**Customer Management System**

It is a **Team Project** that I have created with my classmates and I was leading this project and my work is to give the **design** and **development** process and we all teammates contributed in this **Customer Management System.**

We created Customer Management System using **OOPs** as well as **Tkinter** in which a person can **add**, **search**, **modify**, **delete** and show customer details.

In this, **temporary** **database** also created for accessing all the data that have been entered in this management system.

This system can **save** a lot of **time** of managing data manually into **managing data digitally**.

There are 2 files created for this project as 1st is for all **backend** and whole **code** etc., and 2nd is for **GUI** which user is experiencing.

Both files mentioned below and with all the outputs:

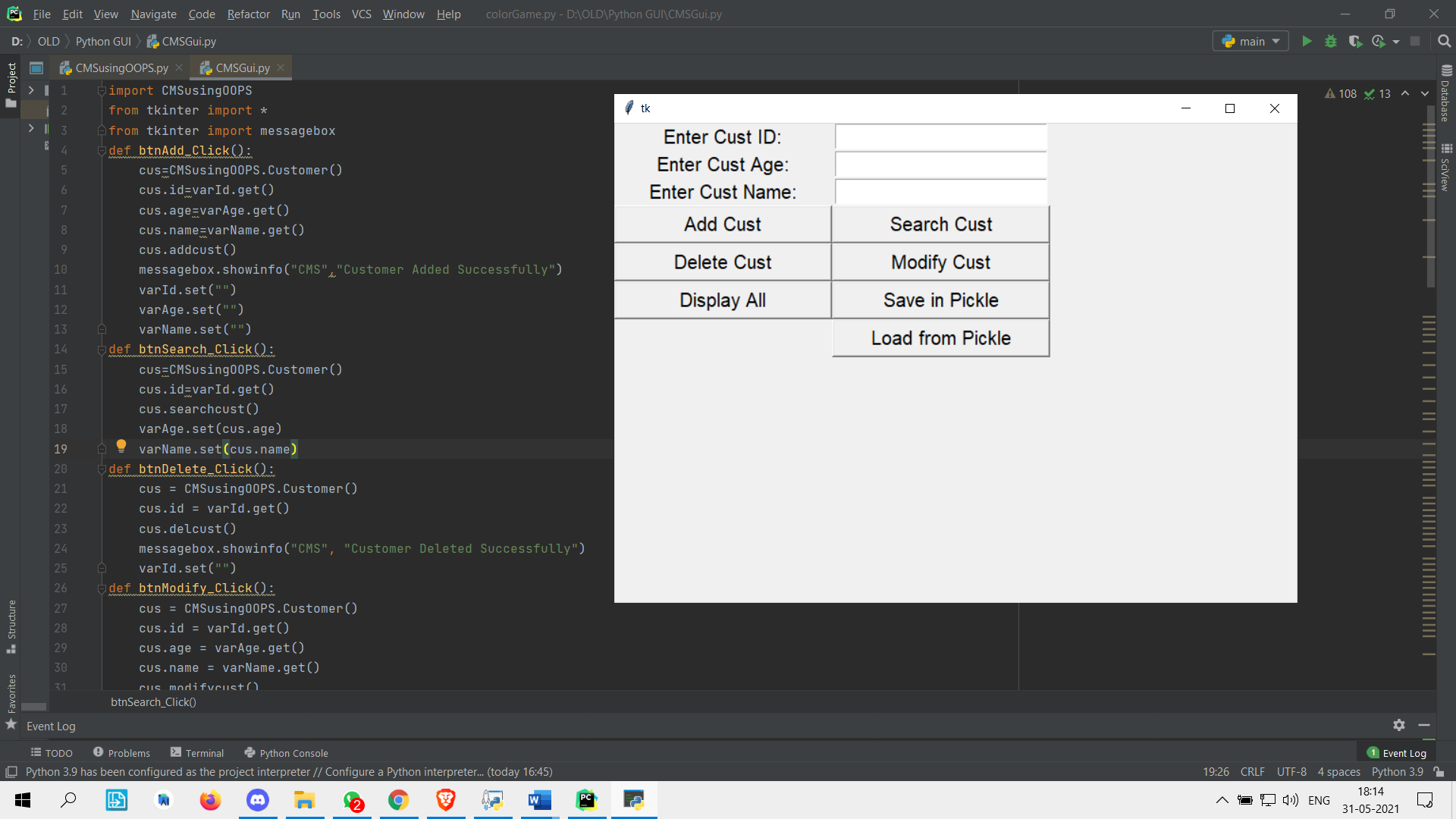
**Main Code:**

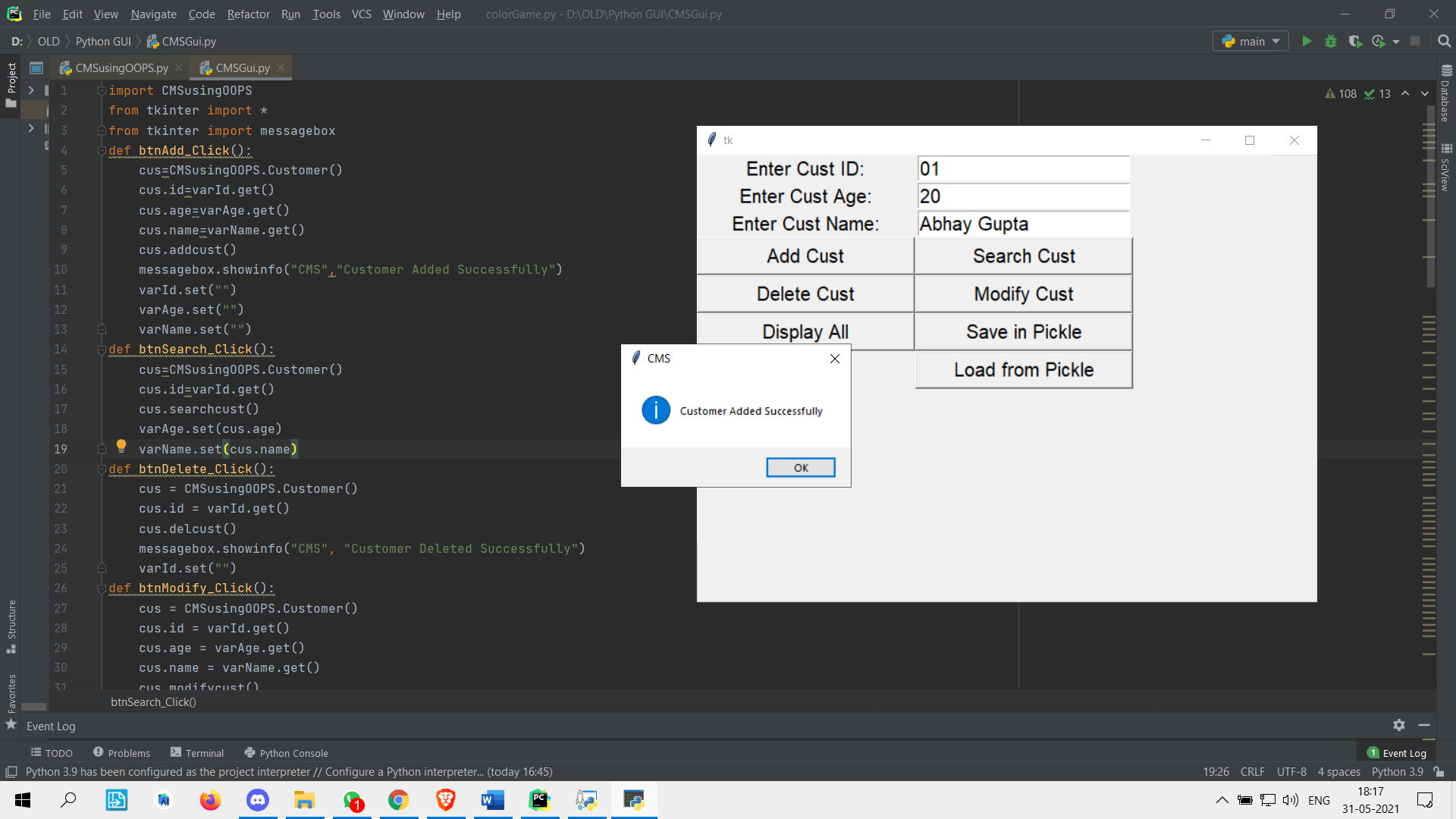
import pickle  
class Customer:  
 cuslist=[]  
 def \_\_init\_\_(self):  
 self.id=0  
 self.age=0  
 self.name=""  
  
 def addcust(self):  
 Customer.cuslist.append(self)  
  
 def searchcust(self):  
 for cus in Customer.cuslist:  
 if(cus.id==self.id):  
 self.age=cus.age  
 self.name=cus.name  
 def modifycust(self):  
 for cus in Customer.cuslist:  
 if(cus.id==self.id):  
 cus.age=self.age  
 cus.name=self.name  
  
 def delcust(self):  
 # for i in range(len(Customer.cuslist)):  
 # if(Customer.cuslist[i].id==self.id):  
 # cus=Customer.cuslist.pop(i)  
 # return  
 for e in Customer.cuslist:  
 if(e.id==self.id):  
 Customer.cuslist.remove(e)  
 return  
 @staticmethod  
 def savetoPickle():  
 f=open("D://cetpa//mypickle.txt","wb")  
 pickle.dump(Customer.cuslist,f)  
 f.close()  
 @staticmethod  
 def loadfromPickle():  
 f = open("D://cetpa//mypickle.txt", "rb")  
 Customer.cuslist=pickle.load(f)  
 f.close()  
  
  
#Presentation Layer  
if(\_\_name\_\_=="\_\_main\_\_"):  
 def displayall():  
 for cus in Customer.cuslist:  
 print("Customer ID:",cus.id, "Customer Age:",cus.age, "Customer Name:",cus.name)  
  
  
 while(1):  
 print("1 to add customer, 2 to delete customer")  
 print("3 to search customer, 4 to display all")  
 print("5 to exit, 6 save to pickle, 7 load from pickle:")  
 c=int(input("Enter Your Choice"))  
 if(c==1): #Add Customer  
 cus=Customer()  
 cus.id=int(input("Enter Customer ID"))  
 cus.age=int(input("Enter Customer Age"))  
 cus.name=input("Enter Customer Name")  
 cus.addcust()  
 elif(c==2): #Delete Customer  
 cus=Customer()  
 cus.id=int(input("Enter Customer ID"))  
 cus.delcust()  
 print("Customer Deleted Successfully")  
 elif(c==3): #Search Customer  
 cus=Customer()  
 cus.id=int(input("Enter Customer ID"))  
 cus.searchcust()  
 print("Cust ID:",cus.id,"Cust Name:",cus.name, "Cust Age:",cus.age)  
 elif(c==4):  
 displayall()  
 elif(c==5):  
 exit()  
 elif (c == 6):  
 Customer.savetoPickle()  
 elif (c == 7):  
 Customer.loadfromPickle()

**GUI Code:**

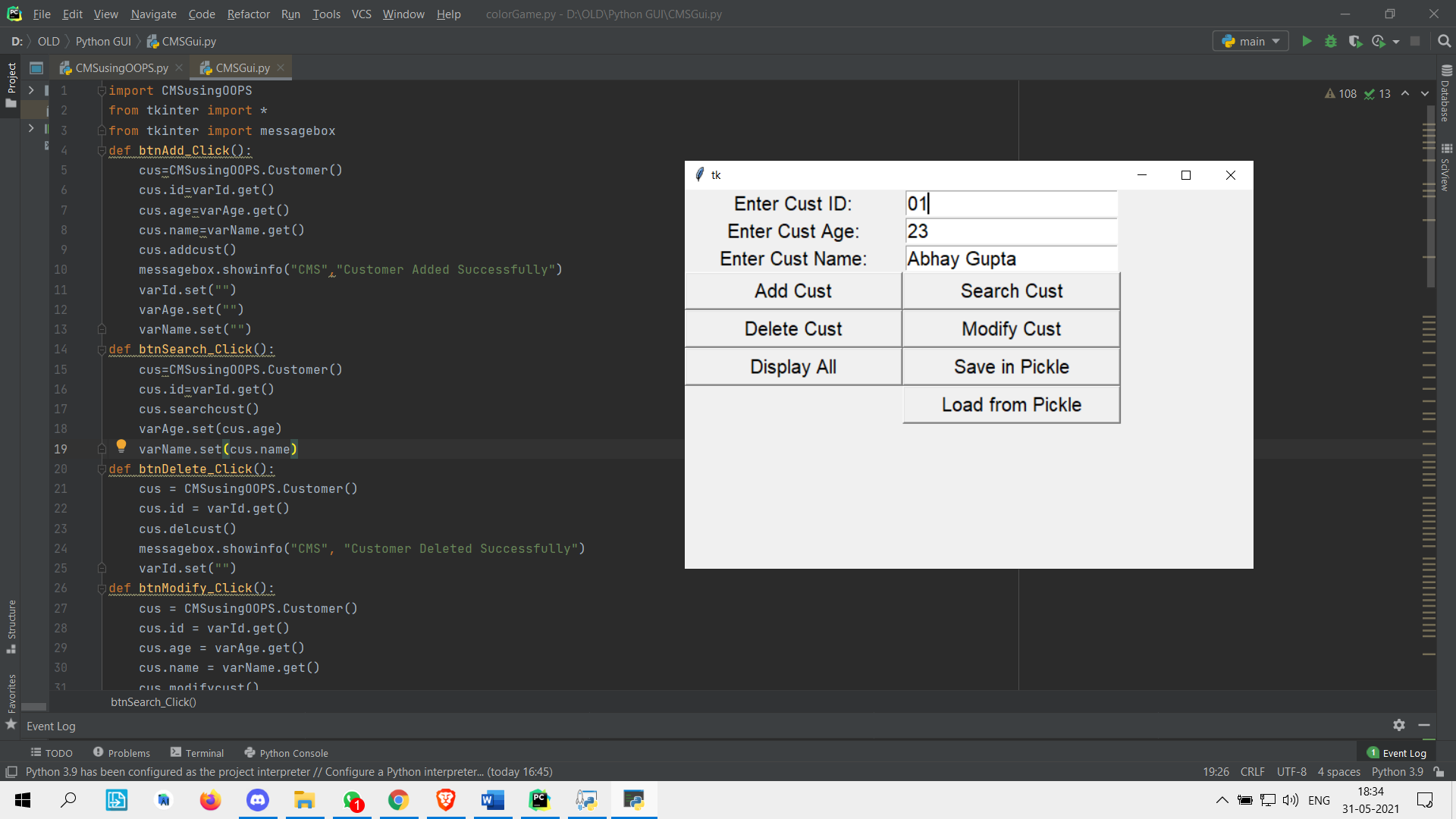
import CMSusingOOPS  
from tkinter import \*  
from tkinter import messagebox  
def btnAdd\_Click():  
 cus=CMSusingOOPS.Customer()  
 cus.id=varId.get()  
 cus.age=varAge.get()  
 cus.name=varName.get()  
 cus.addcust()  
 messagebox.showinfo("CMS","Customer Added Successfully")  
 varId.set("")  
 varAge.set("")  
 varName.set("")  
def btnSearch\_Click():  
 cus=CMSusingOOPS.Customer()  
 cus.id=varId.get()  
 cus.searchcust()  
 varAge.set(cus.age)  
 varName.set(cus.name)  
def btnDelete\_Click():  
 cus = CMSusingOOPS.Customer()  
 cus.id = varId.get()  
 cus.delcust()  
 messagebox.showinfo("CMS", "Customer Deleted Successfully")  
 varId.set("")  
def btnModify\_Click():  
 cus = CMSusingOOPS.Customer()  
 cus.id = varId.get()  
 cus.age = varAge.get()  
 cus.name = varName.get()  
 cus.modifycust()  
 messagebox.showinfo("CMS", "Customer Modified Successfully")  
 varId.set("")  
 varAge.set("")  
 varName.set("")  
def btnAll\_Click():  
 root1=Tk()  
 lblId1=Label(root1,text="CUST ID",bg="orange",font=1,width=20)  
 lblId1.grid(row=0,column=0)  
 lblAge1 = Label(root1, text="CUST AGE", bg="orange", font=1, width=20)  
 lblAge1.grid(row=0, column=1)  
 lblName1 = Label(root1, text="CUST NAME", bg="orange", font=1, width=20)  
 lblName1.grid(row=0, column=2)  
 i=1  
 for e in CMSusingOOPS.Customer.cuslist:  
 lblId2 = Label(root1, text=f"{e.id}", bg="pink", font=1, width=20)  
 lblId2.grid(row=i, column=0)  
 lblAge2 = Label(root1, text=f"{e.age}", bg="pink", font=1, width=20)  
 lblAge2.grid(row=i, column=1)  
 lblName2 = Label(root1, text=f"{e.name}", bg="pink", font=1, width=20)  
 lblName2.grid(row=i, column=2)  
 i+=1  
def btnSave\_Click():  
 CMSusingOOPS.Customer.savetoPickle()  
def btnLoad\_Click():  
 CMSusingOOPS.Customer.loadfromPickle()  
root=Tk()  
root.geometry("600x400")  
lblId=Label(root,text="Enter Cust ID:",font=1)  
lblId.grid(row=0,column=0)  
lblAge=Label(root,text="Enter Cust Age:",font=1)  
lblAge.grid(row=1,column=0)  
lblName=Label(root,text="Enter Cust Name:",font=1)  
lblName.grid(row=2,column=0)  
  
varId=StringVar()  
entrId=Entry(root,textvariable=varId,font=1)  
entrId.grid(row=0,column=1)  
varAge=StringVar()  
entrAge=Entry(root,textvariable=varAge,font=1)  
entrAge.grid(row=1,column=1)  
varName=StringVar()  
entrName=Entry(root,textvariable=varName,font=1)  
entrName.grid(row=2,column=1)  
  
btnAdd=Button(root,text="Add Cust",font=1,command=btnAdd\_Click,width=20)  
btnAdd.grid(row=3,column=0)  
btnSearch=Button(root,text="Search Cust",font=1,command=btnSearch\_Click,width=20)  
btnSearch.grid(row=3,column=1)  
btnDelete=Button(root,text="Delete Cust",font=1,command=btnDelete\_Click,width=20)  
btnDelete.grid(row=4,column=0)  
btnModify=Button(root,text="Modify Cust",font=1,command=btnModify\_Click,width=20)  
btnModify.grid(row=4,column=1)  
btnAll=Button(root,text="Display All",font=1,command=btnAll\_Click,width=20)  
btnAll.grid(row=5,column=0)  
btnSave=Button(root,text="Save in Pickle",font=1,command=btnSave\_Click,width=20)  
btnSave.grid(row=5,column=1)  
btnLoad=Button(root,text="Load from Pickle",font=1,command=btnLoad\_Click,width=20)  
btnLoad.grid(row=6,column=1)  
  
  
root.mainloop()

**Output:**

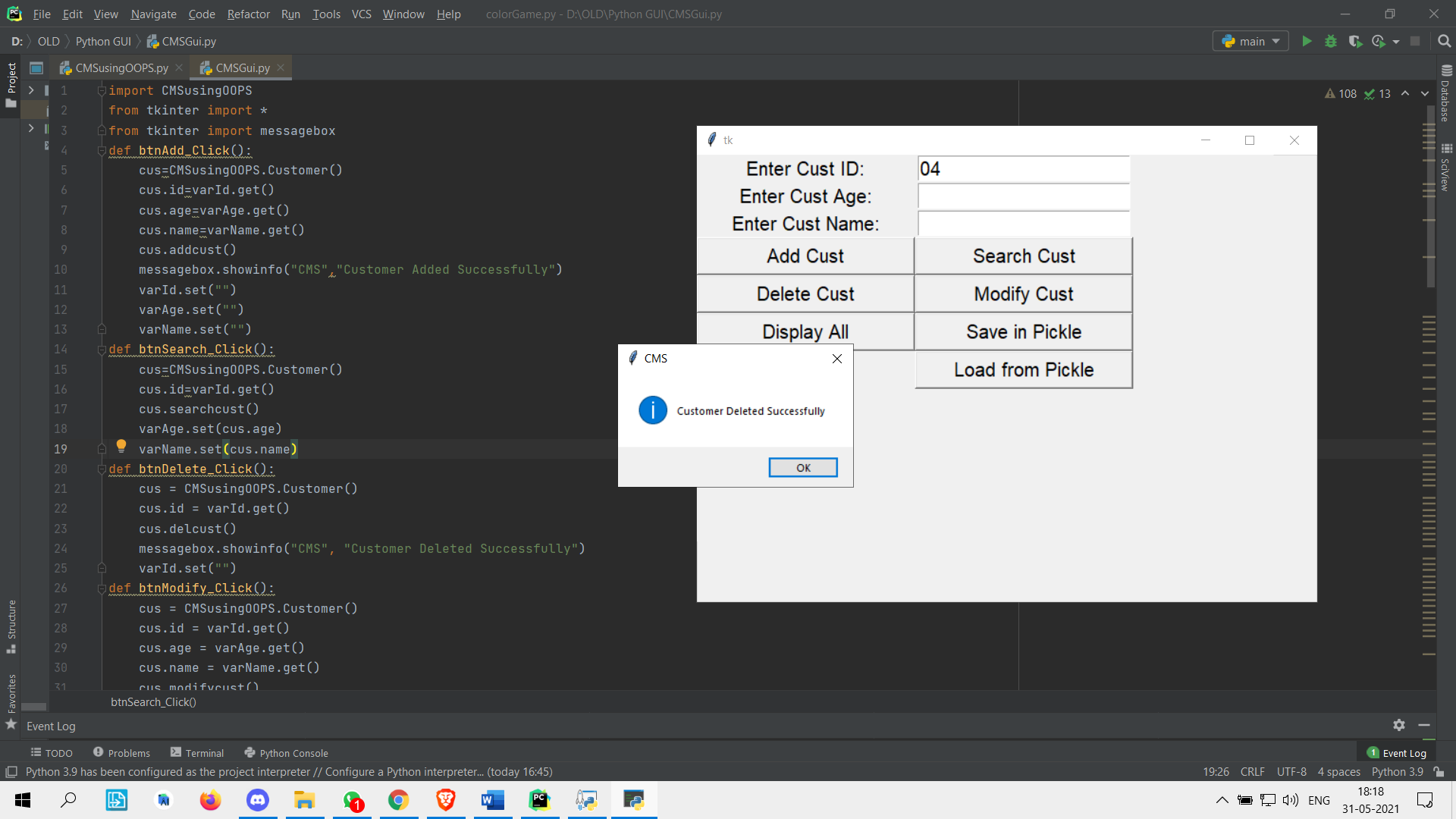


In this we are adding the customers to the list

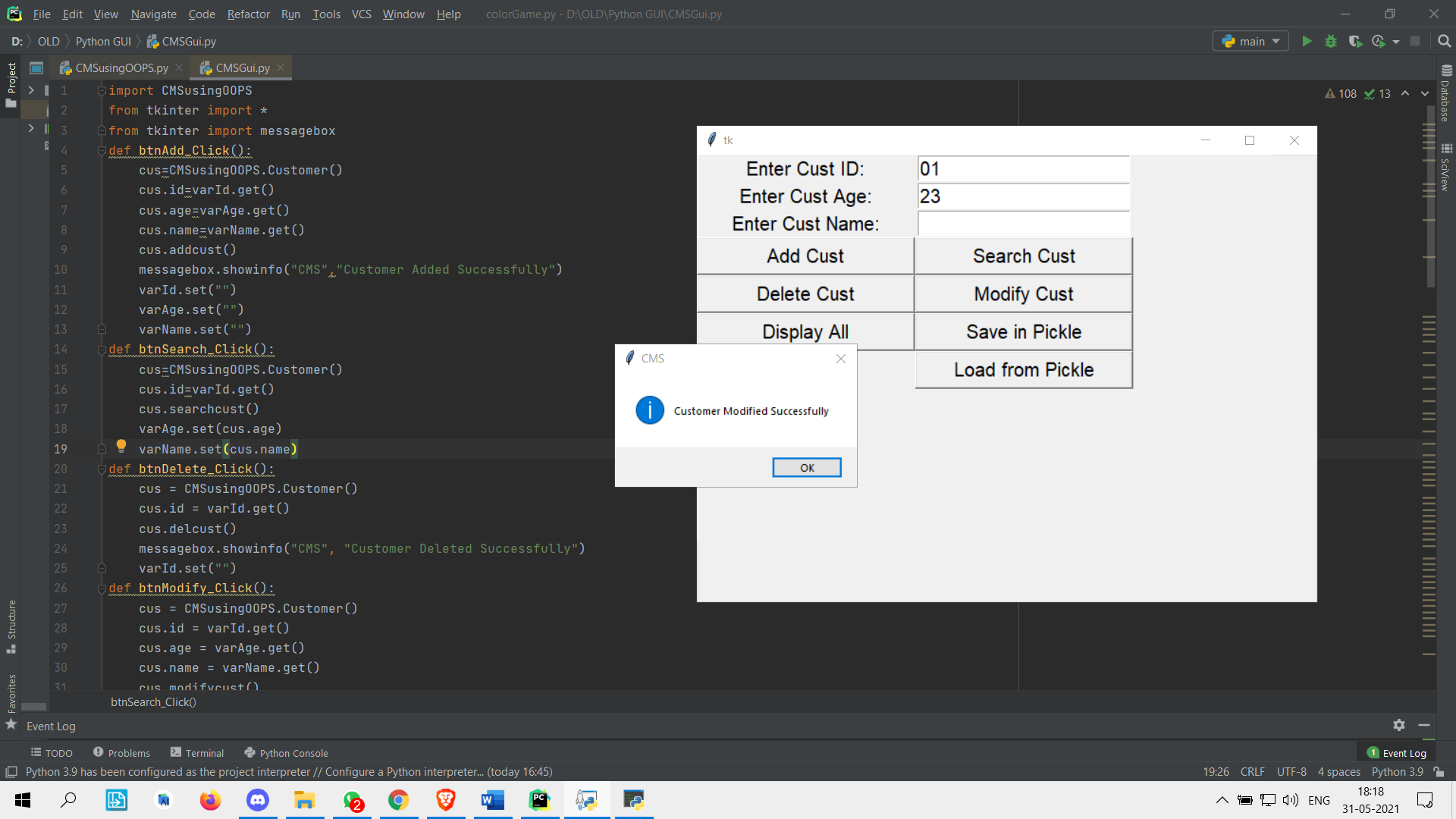
By just putting **custID** and click on **search cust** user can see the **details** of the **particular customer.**



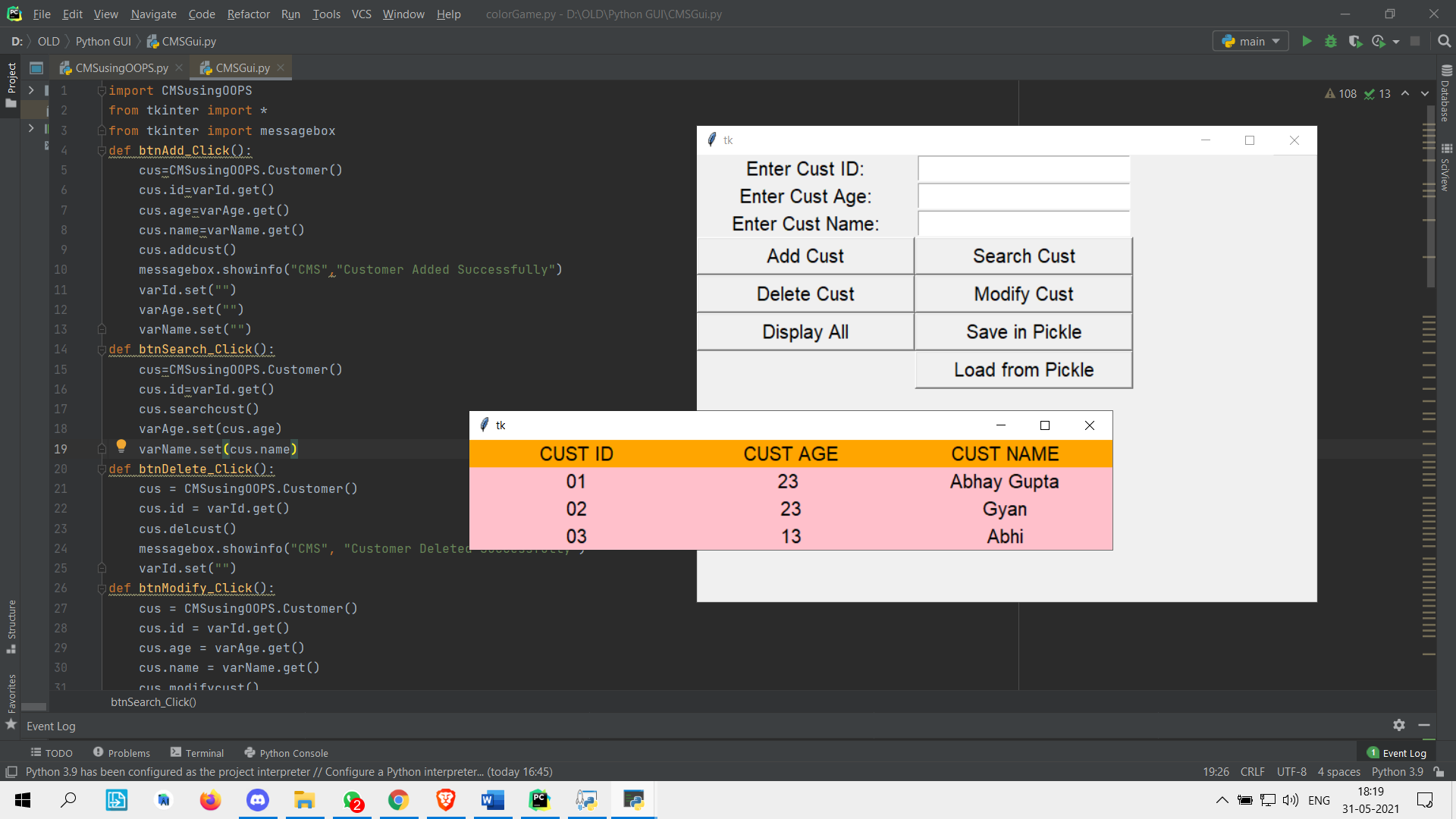
By giving **CustID,** user can **delete** that customer details from the list



By giving details of existing customer and clicking on **modify**, it will change the details of that customer in the list.



By clicking on **Display All**, it will show all the details present in this **database**.



In last is user click on **save in pickle** then it will save all the data to the **temporary storage** that we have created and whenever we restart the system again and we just have to click on **load from pickle**, then it will automatically **import** all the data from **database** and **user** can **access** it.