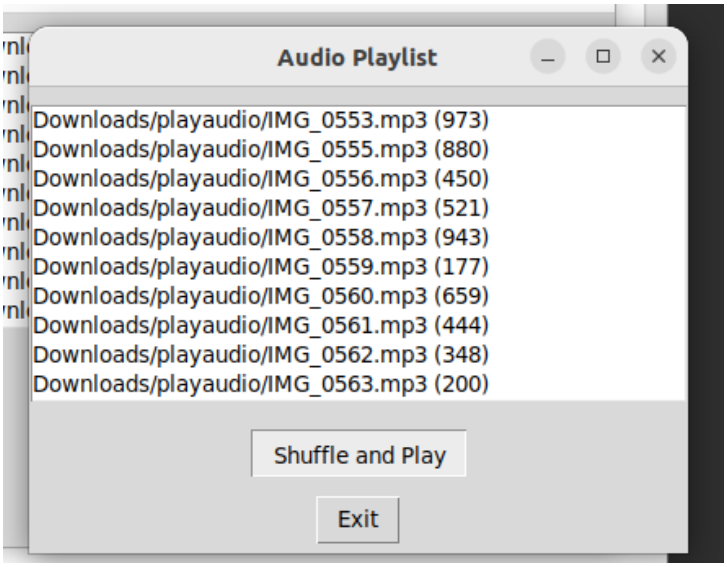

Figure 1: User Interface for the Playlist