AI1110
SOFTWARE ASSIGNMENT

Nalavolu Chetana
CS22BTECH11042

1 INTRODUCTION

In this project we made a list of songs to play randomly using a python code.We included some features like pause,resume,next and quit in addition to autoplay.

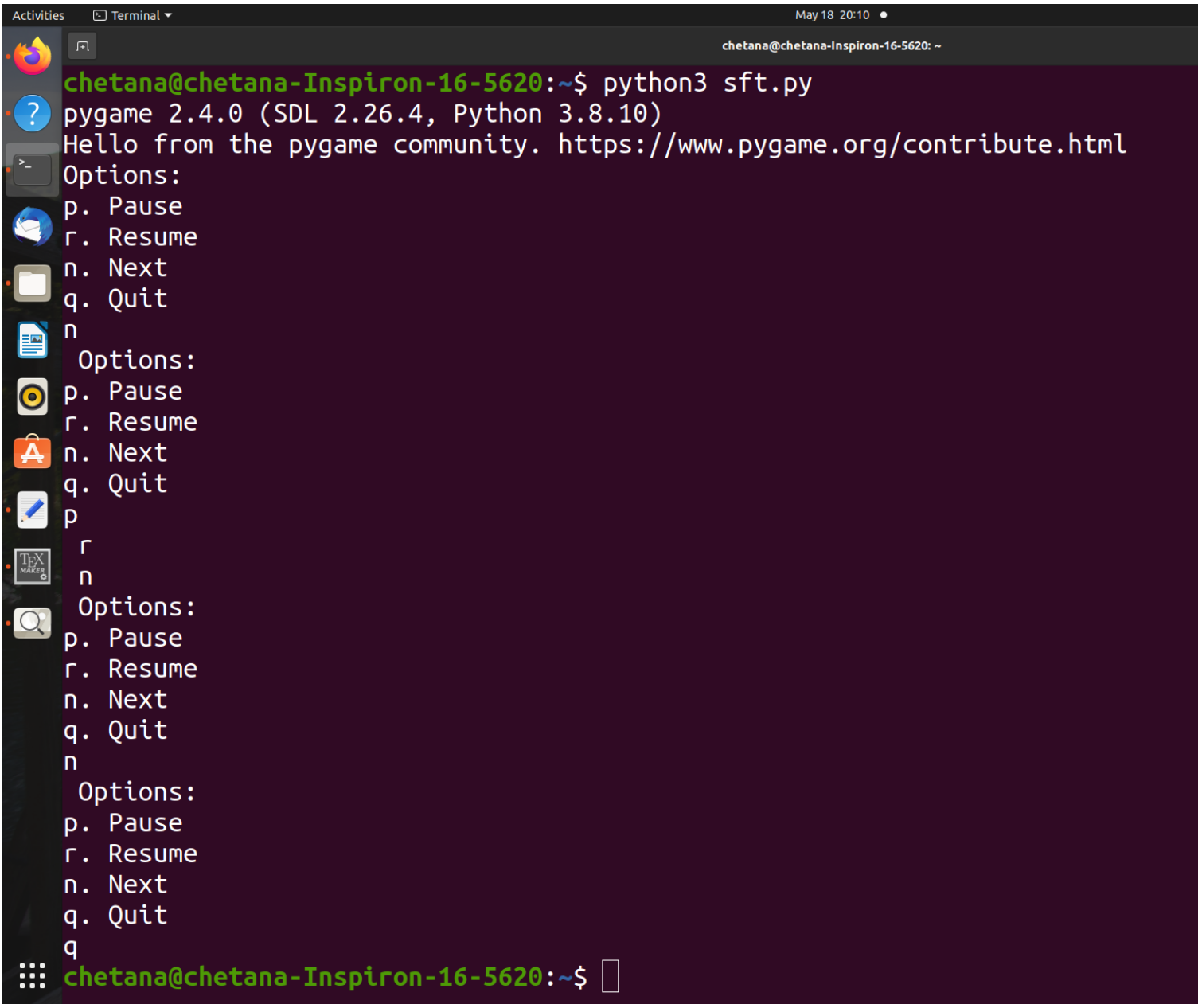
2 DESCRIPTION

1) The script begins by importing the necessary modules, including os, random, pygame, sys, and select.
2) It also defines the directory path where the songs are located. The main logic is enclosed within an infinite while loop to continuously play songs in a new random order.
3) First, the script retrieves a list of all the songs present in the specified directory using the os.listdir() function.
4) Then, the list is shuffled using the random.shuffle() function to randomize the order of songs for playback.
5) The Pygame library is initialized by calling pygame.mixer.init(). So that we can use some functions in this mixer module.
6) The script then iterates through each song in the shuffled list.
7) For each song, the script constructs the full path to the song file using os.path.join().
8) It loads the song using pygame.mixer.music.load(), which prepares the music file for playback. The song is played using pygame.mixer.music.play().
9) At this point, the script enters a loop that handles user input and song control. It first checks if the song is still playing or if it has been paused
10) The select.select() function is used to wait for input from the user within the specified time limit.So that if user do not give any input the playlist just plays automatically.
11) If there is user input available, the script reads the input and checks the corresponding choice. If the choice is 'p', the script pauses the song if it is currently playing. If the choice is 'r', the script resumes the paused song. If the choice is 'n', the script stops the current song and moves to the next song. If the choice is 'q', the script stops the music and exits the program.
12) If no user input available is within the time limit, the script breaks out of the inner loop and proceeds to the next song in the shuffled list.
13) Finally, after playing all the songs in the playlist, the Pygame mixer is terminated by calling pygame.mixer.quit().

3 CONCLUSION

The presented Python script demonstrates the implementation of a music player with user interaction using the Pygame library. By shuffling the playlist and providing options for pause, resume, skip, and quit, users have control over the playback of songs.

4 OUTPUT IMAGE



OUTPUT