**“BUGATTI CAR SHOWROOM”**

**A COMPUTER SCIENCE PROJECT REPORT**

**SUBMITTED BY**

**AYUSH GANGANI**

**IN PARTIAL FULFILMENT OF THE**

**AISSCE – 2022-23**

**IN**

**COMPUTER SCIENCE (083)**

**AT**



**J.B. DIAMONDS & 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 Ayush Gangani** is a student of J. B. Diamonds & KARP Impex Vidya Sankul, who has successfully completed the project work on title **BUGATTI 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.

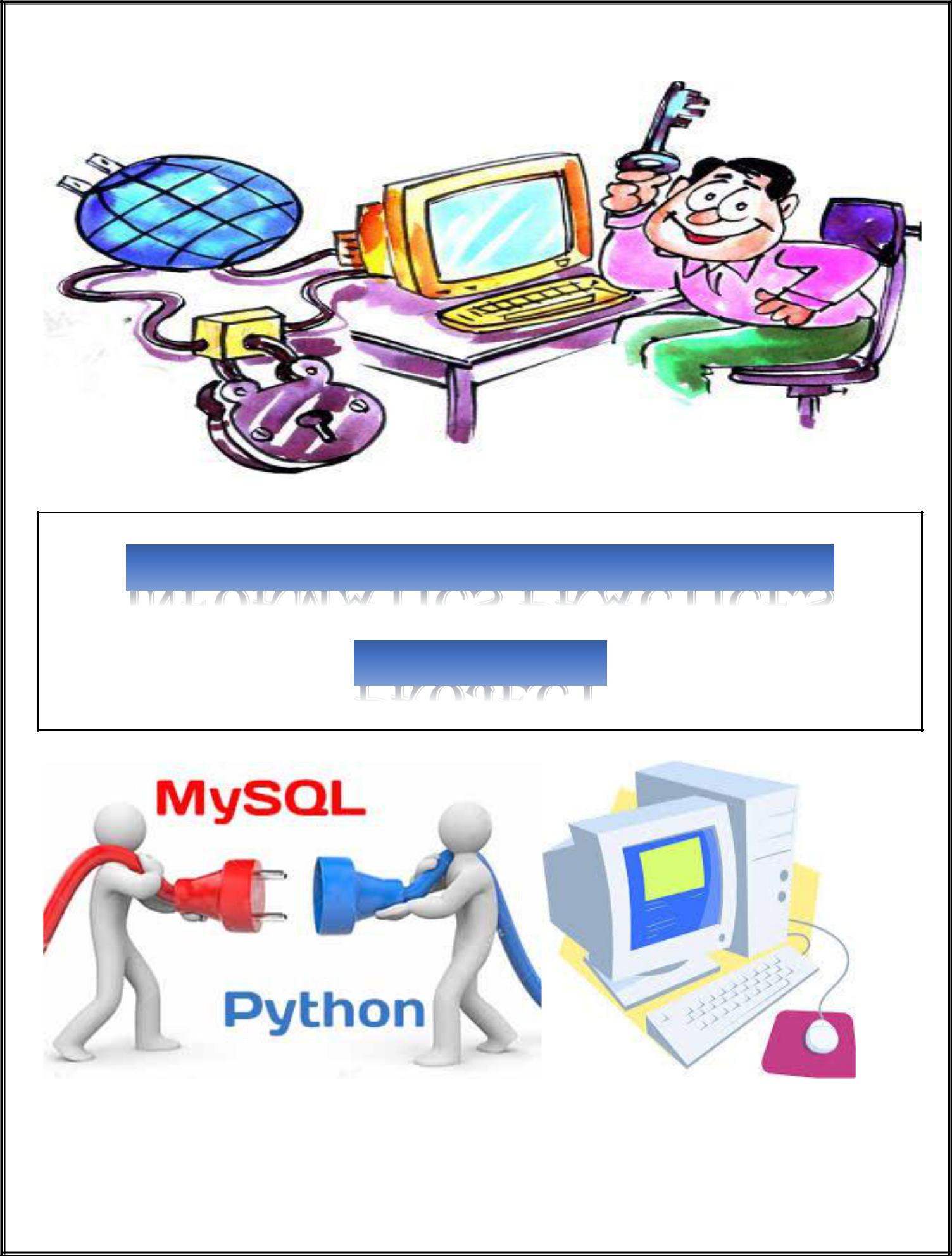
We wish all the success for his future endeavors.

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

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

**………………………………………….**

**Principle 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 **“BUGATTI 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.

****

**Bugatti**

**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.

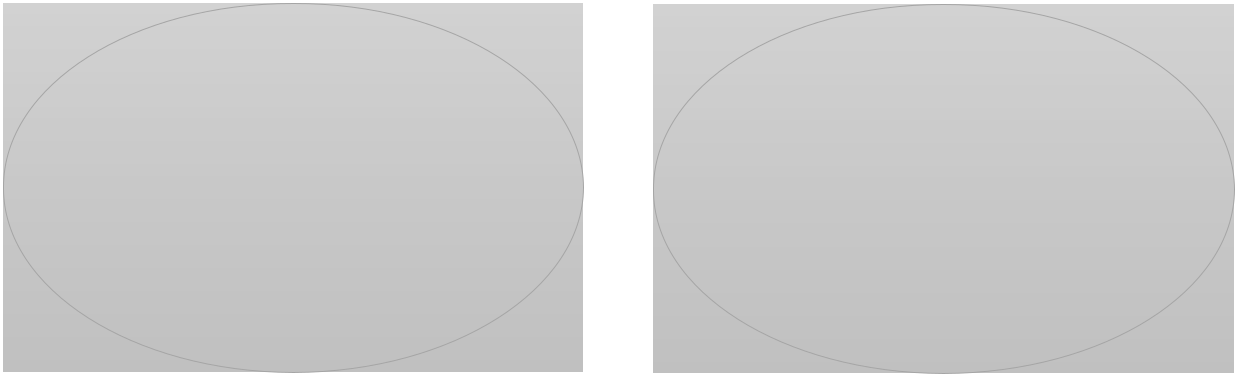
**Step 4** –Create a cursor instance.

**Step 5** –Execute a query.

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

**Step 7** –Clean up the environment.





**Source Code:**

# Login Form:

#LOGIN FORM

from tkinter import\*

import tkinter.messagebox # for messagebox

import os # for stringvariable

from tkinter import ttk # for combobox

import random # for reference

import time

import datetime

def main():

root = Tk()

app = Window\_1(root)

class Window\_1:

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

self.master = master

self.master.title("Login Window")

self.master.geometry('750x550')

self.master.config(bg = 'maroon')

self.Frame = Frame(self.master, bg = 'maroon')

self.Frame.pack()

self.Username = StringVar() # x = StringVar() Holds a string; default value is " "

self.Password = StringVar()

self.Lbl\_Title = Label(self.Frame, text = 'Login Form', font = ('algerian',55,'underline'), bg = 'maroon', fg = 'red')

self.Lbl\_Title.grid(row = 0, column = 0, columnspan =3, pady = 40)

self.Login\_Frame\_1 = LabelFrame(self.Frame, width = 1350, height = 150, relief = 'ridge', bg = 'sky blue', bd = 13, text='Login', fg = 'darkblue',

font = ('lucida calligraphy',25,'bold'))

self.Login\_Frame\_1.grid(row = 1, column =0)

self.Login\_Frame\_2 = LabelFrame(self.Frame, width = 1000, height = 150, relief = 'ridge',bg = 'Orange', bd = 15, text='Events', fg = 'darkblue',

font = ('lucida calligraphy',25,'bold'))

self.Login\_Frame\_2.grid(row = 2, column = 0)

#===================================================LABEL and ENTRIES=======================================================================

self.Label\_Username = Label(self.Login\_Frame\_1, text = 'Username', font = ('lucida handwriting',20,'bold'), bg = 'sky blue', fg = 'black', bd = 20)

self.Label\_Username.grid(row = 0, column = 0)

self.text\_Username = Entry(self.Login\_Frame\_1, font = ('lucida handwriting',20,'bold'), fg="red",textvariable = self.Username)

self.text\_Username.grid(row = 0, column = 1, padx = 50)

self.text\_Username.focus()

self.Label\_Password = Label(self.Login\_Frame\_1, text = 'Password', font = ('lucida handwriting',20,'bold'), bg = 'sky blue', fg = 'black', bd = 20)

self.Label\_Password.grid(row = 1, column = 0)

self.text\_Password = Entry(self.Login\_Frame\_1, font = ('lucida handwriting',20,'bold'), show = '~', fg="green", textvariable = self.Password)

self.text\_Password.grid(row = 1, column = 1)

#=============================================================BUTTONS=======================================================================

self.btnLogin = Button(self.Login\_Frame\_2, text = 'Login', fg = 'green', width = 10, font = ('lucida calligraphy',19,'bold'), command = self.Login)

self.btnLogin.grid(row = 3, column = 0, padx = 8, pady = 20)

self.btnReset = Button(self.Login\_Frame\_2, text = 'Reset', fg = 'black', width = 10, font = ('lucida calligraphy',19,'bold'), command = self.Reset)

self.btnReset.grid(row = 3, column = 1, padx = 8, pady = 20)

self.btnExit = Button(self.Login\_Frame\_2, text = 'Exit', fg = 'red', width = 10, font = ('lucida calligraphy',19,'bold'), command = self.Exit)

self.btnExit.grid(row = 3, column = 2, padx = 8, pady = 20)

#======================================================Code for Login Button==================================================================

def Login(self):

u = (self.Username.get())

p = (self.Password.get())

if (u == str('root') and p == str(12345)):

tkinter.messagebox.askyesno("Login Successfully","Thanks : For using Login Form.")

self.master.destroy()

self.\_\_library\_\_()

else:

tkinter.messagebox.askyesno("Login","Error : Wrong Password")

self.Username.set("")

self.Password.set("")

self.text\_Username.focus()

#======================================================Code for Reset Button==================================================================

def Reset(self):

self.Username.set("")

self.Password.set("")

self.text\_Username.focus()

#======================================================Code for Exit Button==================================================================

def Exit(self):

self.Exit = tkinter.messagebox.askokcancel("Login System", "Confirm if you want to Exit")

if self.Exit > 0:

self.master.destroy()

def \_\_library\_\_(self):

filename = 'Ayush.py'

os.system(filename)

os.system('notepad'+filename)

if \_\_name\_\_ == '\_\_main\_\_': # https://micropyramid.com/blog/understand-self-and-\_\_init\_\_-method-in-python-class/

main()

#Main Page

import tkinter

from tkinter import \*

from tkinter import ttk

from tkinter import ttk

from tkinter import messagebox

import mysql.connector as sql

from Show import \*

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='black',bd=10,relief=SUNKEN)

Label(splashscreen, text='BUGATTI', font='Algerian 35', fg='sky blue', bg='black',bd=10,relief=RAISED).pack()

#Add image

image1 = PhotoImage(file="1001.png")

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

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

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

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

Label(splashscreen, text="12th Science-B", font='Algerian 13', bg='black', fg='sky blue',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='sky blue',fg='black')#, 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="3905")#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 cardetails(cid varchar(30), name varchar(30), lname varchar(30), \

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

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

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

'''

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

"("

"cid varchar(12) primary key,"

"name char(50),"

"mno varchar(10),")

'''

class BugattiCarShowroom:

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

self.root=root

'''self.root.iconbitmap("00.png")'''

self.root.title("CAR WORLD")

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

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

self.cid=StringVar()

self.name=StringVar()

self.lname=StringVar()

self.mno=StringVar()

self.id=StringVar()

self.pno=StringVar()

self.cname=StringVar()

self.model=StringVar()

self.ftype=StringVar()

self.ctype=StringVar()

self.colour=StringVar()

self.price=StringVar()

lblTitle=Label(self.root,text="BUGATTI",bg="sky blue",fg="black",bd=10,relief=RIDGE,\

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

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

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

frame.place(x=0,y=123,width=1060,height=290)

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

DataFrameLeft.place(x=-13,y=7,width=1020,height=250)

lbl2=Label(DataFrameLeft,bg="sky blue",fg="black",text="Car Name",font=("times new roman",14,"bold"),padx=20,pady=3)

lbl2.grid(row=0,column=2,sticky=W)

txtPRN\_No=Entry(DataFrameLeft, textvariable=self.cname,font=("times new roman",14),width=18)

txtPRN\_No.grid(row=0,column=3,sticky=W)

lbl1=Label(DataFrameLeft,bg="sky blue",fg="black",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="sky blue",fg="black",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="sky blue",fg="black",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.name,font=("times new roman",12),width=20)

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

lbllname=Label(DataFrameLeft,bg="sky blue",fg="black",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)

lblp=Label(DataFrameLeft,bg="sky blue",fg="black",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="sky blue",fg="black",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="sky blue",fg="black",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.pno,font=("times new roman",12),width=20)

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

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

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

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

cmbcolour["value"]=("Black/Orange","White","Dark Blue",'Yellow')

cmbcolour.current(0)

cmbcolour.grid(row=6,column=3,sticky=W)

lblprice=Label(DataFrameLeft,bg="sky blue",fg="black",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.price,font=("times new roman",12),width=20)

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

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

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

fram.place(x=1062,y=123,width=305,height=580)

btnAddData1=Button(fram,text="Bugatti Cihron",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnAddData1.place(x=70,y=105)

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

self.lbb.place(x=1080,y=140, width=270, height=97)

self.ig=PhotoImage(file='01.png')

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

btnAddData13=Button(fram,text="Bugatti W16 Mistral",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnAddData13.place(x=70,y=385)

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

self.lbb.place(x=1080,y=275, width=270, height=97)

self.ig1=PhotoImage(file='04.png')

self.lbb.config(image=self.ig1)

btnAddData11=Button(fram,text="Bugatti One Off",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnAddData11.place(x=70,y=238)

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

self.lbb.place(x=1080,y=415, width=270, height=97)

self.ig2=PhotoImage(file='03.png')

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

btnAddData12=Button(fram,text="Bugatti Veyron",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnAddData12.place(x=70,y=525)

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

self.lbb.place(x=1080,y=565, width=270, height=97)

self.ig3=PhotoImage(file='02.png')

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

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

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

FrameDetails.place(x=0,y=473,width=1060,height=230)

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

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

self.Car\_Table=ttk.Treeview(FrameDetails,column=("cid","name","lname","mno","id","pno","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("name",text="Frist Name")

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

self.Car\_Table.heading("mno",text="Mobile Number")

self.Car\_Table.heading("id",text="Proof Type")

self.Car\_Table.heading("pno",text="Id Number")

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("name",width=100)

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

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

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

self.Car\_Table.column("pno",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=RIDGE,padx=20,bg="black")

FrameButton.place(x=0,y=416,width=1060,height=55)

btnAddData=Button(FrameButton,command=self.add\_data,text="ADD",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnAddData.grid(row=0,column=0,padx=22)

btnShowData=Button(FrameButton,command=Display,text="SHOW",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnShowData.grid(row=0,column=1,padx=22)

btnUpdateData=Button(FrameButton,command=self.update\_data,text="UPDATE",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnUpdateData.grid(row=0,column=2,padx=22)

btnDeleteData=Button(FrameButton,command=self.delete\_data,text="DELETE",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnDeleteData.grid(row=0,column=3,padx=22)

btnResetData=Button(FrameButton,command=self.reset\_data,text="RESET",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnResetData.grid(row=0,column=4,padx=22)

btnExitData=Button(FrameButton,command=self.iExit,text="EXIT",font=("arial",8,"bold"),width=15,bg="sky blue",fg="black",relief=RAISED,bd=5)

btnExitData.grid(row=0,column=5,padx=22)

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

mydb=sql.connect(host="localhost",user="root",passwd="3905",database="car")

mycur=mydb.cursor()

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

self.cid.get(),

self.name.get(),

self.lname.get(),

self.mno.get(),

self.id.get(),

self.pno.get(),

self.cname.get(),

self.model.get(),

self.ftype.get(),

self.ctype.get(),

self.colour.get(),

self.price.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="3905",database="car")

mycur=mydb.cursor()

mycur.execute("update cardetails set name=%s,lname=%s,mno=%s,id=%s,pno=%s,cname=%s,\

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

self.name.get(),

self.lname.get(),

self.mno.get(),

self.id.get(),

self.pno.get(),

self.cname.get(),

self.model.get(),

self.ftype.get(),

self.ctype.get(),

self.colour.get(),

self.price.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="3905",database="car")

mycur=mydb.cursor()

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

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]), #To show the values from table to their respective field in the second frame.

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

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

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

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

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

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

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

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

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

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

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

def show\_data(self): #To show data in right-side List Box

self.txtBox.insert(END,"Customer Id Type:\t\t"+self.cid.get()+"\n")

self.txtBox.insert(END,"Frist Name:\t\t"+self.name.get()+"\n")

self.txtBox.insert(END,"Last Name:\t\t"+self.lname.get()+"\n")

self.txtBox.insert(END,"Mobile No.:\t\t"+self.mno.get()+"\n")

self.txtBox.insert(END,"Id Type:\t\t"+self.id.get()+"\n")

self.txtBox.insert(END,"Id No.:\t\t"+self.pno.get()+"\n")

self.txtBox.insert(END,"Car Name:\t\t"+self.cname.get()+"\n")

self.txtBox.insert(END,"Car Model.:\t\t"+self.model.get()+"\n")

self.txtBox.insert(END,"Fuel Type:\t\t"+self.ftype.get()+"\n")

self.txtBox.insert(END,"Car Type:\t\t"+self.ctype.get()+"\n")

self.txtBox.insert(END,"Car Colour:\t\t"+self.colour.get()+"\n")

self.txtBox.insert(END,"Car Price:\t\t"+self.price.get()+"\n")

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

self.cid.set(""),

self.name.set(""),

self.lname.set(""),

self.mno.set(""),

self.id.set(""),

self.pno.set(""),

self.cname.set(""),

self.model.set(""),

self.ftype.set(""),

self.ctype.set(""),

self.colour.set(""),

self.price.set(""),

def iExit(self):

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

if iExit>0:

self.root.destroy()

return

def delete\_data(self):

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

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

else:

mydb=sql.connect(host="localhost",user="root",passwd="

,database="car")

mycur=mydb.cursor()

query="delete from cardetails 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=BugattiCarShowroom(root)

root.mainloop()

from tkinter import \*

#from PIL import ImageTk,Image

from tkinter import messagebox

#import pymysql

import mysql.connector as sql

# Add your own database name and password here to reflect in the code

db="car"

con = sql.connect(host="localhost",user="root",password='3905',database=db)

def Display():

root = Tk()

root.title("car")

root.minsize(width=600,height=500)

root.geometry("600x500")

Canvas1 = Canvas(root)

Canvas1.config(bg="sky blue")

Canvas1.pack(expand=True,fill=BOTH)

headingFrame1 = Frame(root,bg="white",bd=5)

headingFrame1.place(relx=0.25,rely=0.1,relwidth=0.5,relheight=0.13)

headingLabel = Label(headingFrame1, text="Car Details", bg='black', fg='white', font=('Courier',15))

headingLabel.place(relx=0,rely=0, relwidth=1, relheight=1)

labelFrame = Frame(root,bg='black')

labelFrame.place(relx=0.1,rely=0.3,relwidth=0.8,relheight=0.6)

y = 0.25

Label(labelFrame, text="%-10s%-20s%-30s%-30s%-30s%-20s%-20s%-20s%-20s%-20s%-20s%-10s"%('C-id','name','L-Name','ID','IdNo','phno','C-Model','C-Name','Colour','C-type','F-type','C-price'),bg='black',fg='white').place(relx=0.07,rely=0.1)

Label(labelFrame, text="---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------",bg='black',fg='white').place(relx=0.07,rely=0.2)

cur = con.cursor()

getBooks = "select \* from cardetails"

try:

cur.execute(getBooks)

data=cur.fetchall()

con.commit()

for i in data:

Label(labelFrame, text="%-10s%-20s%-20s%-20s%-20s%-20s%-20s%-20s%-20s%-20s%-20s%-10s"%(i[0],i[1],i[2],i[3],i[4],i[5],i[6],i[7],i[8],i[9],i[10],i[11]),bg='black',fg='white').place(relx=0.07,rely=y)

y += 0.1

except:

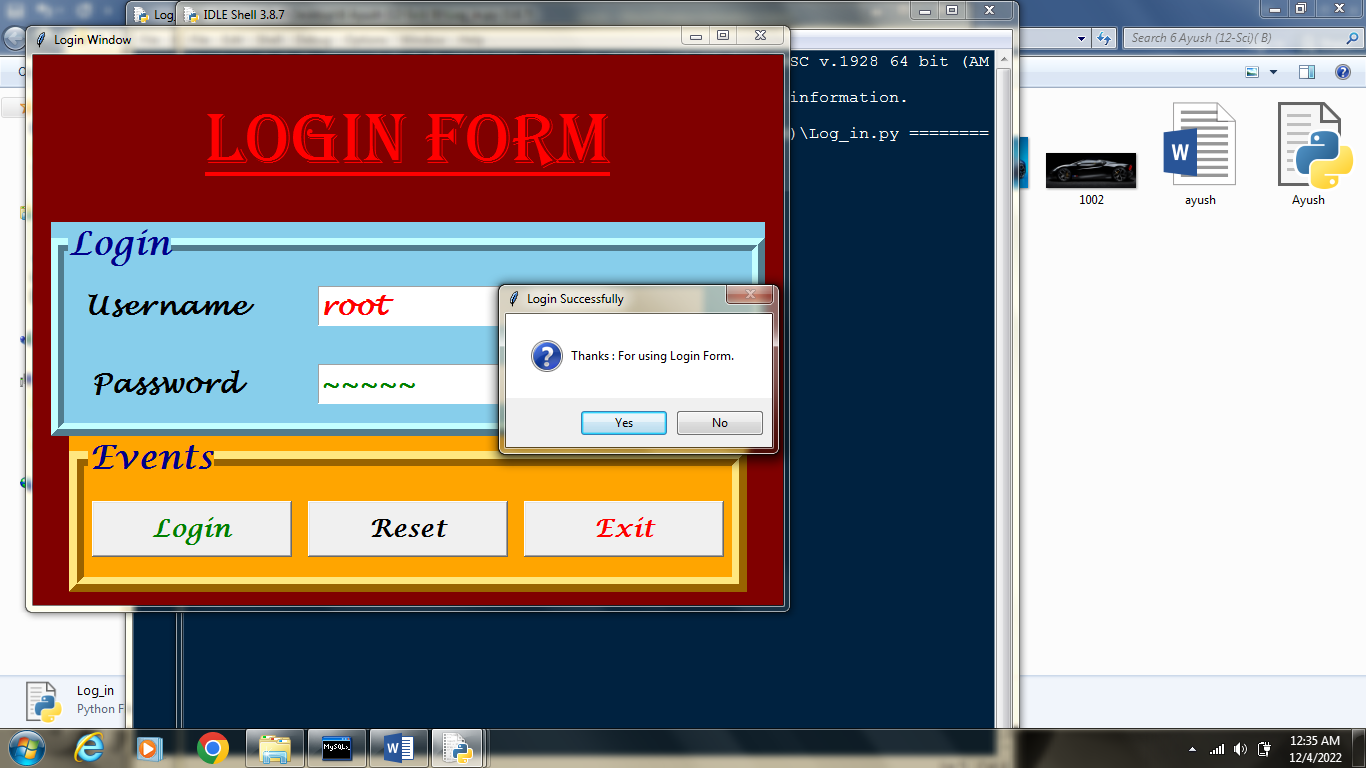
messagebox.showinfo("Failed to fetch files from database")

quitBtn = Button(root,text="Quit",bg='#f7f1e3', fg='black', command=root.destroy)

quitBtn.place(relx=0.4,rely=0.9, relwidth=0.18,relheight=0.08)

root.mainloop()

#LOGIN FORM



#Splashscreen

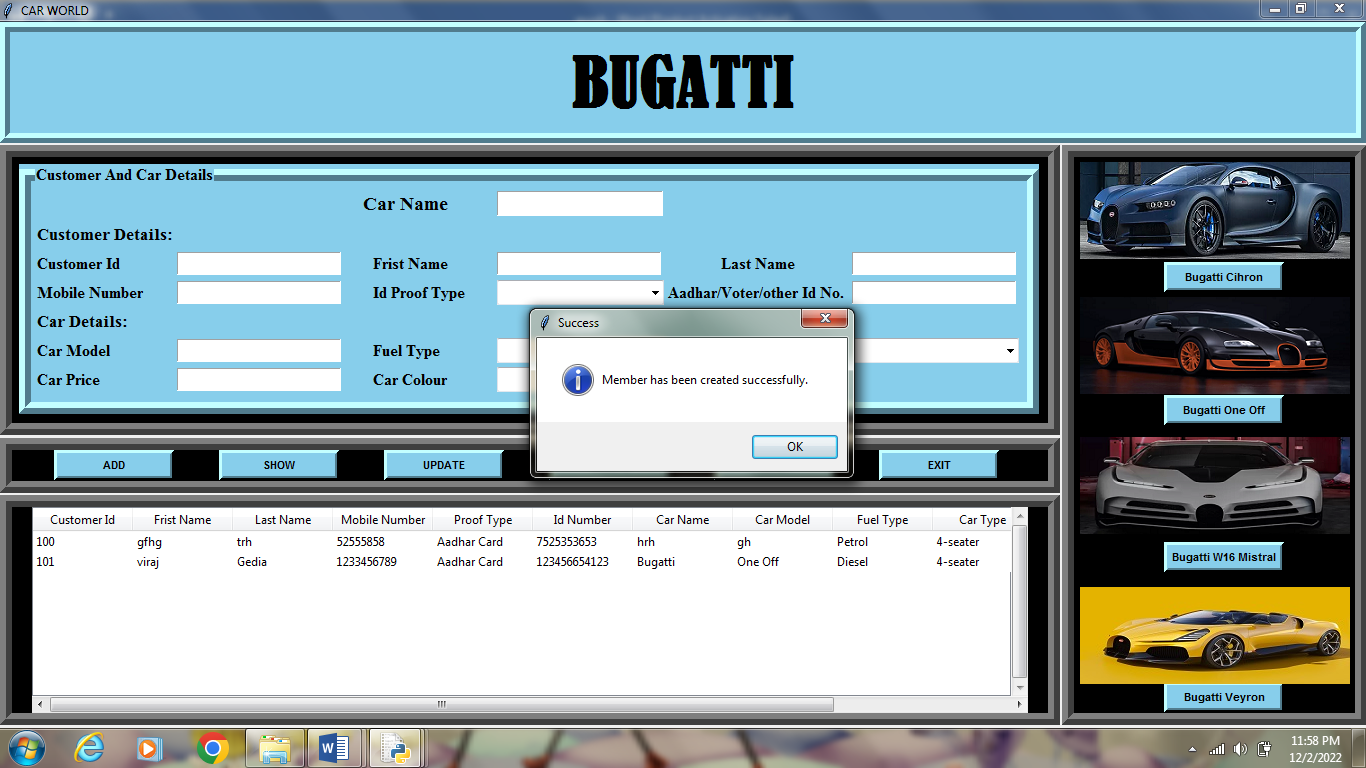


#Main Page

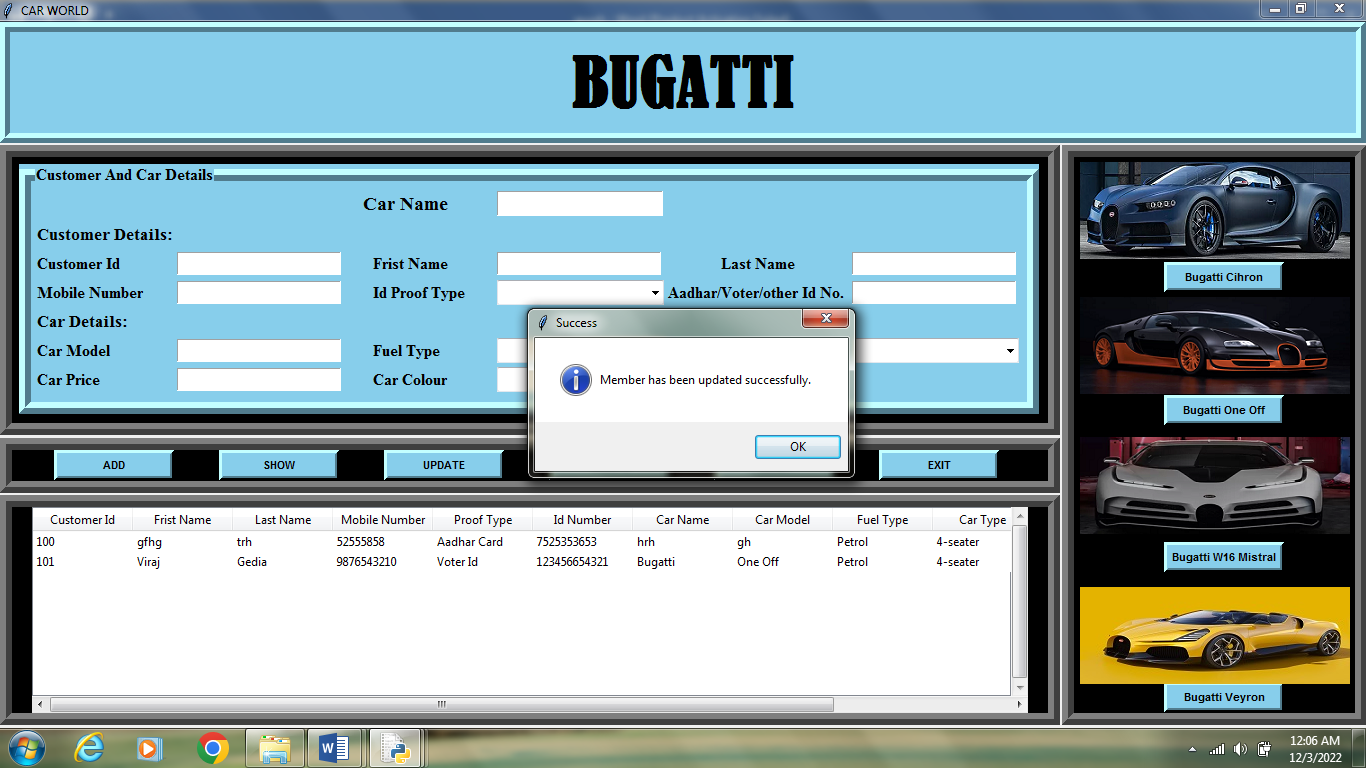


#Add Member





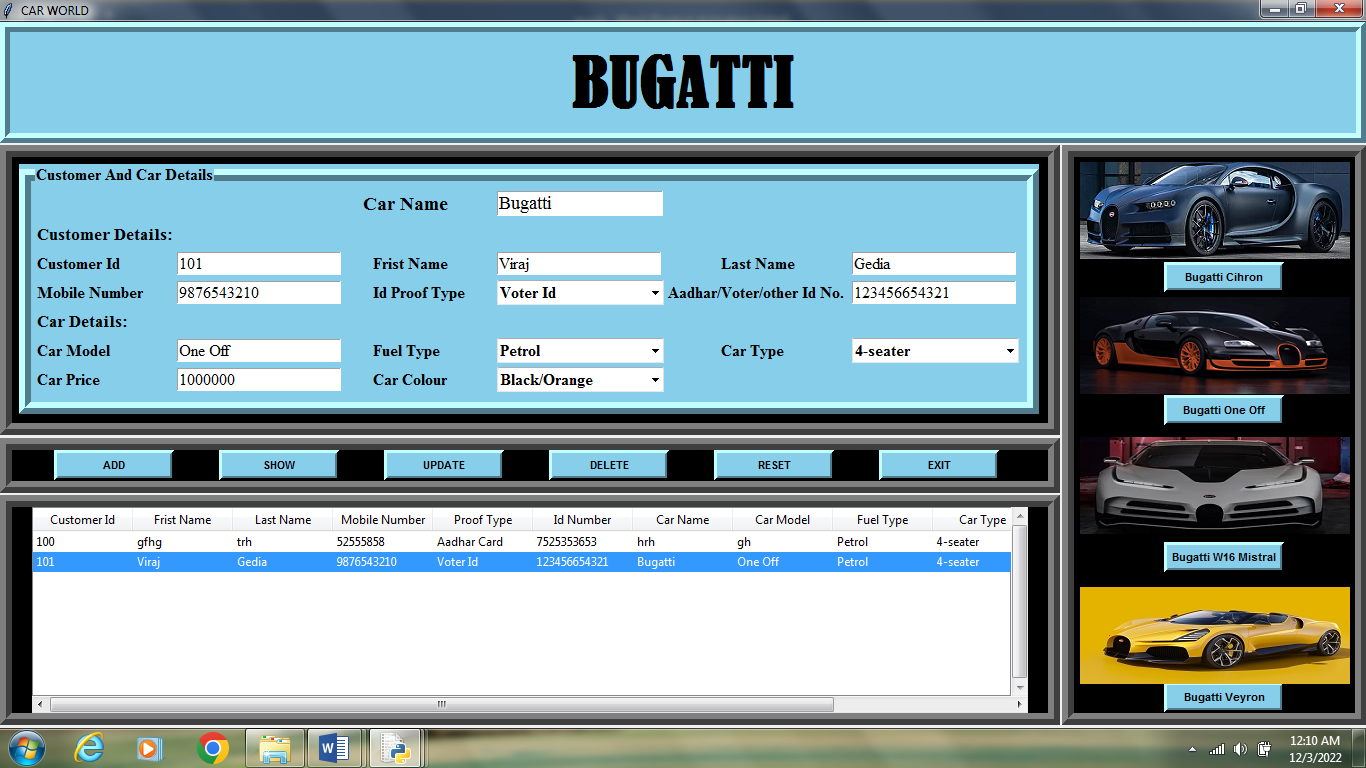
#Update Member

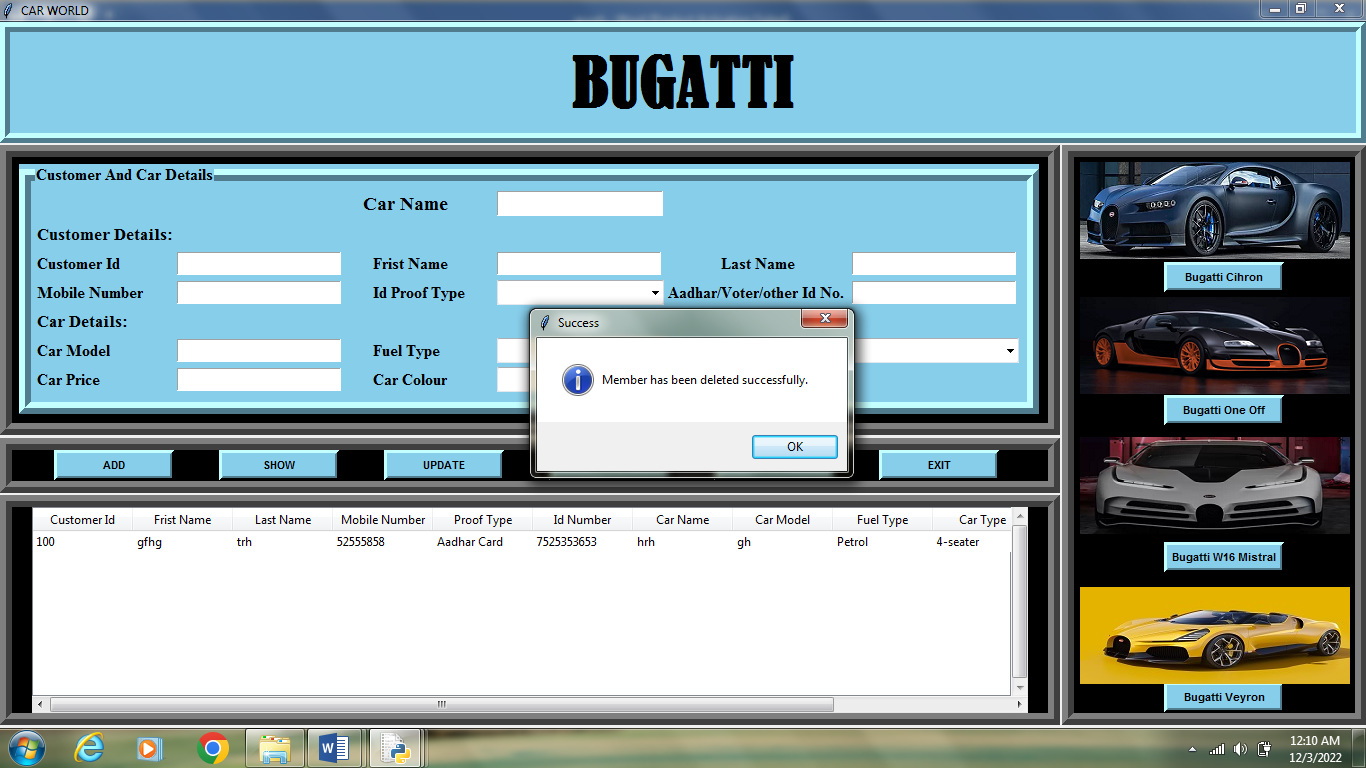


# Reset Data



# Delete Data

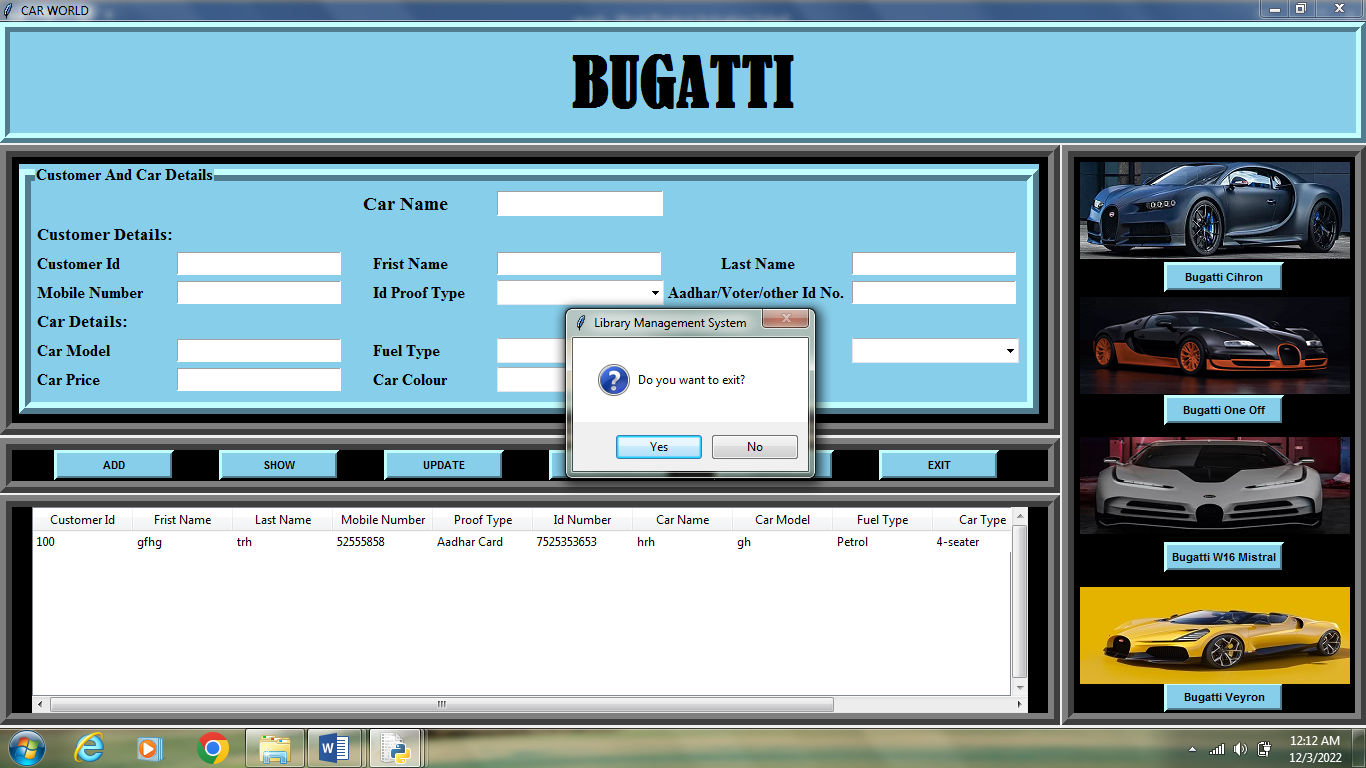


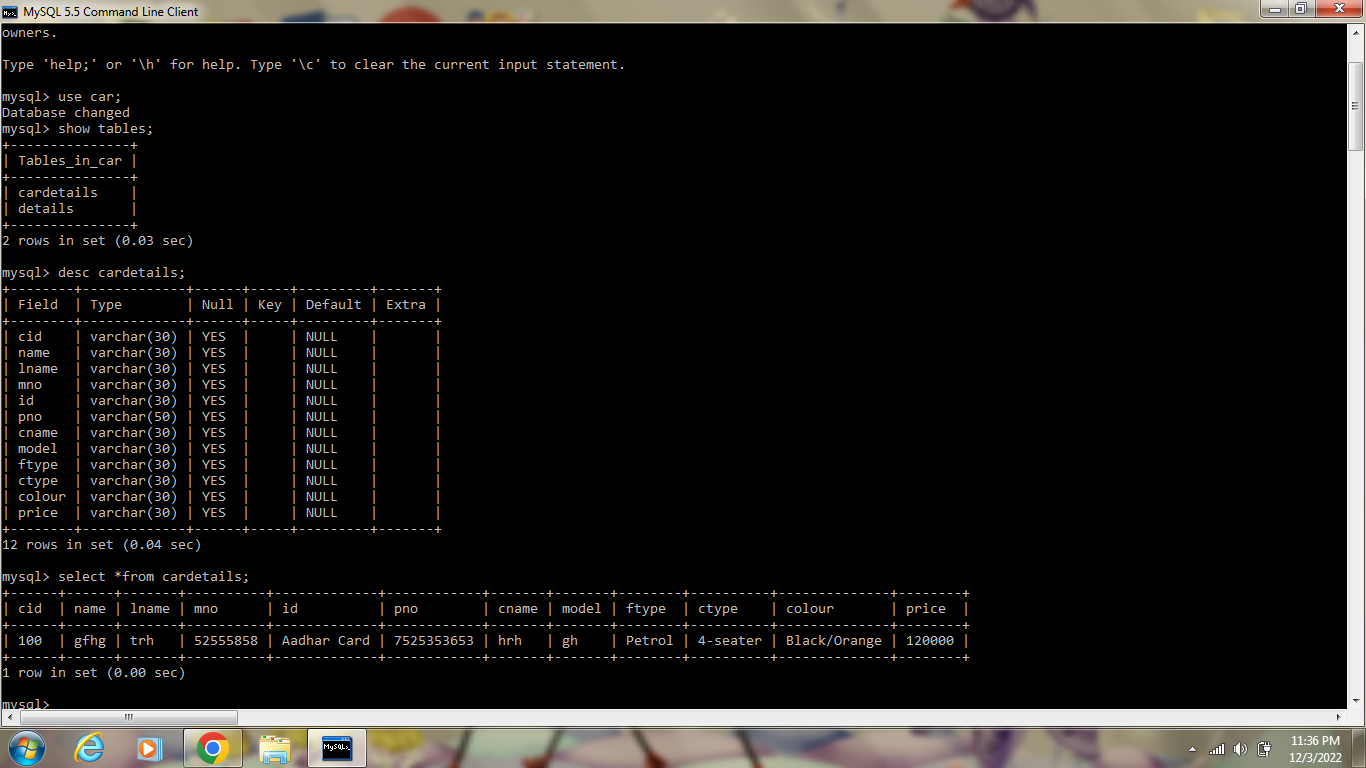


#Show



#Exit







* C.S. Textbook

Class 12.

* Python IDLE Help.
* Tkinter Module Book.



**THANK YOU**

