**“ROLLS ROYCE CAR SHOWROOM”**

**A COMPUTER SCIENCE PROJECT REPORT**

**SUBMITTED BY**

**DIVYAM VANJARA**

**IN PARTIAL FULFILMENT OF THE**

**AISSCE – 2022-23**

**IN**

**COMPUTER SCIENCE (083)**

**AT**



**J.B. DIAMOND & KARP IMPEX VIDYA SANKUL**

**LASKANA, KAMREJ ROAD, SURAT**

**J.B. Diamonds & KARP Impex Vidya Sankul**

Opp. Diamond Nagar, B/H Thakor Dwar Farm, Surat - Kamrej Road, Laskana

**Phone No: 9228025712, Email id: jbkarpschool.cbse@gmail.com**

**Web: www.jbkarpschool.ac.in**

**CBSE-English Medium.**

**Certificate**

This is to certify that **Mr DIVYAM VANJARA** is a student of J. B. Diamonds & KARP Impex Vidya Sankul, who has successfully completed the project work on title **ROLLS ROYCE CAR SHOWROOM** in **COMPUTER SCIENCE (083)** assigned to him as a part of AISSCE curriculum during the academic year **2022-23.**

We found him very sincere, hardworking and disciplined boy.

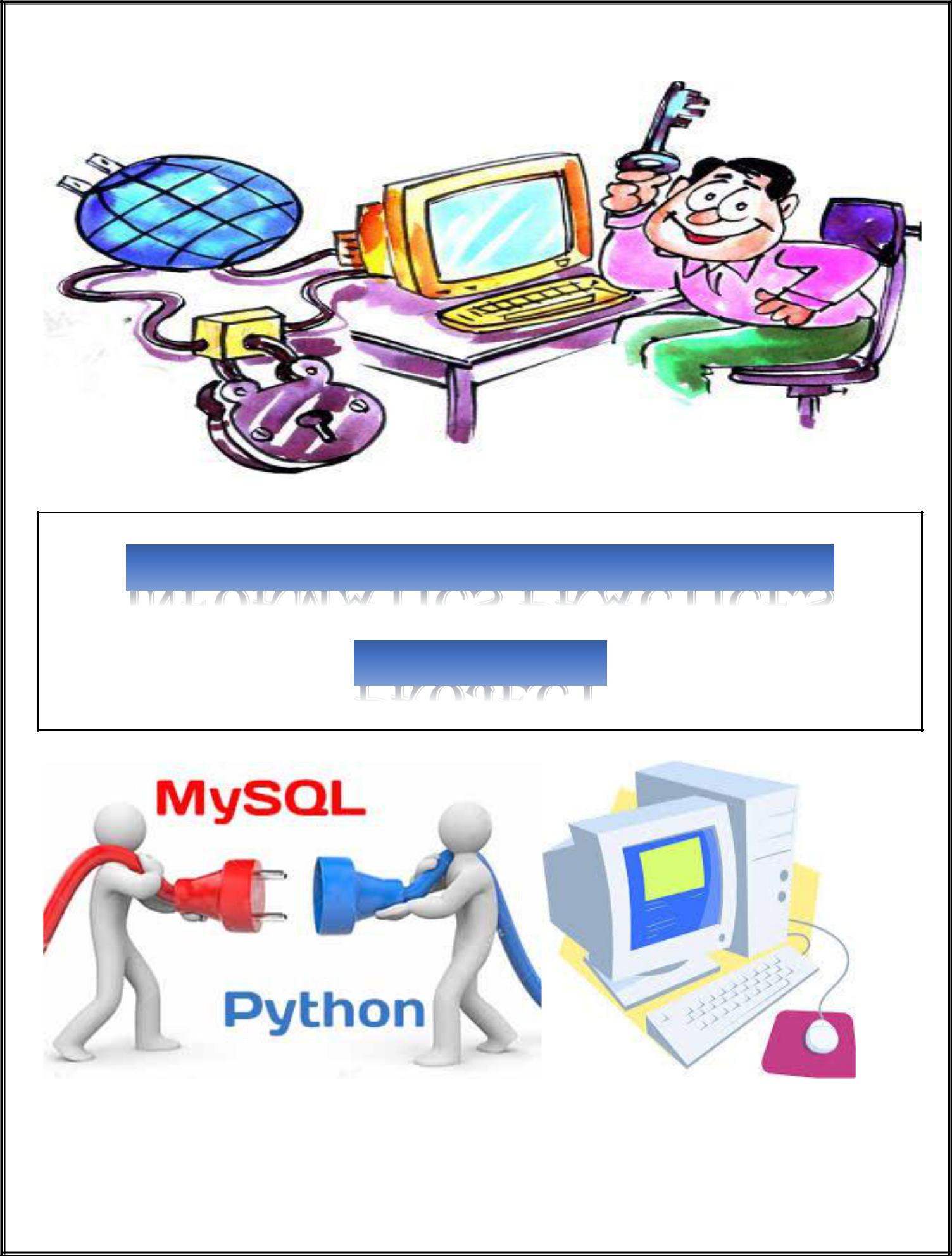
We wish all the success for his future endeavors.

**…………………………………………… ……………………………………………….**

**Signature of the Internal Examiner Signature of the External Examiner**

**………………………………………….**

**Principal Signature**

****

**PROJECT FILE**

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my Computer Science teacher **Mr.** **Ajay Tiwari Sir** as well as our principal **Mr.** **Gaurang Patel Sir** for their guidance and support in completing this wonderful project entitled **“ROLLS ROYCE CAR SHOWROOM”** using **Python - MySQL connectivity**”.

I came to know about many new things. I am really thankful to them.

A debt of gratitude is also owed to my parents and friends who helped me with their valuable suggestions.

Although this report has been prepared with utmost care and deep routed interest, even then I accept respondents and imperfections.

CONTENT

****

**ROLLS ROYCE**

**Car Showroom**

**Using My SQL Connectivity**



* **What is Python?**
  + The Python Programming Language is a recent, general-purpose and higher-level programming language. It has features for database programming also.
  + This project aims on explaining how one can create a MySQL database from within a Python script and create a user interface software.
* **Why Python?**
  + - Due to its open source nature, Python has been ported to many platforms.
    - It is free and open source. It is available for free and runs on almost every current platform.
    - Python provides interfaces to all major commercial databases.
    - It can easily integrated with C, C++, COM, Java, MySQL, etc.
* **What is MySQL?**
  + MySQL is a freely available open source Relational Database Management System (RDBMS) that uses Structured Query Language (SQL).
  + It provides you with a rich set of features that support a secure environment for storing, maintaining, and accessing data.
* **Why MySQL?**
  + It is an open source software and is easily portable.
  + It is easy to use, manage and works quickly and efficiently.
  + It is used to create databases, manage security of a database.
  + It maintains integrity and reduces data redundancy.

|  |  |
| --- | --- |
| **Interface Python with MySQL**  Python is a | MySQL is a |
| Front End | Back End |
| Software | Software |

There are mainly seven steps that must be followed in order to create a database connectivity application.

**Step 1** –Start Python

**Step 2** –Import the packages required for databaseprogramming.

**Step 3** –Open a connection to database.

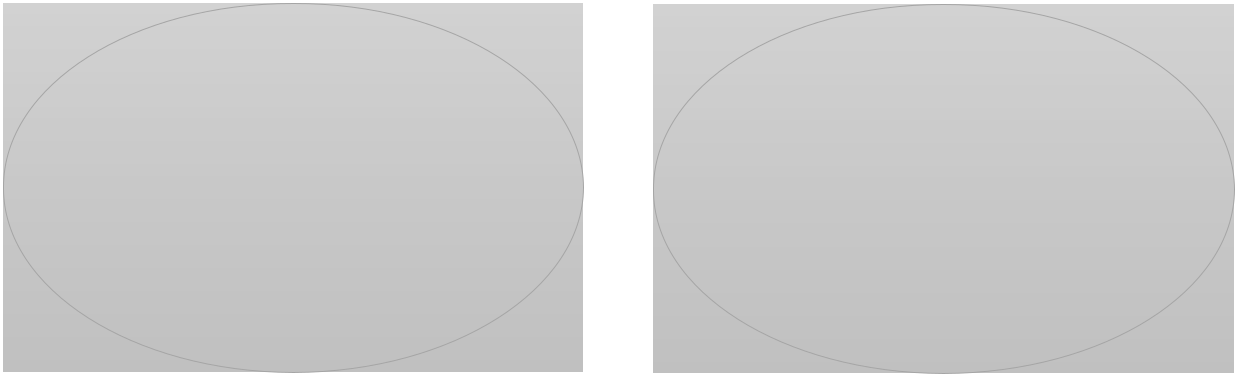
**Step 4** –Create a cursor instance.

**Step 5** –Execute a query.

**Step 6** –Extract data from result set.

**Step 7** –Clean up the environment.





**Source Code:**

import tkinter

from tkinter import \*

from tkinter import ttk

from tkinter import ttk

from tkinter import messagebox

import mysql.connector as sql

import datetime as dt

import time

from subprocess import call

#from PIL import Image, ImageTk

def SplashScreen():

splashscreen = Tk()

splashscreen.overrideredirect(1) # Remove Title Bar

splashscreen.geometry(

f"825x500+{(splashscreen.winfo\_screenwidth() - 825) // 2}+{(splashscreen.winfo\_screenheight() - 500) // 2}")

splashscreen.configure(bg='white',bd=10,relief=SUNKEN)

Label(splashscreen, text='ROLLS ROYCE', font='Algerian 35', fg='black', bg='white',bd=10,relief=RAISED).pack()

#Add image

image1 = PhotoImage(file="aa.png")

label = Label(splashscreen, image=image1, relief = 'raise', bd = 5).pack()

Label(splashscreen, text="Version 2.O", font='ALGERIAN 10 ', bg='white', fg='black',bd=10,relief=RAISED).place(x=695, y=55)

pgbar = ttk.Progressbar(splashscreen, orient='horizontal', length=600, mode='indeterminate')

Label(splashscreen, text="Designed By: Divyam Vanjara ", font='Algerian 13', bg='white', fg='black',bd=10,relief=RAISED).place(x=517, y=350)

Label(splashscreen, text="12th Science-B", font='Algerian 13', bg='white', fg='black',bd=10,relief=RAISED).place(x=640, y=400)

pgbar.place(x=70, y=450)

pgbar['maximum'] = 100

txt=Label(splashscreen,text='0%',relief=GROOVE,bg='black',fg='white')#, bg='#345', fg='#fff')

txt.place(x=675, y=450)

for i in range(101):

time.sleep(0.01)

pgbar['value'] = i

pgbar.update()

txt['text']=pgbar['value'],'%'

splashscreen.destroy()

splashscreen.mainloop()

mydb=sql.connect(host="localhost",user="root",password="12345")#connection to mysql

mycur=mydb.cursor()

mycur.execute("create database if not exists car")

mycur.execute("use car")

mycur.execute('Create table if not exists details(cid varchar(30), fname varchar(30),lname varchar(30), g varchar(30), \

mno varchar(30), id varchar(30), idno varchar(50), \

cname varchar(30), model varchar(30), ftype varchar(30), \

ctype varchar(30), colour varchar(30), cprice varchar(30))')

'''

mycur.execute("create table if not exists appointment"

"("

"idno varchar(12) primary key,"

"name char(50),"

"age char(3),"

"gender char(1),"

"phone varchar(10),"

"bg varchar(3))")

'''

class CarManagementSyste:

def \_\_init\_\_(self,root):

self.root=root

self.root.iconbitmap("ab.png")

self.root.title("CAR WORLD")

self.root.geometry("1350x690+0+0") #Fixing window size according to monitor resolution

#=============================== Variable Details ================================================#

self.model=StringVar()

self.cname=StringVar()

self.colour=StringVar()

self.ctype=StringVar()

self.ftype=StringVar()

self.mno=StringVar()

self.fname=StringVar()

self.lname=StringVar()

self.g=StringVar()

self.cid=StringVar()

self.cprice=StringVar()

self.id=StringVar()

self.idno=StringVar()

lblTitle=Label(self.root,text="ROLLS ROYCE",bg="black",fg="WHite",bd=10,relief=RIDGE,\

font=("Bernard MT Condensed",50,"bold"),padx=10,pady=10)

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

self.lbb=Label(self.root,bg='white')

self.lbb.place(x=25,y=15, width=49, height=90)

self.ig0=PhotoImage(file='ab.png')

self.lbb.config(image=self.ig0)

self.lbb=Label(self.root,bg='white')

self.lbb.place(x=1275,y=15, width=67, height=90)

self.ig8=PhotoImage(file='ac.png')

self.lbb.config(image=self.ig8)

frame=Frame(self.root,bd=12,relief=RIDGE,padx=20,bg="white")

frame.place(x=0,y=120,width=1365,height=275)

DataFrameLeft=LabelFrame(frame,text="Customer And Car Details",bg="black",fg="white",bd=12,relief=RIDGE,font=("times new roman",12,"bold"),padx=2,pady=3)

DataFrameLeft.place(x=-13,y=0,width=1327,height=250)

lbl2=Label(DataFrameLeft,bg="black",fg="White",text="Car Name",font=("times new roman",15,"bold"),padx=20,pady=3)

lbl2.grid(row=0,column=3,sticky=W)

txtPRN\_No=Entry(DataFrameLeft, textvariable=self.cname,font=("times new roman",15),width=18)

txtPRN\_No.grid(row=0,column=4,sticky=W)

lbl1=Label(DataFrameLeft,bg="black",fg="White",text="Customer Details:",font=("times new roman",13,"bold"),padx=2,pady=3)

lbl1.grid(row=1,column=0,sticky=W)

lblid=Label(DataFrameLeft,bg="black",fg="white",text="Customer Id",font=("times new roman",12,"bold"),padx=2,pady=3)

lblid.grid(row=2,column=0,sticky=W)

txtid=Entry(DataFrameLeft, textvariable=self.cid,font=("times new roman",12),width=20)

txtid.grid(row=2,column=1,sticky=W)

lblname=Label(DataFrameLeft,bg="black",fg="white",text="Frist Name",font=("times new roman",12,"bold"),padx=30,pady=3)

lblname.grid(row=2,column=2,sticky=W)

txtname=Entry(DataFrameLeft, textvariable=self.fname,font=("times new roman",12),width=20)

txtname.grid(row=2,column=3,sticky=W)

lbllname=Label(DataFrameLeft,bg="black",fg="white",text="Last Name",font=("times new roman",12,"bold"),padx=55,pady=3)

lbllname.grid(row=2,column=4,sticky=W)

txtlname=Entry(DataFrameLeft, textvariable=self.lname,font=("times new roman",12),width=20)

txtlname.grid(row=2,column=5,sticky=W)

lblg=Label(DataFrameLeft,bg="black",fg="white",text="Gender",font=("times new roman",12,"bold"),padx=30,pady=3)

lblg.grid(row=2,column=6,sticky=W)

cmbg=ttk.Combobox(DataFrameLeft,textvariable=self.g,font=("times new roman",12,"bold"),width=18,state="readonly")

cmbg["value"]=("Male","Female","Other")

cmbg.current(0)

cmbg.grid(row=2,column=7,sticky=W)

lblp=Label(DataFrameLeft,bg="black",fg="white",text="Id Proof Type",font=("times new roman",12,"bold"),padx=30,pady=3)

lblp.grid(row=3,column=2,sticky=W)

cmbp=ttk.Combobox(DataFrameLeft,textvariable=self.id,font=("times new roman",12,"bold"),width=18,state="readonly")

cmbp["value"]=("Aadhar Card","Voter Id","Other")

cmbp.current(0)

cmbp.grid(row=3,column=3,sticky=W)

lblMobilenumber=Label(DataFrameLeft,bg="black",fg="white",text="Mobile Number",font=("times new roman",12,"bold"),padx=2,pady=3)

lblMobilenumber.grid(row=3,column=0,sticky=W)

txtMobilenumber=Entry(DataFrameLeft, textvariable=self.mno,font=("times new roman",12),width=20)

txtMobilenumber.grid(row=3,column=1,sticky=W)

lblMobilepnumber=Label(DataFrameLeft,bg="black",fg="white",text="Aadhar/Voter/other Id No.",font=("times new roman",12,"bold"),padx=2,pady=3)

lblMobilepnumber.grid(row=3,column=4,sticky=W)

txtMobilepnumber=Entry(DataFrameLeft, textvariable=self.idno,font=("times new roman",12),width=20)

txtMobilepnumber.grid(row=3,column=5,sticky=W)

lblCardetails=Label(DataFrameLeft,bg="black",fg="white",text="Car Details:",font=("times new roman",13,"bold"),padx=2,pady=3)

lblCardetails.grid(row=4,column=0,sticky=W)

lblmodel=Label(DataFrameLeft,bg="black",fg="white",text="Car Model",font=("times new roman",12,"bold"),padx=2,pady=3)

lblmodel.grid(row=5,column=0,sticky=W)

txtmodel=Entry(DataFrameLeft, textvariable=self.model,font=("times new roman",12),width=20)

txtmodel.grid(row=5,column=1,sticky=W)

lbltype=Label(DataFrameLeft,bg="black",fg="white",text="Fuel Type",font=("times new roman",12,"bold"),padx=30,pady=3)

lbltype.grid(row=5,column=2,sticky=W)

cmbtype=ttk.Combobox(DataFrameLeft,textvariable=self.ftype,font=("times new roman",12,"bold"),width=18,state="readonly")

cmbtype["value"]=("Petrol","Diesel","Other")

cmbtype.current(0)

cmbtype.grid(row=5,column=3,sticky=W)

lblctype=Label(DataFrameLeft,bg="black",fg="white",text="Car Type",font=("times new roman",12,"bold"),padx=55,pady=3)

lblctype.grid(row=5,column=4,sticky=W)

cmbctype=ttk.Combobox(DataFrameLeft,textvariable=self.ctype,font=("times new roman",12,"bold"),width=18,state="readonly")

cmbctype["value"]=("4-seater","6-seater")

cmbctype.current(0)

cmbctype.grid(row=5,column=5,sticky=W)

lblcolour=Label(DataFrameLeft,bg="black",fg="white",text="Car Colour",font=("times new roman",12,"bold"),padx=30,pady=3)

lblcolour.grid(row=5,column=6,sticky=W)

cmbcolour=ttk.Combobox(DataFrameLeft,textvariable=self.colour,font=("times new roman",12,"bold"),width=18,state="readonly")

cmbcolour["value"]=("Blue","Grey","Dark Blue",'Black')

cmbcolour.current(0)

cmbcolour.grid(row=5,column=7,sticky=W)

lblprice=Label(DataFrameLeft,bg="black",fg="white",text="Car Price",font=("times new roman",12,"bold"),padx=2,pady=3)

lblprice.grid(row=6,column=0,sticky=W)

txtprice=Entry(DataFrameLeft, textvariable=self.cprice,font=("times new roman",12),width=20)

txtprice.grid(row=6,column=1,sticky=W)

#=============================== DataFrame Right ================================================#

FrameButton=Frame(self.root,bd=12,relief=SUNKEN,padx=20,bg="black")

FrameButton.place(x=0,y=395,width=1366,height=145)

btnAddData01=Button(FrameButton,text="Rolls Royce Cullian",font=("arial",7,"bold"),width=25,bg="white",fg="black",relief=RAISED,bd=5)

btnAddData01.place(x=20,y=100)

self.lbb=Label(self.root,bg='white')

self.lbb.place(x=30,y=405, width=210, height=97)

self.ig=PhotoImage(file='12.png')

self.lbb.config(image=self.ig)

btnAddData2=Button(FrameButton,text="Rolls Royce Ghost",font=("arial",7,"bold"),width=25,bg="white",fg="black",relief=RAISED,bd=5)

btnAddData2.place(x=370,y=100)

self.lbb=Label(self.root,bg='white')

self.lbb.place(x=350,y=405, width=262, height=97)

self.ig2=PhotoImage(file='11.png')

self.lbb.config(image=self.ig2)

btnAddData1=Button(FrameButton,text="Rolls Royce Ghost",font=("arial",7,"bold"),width=25,bg="white",fg="black",relief=RAISED,bd=5)

btnAddData1.place(x=730,y=100)

self.lbb=Label(self.root,bg='white')

self.lbb.place(x=720,y=405, width=252, height=100)

self.ig3=PhotoImage(file='13.png')

self.lbb.config(image=self.ig3)

btnAddData3=Button(FrameButton,text="Rolls Royce Cullian",font=("arial",7,"bold"),width=25,bg="white",fg="black",relief=RAISED,bd=5)

btnAddData3.place(x=1070,y=100)

self.lbb=Label(self.root,bg='white')

self.lbb.place(x=1070,y=405, width=224, height=97)

self.ig4=PhotoImage(file='14.png')

self.lbb.config(image=self.ig4)

#================================================================

def cullian(self):

self.cname.set("Rolls Royce Cullian")

self.model.set("Cullian")

self.cprice.set("1200000")

#====================================== Information Frames ======================================#

FrameDetails=Frame(self.root,bd=12,relief=SUNKEN,padx=20,bg="black")

FrameDetails.place(x=0,y=590,width=1366,height=120)

xScroll=ttk.Scrollbar(FrameDetails,orient=HORIZONTAL)

yScroll=ttk.Scrollbar(FrameDetails,orient=VERTICAL)

self.Car\_Table=ttk.Treeview(FrameDetails,column=("cid","fname","lname","g","mno","id","idno","cname",\

"model","ftype","ctype","colour","price"),\

x=xScroll.set,y=yScroll.set) #Creating table to show the books borrowed information in tabular form

xScroll.pack(side=BOTTOM,fill=X) #Adding horizontal scrollbar to the table

yScroll.pack(side=RIGHT,fill=Y) #Adding vertical scrollbar to the table

xScroll.config(command=self.Car\_Table.xview) #Binding scrollbar to the table

yScroll.config(command=self.Car\_Table.yview)

self.Car\_Table.heading("cid",text="Customer id") #Creating heading in table for all fields

self.Car\_Table.heading("fname",text="First name")

self.Car\_Table.heading("lname",text="Last name")

self.Car\_Table.heading("g",text="Gender")

self.Car\_Table.heading("mno",text="Mobile no.")

self.Car\_Table.heading("id",text="ID type")

self.Car\_Table.heading("idno",text="ID no.")

self.Car\_Table.heading("cname",text="Car name")

self.Car\_Table.heading("model",text="Car model")

self.Car\_Table.heading("ftype",text="Fuel type")

self.Car\_Table.heading("ctype",text="Car type")

self.Car\_Table.heading("colour",text="Car colour")

self.Car\_Table.heading("price",text="Car price")

self.Car\_Table["show"]="headings"

self.Car\_Table.pack(fill=BOTH,expand=1)

self.Car\_Table.column("cid",width=100) #Fixing the width of all fields

self.Car\_Table.column("fname",width=100)

self.Car\_Table.column("lname",width=100)

self.Car\_Table.column("g",width=100)

self.Car\_Table.column("mno",width=100)

self.Car\_Table.column("id",width=100)

self.Car\_Table.column("idno",width=100)

self.Car\_Table.column("cname",width=100)

self.Car\_Table.column("model",width=100)

self.Car\_Table.column("ftype",width=100)

self.Car\_Table.column("ctype",width=100)

self.Car\_Table.column("colour",width=100)

self.Car\_Table.column("price",width=100)

self.fetch\_data() #TO show data in the table below.

self.Car\_Table.bind("<ButtonRelease-1>",self.get\_cursor)

#===============================================================================#

FrameButton=Frame(self.root,bd=12,relief=SUNKEN,padx=20,bg="black")

FrameButton.place(x=0,y=540,width=1365,height=50)

btnAddData=Button(FrameButton,command=self.add\_data,text="ADD",font=("arial",8,"bold"),width=20,bg="white",fg="black",relief=RAISED,bd=5)

btnAddData.grid(row=0,column=0,padx=50)

btnUpdateData=Button(FrameButton,command=self.update\_data,text="UPDATE",font=("arial",8,"bold"),width=20,bg="white",fg="black",relief=RAISED,bd=5)

btnUpdateData.grid(row=0,column=1,padx=50)

btnDeleteData=Button(FrameButton,command=self.delete\_data,text="DELETE",font=("arial",8,"bold"),width=20,bg="white",fg="black",relief=RAISED,bd=5)

btnDeleteData.grid(row=0,column=3,padx=50)

btnResetData=Button(FrameButton,command=self.reset\_data,text="RESET",font=("arial",8,"bold"),width=20,bg="white",fg="black",relief=RAISED,bd=5)

btnResetData.grid(row=0,column=4,padx=50)

btnExitData=Button(FrameButton,command=self.iExit,text="EXIT",font=("arial",8,"bold"),width=20,bg="white",fg="black",relief=RAISED,bd=5)

btnExitData.grid(row=0,column=5,padx=50)

#=============================================================

def add\_data(self): #Add\_Data function to save records in Library database

mydb=sql.connect(host="localhost",user="root",passwd="12345",database="car")

mycur=mydb.cursor()

mycur.execute("insert into details values(%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s,%s)",(

self.cid.get(),

self.fname.get(),

self.lname.get(),

self.g.get(),

self.mno.get(),

self.id.get(),

self.idno.get(),

self.cname.get(),

self.model.get(),

self.ftype.get(),

self.ctype.get(),

self.colour.get(),

self.cprice.get()

))

mydb.commit()

self.fetch\_data()

self.reset\_data()

messagebox.showinfo("Success","Member has been created successfully.")

mycur.close()

def update\_data(self): #Update\_Data function to update records in Library database

mydb=sql.connect(host="localhost",user="root",passwd="12345",database="car")

mycur=mydb.cursor()

mycur.execute("update details set fname=%s,lname=%s,g=%s,mno=%s,id=%s,idno=%s,cname=%s,\

model=%s,ftype=%s,ctype=%s,colour=%s,cprice=%s where cid=%s",(

self.fname.get(),

self.lname.get(),

self.g.get(),

self.mno.get(),

self.id.get(),

self.idno.get(),

self.cname.get(),

self.model.get(),

self.ftype.get(),

self.ctype.get(),

self.colour.get(),

self.cprice.get(),

self.cid.get()

))

mydb.commit()

self.fetch\_data()

self.reset\_data()

mydb.close()

messagebox.showinfo("Success","Member has been updated successfully.")

def fetch\_data(self): #Function to access all records from the bookdetails table

mydb=sql.connect(host="localhost",user="root",passwd="12345",database="car")

mycur=mydb.cursor()

mycur.execute("select \* from details")

rows=mycur.fetchall()

if len(rows)!=0: #To delete previous data from the table

self.Car\_Table.delete(\*self.Car\_Table.get\_children())

for i in rows:

self.Car\_Table.insert("",END,values=i)

mydb.commit()

mydb.close()

def get\_cursor(self,event=""): #To focus the curosr on the table

cursor\_row=self.Car\_Table.focus()

content=self.Car\_Table.item(cursor\_row)

row=content["values"]

self.cid.set(row[0]),

self.fname.set(row[1]),

self.lname.set(row[2]),

self.g.set(row[3]),

self.mno.set(row[4]),

self.id.set(row[5]),

self.idno.set(row[6]),

self.cname.set(row[7]),

self.model.set(row[8]),

self.ftype.set(row[9]),

self.ctype.set(row[10]),

self.colour.set(row[11]),

self.cprice.set(row[12]),

def reset\_data(self): #To reset values of all controls

self.cid.set(""),

self.fname.set(""),

self.lname.set(""),

self.g.set(""),

self.mno.set(""),

self.id.set(""),

self.idno.set(""),

self.cname.set(""),

self.model.set(""),

self.ctype.set(""),

self.ftype.set(""),

self.colour.set(""),

self.cprice.set("")

def iExit(self):

iExit=tkinter.messagebox.askyesno("Ford","Do you want to exit?")

if iExit>0:

self.root.destroy()

return

def delete\_data(self):

if self.cid.get()=="" or self.fname.get()=="":

messagebox.showerror("Error!!!","First select the Member.")

else:

mydb=sql.connect(host="localhost",user="root",passwd="12345",database="car")

mycur=mydb.cursor()

query="delete from details where cid=%s"

value=(self.cid.get(),)

mycur.execute(query,value)

mydb.commit()

self.fetch\_data()

self.reset\_data()

mydb.close()

messagebox.showinfo("Success","Member has been deleted successfully.")

#======================================================

if \_\_name\_\_=="\_\_main\_\_": #Infinite loop to run the program

SplashScreen()

root=Tk()

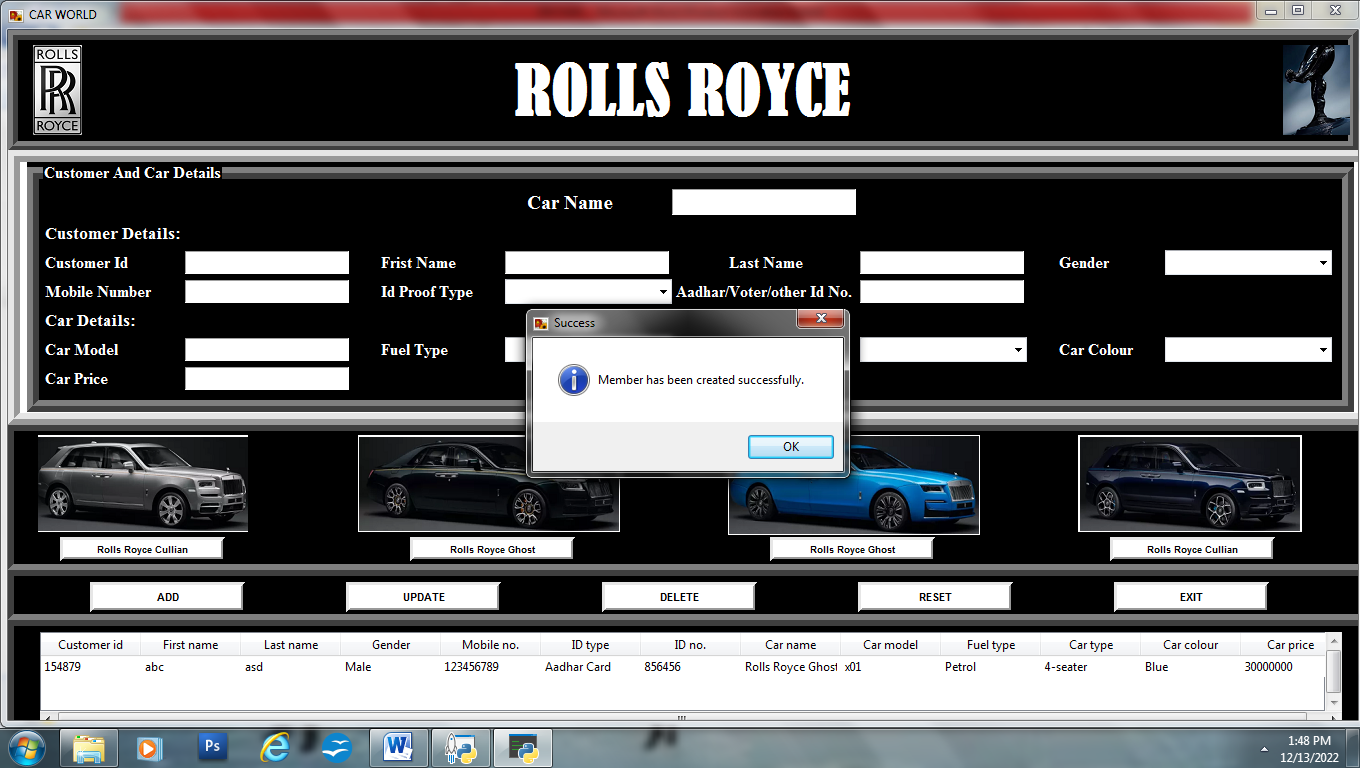
obj=CarManagementSyste(root)

root.mainloop()

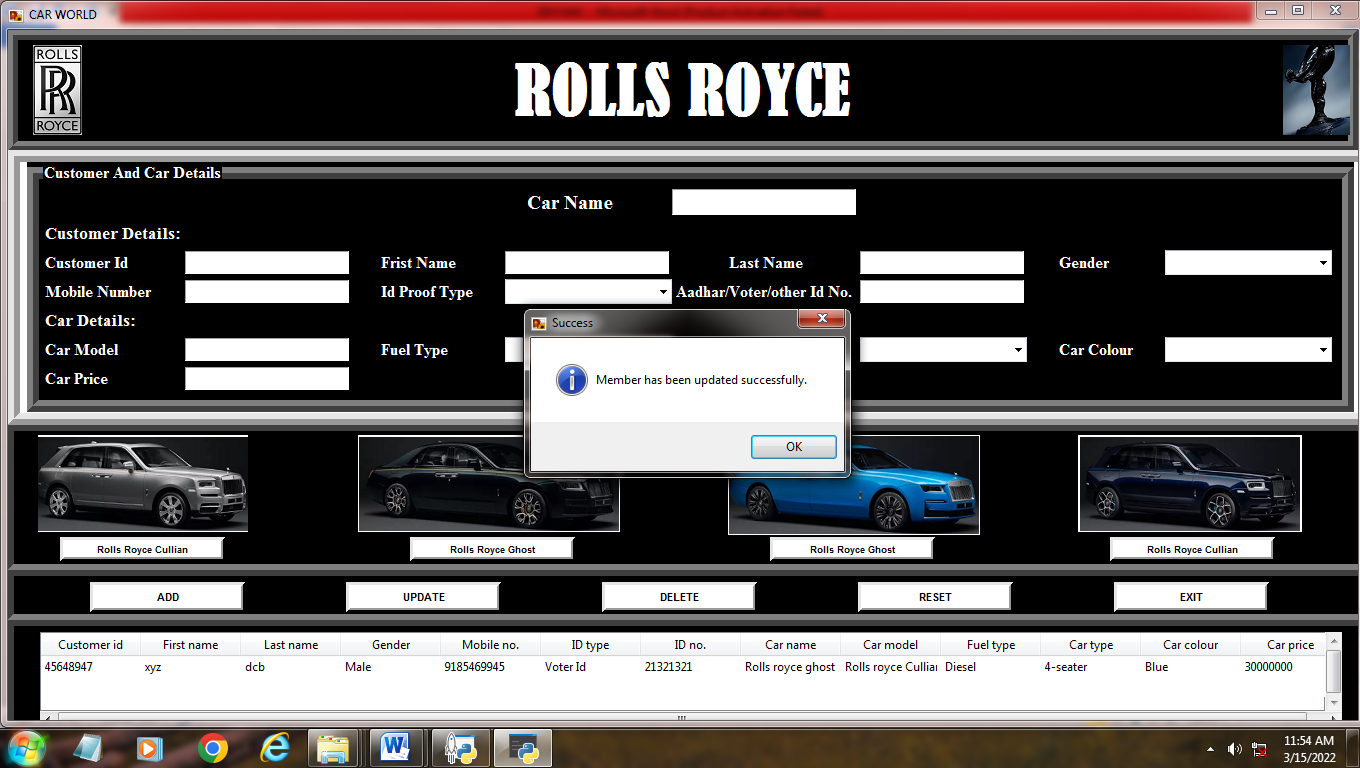
INPUT AND OUTPUT INTERFERENCE



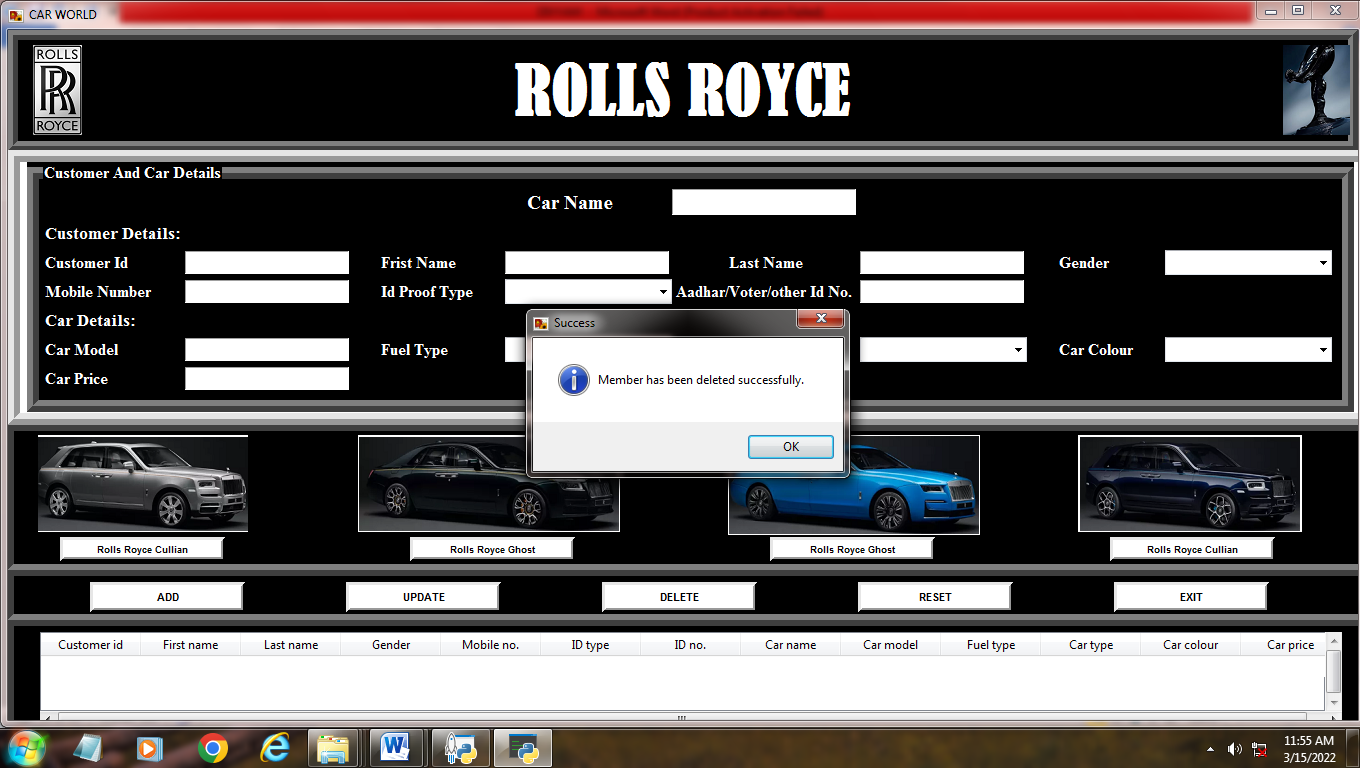
#Add



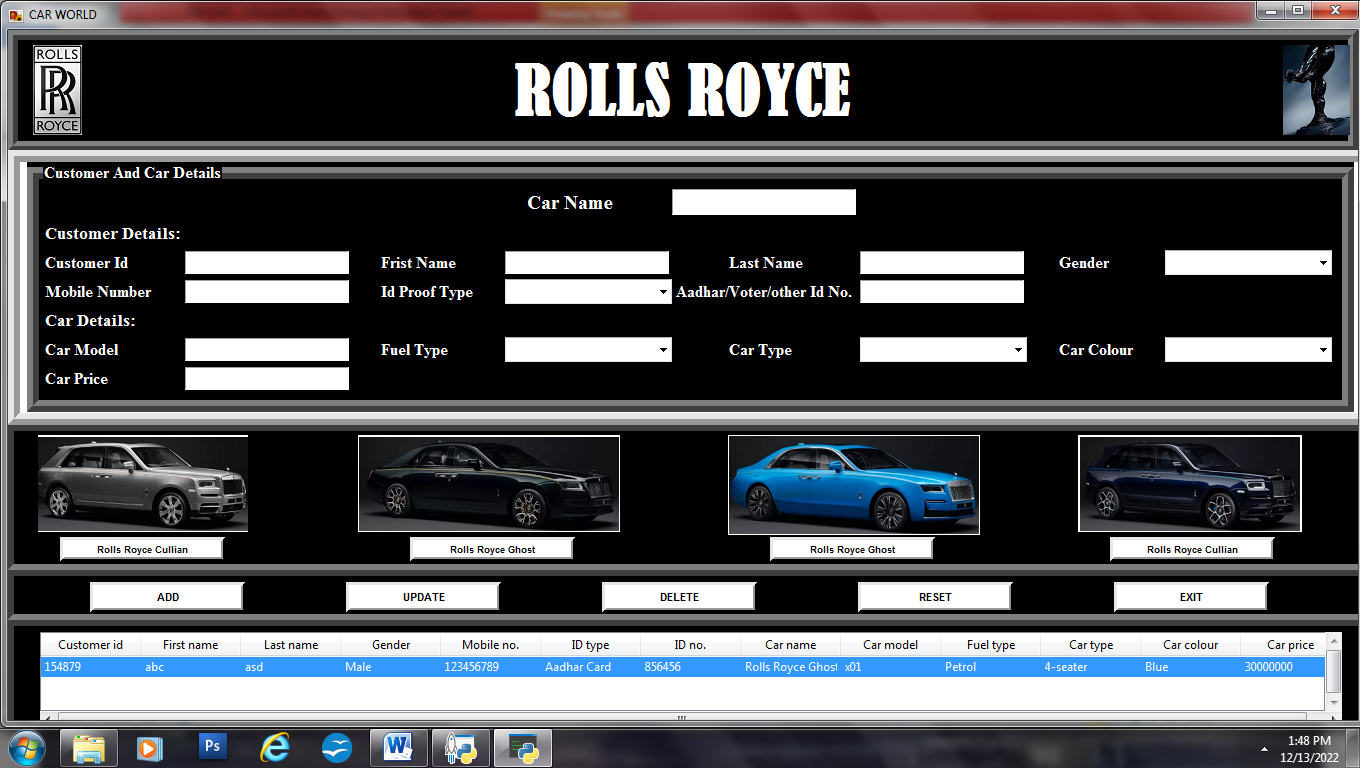
#Update



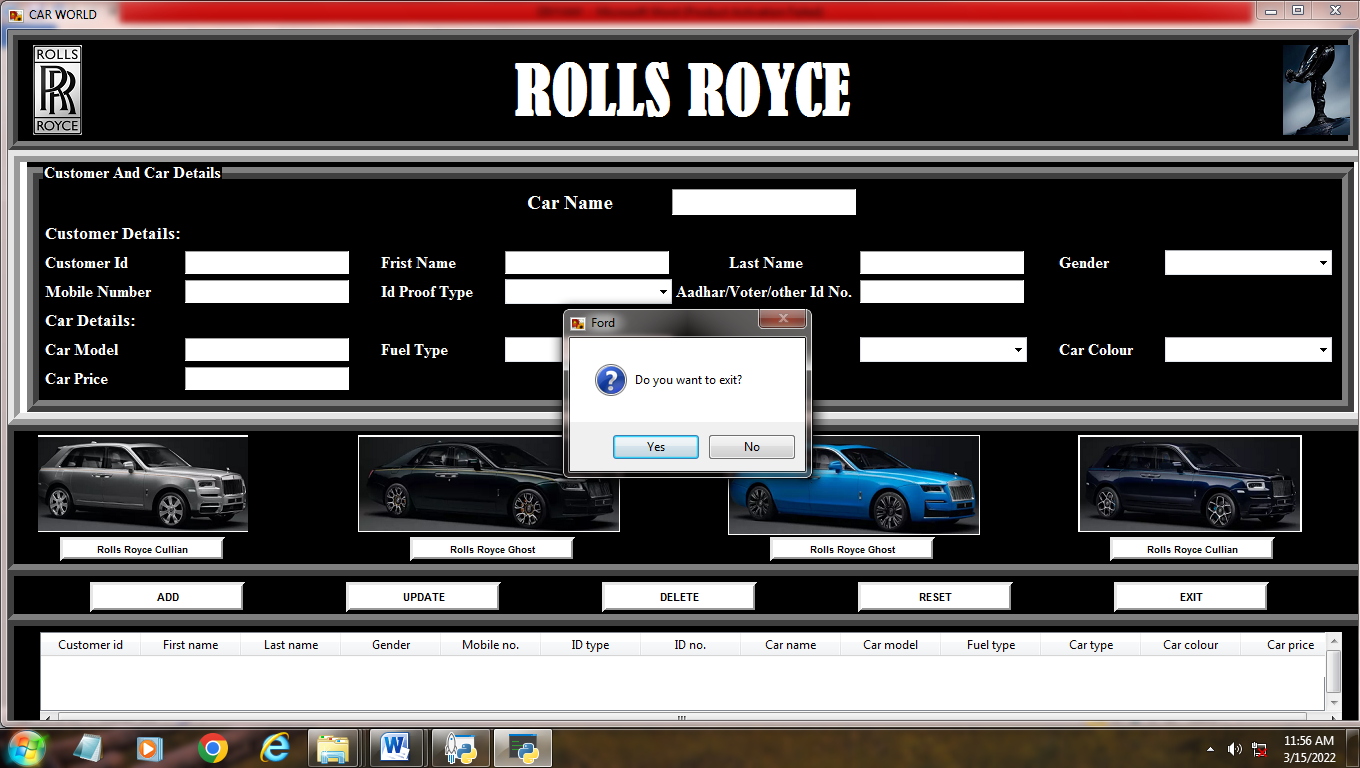
##Delete



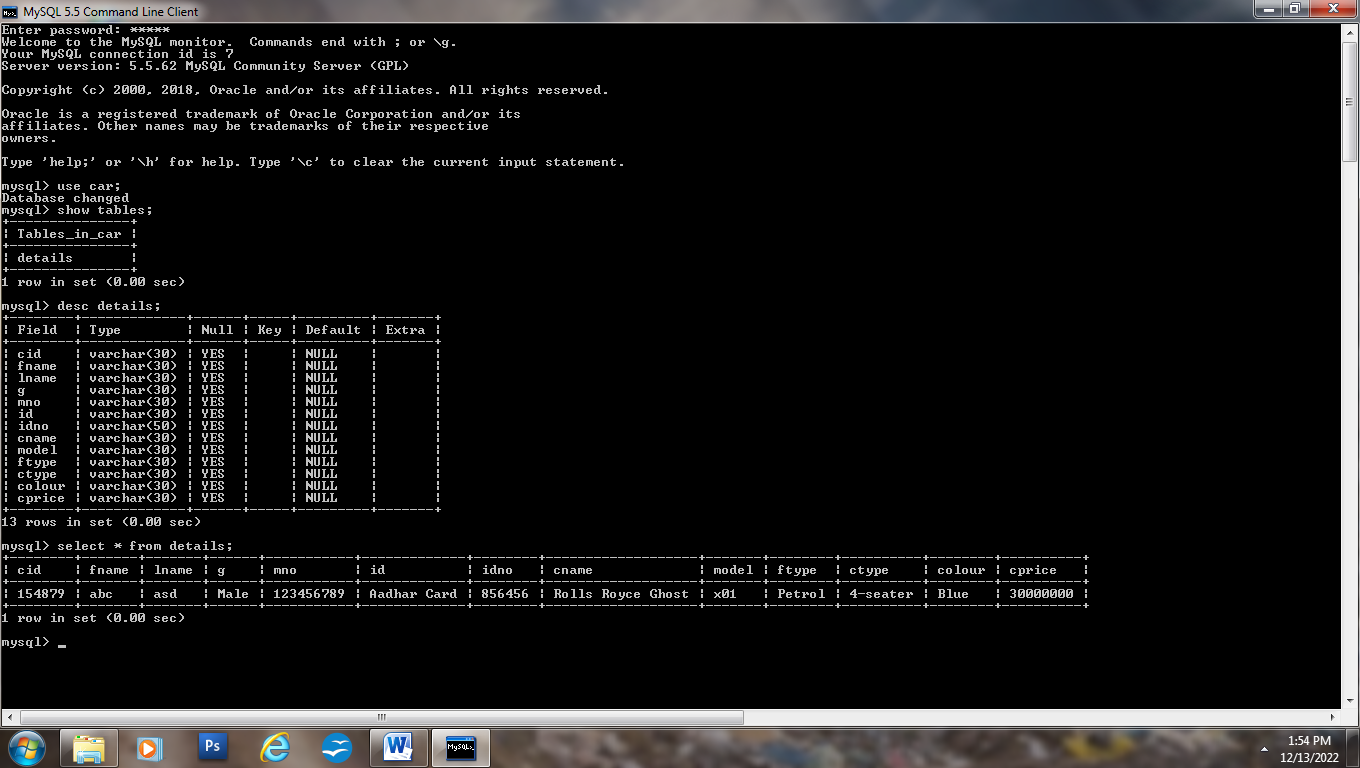
#Reset data

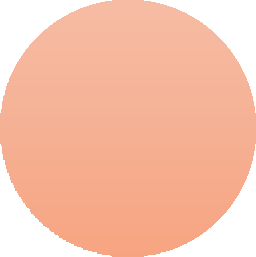


#Exit



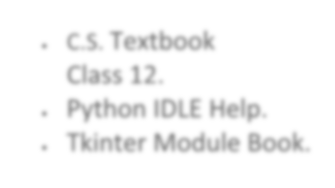
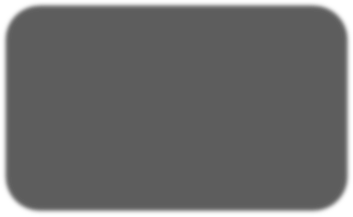
Data structure:





Bibliography





* C.S. Textbook Class 12.
* Python IDLE Help.
* Tkinter Module Book.



