

```
In [ ]: from tkinter import filedialog, messagebox
        from tkinter import *
        from pygame import mixer

        import os
        import tkinter as tk

        class Audio:
            def __init__(self):
                #Vbles
                self.version = "8.0"
                self.coment = "Cambio de libreria de audio"

                self.pause = False # Music has been paused
                self.onPlay = -1

                self.playlist = []
                self.playname = []

                # Init mixer
                mixer.init()

                # Interfaz grafica
                self.root = Tk()
                self.root.geometry('300x340')

            def run(self):
                self.load_buttons()

                self.monta_menu()

                self.root.mainloop()

            def load_music(self):

                self.audio_file = filedialog.askopenfilename(
                    title="Escoje musica", filetypes = (("mp3 files", "*.mp3"),
                    ("wav files", "*.wav"), ("ogg files", "*.ogg"),
                    ("flv files", "*.flv"), ("mp4 files", "*.mp4")))

                if self.audio_file != "":

                    self.audio_f = (self.audio_file.split("/"))[-1]
                    self.audio_name, self.ex = os.path.splitext(self.audio_f)

                    if self.onPlay == -1:
                        mixer.music.load(self.audio_file)

                        self.paz.set(self.audio_name + self.ex)
                        self.onPlay = 0

                    self.playlist.append(self.audio_file)
                    self.playname.append(self.audio_name + self.ex)

                    listToStr = '\n'.join([str(p) for p in self.playname])
                    self.playlistSTR.set(listToStr)

            def play_music(self):

                self.texto_boton.set("Pausa")
                self.pause = False
```

```
if self.onPlay == -1:
    messagebox.showwarning("Sin lista", "No hay lista de reproduccion")
    return False
try:
    mixer.music.load(self.playlist[self.onPlay])

    mixer.music.play()
    self.texto.set("Estatus: Reproduciendo")
except Exception as e:
    messagebox.showwarning("UNEXPECTED ERROR", str(e))

def stop_music(self):
    mixer.music.stop()
    self.texto.set("Estatus: Parado")

def back_music(self):
    self.texto.set("Estatus: Atrás")
    mixer.music.rewind()

def pause_music(self):
    if not self.pause :
        self.pause = True
        self.texto.set("Estatus: Pausado")
        mixer.music.pause()
        self.texto_boton.set("GO")
    else:
        self.pause = False
        self.texto.set("Estatus: DesPausado")
        mixer.music.unpause()
        self.texto_boton.set("Pausa")

def forw(self):
    if self.onPlay == -1:
        messagebox.showwarning("Sin lista", 'Crea una lista')
        return False
    elif self.onPlay >= len(self.playlist) - 1 :
        messagebox.showwarning("Final de lista", 'Te pasaste')
        return False

    playing = self.is_playing()

    self.onPlay += 1
    mixer.music.load(self.playlist[self.onPlay])
    self.paz.set(self.playname[self.onPlay])
    if playing == True:
        mixer.music.play()

def back(self):
    if self.onPlay == -1:
        messagebox.showwarning("Sin lista", 'Crea una lista')
        return False
    elif self.onPlay == 0:
        messagebox.showwarning("Inicio de lista", 'Estas al principio')
        return True

    playing = self.is_playing()

    self.onPlay -= 1
    mixer.music.load(self.playlist[self.onPlay])
    self.paz.set(self.playname[self.onPlay])

    if playing == True:
        mixer.music.play()
```

```

        mixer.music.play()

def stop(self):
    mixer.quit()

def load_buttons(self):

    self.paz = StringVar()
    self.texto = StringVar()
    self.playlistSTR = StringVar()
    self.texto_boton = StringVar()

    self.texto.set("Estatus: Parado")

    self.frame1 = Frame(self.root)
    self.frame1.pack(anchor=N)
    self.frame1.config(width=300,height=40)

    self.frame2 = Frame(self.root)
    self.frame2.pack(anchor=N)
    self.frame2.config(width=300,height=100)

    self.frame3 = Frame(self.root)
    self.frame3.pack(anchor=N)
    self.frame3.config(width=300,height=200,bg="red")

    self.frame4 = Frame(self.root)
    self.frame4.pack(anchor=N)
    self.frame4.config(width=300,height=200,bg="yellow")

    self.frame5 = Frame(self.root)
    self.frame5.pack(anchor=N)
    self.frame5.config(width=300,height=100)

    Label(self.frame1, textvariable=self.paz).pack()
    Label(self.frame1, textvariable=self.texto).pack()

    Button(self.frame2, text="Load", command=self.load_music, anchor="n", width=100, height=30,
           .grid(row=0, column=0, sticky="w", columnspan=4)
    Button(self.frame2, text="Play", command=self.play_music, anchor="n", width=100, height=30,
           .grid(row=1, column=0, sticky="w")
    Button(self.frame3, text="Stop", command=self.stop_music, anchor="n") \
           .grid(row=3, column=1, sticky="w")
    Button(self.frame3, text="Back", command=self.back_music, anchor="n") \
           .grid(row=3, column=2, sticky="w")

    self.texto_boton.set("Pausa")
    Button(self.frame3, textvariable=self.texto_boton, command=self.pause_music, anchor="n", width=100, height=30,
           .grid(row=3, column=3, sticky="w")

    Button(self.frame3, text="<<<", command=self.back, anchor="n") \
           .grid(row=3, column=4, sticky="w")
    Button(self.frame3, text=">>>", command=self.forw, anchor="n") \
           .grid(row=3, column=5, sticky="w")

    Label(self.frame4, text="Play list") \
           .grid(row=0, column=0, sticky="w")
    Label(self.frame5, textvariable=self.playlistSTR) \
           .grid(row=0, column=0, sticky="w")

def monta_menu(self):
    menubar = tk.Menu(self.root)
    self.root.config(menu=menubar)

```

```
auxmenu = tk.Menu(menuubar, tearoff=0)
auxmenu.add_command(label="Load", command=self.load_music)
auxmenu.add_separator()
auxmenu.add_command(label="Play", command=self.play_music)
auxmenu.add_command(label="Stop", command=self.stop_music)
auxmenu.add_command(label="Back", command=self.back_music)
auxmenu.add_command(label="Paus", command=self.pause_music)

menuubar.add_cascade(label="Menu", menu=auxmenu)

menu2 = tk.Menu(menuubar, tearoff=0)
menu2.add_command(label="<<<", command=self.back)
menu2.add_command(label=">>>", command=self.forw)

menuubar.add_cascade(label="Playlist", menu=menu2)

def is_playing(self):
    return mixer.music.get_busy()

audio1 = Audio()
audio1.run()
audio1.stop()
```

In []: