from tkinter import \*

import pymongo

main = Tk()

main.title("MongoDB Car Rental System")

button\_connect = Button(main, text="MongoDB Car Rental Database System" , bg="#f45941", command=lambda:connect())

button\_connect.pack(padx=40 , pady=25)

main.configure(background="#41d6f4")

def connect():

smain = Tk()

smain.geometry("300x100")

smain.configure(background="#a1dbcd")

myclient = pymongo.MongoClient("mongodb://localhost:27017/")

cardb = myclient["RentCar"]

mycars = cardb["allCars"]

rentedcars = cardb["rentedCars"]

label\_ok = Label(smain, text="Connection Established successfully!!!...Enjoy MongoDB")

label\_ok.grid(row=8, column=6)

smain.after(1000 , lambda: home1())

smain.after(2000 , lambda: smain.destroy())

def home1():

button\_ addcar = Button (main, text="Add a Car to garage", bg= "#41f4e2" , command=lambda:addcarfn())

button\_addcar.pack(padx=400, pady=25)

button\_rmcar = Button(main, text="Remove a Car from Garage", bg ="#FFD43B", command=lambda:rmcarfn())

button\_rmcar.pack(padx=400, pady=25)

button\_rent = Button(main, text="Rent a Car", bg="#FF0000", command=lambda:rentcarfn())

button\_rent.pack(padx=400, pady=25)

button\_return = Button(main, text="Return a Car",bg="#0000FF",command=lambda:retcarfn())

button\_return.pack(padx=400, pady=25)

button\_upcar = Button(main, text="Update Car Details",bg="#00FF00",command=lambda:upcarfn())

button\_upcar.pack(padx=400, pady=25)

def addcarfn():

smain = Tk()

smain.geometry("600x300")

smain.configure(background="#42f4c5")

smain.title("Add Car to Garage")

label\_snum = Label(smain, fg="#FF0000" , bg="#42f4c5", text="Enter Serial Number of car")

label\_snum.grid(row=6, column=6)

entry\_snum = Entry(smain)

entry\_snum.grid(row=6, column=100)

label\_aname = Label(smain,fg="#FF0000" , bg="#42f4c5", text="Enter Model Name of car")

label\_aname.grid(row=8, column=6)

entry\_aname = Entry(smain)

entry\_aname.grid(row=8, column=100)

label\_mil = Label(smain, fg="#FF0000" , bg="#42f4c5", text="Enter Mileage")

label\_mil.grid(row=10, column=6)

entry\_mil = Entry(smain)

entry\_mil.grid(row=10, column=100)

label\_rnge = Label(smain, fg="#FF0000" , bg="#42f4c5", text="Enter Range")

label\_rnge.grid(row =12 , column=6)

entry\_rnge = Entry(smain)

entry\_rnge.grid(row=12 , column=100)

label\_ppk = Label(smain, fg="#FF0000" , bg="#42f4c5", text="Enter Price per Km")

label\_ppk.grid(row=14 , column=6)

entry\_ppk = Entry(smain)

entry\_ppk.grid(row=14 , column=100)

button\_submit = Button(smain, text="Add", bg="#FF0000", command=lambda:insertdb(entry\_snum, entry\_aname, entry\_mil , entry\_rnge , entry\_ppk))

button\_submit.grid(row=20, column=10)

def insertdb(snum,aname,mil,rnge,ppk):

snum = snum.get()

aname = aname.get()

mil = mil.get()

ppk = ppk.get()

rnge = rnge.get()

val = {"Serial Number":snum, "Model Name":aname, "Mileage":mil, "Range":rnge, "Price per Km":ppk}

x = mycars.insert\_one(val)

smain.destroy()

def rmcarfn():

smain = Tk()

smain.geometry("800x800")

smain.configure(background="#4183f4")

smain.title("Remove a Car from Garage")

lbox = Listbox(smain, width=150)

for x in mycars.find({} , {"Serial Number":1, "Model Name":1, "Mileage":1, "Range":1, "Price per Km":1}):

x = dict(x)

lbox.insert(1,x)

lbox.pack()

label\_rm = Label(smain, fg="#FFD43B" , bg="#306998", text="Enter Serial number of car to remove from garage")

label\_rm.pack()

entry\_rm = Entry(smain)

entry\_rm.pack()

button\_rm = Button(smain, bg="#FFD43B", text="Remove", command=lambda:rmdb(entry\_rm))

button\_rm.pack()

def rmdb(rmsnum):

snum = rmsnum.get()

myquery = {"Serial Number":snum}

mycars.delete\_one(myquery)

smain.destroy()

def rentcarfn():

smain = Tk()

smain.geometry("800x800")

smain.title("Rent a Car")

smain.configure(background="#ffccff")

lbox = Listbox(smain, width=150)

for x in mycars.find({},{"\_id":0, "Serial Number":1, "Model Name":1, "Mileage":1, "Range":1, "Price per Km":1}):

x = dict(x)

lbox.insert(1,x)

lbox.pack()

label\_rent = Label(smain,fg="#FF0000" , bg="#ffccff", text="Enter Serial number of book to rent")

label\_rent.pack()

entry\_rent = Entry(smain)

entry\_rent.pack()

button\_rent = Button(smain, bg="#FF0000" , text="Rent", command=lambda:rentcar(entry\_rent))

button\_rent.pack()

def rentcar(erent):

rent = erent.get()

for x in mycars.find():

if x["Serial Number"]==rent:

rentedcars.insert\_one(x)

mycars.delete\_one(x)

smain.destroy()

def retcarfn():

smain = Tk()

smain.geometry("800x800")

smain.title("Return Car")

smain.configure(background="#FFD43B")

lbox = Listbox(smain, width=150)

for x in rentedcars.find({},{"\_id":0, "Serial Number":1,"Model Name":1, "Mileage":1 , "Range":1, "Price per Km":1}):

x = dict(x)

lbox.insert(1,x)

lbox.pack()

label\_ret = Label(smain, fg="#FF0000" , bg="#FFD43B" , text="Enter Serial Number to return")

label\_ret.pack()

entry\_ret = Entry(smain)

entry\_ret.pack()

button\_ret = Button(smain, bg="FF0000", text="Return", command=lambda:retcar(entry\_ret))

button\_ret.pack()

def retcar(eret):

ret = eret.get()

for x in rentedcars.find():

if x["Serial Number"]==ret:

rentedcars.delete\_one(x)

mycars.insert\_one(x)

smain.destroy()

def upcarfn():

smain = Tk()

smain.geometry("600x600")

smain.configure(background="#a1dbcd")

smain.title("Update Car Details")

lbox = Listbox(smain, width=150)

for x in mycars.find({},{"\_id":0, "Serial Number":1,"Model Name":1, "Mileage":1 , "Range":1, "Price per Km":1}):

x = dict(x)

lbox.insert(1,x)

lbox.pack()

label\_snum = Label(smain, fg="#FF0000" , bg="#a1dbcd", text="Enter Serial Number to update")

label\_snum.pack()

entry\_snum = Entry(smain)

entry\_snum.pack()

label\_mil = Label(smain, fg="#FF0000" , bg="#a1dbcd", text="Enter Mileage")

label\_mil.pack()

entry\_mil = Entry(smain)

entry\_mil.pack()

label\_rnge = Label(smain , fg="#FF0000" , bg="#a1dbcd", text="Enter Range")

label\_rnge.pack()

entry\_rnge = Entry(smain)

entry\_rnge.pack()

label\_ppk = Label(smain, fg="#FF0000" , bg="#a1dbcd" ,text="Enter Price per Km")

label\_ppk.pack()

entry\_ppk = Entry(smain)

entry\_ppk.pack()

button\_up = Button(smain, bg="#FF0000" , text="Update", command=lambda:updb(entry\_snum, entry\_mil, entry\_rnge, entry\_ppk))

button\_up.pack()

def updb(snum, mil, rnge , ppk):

snum = snum.get()

mil = mil.get()

rnge = rnge.get()

ppk = ppk.get()

for x in mycars.find():

if x["Serial Number"]==snum:

oldval = {"Mileage":x["Mileage"], "Range":x["Range"], "Price per Km":x["Price per Km"]}

newval = {"$set":{"Mileage":mil, "Range":rnge, "Price per Km":ppk}}

mycars.update\_one(oldval, newval)

smain.destroy()