MP3 Player using Python

Requirements:-
pygame: music playing
tKinter: to make the basic GUI for the app
mutagen

Pygame is basically a game developing module from which we use the mixer.music method in order to load and play mp3 songs.

Mutagen is a complex python module which supports audio metadata from a particular file (metadata is additional information which comes with a file).

Tkinter:

root = Tk( ) // This command creates a blank window which is created by the root window. So anytime you see a root then think about blank window.

theLabel = Label(root, text="My name's Vivek") // This will allow us to print something on the screen. It takes a couple of parameters like where to put the text, and what text we want to put.

theLabel.pack( ) // This commands packs the text and places it somewhere in the blank window.

root.mainloop( ) // If we don't give this command then our computer will print the text and closes the window immediately after displaying. But if we give this command then all the commands will be put in an infinite loop which prevents the window from closing. It will only close once the close button is pressed.

// By default whenever we pack things, they get packed on top of each other.

pack(side=LEFT) // This command packs the object to the left side of the window.

topFrame = Frame(root)
topFrame.pack(side=TOP) // These commands create a top frame and divides the window into two parts.

button1 = Button(topFrame, text="Button 1", fg="red") // Here fg represents Foreground colour, bg represents Background colour and bd represents border.

One = Label(root, text="One", bg="red", fg="white")
One.pack(fill=X) // These commands creates a window with the text One inside it and also it expands in X direction when the window is widened. The same is also possible with Y.

<u>Grid Layout</u>:-

```
from tkinter import *
root = Tk( )
```

```
Label1 = Label(root, text="Name:")
Label2 = Label(root, text="Password:")
Entry1 = Entry(root)
Entry2 = Entry(root)
```

Label1.grid(row=0, sticky=E) // Creates a grid layout and places Label1 at row=0 and column=0
Label2.grid(row=1, sticky=E) // Here sticky commands allows us to align some text according to our need. Here we have given sticky=E. So it will right align the text.
Entry1.grid(row=0, column=1)
Entry2.grid(row=1, column=1)
c = Checkbutton(root, text="Keep me logged in") // This command creates a basic checkbox.

```
c.grid(columnspan=2) // This commands prints the text in only 1 row but in 2 different columns.

root.mainloop( )
```

Binding Functions to Widgets:-

1)

```
from tkinter import *
root = Tk( )

def myName( ):
        print(“My name is Vivek!”)

button1 = Button(root, text=“Print my name”, command=myName) // This command binds the function to the button and whenever button is pressed the function is called.
button1.pack( )

root.mainloop( )
```

2)

```
from tkinter import *
root = Tk( )

def myName(event):
        print(“My name is Vivek!”)

button1 = Button(root, text=“Print my name”)
button1.bind(“<Button-1>”, myName) // Here Button-1 represents left-click on the mouse, Button_2 represents the middle-click and Button-3 represents the right-click.
button1.pack( )

root.mainloop( )
```

Example:-

```
from tkinter import *
root = Tk()

def printMessage(event):
   print("Akshay is an asshole")

frame = Frame(root, width=300, height=250)
frame.pack(side=TOP)
button1 = Button(frame, text="Press Me")
button1.bind("<Button-1>", printMessage)
button1.pack()

button2 = Button(frame, text="Quit", command=root.destroy) // This command quits the window.
button2.pack()

root.mainloop()
```

A live Application:-

```
from tkinter import *
root = Tk()

# *** Main Menu ***
menu = Menu(root)
root.config(menu=menu)
```

```
subMenu = Menu(menu)
menu.add_cascade(label="File", menu=subMenu)
subMenu.add_command(label="New Project…")
subMenu.add_command(label="New…")
subMenu.add_separator()
subMenu.add_command(label="Exit")

editMenu = Menu(menu)
menu.add_cascade(label="Edit", menu=editMenu)
editMenu.add_command(label="Redo")

# *** Tool Bar ***
toolbar = Frame(root, bg="blue")

insertButton = Button(toolbar, text="Insert Image")
insertButton.pack(side=LEFT, padx=2, pady=2)
printButton = Button(toolbar, text="Print")
printButton.pack(side=LEFT, padx=2, pady=2)

toolbar.pack(side=TOP, fill=X)

# *** Status Bar ***
statusbar = Label(root, text="Preparing to do nothing…", bd=1, anchor=W, relief=SUNKEN)
statusbar.pack(side=BOTTOM, fill=X)

root.mainloop()
```

Message Boxes in tkinter:-

This is the simple code for displaying a message box in python using tkinter.

```
from tkinter import *
import tkinter.messagebox
root = Tk()

tkinter.messagebox.showinfo("Window Title", "Lions live for 20 years!") // Here 1st column
represents the title of the window and the 2nd one for the message to be displayed.

answer = tkinter.messagebox.askquestion("Question1", "Do you like Pok'emon")

if answer == 'yes':
        print("Charizard")

root.mainloop()
```

Canvas in tkinter:-

Canvas is the place on which we draw some shapes in tkinter.

```
from tkinter import *
root = Tk()

canvas = Canvas(root, width=200, height=100)
canvas.pack()

blackline = canvas.create_line(0, 0, 200, 50)
redline = canvas.create_line(0, 100, 200, 50, fill="red")
greenbox = canvas.create_rectangle(25, 25, 130, 65, fill="green")

canvas.delete(redline) // In case you want to delete it.
```

```
root.mainloop()
```

Photos in tkinter:-

We cannot display photos with gooey in tkinter, so we need to set them inside a label. So, we need to take a photo and put it in a label and as we know a label can be placed anywhere inside a window.

```
from tkinter import *
root = Tk()

photo = PhotoImage(file="vivek.jpg") // Remember that a photo must always be inside the python file folder.
label = Label(root, image=photo)
label.pack()

root.mainloop()
```