# ACKNOWLEDGEMENT

We would like to sincerely thank our 11th grade computer teacher, Ms. Reshmi Nambiar, who unfortunately left last year, for her guidance and inspiration.

We would like to thank our computer teacher Ms. Remya, for her staunch support, guidance and constant vigilance, without which our project would not have been possible. We would also like to thank the school for its support and the facilities it has provided us with to develop on the project, which assisted in allowing us to understand the concept in an effective manner.

My team members were an integral part of this project. Without their support and guidance nothing would have been possible. We, as a team, have worked towards this project's perfection.

# Index

# AIM

The resource crisis that the world is currently facing is a matter of increasing concern.

Our project not only aims at creating a virtual fuel station, to allow the user to purchase fuel through the program and even locate nearby fuel stations, but also attempts at bringing about a sense of environmental consciousness by keeping the user updated on the amount of pollutants he releases into the atmosphere and the amount of money he spends on the same, on a monthly basis.

Along with this, to enhance our fuel station experience, we've designed an auxiliary convenience store (or C-store), to allow the user to further purchase everyday essentials such as tools, food and drinks.

# User Documentation

## Modules Used In The Program.

| Module Name | It's Function |
|---|---|
| **Tkinter** | This Module Provides A GUI (Graphic User Interface) For The Python Program. This Make Python More User Friendly .It Also Uses Data Hiding. |
| **tkFont** | This Module Has Been Imported To Enable The Use Of A Custom Fonts. |
| **tkMessageBox** | This Module Has Been Imported To Display A Message To User When An Error (Here Invalid Entries) Occurs. |
| **pickle** | This Module Is Used To Add, View And Modify Data In A Binary File. |
| **os** | The Module Has Been Imported To Rename And Remove Files While Deleting And Updating Data. |
| **random** | This Module Is Used To Randomize Captcha Images And Ads. |
| **datetime** | This Module Allows To Know Date And Time At Which The Purchase Has Taken Place. |

| | |
|---|---|
| | |
| **smtplib** | This Module Allows The Program To Mail The User Regarding The Sign Up Status. |
| **webbrower** | This Module Allows User To Link Webpages (Google Maps) With The Program. |
| **string** | This Module Contains A Number Of Functions To Process Standard Python Strings. |
| **sys** | This Module Provides Information About Constants, Functions And Methods Of The Python Interpreter And Provides A Number Of Functions And Variables That Can Be Used To Manipulate Different Parts Of The Python Runtime Environment. |

# Functions Used In The Program

| FUNCTION | PURPOSE |
|---|---|
| **mainfuelapp()** | This Function Is Used To Display The Main Page Of The Program, Through Which The User/Admin Can Choose To Sign-Up/Log In. |

| | |
|---|---|
| **signupcus()** | Used To Display The Sign Up Page For Users. |
| **createaccountcus(username,password,fuelcapacity,fuelamt)** | Used To Create The Customer Object With Parameters And Display The Confirmation Page If All Values Entered Are Correct. |
| **profilepagecus(un)** | Used To Display The User's Profile Page. |
| **fuelexppage(name)** | Used To Display The Total Amount Of Money Spent On Fuel Every Month. |
| **co2emm(name)** | Used To Display The Total Amount Of Fuel Released By The User Through Consumption Of Fuel. |
| **purchase(un)** | Used To Display The Page Through Which The Purchase Of Fuel Is Done And To Input Values Of Current Fuel Amount, Fuel To Be Filled, And Month And Fuel Type. |
| **purchase1(name,amt,fill,monthentry,fueltype)** | Used To Take The Values Of Fuel Type, Amount To |

| | |
|---|---|
| | Be Filled, Current Amount Of Fuel And Month And Carry Out Calculations For Billing. |
| **calculate(fueltype1,fillup1)** | Used To Calculate The Cost Of The Fuel Filled. |
| **monthlyexp(name,month,x)** | Used To Update The Expenditure On Fuel For The Month Entered. |
| **monthlyem(name, month, currentamt,fueltype)** | Used To Update The Amount Of $CO_2$ Released Through The User's Fuel Consumption. |
| **fill(name,fillup,currentamt)** | Used To Update The Current Amount Of Fuel Present In The User's Tank. |
| **locate()** | Used To Locate The Fuel Stations In The Emirate That Is Entered. |
| **printadmin()** | This Function Enables The Admin To Print His/Her Details From Their Profile Page. |

| | |
|---|---|
| **createaccountadmin()** | This Function Enables The User To Save His/ Her Details, After He/She Has Created A New Account As Admin, Into The Binary File "admin1.Dat". |
| **adminsignuppage()** | This Function Is Called When A User Wants To Sign Up As An Admin (New User). |
| **adminlogin()** | This Function Is Called When A User Wants To Login As An Admin (Existing User). |
| **confirmaccountadmin()** | This Function Checks Whether The Admin Has Entered His/Her Correct Details When He/She Logins As Admin. |
| **printall()** | This Function Enables The Admin To Print All The Customer's Details. |
| **delwindow()** | This Function Provides The GUI For The deletecustomer() And Calls The Same. |

| | |
|---|---|
| **deletecustomer()** | This Function Enables The Admin To Delete A Certain Customer Using His Username. |
| **profilepageadmin()** | This Function Enables The Admin To Call The Following Functions: delwindow(), printall() And printadmin(). |
| **quit3()** | This Function Destroys The Page Created By createaccountadmin() And Opens The User's Profile Page (Calls The profilepageadmin()). |
| **Displaymenu()** | Display Items In The Store As Buttons |
| **Displayinfo()** | Display's Information Regarding The Item In The Store When Button Is Clicked |
| **Additem()** | Allows Admin To Add A New Item And Its Related Details |
| **Addfunction()** | Adds The Items To Their Respective Database. |
| **Sername()** | Allows Admin To Search For An Item By Its Name. |

| | |
|---|---|
| **Sercost()** | Allows Admin To Search For An Item By It Cost |
| **Serstock()** | Allows Admin To Search For An Item By Its Stock |
| **Sersold()** | Allows Admin To Search For An Item By Amount Of Items Sold |
| **Serupdate()** | Searches For The Item To Be Updated |
| **Updateinput()** | Takes In The Update And Saves The Updated Item Details In The Database |
| **Ads()** | Displays Ads In The Customer Dashboard |
| **Displayitems()** | Display Items In The Store As Buttons |
| **Displayinfo()** | Display's Information Regarding The Item In The Store When Button Is Clicked |
| **Addqty()** | Increases Quantity To Be Purchased |
| **Subqty()** | Decreases Quantity To Be Purchased |
| **Buyitem()** | Confirms The Purchase Of Item And Updates The Database |
| **Payment()** | Provides Payment Options |
| **Deliverytime()** | Provides Token For Purchase. |

# CLASS FUNCTIONS

| FUNCTION | PURPOSE |
|---|---|
| **assigndatacus(self,username,password,fuelcapacity,fuelamt)** | Used To Assign The Values Of Username, Password, User's Fuel Capacity, And User's Current Amount Of Fuel To Customer Object. |
| **assigndataadmin()** | This Function Assigns The Value For A Particular Admin To Its Respective Variable. |

# Program Algorithm

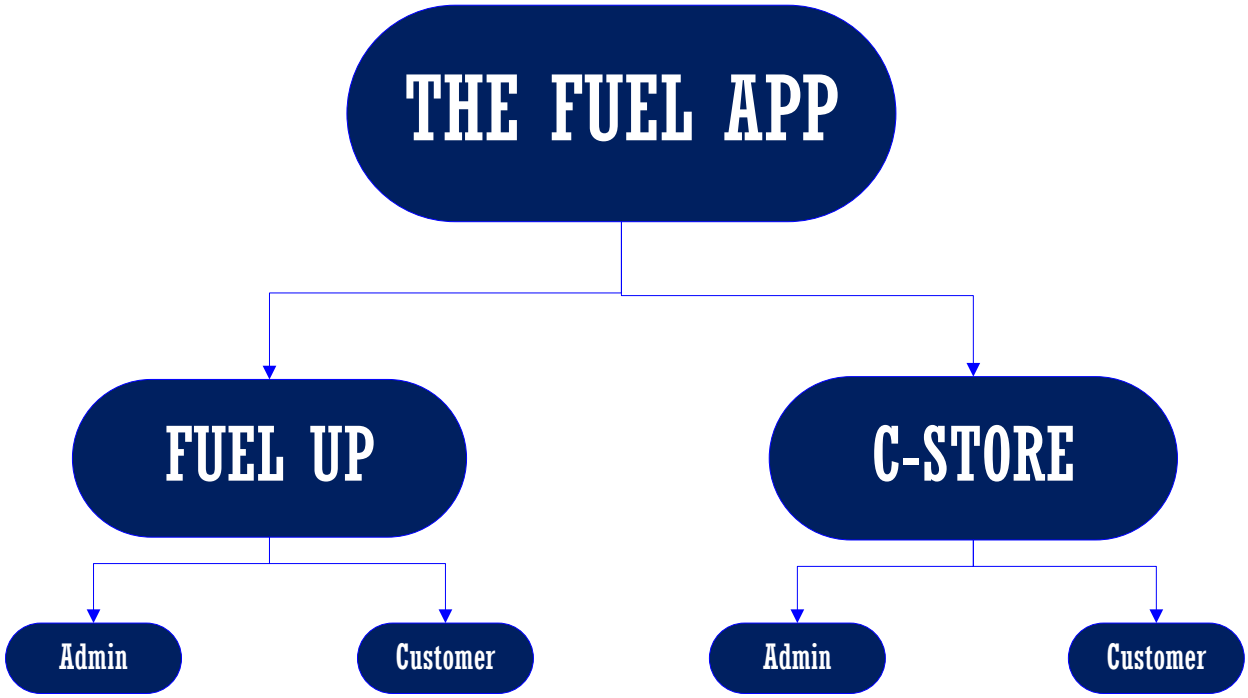
## Main Algorithm

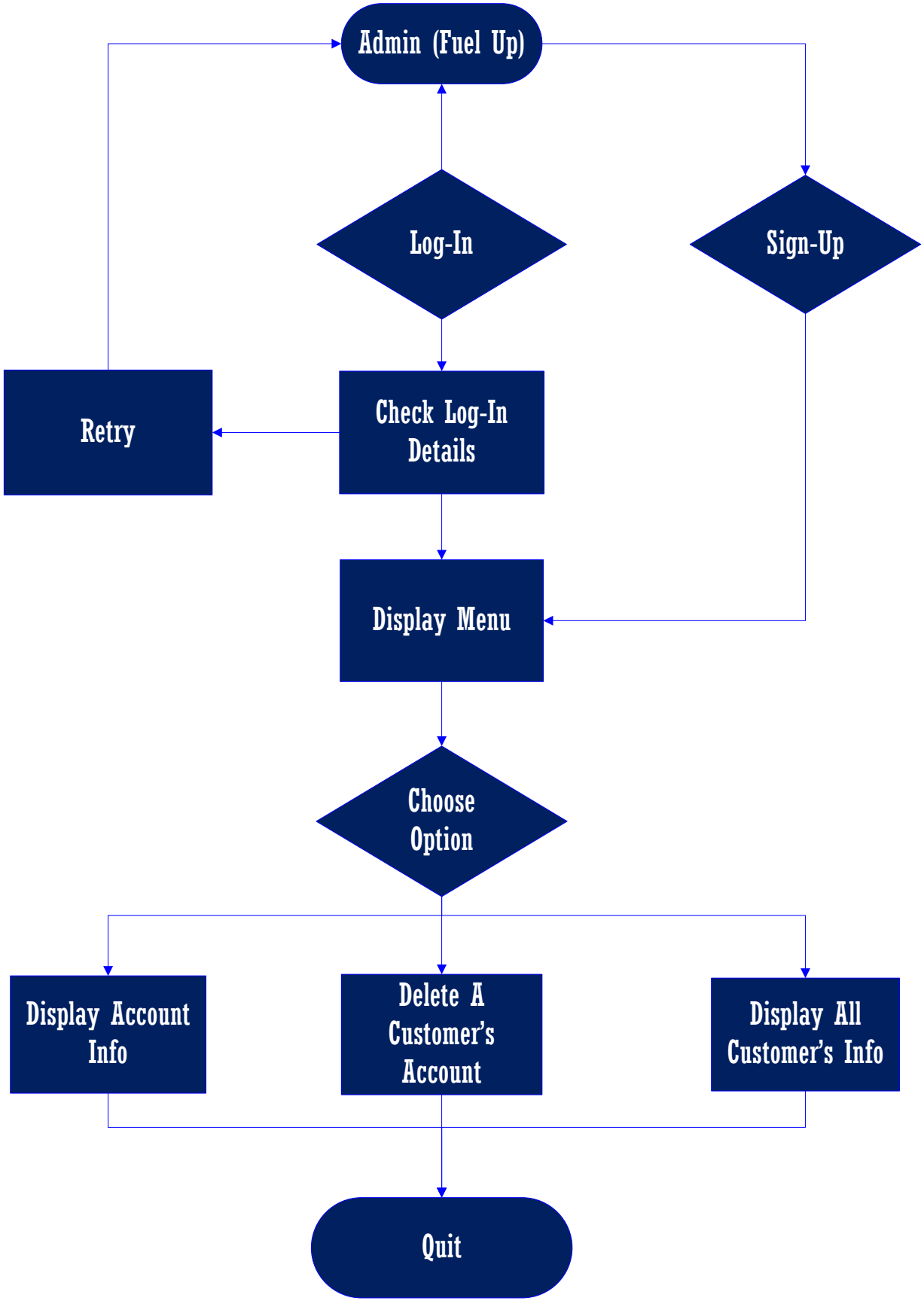
Step 1: The Fuel App, Fuel Up Or C-Store. If Fuel App Is Selected Proceed To Step 2, Else If C-Store Is Selected Proceed To Step 27.
Step 2: Fuel Up, Sign Up Or Login As Admin Or Sign Up Or Login As Customer. For Admin, Proceed To Step 3, Else If For Customer, Proceed To Step 14.
Step 3: To Sign Up As Admin, Enter Username, Password And Designation. Proceed To Step 7.
Step 4: To Login Enter Username And Password.
Step 5: If Correct Details Are Entered, Profile Page Is Opened. Proceed To Step 7.
Step 6: If Incorrect Details Are Entered Repeat Step 4.
Step 7: To Display Customer Data Proceed To Step 8 Or To Delete Customer Data Proceed To Step 9 Or To Display Admin Profile Proceed To Step 10 Or To Quit Proceed To Step 13.
Step 8: To Display Customer Data Click "Display Now". Data Is Displayed.

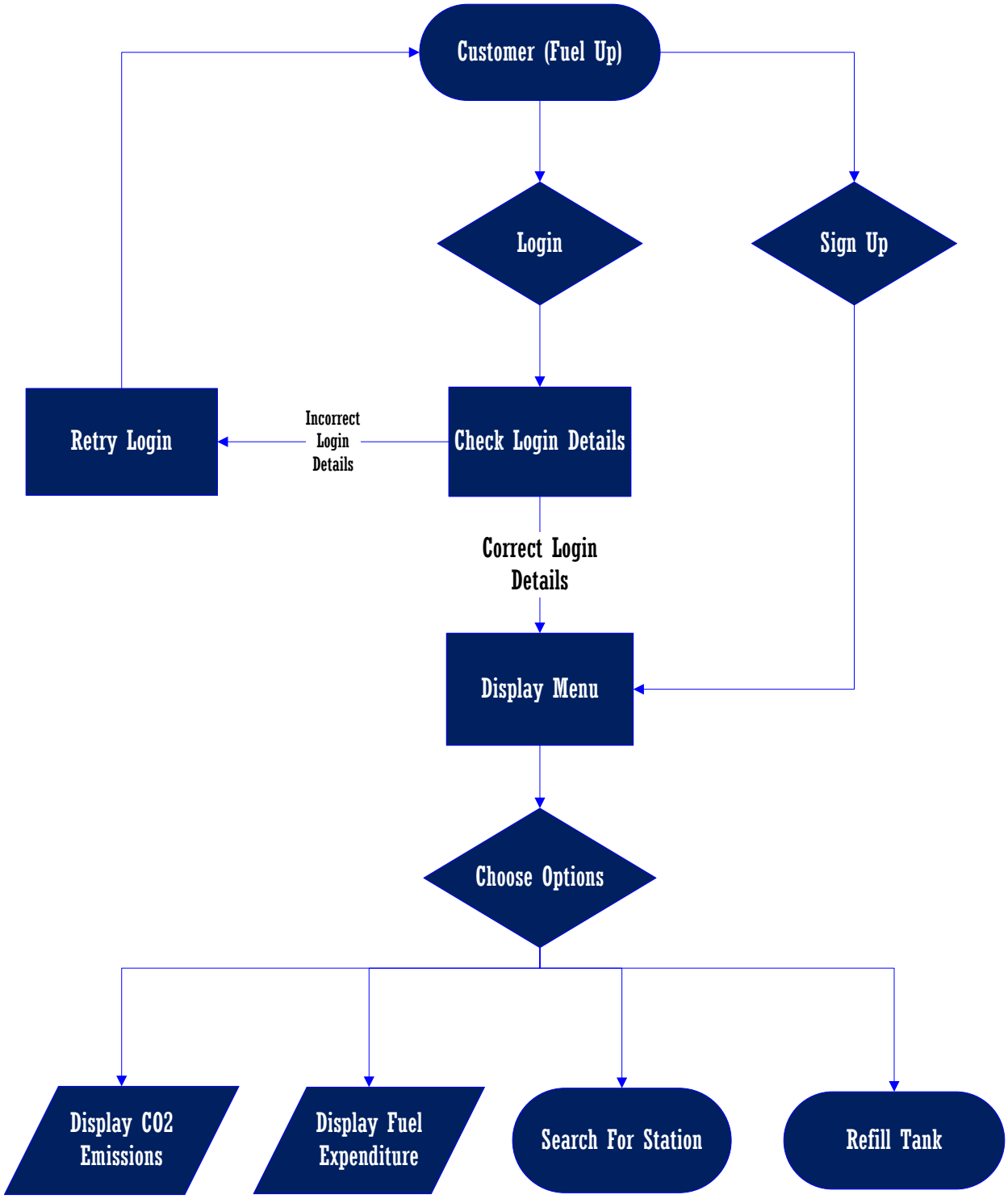
Step 9: To Delete Customer Data, Enter Customer Name And Click “Delete”
Step 10: To Display Admin Profile Click “Display Now”. Admin Details Are Displayed.
Step 11: Proceed To Step 14 To Continue Or Proceed To Step 26 To Quit.
Step 12: To Continue Return To The Profile Page. Return To Step 7
Step 13: Sign Up Or Login As Customer.
Step 14: To Sign Up Enter Username, Password, Tank Maximum Capacity And Current Fuel In Tank. Proceed To Step 18.
Step 15: To Login Enter Username And Password.
Step 16: If Correct Details Are Entered, Profile Page Is Opened. Proceed To Step 18.
Step 17: If Incorrect Details Are Entered Repeat Step 14.
Step 18: To Check $CO_2$ Emissions Over The Year Proceed To Step 19 Or To Check Fuel Expenditure Over The Year Proceed To Step 20 Or To Locate Station Proceed To Step 21 Or To Fill Tank Proceed To Step 22 Or To Display All Data Proceed To Step 23 Or To Quit Proceed To Step 26.
Step 19: To Check $CO_2$ Emissions Over The Years Click “Check Now” And Data Is Displayed.
Step 20: To Check Fuel Expenditure Over The Year, Click “Check Now” And Data Is Displayed.
Step 21: To Locate Station Click “Locate”, Then Enter Desired Location And Click “View”.
Step 22: To Fill Tank Click “Fuel Up”, Then Enter Details And Click “Done”.
Step 23: To Display All Data Click “Check Now” And Data Is Displayed.
Step 24: Proceed To Step 18 To Continue Or Proceed To Step 26 To Quit.
Step 25: To Continue Return To The Profile Page. Return To Step 6.
Step 26: To Quit Click “Quit”. Display “Thank You For Using Fuel App”.
Step 27: C-Store, Sign Up Or Login As Admin Or Sign Up Or Login As Customer. For Admin, Proceed To Step 28, Else If For Customer, Proceed To Step 47.
Step 28: Sign Up Or Login As Admin.
Step 29: To Sign Up Enter Username, Password And Designation. Proceed To Step 33.
Step 30: To Login Enter Username And Password.
Step 31: If Correct Details Are Entered, Profile Page Is Opened. Proceed To Step 33.
Step 32: If Incorrect Details Are Entered Repeat Step 30.
Step 33: To Display The Details Of An Item, Proceed To Step 34, To Add An Item, Move To Step 38, To Search For An Item, Move To Step 39, To Update The Details Of An Item, Move To Step 44, To Remove An Item, Move To Step 46.
Step 34: If Snacks Are Selected, Proceed To Step 35, If Juices Are Selected, Proceed To Step 36, If E-Tools Are Selected, Proceed To Step 37.
Step 35: Display All Details Of Every Item In Snacks, Including Number Of Items Sold, Stock Remaining, Amount Gained From Sales Of The Item, Price Of The Item, Name Of The Item And Item Number.
Step 36: Display All Details Of Every Item In Juices, Including Number Of Items Sold, Stock Remaining, Amount Gained From Sales Of The Item, Price Of The Item, Name Of The Item And Item Number.
Step 37: Display All Details Of Every Item In E-Tools, Including Number Of Items Sold, Stock Remaining, Amount Gained From Sales Of The Item, Price Of The Item, Name Of The Item And Item Number.

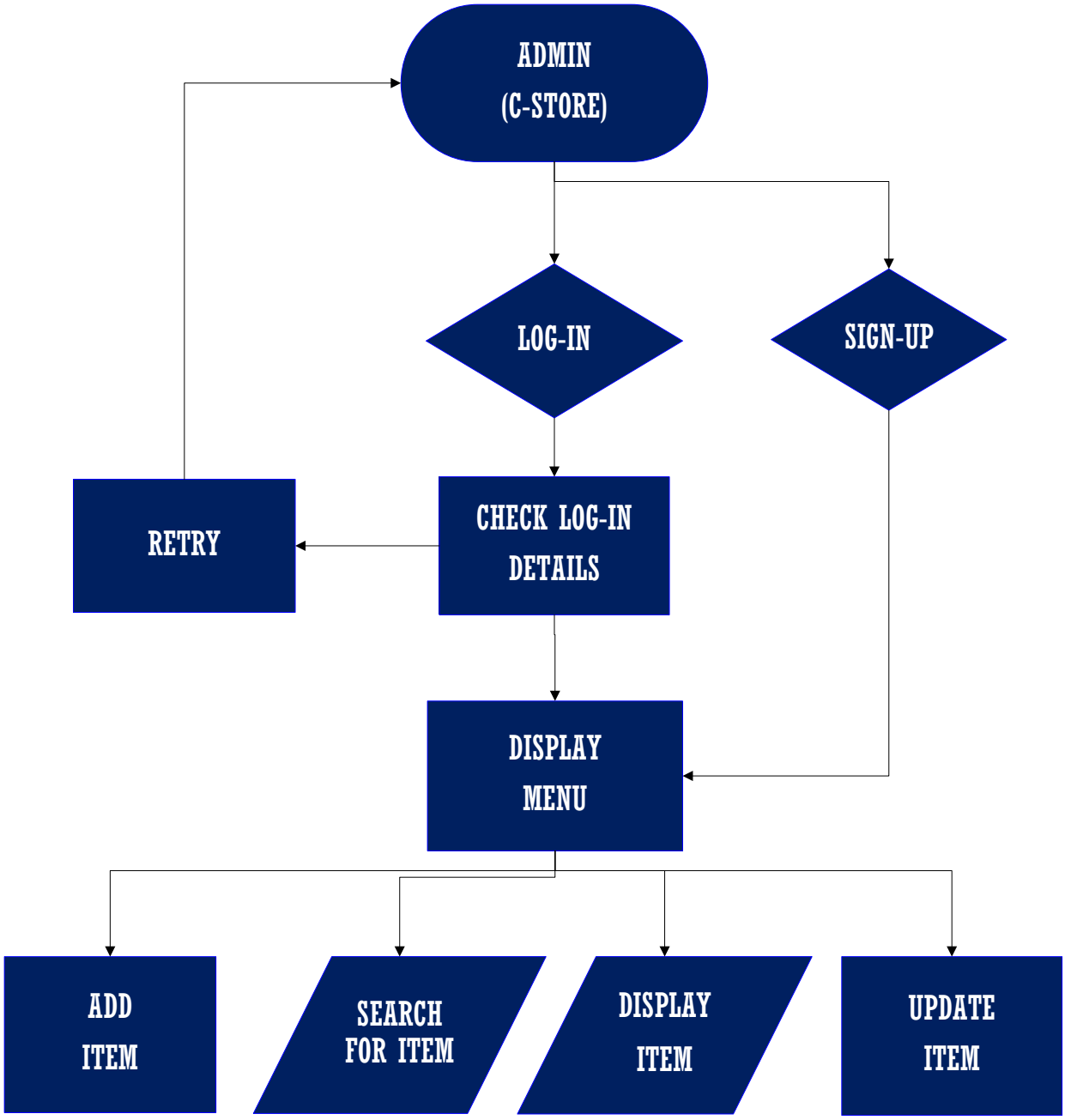
Step 38: To Add An Item, Enter The Details Of The Item, Including Number Of Items Sold, Stock Remaining, Amount Gained From Sales Of The Item, Price Of The Item, Name Of The Item And Item Number And Choose Whether The Object Is A Snack, Juice Or E-Tool.
Step 39: To Search For An Item, Proceed To Step 40 To Search By Name, Proceed To Step 41 To Search By Item Number, Proceed To Step 42 To Search By Amount Sold, Proceed To Step 43 To Search By Amount Of Money Gained, Proceed To Step 43 To Search By Stock Left.
Step 40: Enter The Name Of The Item, Display The Item With That Item Name.
Step 41: Enter The Item Number, Display The Item With That Item Number.
Step 42: Enter The Amount Of The Item Sold, Display All Items That Have Been Sold That Many Times.
Step 43: Enter The Amount Of Money Gained, Display All Items Whose Sales Have Given That Amount.
Step 44: Enter The Item You Wish To Update.
Step 45: Re-Enter The Details Of The Item You Wish To Update, Such As Number Of Items Sold, Stock Remaining, Amount Gained From Sales Of The Item, Price Of The Item, Name Of The Item And Item Number And Choose Whether The Object Is A Snack, Juice Or E-Tool.
Step 46: To Delete An Item, Choose If It Is A Snack, Juice Or E-Tool And Enter The Name Of The Item.
Step 47: Sign Up Or Login As Customer.
Step 48: To Sign Up Enter Username, Password And Designation. Proceed To Step 52
Step 49: To Login Enter Username And Password.
Step 50: If Correct Details Are Entered, Profile Page Is Opened. Proceed To Step 52.
Step 51: If Incorrect Details Are Entered Repeat Step 49.
Step 52: Choose Whether To Purchase Snacks, Juices Or E-Tools.
Step 53: Select The Items You Wish To Purchase, If The Item You Select Is Out Of Stock, Move To Step 54, Else, Proceed To Step 55.
Step 54: Display A Message Saying The Item Is Out Of Stock, And Repeat Step 53.
Step 55: Display Details Of Items That Have Been Purchased.
Step 56: Display Bill.
Step 57: Calculate Approximate Time Required Before Customer Can Obtain Items.
Step 58: Display Time Required Before Customer Can Obtain Items.

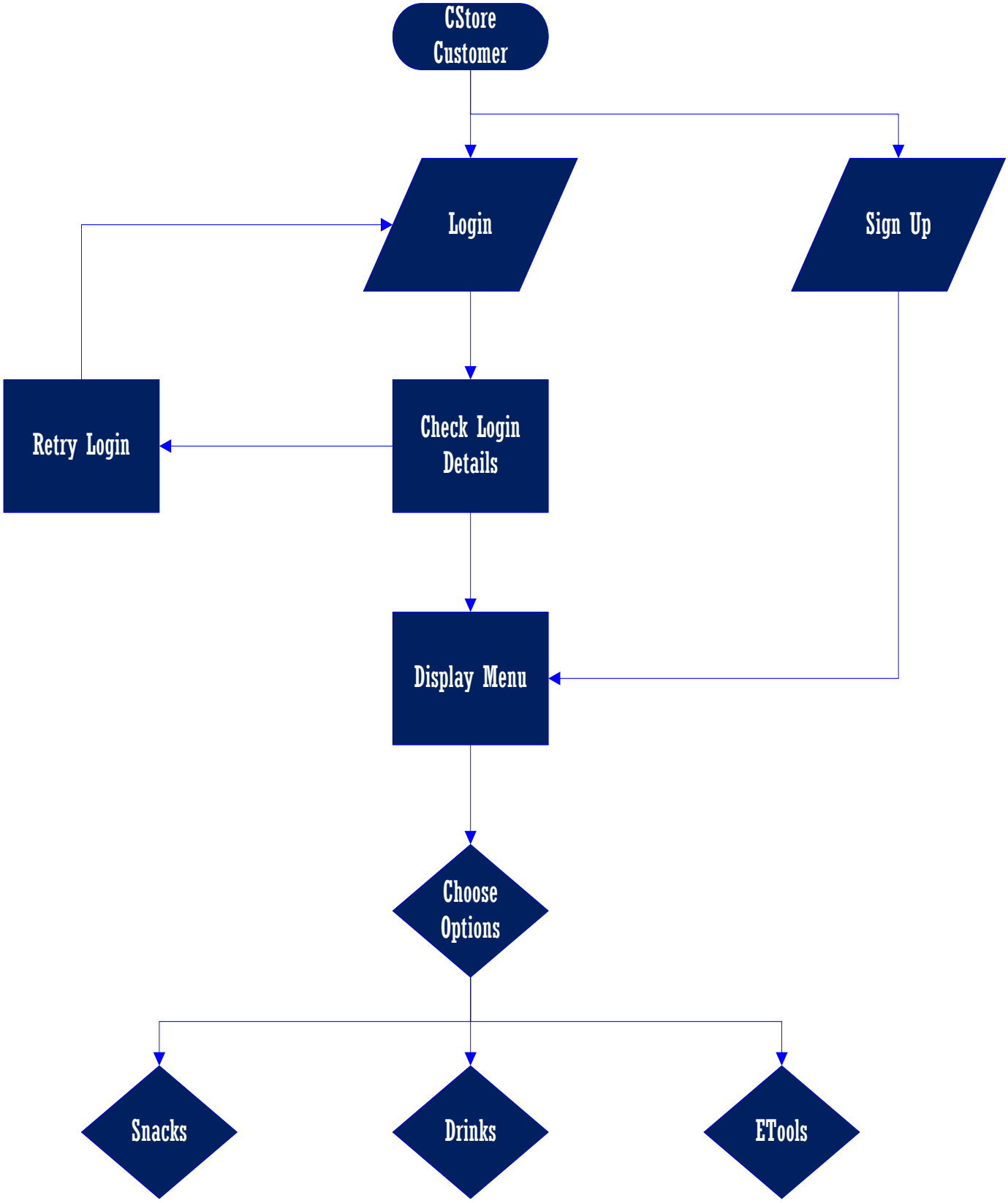
# System Flowchart

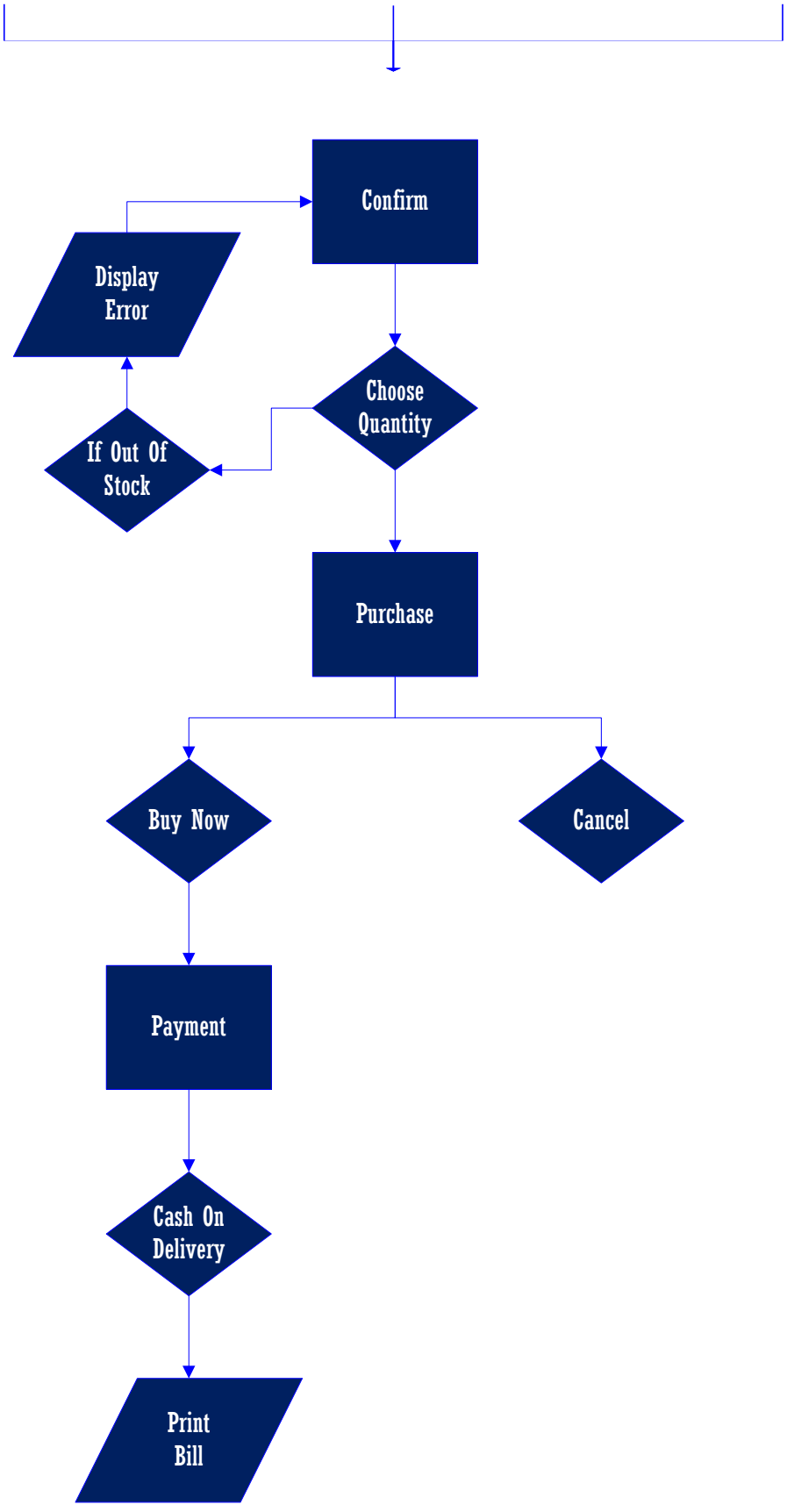



Admin (Fuel Up)
Log-In
Sign-Up
Retry
Check Log-In Details
Display Menu
Choose Option
Display Account Info
Delete A Customer's Account
Display All Customer's Info
Quit


Customer (Fuel Up)
Login
Sign Up
Retry Login
Incorrect Login Details
Check Login Details
Correct Login Details
Display Menu
Choose Options
Display CO2 Emissions
Display Fuel Expenditure
Search For Station
Refill Tank


ADMIN
(C-STORE)
LOG-IN
SIGN-UP
RETRY
CHECK LOG-IN
DETAILS
DISPLAY
MENU
ADD
ITEM
SEARCH
FOR ITEM
DISPLAY
ITEM
UPDATE
ITEM


CStore Customer
Login
Sign Up
Retry Login
Check Login Details
Display Menu
Choose Options
Snacks
Drinks
ETools


Confirm
Display Error
Choose Quantity
If Out Of Stock
Purchase
Buy Now
Cancel
Payment
Cash On Delivery
Print Bill

# Program Code

```
from Tkinter import *
import tkFont
import pickle
import os
import tkMessageBox
import string
import webbrowser
import datetime
import sys
import random
import smtplib
import time

fuelprices={'fuel oil':30,'diesel':40,'natural gas':15}

class admin:
    def assigndataadmin(self,username,password,designation):
        self.username=str(username)
        self.password=password
        self.designation=designation
class customer: #on sign-up
    def assigndatacus(self,username,password,fuelcapacity,fuelamt):
        self.username=str(username)
        self.password=password
        self.fuelcapacity=fuelcapacity
        self.fuelamt=fuelamt #this vairable is used in the calculation of CO2 emmission

self.fuelexp={'january':0,'march':0,'april':0,'may':0,'june':0,'july':0,'august':0,'september':0,'october':0,'november':0,'december'
:0}

self.co2em={'january':0,'march':0,'april':0,'may':0,'june':0,'july':0,'august':0,'september':0,'october':0,'november':0,'december':
0}
    def setdatacus(self,username,password,fuelcapacity,new1,new2,new3):
        self.username=username
        self.password=password
        self.fuelcapacity=fuelcapacity
        self.fuelamt=new1 #this vairable is used in the calculation of CO2 emmission
```

```
            self.fuelexp=new2
            self.co2em=new3
def printadmin(un):
    f=open("admin1.dat","rb")
    flag=0
    try:
        while True:
            y=pickle.load(f)
            if y.username==un:
                flag=1
                global printadmin
                printadmin=Toplevel()
                customfontx= tkFont.Font(printadmin,family="JLS Data GothicR  NC", size=50)
                customfont= tkFont.Font(printadmin,family="JLS Data GothicR  NC", size=25)
                titlelabel=Label(printadmin,text="Admin Details",font=customfontx,fg='white',bg='black')
                namelabel=Label(printadmin,text='Name: '+str(un),font=customfont,fg='white',bg='black')
                designationlabel=Label(printadmin,text='Designation:
'+str(y.designation),font=customfont,fg='white',bg='black')
                titlelabel.pack()
                namelabel.pack()
                designationlabel.pack()
                printadmin.configure(bg='black')
                printadmin.mainloop()
    except EOFError:
        pass
    if flag==0:
        tkMessageBox.showinfo('Error','Username Not Found')
def confirmaccountcus(un,pw):
    f=open('customer1.dat','rb')
    flag=0
    try:
        while True:
            x=pickle.load(f)
            if x.username==un and x.password==pw:
                Userlogin.destroy()
                f.close()
                profilepagecus(un)
                flag=1
                break
            elif x.username!=un and x.password==pw:
```

```
                f.close()
                tkMessageBox.showinfo('Error','Incorrect Username')
                flag=1
                break
            elif x.password!=pw and x.username==un:
                f.close()
                tkMessageBox.showinfo('Error','Incorrect Password')
                flag=1
                break
            else:
                pass
    except EOFError:
        tkMessageBox.showinfo('Error','Server does not recognize this user')
        f.close()
    if flag==0:
        print 'Do not recognize this user'
def updatecus(un,new1,new2,new3):
    f=open('customer1.dat','rb')
    g=open('temp1.dat','wb')
    try:
        while True:
            y=pickle.load(f)
            if y.username==un:
                x=customer()
                x.setdatacus(y.username,y.password,y.fuelcapacity,new1,new2,new3)
                pickle.dump(x,g)
            else:
                pickle.dump(y,g)
    except EOFError:
        f.close()
        g.close()
        os.remove('customer1.dat')
        os.rename('temp1.dat','customer1.dat')
def createaccountadmin(username,password,designation):
    f=open('admin1.dat','ab')
    signup.destroy()
    x=admin()
    x.assigndataadmin(username,password,designation)
    pickle.dump(x,f)
    f.close()
```

```
    global confirmation
    confirmation=Toplevel()
    customfont= tkFont.Font(confirmation,family="JLS Data GothicR  NC", size=25)
    message=Label(confirmation,text='Congratulations!',font=customfont,fg='white',bg='black')
    confirm=Button(confirmation,text='Go To Profile',
font=customfont,fg='white',bg='black',command=lambda:quit3(username))
    message.pack(side=TOP,fill=X)
    confirm.pack(side=TOP)
    confirmation.configure(bg='black')
def adminsignuppage():
    x=admin()
    root1.destroy()
    global signup
    signup=Toplevel()
    customfonts= tkFont.Font(signup,family="JLS Data GothicR  NC", size=25)
    customfontl= tkFont.Font(signup,family="JLS Data GothicR  NC", size=50)
    customfontx= tkFont.Font(signup,family="JLS Data GothicR  NC", size=20)
    title=Label(signup,text='Admin Sign Up',fg='white',bg='black',font=customfontl)
    namelabel=Label(signup, text='Enter Your Username:', fg='white', bg='black', font=customfonts)
    passwordlabel=Label(signup, text='Enter Your Password:',fg='white',bg='black',font=customfonts)
    designationlabel=Label(signup, text='Enter Your Designation:', fg='white',bg='black',font=customfonts)
    ne=Entry(signup)
    pe=Entry(signup,show='*')
    de=Entry(signup)
    title.grid(row=0, columnspan=6)
    namelabel.grid(row=1, column=0, sticky=W)
    passwordlabel.grid(row=2, column=0, sticky=W)
    designationlabel.grid (row=3, column=0, sticky=W)
    ne.grid(row=1, column=1, sticky=E)
    pe.grid(row=2, column=1, sticky=E)
    de.grid(row=3, column=1, sticky=E)
    ed=Button(signup,text="Sign up", bg='black', fg='white', font=customfontx,
command=lambda:createaccountadmin(ne.get(),pe.get(),de.get()))
    ed.grid(row=5, column=0, columnspan=2)
    signup.configure(bg='black')
def adminlogin():
    global Adminlogin
    Adminlogin=Toplevel()
    root1.destroy()
    customfont= tkFont.Font(Adminlogin,family="JLS Data GothicR  NC", size=50)
```

```
    customfontx= tkFont.Font(Adminlogin,family="JLS Data GothicR  NC", size=25)
    customfont1= tkFont.Font(Adminlogin,family="JLS Data GothicR  NC", size=20)
    titlelabel=Label(Adminlogin,text='Admin Login', font=customfont, bg='black', fg='white')
    usernamelabel=Label(Adminlogin,text='Username:', font=customfontx, bg='black', fg='white')
    usernameentry=Entry(Adminlogin)
    passwordlabel=Label(Adminlogin, text='Password:', font=customfontx, bg='black',fg='white')
    passwordentry=Entry(Adminlogin,show='*')
    okbutton=Button(Adminlogin,text='Login', bg='black', fg='white',
font=customfont1,command=lambda:confirmaccountadmin(usernameentry.get(),passwordentry.get()))
    titlelabel.grid(row=0,columnspan=6)
    usernamelabel.grid(row=1,column=0,sticky=W)
    usernameentry.grid(row=1,column=1,sticky=E)
    passwordlabel.grid(row=2,column=0,sticky=W)
    passwordentry.grid(row=2,column=1,sticky=E)
    okbutton.grid(row=4,columnspan=6)
    Adminlogin.configure(bg='black')
    Adminlogin.mainloop()
def confirmaccountadmin(un,pw):
    f=open('admin1.dat','rb')
    flag=0
    try:
        while True:
            x=pickle.load(f)
            if x.username==un and x.password==pw:
                Adminlogin.destroy()
                f.close()
                profilepageadmin(un)
                flag=1
                break
            elif x.username!=un and x.password==pw:
                f.close()
                tkMessageBox.showinfo('Error','Incorrect Username')
                flag=1
                break
            elif x.password!=pw and x.username==un:
                f.close()
                tkMessageBox.showinfo('Error','Incorrect Password')
                flag=1
                break
            else:
```

```
                pass
        except EOFError:
            tkMessageBox.showinfo('Error','Server Does Not Recognize This User.')
            f.close()
        if flag==0:
            print 'Does Not Recognize This User.'
def printall():
    f=open('customer1.dat','rb')
    printcus=Tk()
    customfont=tkFont.Font(printcus,family="JLS Data GothicR  NC", size=25)
    cfont=tkFont.Font(printcus,family="JLS Data GothicR  NC", size=40)
    try:
        while True:
            sum1=0
            sum2=0
            y=pickle.load(f)
            tlabel= Label(printcus,text="Customer Details",font=cfont,fg='white',bg='black')
            namelabel= Label(printcus,text='Name:'+str(y.username),font=customfont,fg='white',bg='black')
            for i in y.co2em.keys():
                sum1+=y.co2em[i]
                sum2+=y.fuelexp[i]
            sum1label=Label(printcus,text='Total CO2 Emissions Throughout The
Year:'+str(sum1),font=customfont,fg='white',bg='black')
            sum2label=Label(printcus,text='Total Expenditure For The
Year:'+str(sum2),font=customfont,fg='white',bg='black')
            tlabel.pack()
            namelabel.pack()
            sum1label.pack()
            sum2label.pack()
    except EOFError:
        pass
    printcus.configure(bg='black')
    printcus.mainloop()
    f.close()
def delwindow():
    deletecus=Toplevel()
    customfont=tkFont.Font(deletecus,family="JLS Data GothicR  NC", size=20)
    customfontl= tkFont.Font(deletecus,family="JLS Data GothicR  NC", size=50)
    title1=Label(deletecus,text='Delete Customer',fg='white',bg='black',font=customfontl)
    who=Label(deletecus,text='Enter The Customer You Wish To Delete:',font=customfont,bg='black',fg='white')
```

```
    delentry=Entry(deletecus)

delbutton=Button(deletecus,text='Delete',fg='white',bg='black',font=customfont,command=lambda:deletecustomer(str(del
entry.get())))
    title1.grid(row=0, columnspan=6)
    who.grid(row=1,column=0)
    delentry.grid(row=1,column=1)
    delbutton.grid(row=2, columnspan=6)
    deletecus.configure(bg='black')
    deletecus.mainloop()
def deletecustomer(who):
    f=open('customer1.dat','rb')
    g=open('temp1.dat' ,'ab')
    try:
        while True:
            y=pickle.load(f)
            if y.username==who:
                pass
                tkMessageBox.showinfo('Delete Successful','The Customer Has Been Successfully Deleted From The
Database.')
            else:
                pickle.dump(y,g)
    except EOFError:
        pass
    f.close()
    g.close()
    os.remove('customer1.dat' )
    os.rename('temp1.dat','customer1.dat')
def profilepageadmin(un):
    global profile
    profile=Toplevel()
    intro=str(un)
    customfontxl= tkFont.Font(profile,family="JLS Data GothicR  NC", size=30)
    customfontl= tkFont.Font(profile,family="JLS Data GothicR  NC", size=12)
    customfonts= tkFont.Font(profile,family="JLS Data GothicR  NC", size=15)
    welcome=Label(profile,text='Welcome, '+intro, bg='black', fg='white', font=customfontxl)
    title1=Label(profile,text="Display All Customers' Details", bg='black', fg='white', font=customfontxl)
    title2=Label(profile,text="Delete A Customer", bg='black', fg='white', font=customfontxl)
    title3=Label(profile,text='Display Admin Details', bg='black', fg='white', font=customfontxl)
    button1=Button(profile,text='Diplay Now', fg='white', bg='black', font=customfontl, command=lambda:printall())
```

```
    button2=Button(profile,text='Delete Now', fg='white', bg='black', font=customfontl,command=lambda:delwindow())
    button3=Button(profile,text='Display Now', bg='black', fg='white',
font=customfontl,command=lambda:printadmin(un))
    button4=Button(profile,text='Quit', fg='white', bg='black', font=customfonts, command=quit1)
    Img1=PhotoImage(file='ic.gif')
    labelimg1=Label(profile,image=Img1,bg='black',font=customfonts, fg='white')
    labelimg1.image =Img1
    Img2=PhotoImage(file='id.gif')
    labelimg2=Label(profile,image=Img2,bg='black',font=customfonts, fg='white')
    labelimg2.image =Img2
    Img3=PhotoImage(file='ia.gif')
    labelimg3=Label(profile,image=Img3,bg='black',font=customfonts, fg='white')
    labelimg3.image =Img3
    welcome.grid(row=0,columnspan=6)
    title1.grid(row=2,column=1,sticky=W)
    button1.grid(row=2,column=2,sticky=E)
    labelimg1.grid(row=2,column=0)
    title2.grid(row=3,column=1,sticky=W)
    button2.grid(row=3,column=2,sticky=E)
    labelimg2.grid(row=3,column=0)
    title3.grid(row=4,column=1,sticky=W)
    button3.grid(row=4,column=2,sticky=E)
    labelimg3.grid(row=4,column=0)
    button4.grid(row=6,columnspan=6)
    profile.configure(bg='black')
def quit1():
    profile.destroy()
    tkMessageBox.showinfo('Quit The Fuel App','Thank You For Using The Fuel App!')
def quit2(un):
    confirmation.destroy()
    profilepagecus(un)
def quit3(un):
    confirmation.destroy()
    profilepageadmin(un)
def createaccountcus(username,password,fuelcapacity,fuelamt):
    if str(fuelcapacity).isdigit() and str(fuelamt).isdigit():
        if int(fuelcapacity)>=int(fuelamt):
            f=open('customer1.dat', 'ab')
            signup.destroy()
            x=customer()
```

```
            x.assigndatacus(username,password,fuelcapacity,fuelamt)
            pickle.dump(x,f)
            f.close()
            global confirmation
            confirmation=Toplevel()
            customfont= tkFont.Font(confirmation,family="JLS Data GothicR  NC", size=25)
            message=Label(confirmation,text='Congratulations!',font=customfont,fg='white',bg='black')
            confirm=Button(confirmation,text='Go to profile',
font=customfont,fg='white',bg='black',comman=lambda:quit2(username))
            message.pack(side=TOP,fill=X)
            confirm.pack(side=TOP)
            confirmation.configure(bg='black')
            confirmation.mainloop()
        else:
            tkMessageBox.showinfo('Error','Your current amount fuel exceeds your full capacity')
    else:
        tkMessageBox.showinfo('Error','Ensure you have entered numerical values for capacity and amount')
def profilepagecus(un):
    global profile
    profile=Toplevel()
    intro=str(un)
    customfontxl= tkFont.Font(profile,family="JLS Data GothicR  NC", size=30)
    customfontl= tkFont.Font(profile,family="JLS Data GothicR  NC", size=12)
    customfonts= tkFont.Font(profile,family="JLS Data GothicR  NC", size=15)
    welcome=Label(profile,text='Welcome, '+intro, bg='black', fg='white', font=customfontxl)
    title1=Label(profile,text='CO2 emmisions', bg='black', fg='white', font=customfontxl)
    title2=Label(profile,text='Expenditure', bg='black', fg='white', font=customfontxl)
    title3=Label(profile,text='Fill tank', bg='black', fg='white', font=customfontxl)
    title4=Label(profile,text='Display all data', bg='black', fg='white', font=customfontxl)
    title5=Label(profile,text='Locate station',bg='black',fg='white',font=customfontxl)
    title6=Label(profile,text='Edit',bg='black',fg='white',font=customfontxl)
    label1=Label(profile,text='Check Your CO2 Emissions over the year', bg='black', fg='white', font=customfonts)
    button1=Button(profile,text='Check Now', fg='white', bg='black', font=customfontl,command=lambda:co2emm(un))
    Img1=PhotoImage(file='co2 2.gif')
    labelimg1=Label(profile,image=Img1,bg='black',font=customfonts, fg='white')
    labelimg1.image =Img1
    label2=Label(profile,text="Check your fuel expenditure over the year", bg='black', fg='white', font=customfonts)
    button2=Button(profile,text='Check Now', bg='black', fg='white',
font=customfontl,command=lambda:fuelexppage(un))
    Img2=PhotoImage(file='paisa 3.gif')
```

```
    labelimg2=Label(profile,image=Img2,bg='black',font=customfonts, fg='white')
    labelimg2.image =Img2
    label3=Label(profile,text="Refuel your tank", bg='black', fg='white', font=customfonts)
    button3=Button(profile,text='Fuel Up', bg='black', fg='white', font=customfontl,command=lambda:purchase(un))
    Img3=PhotoImage(file="fuel dial.gif")
    labelimg3=Label(profile,image=Img3,bg='black',font=customfonts, fg='white')
    labelimg3.image =Img3
    label5=Label(profile,text='Locate the nearest available fuel station in your
emirate',bg='black',fg='white',font=customfontl)
    button5=Button(profile,text='Locate',bg='black',fg='white',font=customfontl,command=lambda:locate())
    Img5=PhotoImage(file='car battery.gif')
    labelimg5=Label(profile,image=Img5,bg='black',fg='white')
    labelimg5.image=Img5
    welcome.grid(row=0,columnspan=6)
    title1.grid(row=1,column=1,sticky=W)
    label1.grid(row=2,column=1,sticky=W)
    labelimg1.grid(row=1,rowspan=2,column=0)
    button1.grid(row=1,column=2)
    title2.grid(row=3,column=1,sticky=W)
    label2.grid(row=4,column=1,sticky=W)
    labelimg2.grid(row=3,rowspan=2,column=0)
    button2.grid(row=3,column=2)
    title3.grid(row=1,column=4,sticky=W)
    label3.grid(row=2,column=4,sticky=W)
    labelimg3.grid(row=1,rowspan=2,column=3)
    button3.grid(row=1,column=5,padx=20)
    title5.grid(row=3,column=4,sticky=W)
    label5.grid(row=4,column=4,sticky=W)
    labelimg5.grid(row=3,rowspan=2,column=3)
    button5.grid(row=3,column=5,sticky=W,padx=20)
    profile.configure(bg='black')
def purchase(un):
    profile.destroy()
    global purchase
    purchase=Tk()
    customfont= tkFont.Font(purchase,family="JLS Data GothicR  NC", size=30)
    title=Label(purchase,text='FUEL UP NOW!', bg='darkgreen', fg='white',font=customfont)
    fuelamtlabel=Label(purchase,text='Enter the amt of fuel left in tank', bg='darkgreen', fg='white',font=customfont)
    fuelamt=Entry(purchase)
```

```
    Filluplabel=Label(purchase,text='Enter the amount of fuel you wish to fill', bg='darkgreen',
fg='white',font=customfont)
    fillup=Entry(purchase)
    monthlabel=Label(purchase,text='Enter the month',bg='darkgreen', fg='white',font=customfont)
    monthentry=Entry(purchase)
    d=StringVar()
    fueltypelabel=Label(purchase,text='Enter the type of fuel', bg='darkgreen', fg='white',font=customfont)
    diesel=Radiobutton(purchase,text='Diesel',variable=d,value='diesel',bg='darkgreen', fg='white',font=customfont)
    fuel=Radiobutton(purchase,text='Fuel', variable=d, value='fuel oil',bg='darkgreen', fg='white',font=customfont)
    naturalgas=Radiobutton(purchase,text='Natural Gas',variable=d, value='natural gas',bg='darkgreen',
fg='white',font=customfont)
    confirm=Button(purchase,text='Done', font=customfont, bg='darkgreen',
fg='white',command=lambda:purchase1(un,fuelamt.get(),fillup.get(),monthentry.get(),'fuel oil'))
    title.grid(row=0,columnspan=2)
    fuelamtlabel.grid(row=2, column=0, sticky=E)
    fuelamt.grid(row=2,column=1,sticky=W)
    Filluplabel.grid(row=3,column=0,sticky=E)
    fillup.grid(row=3,column=1,sticky=W)
    fueltypelabel.grid(row=4,column=0,sticky=E)
    diesel.grid(row=4,column=1,sticky=W)
    fuel.grid(row=5,column=1,sticky=W)
    naturalgas.grid(row=6,column=1,sticky=W)
    monthlabel.grid(row=7,column=0,sticky=E)
    monthentry.grid(row=7,column=1,sticky=W)
    confirm.grid(row=8,columnspan=2,pady=15)
    purchase.configure(bg='darkgreen')
    purchase.mainloop()
def purchase1(name,amt,fill,monthentry,fueltype):#on every purchase this function is called
    if str(amt).isdigit() and str(fill).isdigit():
        if str(monthentry) in
['january','march','may','april','june','july','august','september','october','november','december']:
            purchase.destroy()
            currentamt=int(amt)
            fillup=int(fill)
            month=monthentry
            print fueltype
            cost=calculate(str(fueltype),fillup)
            newexp=monthlyexp(name,month,cost)
            newem=monthlyem(name,month,currentamt,fueltype)
            f=open('customer.dat','rb')
```

```
            try:
                while True:
                    x=pickle.load(f)
                    if x.username==name:
                        newamt=int(x.fuelcapacity)
                        if fillup<(int(x.fuelcapacity)-currentamt) or fillup==(int(x.fuelcapacity)-currentamt):
                            newamt+=fillup
                        else:
                            print 'Enter a value less than your max capacity',x.fuelcapacity
            except EOFError:
                pass
            f.close()
            update(name,newamt,newexp,newem)
            totalbill=cost
            billing=Toplevel()
            customfont= tkFont.Font(billing,family="JLS Data GothicR  NC", size=30)
            title1=Label(billing,text='Item',fg='white',bg='black',font=customfont)
            title2=Label(billing,text='Quantity',fg='white',bg='black',font=customfont)
            title3=Label(billing,text='Price',fg='white',bg='black',font=customfont)
            img=PhotoImage(file='edit info.gif')
            labelimg=Label(billing,image=img,bg='black')
            labelimg.img=img
            label1=Label(billing,text='Fuel',fg='white',bg='black',font=customfont)
            label2=Label(billing,text=str(fill),fg='white',bg='black',font=customfont)
            label3=Label(billing,text=str(cost),fg='white',bg='black',font=customfont)
            title1.grid(row=0,column=0)
            title2.grid(row=0,column=1)
            title3.grid(row=0,column=2)
            label1.grid(row=1,column=0)
            label2.grid(row=1,column=1)
            label3.grid(row=1,column=2)
            labelimg.grid(row=0,rowspan=5,column=3)
            billing.configure(bg='black')
            billing.mainloop()
        else:
            tkMessageBox.showinfo('Error','Ensure you have entered the value for month taking into account case sensitivity as january, march, may, april, june, july, august, september, october, november or december  ')
    else:
        tkMessageBox.showinfo('Error','Ensure you have entered numerical values for current amount and amount to be filled')
```

```
def purchase1(name,amt,fill,monthentry,fueltype):#on every purchase this function is called
    if str(amt).isdigit() and str(fill).isdigit():
        if str(monthentry) in
['january','march','may','april','june','july','august','september','october','november','december']:
            purchase.destroy()
            currentamt=int(amt)
            fillup=int(fill)
            month=monthentry
            print fueltype
            cost=calculate(str(fueltype),fillup)
            newexp=monthlyexp(name,month,cost)
            newem=monthlyem(name,month,currentamt,fueltype)
            f=open('customer.dat','rb')
            try:
                while True:
                    x=pickle.load(f)
                    if x.username==name:
                        newamt=int(x.fuelcapacity)
                        if fillup<(int(x.fuelcapacity)-currentamt) or fillup==(int(x.fuelcapacity)-currentamt):
                            newamt+=fillup
                        else:
                            print 'Enter a value less than your max capacity',x.fuelcapacity
            except EOFError:
                pass
            f.close()
            update(name,newamt,newexp,newem)
            totalbill=cost
            billing=Toplevel()
            customfont= tkFont.Font(billing,family="JLS Data GothicR  NC", size=30)
            title1=Label(billing,text='Item',fg='white',bg='black',font=customfont)
            title2=Label(billing,text='Quantity',fg='white',bg='black',font=customfont)
            title3=Label(billing,text='Price',fg='white',bg='black',font=customfont)
            img=PhotoImage(file='edit info.gif')
            labelimg=Label(billing,image=img,bg='black')
            labelimg.img=img
            label1=Label(billing,text='Fuel',fg='white',bg='black',font=customfont)
            label2=Label(billing,text=str(fill),fg='white',bg='black',font=customfont)
            label3=Label(billing,text=str(cost),fg='white',bg='black',font=customfont)
            title1.grid(row=0,column=0)
            title2.grid(row=0,column=1)
```

```
            title3.grid(row=0,column=2)
            label1.grid(row=1,column=0)
            label2.grid(row=1,column=1)
            label3.grid(row=1,column=2)
            labelimg.grid(row=0,rowspan=5,column=3)
            billing.configure(bg='black')
            billing.mainloop()
        else:
            tkMessageBox.showinfo('Error','Ensure you have entered the value for month taking into account case sensitivity as january, march, may, april, june, july, august, september, october, november or december  ')
    else:
        tkMessageBox.showinfo('Error','Ensure you have entered numerical values for current amount and amount to be filled')
def monthlyexp(name,month,x): #this updates the monthly values of expenditures
    f=open('customer1.dat','rb')
    try:
        while True:
            query=pickle.load(f)
            if name==query.username:
                newexp=query.fuelexp
                for i in newexp:
                    if i==month:
                        newexp[i]+=x
    except EOFError:
                pass
    f.close()
    return newexp
def monthlyem(name, month, currentamt,fueltype):#this updates the monthly values of CO2 emmissions
    f=open('customer1.dat','rb')
    try:
        while True:
            query=pickle.load(f)
            if name==query.username:
                newem=query.co2em
                for i in newem:
                    if i==month:
                        if fueltype=='fuel oil':
                            x=(int(query.fuelamt)-currentamt)*77.4
                        elif fueltype=='diesel':
                            x=(int(query.fuelamt)-currentamt)*74.1
```

```
                    elif fueltype=='natural gas':
                        x=(int(query.fuelamt)-currentamt)*56.1
                    newem[i]+=x
    except EOFError:
        pass
    f.close()
    return newem
def fill(name,fillup,currentamt): #this updates the current value of fuel in the tank
    f=open('customer1.dat','rb')
    try:
        while True:
            x=pickle.load(f)
            if x.username==name:
                newamt=x.fuelcapacity
                if fillup<(int(x.fuelcapacity)-currentamt) or fillup==(int(x.fuelcapacity)-currentamt):
                   newamt+=fillup
                else:
                    print 'Enter a value less than your max capacity',x.fuelcapacity
            else:
                pickle.dump(x,g)
    except EOFError:
        pass
    f.close()
    return newamt
def fuelexppage(name):
    exppage=Toplevel()
    customfont= tkFont.Font(exppage,family="JLS Data GothicR  NC", size=30)
    f=open('customer1.dat','rb')
    try:
        while True:
            x=pickle.load(f)
            if x.username==name:
                exp=x.fuelexp
    except EOFError:
        pass
    f.close()
    l=['january','march','may','april','june','july','august','september','october','november','december']
    month=Label(exppage,text='Month',bg='black',fg='white',font=customfont)
    amtlabel=Label(exppage,text='Amount spent on fuel',bg='black',fg='white',font=customfont)
    month.grid(row=0,column=0)
```

```
    amtlabel.grid(row=0,column=1)
    for i in range(11):
        label=Label(exppage,text=l[i],bg='black',fg='white',font=customfont)
        amt=Label(exppage,text=exp[l[i]],bg='black',fg='white',font=customfont)
        label.grid(row=i+1,column=0)
        amt.grid(row=i+1,column=1)
    exppage.configure(bg='black')
    exppage.mainloop()
def co2emm(name):
    co2emmpage=Toplevel()
    customfont= tkFont.Font(co2emmpage,family="JLS Data GothicR  NC", size=30)
    f=open('customer1.dat','rb')
    try:
        while True:
            x=pickle.load(f)
            if x.username==name:
                exp=x.co2em
    except EOFError:
        pass
    f.close()
    l=['january','march','may','april','june','july','august','september','october','november','december']
    month=Label(co2emmpage,text='Month',bg='black',fg='white',font=customfont)
    amtlabel=Label(co2emmpage,text='Amount of CO2 released',bg='black',fg='white',font=customfont)
    month.grid(row=0,column=0)
    amtlabel.grid(row=0,column=1)
    for i in range(11):
        label=Label(co2emmpage,text=l[i],bg='black',fg='white',font=customfont)
        amt=Label(co2emmpage,text=exp[l[i]],bg='black',fg='white',font=customfont)
        label.grid(row=i+1,column=0)
        amt.grid(row=i+1,column=1)
    co2emmpage.configure(bg='black')
    co2emmpage.mainloop()
def openpage(location):
    if location=='sharjah':
        webbrowser.open('petrol station - Google Maps.html')
    elif location=='dubai':
        webbrowser.open('dubai fuel stations - Google Search')
    elif location=='abu dhabi':
        webbrowser.open('abu dhabi fuel stations - Google Search')
    elif location=='fujairah':
```

```
        webbrowser.open('fujairah fuel stations - Google Search')
    elif location=='umm al quwain':
        webbrowser.open('umm al quwain fuel stations - Google Search')
    elif location=='ras al khaimah':
        webbrowser.open('ras al khaimah fuel stations - Google Search')
    else:
        tkMessageBox.showinfo('Error','Please ensure you have entered a valid value for emirate which includes the following: sharjah, dubai, abu dhabi, umm al quwain, ras al khaimah, ajman and fujairah')
def locate():
    new=Toplevel()
    customfont= tkFont.Font(new,family="JLS Data GothicR  NC", size=30)
    img=PhotoImage(file='pump2.gif')
    label1=Label(new, image=img,bg='black', fg='white')
    label1.img=img
    label2=Label(new,text='Enter the location:  ',bg='black',fg='white',font=customfont)
    location=Entry(new)
    button=Button(new,text='View',bg='black',fg='white',font=customfont, command=lambda:openpage(location.get()))
    label1.grid(column=0,rowspan=3,padx=20,pady=20)
    label2.grid(column=1,row=0)
    location.grid(column=1,row=1)
    button.grid(column=1,row=2,)
    new.configure(bg='black')
    new.mainloop()
def signupcus():
    x=customer()
    root1.destroy()
    global signup
    signup=Toplevel()
    customfonts= tkFont.Font(signup,family="JLS Data GothicR  NC", size=25)
    customfontl= tkFont.Font(signup,family="JLS Data GothicR  NC", size=35)
    title=Label(signup,text='Sign Up',fg='white',bg='black',font=customfontl)
    Namelabel=Label(signup, text='Enter your username', fg='white', bg='black', font=customfonts)
    Passwordlabel=Label(signup, text='Enter your password',fg='white',bg='black',font=customfonts)
    fuelcaplabel=Label(signup, text='Enter your tank max. capacity', fg='white',bg='black',font=customfonts)
    fuelamt=Label(signup, text='Enter your tank current fuel',fg='white',bg='black',font=customfonts)
    Nameentry=Entry(signup)
    passwordentry=Entry(signup,show='*')
    fuelcapentry=Entry(signup)
    fuelamtentry=Entry(signup)
    signupimg=PhotoImage(file='signup page.gif')
```

```
    bglabel=Label(signup, image=signupimg,bg='black')
    bglabel.img=signupimg
    bglabel.grid(rowspan=6,column=2)
    title.grid(row=0,columnspan=2)
    Namelabel.grid(row=1,column=0,sticky=E)
    Passwordlabel.grid(row=2,column=0,sticky=E)
    Nameentry.grid(row=1,column=1)
    passwordentry.grid(row=2,column=1)
    fuelcaplabel.grid(row=3,column=0,sticky=E)
    fuelamt.grid(row=4,column=0,sticky=E)
    fuelcapentry.grid(row=3,column=1)
    fuelamtentry.grid(row=4,column=1)
    enterdata=Button(signup,text="Ok", bg='black', fg='white',
font=customfonts,command=lambda:createaccountcus(Nameentry.get(),passwordentry.get(),fuelcapentry.get(),fuelamtentry.
get()))
    enterdata.grid(row=5,column=0,columnspan=2)
    signup.configure(bg='black')
def calculate(fueltype1,fillup1): #to calculate the price for fuel filled only
    for i in fuelprices:
        print i
        if fueltype1==i:
            x=fillup1*fuelprices[i]
            print i
            return int(x)
def userlogin(): #help
    global Userlogin
    Userlogin=Toplevel()
    root1.destroy()
    customfont= tkFont.Font(Userlogin,family="JLS Data GothicR  NC", size=50)
    usernamelabel=Label(Userlogin,text='Username:', font=customfont, bg='black', fg='white')
    usernameentry=Entry(Userlogin)
    passwordlabel=Label(Userlogin, text='Password:', font=customfont, bg='black',fg='white')
    passwordentry=Entry(Userlogin,show='*')

confirm=Button(Userlogin,text='Done',font=customfont,bg='black',fg='white',command=lambda:confirmaccountcus(usern
ameentry.get(),passwordentry.get()))
    usernamelabel.pack(fill=X)
    usernameentry.pack()
    passwordlabel.pack(fill=X)
    passwordentry.pack()
```

```
    confirm.pack()
    Userlogin.configure(bg='black')
    Userlogin.mainloop()
def mainfuelapp():
    global root1
    root1=Toplevel()
    customfonts= tkFont.Font(family="JLS Data GothicR  NC", size=15)
    customfontl= tkFont.Font(family="JLS Data GothicR  NC", size=50)
    img100=PhotoImage(file='fuelapp.gif')
    img101=PhotoImage(file='fuelstationcoverleft.gif')
    img102=PhotoImage(file= 'fuelstationcoverright.gif')
    labelleft=Label(root1,image=img101, font=customfonts, fg='white', bg='black')
    labelright=Label(root1,image=img102, font=customfonts, fg='white', bg='black')
    label=Label(root1,image=img100, font=customfonts, fg='white', bg='black')
    label2=Label(root1,text='FUEL UP', font=customfontl, fg='white', bg='black')
    adminloginbutton=Button(root1,text='Login as Admin', bg='black',fg='white',
font=customfonts,command=lambda:adminlogin())
    adminsignup=Button(root1,text='Sign up as
Admin',bg='black',fg='white',font=customfonts,command=lambda:adminsignuppage())
    cusloginbutton=Button(root1,text='Login as Customer', bg='black',fg='white',
font=customfonts,command=lambda:userlogin())
    cussignup=Button(root1,text='Sign up as
Customer',bg='black',fg='white',font=customfonts,command=lambda:signupcus())
    labelleft.grid(column=0,rowspan=5)
    label.grid(column=1,row=0)
    label2.grid(row=0,column=2)
    adminloginbutton.grid(row=1,column=1,columnspan=2)
    adminsignup.grid(row=2,column=1,columnspan=2)
    cusloginbutton.grid(row=3,column=1,columnspan=2)
    cussignup.grid(row=4,column=1,columnspan=2)
    labelright.grid(column=3,row=0,rowspan=5)
    root1.configure(bg='black')
    root1.mainloop()
class Item:
    def itfn(self,itemno,itemname,itemcost,itemstock,itemimage):
            self.itemno=int(itemno)
            self.itemname=itemname
            self.itemcost=int(itemcost)
            self.itemstock=int(itemstock)
            self.itemsold=0
```

```
            self.itemgain=self.itemcost*self.itemsold
            self.itemimage=itemimage
class newcustomer:
    def creatingcustomer(self,name,mail,carn):
                self.name=name
                self.cemail=mail
                self.carn=carn
                self.tempbill=["Your Bill"]
def customerentry(name,mail,carn):
            f=open("Customer.dat","rb")
            n=carn
            try:
                while True:
                        obj=pickle.load(f)
                        if obj.carn!= n:
                            pass
                        elif obj.carn==n:
                            f.close()
                            return 0
            except EOFError:
                pass
            f=open("Customer.dat","ab")
            newvisitor=newcustomer()
            newvisitor.creatingcustomer(name,mail,carn)
            pickle.dump(newvisitor,f)
            server = smtplib.SMTP('smtp.gmail.com', 587)
            server.starttls()
            server.login("CstoreFA@gmail.com", "fuelapp!")
            msg = """
            Hello!
            You have completed your Sign up with Fuel App C-Store.
            Hope you enjoy our Services.
            regards,
            C-Store Admin"""
            server.sendmail("CstoreFA@gmail.com", mail, msg)
            f.close()
            return 1
class customers:
    def customermain(s):
        s.cmw=Toplevel()
```

```
        s.cmw.configure(background="brown")
        s.cmw.title("Welcome to C-Store!")
        BGEntry=PhotoImage(file='BG.gif')
        captcha1=PhotoImage(file='Captcha1.gif')
        captcha2=PhotoImage(file='Captcha2.gif')
        captcha3=PhotoImage(file='Captcha3.gif')
        s.f2=Frame(s.cmw)
        label1=Label(s.f2,image=BGEntry)
        label1.pack()
        s.f2.pack()
        s.cmw.geometry("1080x641")
        Name=Label(s.cmw,text="Name",bg="#AEB6BF",font="Narkisim
25",fg="white").place(x=160,y=260,width=159,height=40)
        EMail=Label(s.cmw,text="E-Mail",bg="#AEB6BF",font="Narkisim
25",fg="white").place(x=160,y=304,width=159,height=40)
        Carnum=Label(s.cmw,text="Car Number",bg="#AEB6BF",font="Narkisim
25",fg="white").place(x=160,y=348,width=159,height=40)
        Captchainstruct=Label(s.cmw,text="Enter the text you see in the image
below",bg="#AEB6BF",font="Narkisim 11",fg="white").place(x=160,y=400,width=250,height=10)
        Lemail=Label(s.cmw,text="Email",bg="#AEB6BF",font="Narkisim
25",fg="white").place(x=690,y=250,width=159,height=40)
        LCnum=Label(s.cmw,text="Car Number",bg="#AEB6BF",font="Narkisim
25",fg="white").place(x=690,y=350,width=159,height=40)
        s.i=random.randint(0,2)
        if s.i==0:

captchapic=Label(s.cmw,image=captcha1,bg="#AEB6BF",fg="white").place(x=160,y=415,width=200,height=70)
        elif s.i==1:

captchapic=Label(s.cmw,image=captcha2,bg="#AEB6BF",fg="white").place(x=160,y=415,width=200,height=70)
        elif s.i==2:

captchapic=Label(s.cmw,image=captcha3,bg="#AEB6BF",fg="white").place(x=160,y=415,width=200,height=70)
        s.newname=Entry(s.cmw,bg="#EDBB99",font="Narkisim 25",fg="white")
        s.newemail=Entry(s.cmw,bg="#EDBB99",font="Narkisim 25",fg="white")
        s.newcarnum=Entry(s.cmw,bg="#EDBB99",font="Narkisim 25",fg="white")
        s.captchae=Entry(s.cmw,bg="#EDBB99",font="Narkisim 25",fg="white")
        s.LemailE=Entry(s.cmw,bg="#EDBB99",font="Narkisim 25",fg="white")
        s.LCnumE=Entry(s.cmw,bg="#EDBB99",show="*",font="Narkisim 25",fg="white")
```

```
        NewCustomer=Button(s.cmw,text="Sign Up",command=s.Sign_up,bg="#808080",font="Narkisim
35",fg="white").place(x=220,y=496,width=200,height=60)
        s.newname.place(x=340,y=260,width=159,height=40)
        s.newemail.place(x=340,y=304,width=159,height=40)
        s.newcarnum.place(x=340,y=348,width=159,height=40)
        s.captchae.place(x=379,y=430,width=120,height=40)
        s.LemailE.place(x=680,y=295,width=180,height=40)
        s.LCnumE.place(x=680,y=395,width=180,height=40)
        Login=Button(s.cmw,text="Login",command=s.Login,bg="#808080",font="Narkisim
35",fg="white").place(x=670,y=476,width=200,height=60)
        s.cmw.mainloop()##############################################
    def confirmclear(s):##############################################
        s.displayimagelble.destroy()
        s.displayitemname.destroy()
        s.displayitemcost.destroy()
        s.displayitemstock.destroy()
        s.plus.destroy()
        s.qtylabel.destroy()
        s.minus.destroy()
        s.atc.destroy()
        s.qtylabel.destroy()
        for i in s.button:
            i.destroy()
    def Sign_up(s):
        x=s.captchae.get()
        name=s.newname.get()
        email=s.newemail.get()
        carn=s.newcarnum.get()
        if s.i==0 and x=="W68HP":
            if customerentry(name,email,carn)==0:
                Error=Label(s.cmw,text="Car Number Already exists",bg="#AEB6BF",font="Narkisim
11",fg="#E74C3C").place(x=160,y=240,width=250,height=10)
            else:
                Success=Label(s.cmw,text="Sign up was successful",bg="#AEB6BF",font="Narkisim
11",fg="#2ECC71").place(x=160,y=240,width=250,height=10)
        elif s.i==1 and x=="EBFQY":
            if customerentry(name,email,carn)==0:
                Error=Label(s.cmw,text="Car Number Already exists",bg="#AEB6BF",font="Narkisim
11",fg="#E74C3C").place(x=160,y=240,width=250,height=10)
            else:
```

```
                Success=Label(s.cmw,text="Sign up was successful",bg="#AEB6BF",font="Narkisim
11",fg="#2ECC71").place(x=160,y=240,width=250,height=10)
        elif s.i==2 and x=="NSF":
            if customerentry(name,email,carn)==0:
                Error=Label(s.cmw,text="Car Number Already exists",bg="#AEB6BF",font="Narkisim
11",fg="#E74C3C").place(x=160,y=240,width=250,height=10)
            else:
                Success=Label(s.cmw,text="Sign up was successful",bg="#AEB6BF",font="Narkisim
11",fg="#2ECC71").place(x=160,y=240,width=250,height=10)
        else:
            Error=Label(s.cmw,text="Wrong Captcha:Retry",bg="#AEB6BF",font="Narkisim
11",fg="#E74C3C").place(x=560,y=240,width=250,height=10)
    def Login(s):
            f=open("Customer.dat","rb")
            n=s.LemailE.get()
            x=s.LCnumE.get()
            try:
                while True:
                        obj=pickle.load(f)
                        if obj.carn!= x:
                            pass
                        elif obj.carn==x and obj.cemail==n:
                            s.currentcustomer=obj
                            obj.tempbill=["Your Bill"]
                            s.customerdash()
                else:
                    Error=Label(s.admw,text="Invalid Entry",bg="#010F16",font="Narkisim
11",fg="#E74C3C").place(x=620,y=140,width=250,height=10)
            except EOFError:
                    f.close()
    def ads(s):
        adscount=random.randint(1,7)
        s.f1=Frame(s.custdash)
        if adscount==1:
            dash=PhotoImage(file='MCAD.gif')
        elif adscount==2:
            dash=PhotoImage(file='MBAD.gif')
        elif adscount==3:
            dash=PhotoImage(file='EM.gif')
        elif adscount==4:
```

```
            dash=PhotoImage(file='DR1.gif')
        elif adscount==5:
            dash=PhotoImage(file='CC.gif')
        elif adscount==6:
            dash=PhotoImage(file='AD.gif')
        elif adscount==7:
            dash=PhotoImage(file='SS.gif')
        label1=Label(s.f1,image=dash)
        label1.image=dash
        label2=Label(s.f1,text="toggle ads",font="Narkisim 12",fg="#BFC9CA")
        label1.pack()
        label2.pack()
        s.f1.pack()
        s.f1.place(x=32,y=300,width=266,height=240)
    def customerdash(s):
        s.custdash=Toplevel()
        s.custdash.configure(background="black")
        s.custdash.title("Dashboard")
        s.custdash.geometry("1080x560")
        s.clearflag=0
        dash=PhotoImage(file='customerdash.gif')
        s.f=Frame(s.custdash)
        label1=Label(s.f,image=dash)
        label1.image=dash
        label1.pack()
        s.f.pack()
        s.clearitem=1
        s.snackbutton=Button(s.custdash,text="Snacks",command=s.snackselect,font="Narkisim
24",bg="black",activebackground="white",activeforeground="red",fg="#F4D03F",relief="flat")
        s.juicebutton=Button(s.custdash,text="Juices",command=s.juiceselect,font="Narkisim
24",bg="black",activebackground="white",activeforeground="red",fg="#3498DB",relief="flat")
        s.etbutton=Button(s.custdash,text="Emergency tools",command=s.etoolsselect,font="Narkisim
20",bg="black",activebackground="white",fg="#E74C3C",activeforeground="blue",relief="flat")
        s.cpurchase=Button(s.custdash,text=" Confirm ",command=s.confirmation,font="Narkisim
24",bg="black",activebackground="white",fg="#AF7AC5",activeforeground="red",relief="flat")
        s.payment=Button(s.custdash,text=" Payment ",command=s.Payment,font="Narkisim
24",bg="black",activebackground="red",fg="#58D68D",activeforeground="yellow",relief="flat")
        s.snackbutton.place(x=37,y=54,height=40,width=190)
        s.juicebutton.place(x=37,y=143,height=40,width=190)
        s.etbutton.place(x=37,y=233,height=40,width=190)
```

```
        s.cpurchase.place(x=440,y=140,height=40,width=190)
        s.payment.place(x=846,y=140,height=40,width=190)
        s.ads()
    def snackselect(s):
        try:
            for i in s.button:
                i.destroy
        except AttributeError :
            pass
        s.select=0
        s.displayitems()
    def juiceselect(s):
        try:
            for i in s.button:
                i.destroy()
        except AttributeError :
            pass
        s.select=1
        s.displayitems()
    def etoolsselect(s):
        try:
            for i in s.button:
                i.destroy()
        except AttributeError :
            pass
        s.select=2
        s. displayitems()

    def displayitems(s):
        if s.select==0:
            f=open("Snacks.dat","rb")
        elif s.select==1:
            f=open("Juices.dat","rb")
        elif s.select==2:
            f=open("Etools.dat","rb")
        s.itemdetails=[]
        s.button=[]
        count=0
        try:
            while True:
```

```
                    x=pickle.load(f)
                    s.itemdetails.append(x)
                    txt=x.itemname
                    s.button.append(Button(s.custdash,bg="white",anchor='w',text=txt,command=lambda
count=count:s.displayinfo(count),justify="center",relief="flat",fg="#607D8B",font="Narkisim 18"))
                    s.button[count].place(x=320,y=(220+(count*25)),width=175,height=24)
                    count+=1
            except EOFError:
                f.close()
                count=1
        def displayinfo(s,i):
            if s.clearitem==2:
                s.displayimagelble.destroy()
                s.displayitemname.destroy()
                s.displayitemcost.destroy()
                s.displayitemstock.destroy()
                s.plus.destroy()
                s.qtylabel.destroy()
                s.minus.destroy()
                s.atc.destroy()
            s.qty=1
            s.cart=[]
            s.insertitem=""
            s.noofitems=1
            itemdisplay=PhotoImage(file=s.itemdetails[i].itemimage)
            s.qtyflag=0
            s.displayimagelble=Label(s.custdash,image=itemdisplay,bg="#050518",font="Narkisim 36",fg="#22D8C0")
            s.displayimagelble.image=itemdisplay
            s.displayitemname=Label(s.custdash,text=s.itemdetails[i].itemname,bg="white",font="Narkisim
16",fg="#1A237E")
            s.displayitemcost=Label(s.custdash,text="AED: "+str(s.itemdetails[i].itemcost),bg="white",font="Narkisim
16",fg="#1A237E")
            if s.itemdetails[i].itemstock==0:
                s.displayitemstock=Label(s.custdash,text="Out Of Stock",bg="white",font="Narkisim 16",fg="#1A237E")
            elif s.itemdetails[i].itemstock > 10:
                s.displayitemstock=Label(s.custdash,text="Available",bg="white",font="Narkisim 16",fg="#1A237E")
            else:
                s.displayitemstock=Label(s.custdash,text="Few Left",bg="white",font="Narkisim 16",fg="#1A237E")
            s.currentitem=s.itemdetails[i]
            s.clearitem=2
```

```
        s.displayimagelble.place(x=710,y=214,height=200,width=200)
        s.displayitemname.place(x=710,y=445,height=30,width=200)
        s.displayitemcost.place(x=710,y=475,height=30,width=200)
        s.displayitemstock.place(x=710,y=505,height=30,width=200)
        s.addtocart()
    def addqty(s):
        if s.qty>=12:
            pass
        elif s.currentitem.itemstock > 0:
            s.qtylabel.destroy()
            if s.qtyflag==1:
                                s.qtylabel.destroy()
            s.qty+=1
            s.qtylabel=Label(s.custdash,text=s.qty,bg="white",fg="#2C3E50",font="Narkisim 38",relief="flat")
            s.qtylabel.place(x=950,y=350,height=50,width=50)
            s.qtyflag=1
    def subqty(s):
        if s.qty==1:
            pass
        else:
            s.qtylabel.destroy()
            if s.qtyflag==1:
                s.qtylabel.destroy()
            s.qty-=1
            s.qtylabel=Label(s.custdash,text=s.qty,bg="white",fg="#2C3E50",font="Narkisim 38",relief="flat")
            s.qtylabel.place(x=950,y=350,height=50,width=50)
            s.qtyflag=1
    def addtocart(s):
        s.qtylabel=Label(s.custdash,text=s.qty,bg="white",fg="#2C3E50",font="Narkisim 38",relief="flat")
        s.plus=Button(s.custdash,text="+",command=s.addqty,bg="white",fg="#16A085",font="Narkisim
28",relief="groove")
        s.minus=Button(s.custdash,text="-",bg="white",command=s.subqty,fg="#EC7063",font="Narkisim
50",relief="groove")
        s.atc=Button(s.custdash,text=" Purchase ",command=s.Purchase,bg="white",fg="#F39C12",font="Narkisim
11",relief="groove")
        s.plus.place(x=960,y=314,height=30,width=30)
        s.qtylabel.place(x=950,y=350,height=50,width=50)
        s.minus.place(x=960,y=405,height=30,width=30)
        s.atc.place(x=930,y=455,height=30,width=100)
    def Purchase(s):
```

```
        s.confirmlabel=Label(s.custdash,text=" Please confirm ",bg="white",fg="#2C3E50",font="Narkisim
11",relief="flat")
        s.confirmlabel.place(x=480,y=110,height=14,width=100)
    def confirmation(s):
        s.confirmclear()
        itemdisplay=PhotoImage(file=s.currentitem.itemimage)
        s.displayimagelble=Label(s.custdash,image=itemdisplay,bg="#050518",font="Narkisim 36",fg="#22D8C0")
        s.displayimagelble.image=itemdisplay
        s.displayitemname=Label(s.custdash,text=s.currentitem.itemname,bg="white",font="Narkisim
16",fg="#1A237E")
        s.displayqty=Label(s.custdash,text="Quantity :"+str(s.qty),bg="white",font="Narkisim 16",fg="#1A237E")
        s.tcost=s.currentitem.itemcost*s.qty
        s.displayitemcost=Label(s.custdash,text="AED: "+str(s.tcost),bg="white",font="Narkisim 16",fg="#1A237E")
        s.buy=Button(s.custdash,text=" Buy Now ",command=s.buyitem,bg="white",fg="#F39C12",font="Narkisim
11",relief="groove")
        s.cancelb=Button(s.custdash,text="Cancel",command=s.clearbuy,bg="white",fg="red",font="Narkisim
11",relief="groove")
        s.buy.place(x=810,y=414,height=30,width=200)
        s.cancelb.place(x=810,y=454,height=30,width=200)
        s.displayimagelble.place(x=510,y=214,height=200,width=200)
        s.displayitemname.place(x=510,y=445,height=30,width=200)
        s.displayitemcost.place(x=510,y=475,height=30,width=200)
        s.displayqty.place(x=510,y=505,height=30,width=200)
    def clearbuy(s):
        s.displayimagelble.destroy()
        s.displayitemname.destroy()
        s.displayitemcost.destroy()
        s.displayqty.destroy()
    def buyitem(s):
        if s.select==0:
            f=open("Snacks.dat","rb")
        elif s.select==1:
            f=open("Juices.dat","rb")
        elif s.select==2:
            f=open("Etools.dat","rb")
        g=open("Temp.dat","wb")
        try:
            while True:
                    H=pickle.load(f)
                    if H.itemname.upper()!= s.currentitem.itemname.upper():
```

```
                        pickle.dump(H,g)
                    elif H.itemname.upper()==s.currentitem.itemname.upper():
                        s.qty=s.tcost/s.currentitem.itemcost
                        H.itemsold+=s.qty
                        H.itemstock-=s.qty
                        x=s.qty*H.itemcost
                        H.itemgain+=x
                        pickle.dump(H,g)
                        s.confirmlabel=Label(s.custdash,text=" Please Proceed with Payment
",bg="white",fg="#2C3E50",font="Narkisim 8",relief="flat")
                        s.confirmlabel.place(x=846,y=110,height=20,width=200)
            except EOFError:
                f.close()
                g.close()
            if s.select==0:
                os.remove("Snacks.dat")
                os.rename("Temp.dat","Snacks.dat")
            elif s.select==1:
                os.remove("Juices.dat")
                os.rename("Temp.dat","Juices.dat")
            elif s.select==2:
                os.remove("Etools.dat")
                os.rename("Temp.dat","Etools.dat")


        def Payment(s):
            s.clearbuy()
            s.cancelb.destroy()
            s.buy.destroy()
            s.cod=Button(s.custdash,text=" Cash On Delivery
",command=s.deliverytime,bg="white",fg="#F39C12",font="Narkisim 21",relief="groove")
            s.cod.place(x=330,y=340,height=40,width=400)
        def deliverytime(s):
            #s.currentitem,s.currentcustomer
            s.cod.destroy()
            carno=str(s.currentcustomer.carn)
            itemdet=s.currentitem.itemname+"  "+str(s.currentitem.itemcost)+"AED"
            qty=s.qty
            current_time = str(datetime.datetime.now().time().isoformat())
            current_date = str(datetime.datetime.now().date().isoformat())
```

```
        L=current_date.split("-")
        cd=""
        for i in range(len(L)-1,-1,-1):
            cd+=L[i]+"-"
        ct=current_time[:5]
        f=Frame(s.custdash)
        s.token=Label(s.custdash,text="Token",bg="#AED6F1",font="Narkisim 26",fg="#1A237E")
        s.carnum=Label(s.custdash,text="Car number: "+carno,bg="#AED6F1",font="Narkisim 16",fg="#1A237E")
        s.itemd=Label(s.custdash,text=itemdet,bg="#AED6F1",font="Narkisim 16",fg="#1A237E")
        s.quant=Label(s.custdash,text="Quantity: "+str(qty),bg="#AED6F1",font="Narkisim 16",fg="#1A237E")
        s.ctime=Label(s.custdash,text=ct,bg="#AED6F1",font="Narkisim 16",fg="#1A237E")
        s.cdate=Label(s.custdash,text=cd[:-1],bg="#AED6F1",font="Narkisim 16",fg="#1A237E")
        dash=PhotoImage(file='cart.gif')
        label1=Label(f,image=dash)
        label1.image=dash
        label1.pack()
        f.place(x=530,y=290)
        s.token.place(x=560,y=330,height=30,width=200)
        s.carnum.place(x=560,y=360,height=30,width=200)
        s.itemd.place(x=560,y=390,height=30,width=200)
        s.quant.place(x=560,y=420,height=30,width=200)
        s.ctime.place(x=560,y=450,height=30,width=200)
        s.cdate.place(x=560,y=480,height=30,width=200)
        s.delivery=Label(s.custdash,text="Thank You For Using C-Store",bg="white",font="Narkisim
36",fg="#F39C12")
        s.delivery.place(x=360,y=240,height=40,width=600)
class newadmin:
    def creatingadmin(self,name,mail,adpass):
                self.name=name
                self.ademail=mail
                self.adpass=adpass
def adminentry(name,mail,adpass):
            f=open("Admin.dat","rb")
            n=adpass
            try:
                while True:
                        obj=pickle.load(f)
                        if obj.ademail!=mail:
                            pass
                        elif obj.ademail==mail :
```

```
                        f.close()
                        return 0
            except EOFError:
                f.close()
            f=open("Admin.dat","ab")
            newad=newadmin()
            newad.creatingadmin(name,mail,adpass)
            pickle.dump(newad,f)
            server = smtplib.SMTP('smtp.gmail.com', 587)
            server.starttls()
            server.login("CstoreFA@gmail.com", "fuelapp!")
            msg = """
            Hello!
            You have completed your Sign up with Fuel App C-Store.
            Hope you lead our company to success.
            regards,
            C-Store Boss"""
            server.sendmail("CstoreFA@gmail.com", mail, msg)
            f.close()
            return 1
class Item:
    def itfn(self,itemno,itemname,itemcost,itemstock,itemimage):
            self.itemno=int(itemno)
            self.itemname=itemname
            self.itemcost=int(itemcost)
            self.itemstock=int(itemstock)
            self.itemsold=0
            self.itemgain=self.itemcost*self.itemsold
            self.itemimage=itemimage
    def snfn(self,itno,itn,itc,itsk):
            it=open("Snacks.dat","ab")
            x=Item()
            x.itfn(itno,itn,itc,itsk)
            pickle.dump(x,it)
            it.close()
    def Jfn(self,itno,itn,itc,itsk):
            it=open("Juices.dat","ab")
            x=Item()
            x.itfn(itno,itn,itc,itsk)
            pickle.dump(x,it)
```

```
                it.close()
class Admin:
    def cleardash(s):
        if s.cflag=="addf":
            s.addhead.destroy()
            s.itno.destroy()
            s.itemname.destroy()
            s.itemcost.destroy()
            s.itemstock.destroy()
            s.itnoe.destroy()
            s.itemnamee.destroy()
            s.itemcoste.destroy()
            s.itemstocke.destroy()
            s.imagelble.destroy()
            s.imglink.destroy()
            s.imglinke.destroy()
            s.left1.destroy()
            s.right1.destroy()
            s.checkbutton.destroy()
            s.addbutton.destroy()
        if s.cflag=="adds":
            s.addhead.destroy()
            s.itno.destroy()
            s.itemname.destroy()
            s.itemcost.destroy()
            s.itemstock.destroy()
            s.itnoe.destroy()
            s.itemnamee.destroy()
            s.itemcoste.destroy()
            s.itemstocke.destroy()
            s.imagelble.destroy()
            s.imglink.destroy()
            s.imglinke.destroy()
            s.left1.destroy()
            s.right1.destroy()
            s.checkbutton.destroy()
            s.addbutton.destroy()
            s.status.destroy()
            s.imagelbl.destroy()
        if s.cflag=="disf":
```

```
        for i in s.buttonslist:
            i.destroy()
    if s.cflag=="diss":
        for i in s.buttonslist:
            i.destroy()
        for i in range(s.ccount):
            s.infoframe.destroy()
            s.displayimagelble.destroy()
            s.displayitemname.destroy()
            s.displayitemcost.destroy()
            s.displayitemstock.destroy()
            s.displayitemsold.destroy()
            s.displayitemgain.destroy()
    if s.cflag=="serf":
        s.sertaskbar.destroy()
        s.serbyname.destroy()
        s.serbycost.destroy()
        s.serbystock.destroy()
        s.serbysold.destroy()
        s.serbynamee.destroy()
        s.serbycoste.destroy()
        s.serbystocke.destroy()
        s.serbysolde.destroy()
    if s.cflag=="sers":
        s.sertaskbar.destroy()
        s.serbyname.destroy()
        s.serbycost.destroy()
        s.serbystock.destroy()
        s.serbysold.destroy()
        s.serbynamee.destroy()
        s.serbycoste.destroy()
        s.serbystocke.destroy()
        s.serbysolde.destroy()
        s.status.destroy()
        for i in s.buttonslist:
            i.destroy()
        for i in range(s.ccount):
            s.serimagelble.destroy()
            s.seritemname.desroy()
            s.seritemcost.destroy()
```

```
            s.seritemstock.destroy()
            s.seritemsold.destroy()
            s.seritemgain.destroy()
            s.infoframe.destroy()
    if s.cflag=="updf":
        s.uptaskbar.destroy()
        s.uphead.destroy()
        s.updateitemnamee.destroy()
        s.updateitemnume.destroy()
        s.updateser.destroy()
    if s.cflag=="upds":
        s.uptaskbar.destroy()
        s.uphead.destroy()
        s.updateitemnamee.destroy()
        s.updateitemnume.destroy()
        s.updateser.destroy()
        s.imglinke.destroy()
        s.checkimg.destroy()
        s.displayimagelble.destroy()
        s.updateitemname.destroy()
        s.updateitemcost.destroy()
        s.updateitemstock.destroy()
        s.updateitemsold.destroy()
        s.updateitemgain.destroy()
        s.updateitemnamen.destroy()
        s.updateitemcostn.destroy()
        s.updateitemstockn.destroy()
        s.updateitemsoldn.destroy()
        s.updateitemgainn.destroy()
        s.updateitemnamee.destroy()
        s.updateitemcoste.destroy()
        s.updateitemstocke.destroy()
        s.updateitemsolde.destroy()
        s.updateitemgaine.destroy()
        s.updatebutton.destroy()
        for i in range(s.ccount):
            s.status.destroy()
            s.imagelbl.destroy()
    if s.cflag=="delf":
        s.deltaskbar.destroy()
```

```
            s.delhead.destroy()
            s.delitemnamee.destroy()
            s.delitemnume.destroy()
            s.delser.destroy()
        if s.cflag=="dels":
            s.deltaskbar.destroy()
            s.delhead.destroy()
            s.delitemnamee.destroy()
            s.delitemnume.destroy()
            s.delser.destroy()
            s.displayimagelble.destroy()
            s.updateitemname.destroy()
            s.updateitemcost.destroy()
            s.updateitemstock.destroy()
            s.updateitemsold.destroy()
            s.updateitemgain.destroy()
            s.deletebutton.destroy()
    def adminmain(s):
        s.cflag=""
        s.snackimage=[]
        s.juiceimage=[]
        s.etoolimage=[]
        s.admw=Toplevel()
        s.admw.configure(background="black")
        s.admw.title("Welcome  Admin!")
        BGEntry=PhotoImage(file='Admin1.gif')
        captcha1=PhotoImage(file='Captcha1.gif')
        captcha2=PhotoImage(file='Captcha2.gif')
        captcha3=PhotoImage(file='Captcha3.gif')
        s.f2=Frame(s.admw)
        label1=Label(s.f2,image=BGEntry)
        label1.pack()
        s.f2.pack()
        s.admw.geometry("1080x641")
        Name=Label(s.admw,text="Name",bg="#010F16",font="Narkisim
25",fg="#3498DB").place(x=100,y=180,width=159,height=40)
        EMail=Label(s.admw,text="E-Mail",bg="#010F16",font="Narkisim
25",fg="#3498DB").place(x=100,y=230,width=159,height=40)
        Password=Label(s.admw,text="Password",bg="#010F16",font="Narkisim
25",fg="#3498DB").place(x=100,y=280,width=159,height=40)
```

```
        Captchainstruct=Label(s.admw,text="Enter the text you see in the image
below",bg="#010F16",font="Narkisim 11",fg="#3498DB").place(x=160,y=330,width=250,height=10)
        LEmail=Label(s.admw,text="E-Mail",bg="#010F16",font="Narkisim
25",fg="#3498DB").place(x=690,y=180,width=159,height=40)
        LPass=Label(s.admw,text="Password",bg="#010F16",font="Narkisim
25",fg="#3498DB").place(x=690,y=295,width=159,height=40)
        s.i=random.randint(0,2)
        if s.i==0:

captchapic=Label(s.admw,image=captcha1,bg="#010F16",fg="#3498DB").place(x=190,y=355,width=200,height=70)
        elif s.i==1:

captchapic=Label(s.admw,image=captcha2,bg="#010F16",fg="#3498DB").place(x=190,y=355,width=200,height=70)
        elif s.i==2:

captchapic=Label(s.admw,image=captcha3,bg="#010F16",fg="#3498DB").place(x=190,y=355,width=200,height=70)
        s.newname=Entry(s.admw,bg="#0A5073",font="Narkisim 17",fg="#3498DB")
        s.newemail=Entry(s.admw,bg="#0A5073",font="Narkisim 17",fg="#3498DB")
        s.newpassword=Entry(s.admw,bg="#0A5073",show="*",font="Narkisim 17",fg="#3498DB")
        s.captchae=Entry(s.admw,bg="#0A5073",font="Narkisim 17",fg="#3498DB")
        s.LEmailE=Entry(s.admw,bg="#0A5073",font="Narkisim 17",fg="#3498DB")
        s.LPassE=Entry(s.admw,bg="#0A5073",show="*",font="Narkisim 17",fg="#3498DB")
        NewAdmin=Button(s.admw,text="Sign Up",bg="#0B1056",command=s.Sign_up,font="Narkisim
35",fg="#3498DB").place(x=140,y=496,width=300,height=60)
        s.newname.place(x=270,y=180,width=200,height=40)
        s.newemail.place(x=270,y=230,width=200,height=40)
        s.newpassword.place(x=270,y=280,width=200,height=40)
        s.captchae.place(x=220,y=430,width=140,height=40)
        s.LEmailE.place(x=660,y=215,width=220,height=40)
        s.LPassE.place(x=660,y=335,width=220,height=40)
        Login=Button(s.admw,text="Login",command=s.Login,bg="#0B1056",font="Narkisim
35",fg="#3498DB").place(x=670,y=430,width=200,height=60)
        s.admw.mainloop()
    def Sign_up(s):
        x=s.captchae.get()
        name=s.newname.get()
        email=s.newemail.get()
        adpass=s.newpassword.get()
        if s.i==0 and x=="W68HP":
            if adminentry(name,email,adpass)==0:
```

```
                Error=Label(s.admw,text="Account Already exists",bg="#010F16",font="Narkisim
11",fg="#E74C3C").place(x=120,y=140,width=250,height=10)
            else:
                Success=Label(s.admw,text="Sign up was successful",bg="#010F16",font="Narkisim
11",fg="#2ECC71").place(x=120,y=140,width=250,height=10)
        elif s.i==1 and x=="EBFQY":
            if adminentry(name,email,adpass)==0:
                Error=Label(s.admw,text="Account Already exists",bg="#010F16",font="Narkisim
11",fg="#E74C3C").place(x=120,y=140,width=250,height=10)
            else:
                Success=Label(s.admw,text="Sign up was successful",bg="#010F16",font="Narkisim
11",fg="#2ECC71").place(x=120,y=140,width=250,height=10)
        elif s.i==2 and x=="NSF":
            if adminentry(name,email,adpass)==0:
                Error=Label(s.admw,text="Account Already exists",bg="#010F16",font="Narkisim
11",fg="#E74C3C").place(x=120,y=140,width=250,height=10)
            else:
                Success=Label(s.admw,text="Sign up was successful",bg="#010F16",font="Narkisim
11",fg="#2ECC71").place(x=120,y=140,width=250,height=10)
        else:
            Error=Label(s.admw,text="Wrong Captcha Entry:Retry",bg="#010F16",font="Narkisim
11",fg="#E74C3C").place(x=120,y=140,width=250,height=10)
    def admindash(s):
        s.admdash=Toplevel()
        s.admdash.configure(background="black")
        s.admdash.title("Dashboard")
        s.admdash.geometry("1080x690")
        s.clearflag=0
        dash=PhotoImage(file='Admindash.gif')
        s.f3=Frame(s.admdash)
        label1=Label(s.f3,image=dash)
        label1.pack()
        s.f3.pack()

display=Button(s.admdash,command=s.displaymenu,text="Display",activebackground="black",bg="#010F16",font="Na
rkisim 25",fg="#3498DB",relief="flat").place(y=18,x=15,width=188,height=65)
        add=Button(s.admdash,command=s.additem,text="Add
Item",activebackground="black",bg="#010F16",font="Narkisim
25",fg="#3498DB",relief="flat").place(y=108,x=15,width=188,height=65)
```

```
search=Button(s.admdash,command=s.searchitem,text="Search",activebackground="black",bg="#010F16",font="Narki
sim 25",fg="#3498DB",relief="flat").place(y=198,x=15,width=188,height=65)

update=Button(s.admdash,command=s.updateitems,text="Update",activebackground="black",bg="#010F16",font="Nar
kisim 25",fg="#3498DB",relief="flat").place(y=278,x=15,width=188,height=65)

delete=Button(s.admdash,command=s.deleteitem,text="Delete",activebackground="black",bg="#010F16",font="Narkisi
m 25",fg="#3498DB",relief="flat").place(y=368,x=15,width=188,height=65)

Snacks=Button(s.admdash,text="Snacks",command=s.selectsnack,activebackground="black",bg="black",font="Narkisim
25",fg="#B52529",relief="flat").place(y=530,x=15,width=180,height=30)

Juices=Button(s.admdash,text="Juices",command=s.selectjuice,activebackground="black",bg="black",font="Narkisim
25",fg="#B52529",relief="flat").place(y=585,x=15,width=180,height=30)
        ETools=Button(s.admdash,text="E-
Tools",command=s.selectetool,activebackground="black",bg="black",font="Narkisim
25",fg="#B52529",relief="flat").place(y=640,x=15,width=180,height=30)
        s.f3.pack()
        s.admdash.mainloop()
    def selectsnack(s):
        s.select=0
        select=Label(s.admdash,text="Snacks",bg="#010F16",font="Narkisim
21",fg="#E74C3C").place(x=50,y=460,width=100,height=50)
    def selectjuice(s):
        s.select=1
        select=Label(s.admdash,text="Juices",bg="#010F16",font="Narkisim
21",fg="#E74C3C").place(x=50,y=460,width=100,height=50)
    def selectetool(s):
        s.select=2
        select=Label(s.admdash,text="E-Tools",bg="#010F16",font="Narkisim
21",fg="#E74C3C").place(x=50,y=460,width=100,height=50)
#Add Option
    def additem(s):
        s.cleardash()
        s.cflag="addf"
        left=PhotoImage(file='Left1.gif')
        s.left1=Frame(s.admdash,bg="black")
        label1=Label(s.left1,image=left)
        label1.image = left
```

```
        label1.pack()
        s.left1.place(x=225,y=100)
        right=PhotoImage(file='Right.gif')
        s.right1=Frame(s.admdash,bg="black")
        label2=Label(s.right1,image=right)
        label2.image = right
        label2.pack()
        s.right1.place(x=655,y=100)
        s.addhead=Label(s.admdash,text="Add New Item",bg="#050518",font="Narkisim 48",fg="#22D8C0")
        s.itno=Label(s.admdash,text="Item Number",bg="#050518",fg="#1717BB",font="Narkisim 24")
        s.itemname=Label(s.admdash,text="Item Name",bg="#050518",fg="#1717BB",font="Narkisim 24")
        s.itemcost=Label(s.admdash,text="Cost",bg="#050518",fg="#1717BB",font="Narkisim 24")
        s.itemstock=Label(s.admdash,text="Stock left",bg="#050518",fg="#1717BB",font="Narkisim 24")
        s.imglink=Label(s.admdash,text="Image Link",bg="#050518",fg="#1717BB",font="Narkisim 24")
        s.itnoe=Entry(s.admdash,fg="#050518",bg="#3292A0",font="Narkisim 24")
        s.itemnamee=Entry(s.admdash,fg="#050518",bg="#3292A2",font="Narkisim 24")
        s.itemcoste=Entry(s.admdash,fg="#050518",bg="#3292A2",font="Narkisim 24")
        s.itemstocke=Entry(s.admdash,fg="#050518",bg="#3292A2",font="Narkisim 24")
        s.imagelble=Label(s.admdash,text="image here",bg="#050518",font="Narkisim 36",fg="#22D8C0")
        s.imglinke=Entry(s.admdash,fg="#050518",bg="#3292A2",font="Narkisim 24")
        s.checkbutton=Button(s.admdash,command=s.imagecheck,text="Check Link",bg="#0B1056",font="Narkisim
24",fg="#3498DB")
        s.checkbutton.place(x=770,y=470)
        s.addbutton=Button(s.admdash,command=s.addfunction,text="Add Item",bg="#0B1056",font="Narkisim
24",fg="#3498DB")
        s.addbutton.place(x=570,y=610)
        s.addhead.place(x=450,y=25)
        s.itno.place(x=235,y=185,height=40,width=200)
        s.itemname.place(x=235,y=285,height=40,width=200)
        s.itemcost.place(x=235,y=385,height=40,width=200)
        s.itemstock.place(x=235,y=485,height=40,width=200)
        s.itnoe.place(x=440,y=185,height=40,width=200)
        s.itemnamee.place(x=440,y=285,height=40,width=200)
        s.itemcoste.place(x=440,y=385,height=40,width=200)
        s.itemstocke.place(x=440,y=485,height=40,width=200)
        s.imagelble.place(x=750,y=185,height=200,width=200)
        s.imglink.place(x=780,y=390)
        s.imglinke.place(x=680,y=435,width=350,height=25)
        s.ccount=0
    def addfunction(s):
```

```
        if s.select==0:
            f=open("Snacks.dat","ab")
        elif s.select==1:
            f=open("Juices.dat","ab")
        elif s.select==2:
            f=open("Etools.dat","ab")
        x=Item()
        itemno=int(s.itnoe.get())
        itemname=s.itemnamee.get()
        itemcost=int(s.itemcoste.get())
        itemstock=int(s.itemstocke.get())
        itemimage=s.imglinke.get()
        x.itfn(itemno,itemname,itemcost,itemstock,itemimage)
        pickle.dump(x,f)
        f.close()
        s.status=Label(s.admdash,text="Item Added successfully",bg="#010F16",font="Narkisim 11",fg="#2ECC71")
        s.status.place(x=540,y=580,width=250,height=10)
        s.clflag="adds"
        s.ccount+=1
    def imagecheck(s):
        ilink=s.imglinke.get()
        y='\ '
        x=y.strip(" ")
        imagelink=ilink.replace (x,"\\" )
        itemimage=PhotoImage(file=imagelink)
        s.imagelbl=Label(s.admdash,image=itemimage)
        s.imagelbl.image=itemimage
        s.imagelbl.place(x=750,y=185,height=200,width=200)
#Display Option
    def displaymenu(s):
        s.cleardash()
        if s.select==0:
            f=open("Snacks.dat","rb")
        elif s.select==1:
            f=open("Juices.dat","rb")
        elif s.select==2:
            f=open("Etools.dat","rb")
        s.detailslist=[]
        s.buttonslist=[]
        count=0
```

```
        try:
            while True:
                x=pickle.load(f)
                s.detailslist.append(x)
                txt=str(s.detailslist[count].itemno)+"    "+s.detailslist[count].itemname
                s.buttonslist.append(Button(s.admdash,command=lambda
count=count:s.displayinfo(count),text=txt,fg="#1536CF",bg="#030C34",font="Narkisim 17"))
                s.buttonslist[count].place(x=225,y=(185+40*(count-1)),width=400)
                count+=1
        except EOFError:
                pass
                f.close()
        s.cflag="disf"
        s.ccount=0
    def displayinfo(s,count):
        displaybg=PhotoImage(file='Right2.gif')
        s.infoframe=Frame(s.admdash,bg="black")
        label2=Label(s.infoframe,image=displaybg)
        label2.image = displaybg
        label2.pack()
        s.infoframe.place(x=650,y=100,width=350)
        itemdisplay=PhotoImage(file=s.detailslist[count].itemimage)
        s.displayimagelble=Label(s.admdash,image=itemdisplay,bg="#050518",font="Narkisim 36",fg="#22D8C0")
        s.displayimagelble.image=itemdisplay
        s.displayitemname=Label(s.admdash,text="Name:
"+s.detailslist[count].itemname,bg="#000415",font="Narkisim 16",fg="#22D8C0")
        s.displayitemcost=Label(s.admdash,text="Cost:
"+str(s.detailslist[count].itemcost),bg="#000415",font="Narkisim 16",fg="#22D8C0")
        s.displayitemstock=Label(s.admdash,text="Stock:
"+str(s.detailslist[count].itemstock),bg="#000415",font="Narkisim 16",fg="#22D8C0")
        s.displayitemsold=Label(s.admdash,text="Sold:
"+str(s.detailslist[count].itemsold),bg="#000415",font="Narkisim 16",fg="#22D8C0")
        s.displayitemgain=Label(s.admdash,text="Gain:
"+str(s.detailslist[count].itemgain),bg="#000415",font="Narkisim 16",fg="#22D8C0")
        s.displayimagelble.place(x=725,y=135,height=200,width=200)
        s.displayitemname.place(x=725,y=345,height=30,width=200)
        s.displayitemcost.place(x=725,y=385,height=30,width=200)
        s.displayitemstock.place(x=725,y=425,height=30,width=200)
        s.displayitemsold.place(x=725,y=465,height=30,width=200)
        s.displayitemgain.place(x=725,y=505,height=30,width=200)
```

```
            s.cflag="diss"
            s.ccount+=1
    #Search Options
        def searchitem(s):
            s.cleardash()
            s.clearflag="serf"
            s.seract=1
            s.serptr=1
            serframe=PhotoImage(file='serframe.gif')
            s.sertaskbar=Frame(s.admdash,bg="black")
            label=Label(s.sertaskbar,image=serframe)
            label.image = serframe
            label.pack()
            s.sertaskbar.place(x=240,y=40,width=780)
            s.serbyname=Button(s.admdash,command=s.sername,text="Name",font="Narkisim
    18",bg="red",fg="#28B463",relief="flat")
            s.serbycost=Button(s.admdash,command=s.sercost,text="Cost",font="Narkisim
    18",bg="red",fg="#28B463",relief="flat")
            s.serbystock=Button(s.admdash,command=s.serstock,text="Stock Left",font="Narkisim
    18",bg="red",fg="#28B463",relief="flat")
            s.serbysold=Button(s.admdash,command=s.sersold,text="Sold",font="Narkisim
    18",bg="red",fg="#28B463",relief="flat")
            s.serbynamee=Entry(s.admdash,text="Name",font="Narkisim 11",bg="Black",fg="#5DADE2")
            s.serbycoste=Entry(s.admdash,text="Cost",font="Narkisim 18",bg="Black",fg="#5DADE2")
            s.serbystocke=Entry(s.admdash,text="Stock Left",font="Narkisim 18",bg="Black",fg="#5DADE2")
            s.serbysolde=Entry(s.admdash,text="Sold",font="Narkisim 18",bg="Black",fg="#5DADE2")
            s.serbyname.place(x=252,y=48,height=38,width=80)
            s.serbycost.place(x=530,y=48,height=38,width=64)
            s.serbystock.place(x=675,y=48,height=38,width=100)
            s.serbysold.place(x=898,y=48,height=38,width=60)
            s.serbynamee.place(x=354,y=52,height=28,width=130)
            s.serbycoste.place(x=597,y=52,height=28,width=50)
            s.serbystocke.place(x=817,y=52,height=28,width=50)
            s.serbysolde.place(x=976,y=52,height=28,width=40)
            s.cflag="serf"
            s.ccount=0
        def sername(s):
            if s.serptr==1:
                s.serbycost.state="DISABLE"
                s.serbystock.state="DISABLE"
```

```
            s.serbysold.state="DISABLE"
            s.serbycoste.state="DISABLE"
            s.serbystocke.state="DISABLE"
            s.serbysolde.state="DISABLE"
            s.status=Label(s.admdash,text="Press Again to confirm Search",bg="#010F16",font="Narkisim
11",fg="#2ECC71")
            s.status.place(x=225,y=25)
            s.serptr*=-1
        elif s.serptr==-1:
            if s.select==0:
                f=open("Snacks.dat","rb")
            elif s.select==1:
                f=open("Juices.dat","rb")
            elif s.select==2:
                f=open("Etools.dat","rb")
            serindex=s.serbynamee.get()
            didyoumean=""
            try:
                while True:
                        x=pickle.load(f)
                        if x.itemname.upper() != serindex.upper():
                            n=len(serindex)
                            if serindex[0:(n/4)].upper() in x.itemname.upper() or serindex[(n/4):(n/2)].upper() in
x.itemname.upper() or serindex[(n/2):(3*n/4)].upper() in x.itemname.upper() or serindex[(3*n/4):n].upper() in
x.itemname.upper() :
                                didyoumean=x
                        elif x.itemname.upper() == serindex.upper():
                            s.seritemname=Label(s.admdash,text="Name:
"+x.itemname,bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.seritemcost=Label(s.admdash,text="Cost:
"+str(x.itemcost),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.seritemstock=Label(s.admdash,text="Stock:
"+str(x.itemstock),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.seritemsold=Label(s.admdash,text="Sold:
"+str(x.itemsold),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.seritemgain=Label(s.admdash,text="Gain:
"+str(x.itemgain),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.serinfoframe=Frame(s.admdash,bg="black")
                            s.serinfoframe.place(x=650,y=100,width=350)
                            itemdisplay=PhotoImage(file=x.itemimage)
```

```
                        s.serimagelble=Label(s.admdash,image=itemdisplay,bg="#050518",font="Narkisim
36",fg="#22D8C0")
                        s.serimagelble.image=itemdisplay
                        s.serimagelble.place(x=725,y=135,height=200,width=200)
                        s.seritemname.place(x=725,y=345,height=30,width=200)
                        s.seritemcost.place(x=725,y=385,height=30,width=200)
                        s.seritemstock.place(x=725,y=425,height=30,width=200)
                        s.seritemsold.place(x=725,y=465,height=30,width=200)
                        s.seritemgain.place(x=725,y=505,height=30,width=200)
            except EOFError:
                    f.close()
                    s.cflag="sers"
    def sercost(s):
        if s.serptr==1:
            s.serbyname.state="DISABLE"
            s.serbystock.state="DISABLE"
            s.serbysold.state="DISABLE"
            s.serbynamee.state="DISABLE"
            s.serbystocke.state="DISABLE"
            s.serbysolde.state="DISABLE"
            s.status=Label(s.admdash,text="Press Again to confirm Search",bg="#010F16",font="Narkisim
11",fg="#2ECC71")
            s.status.place(x=225,y=25)
            s.serptr*=-1
        elif s.serptr==-1:
            if s.select==0:
                f=open("Snacks.dat","rb")
            elif s.select==1:
                f=open("Juices.dat","rb")
            elif s.select==2:
                f=open("Etools.dat","rb")
            serindex=int(s.serbycoste.get())
            s.detailslist=[]
            s.buttonslist=[]
            didyoumean=""
            try:
                while True:
                        x=pickle.load(f)
                        if x.itemcost != serindex:
                            pass
```

```
                    elif x.itemcost == serindex:
                        s.detailslist.append(x)
                        s.buttonslist.append(random.random())
                        for s.displayobj in range(len(s.buttonslist)):
                            txt=str(s.detailslist[s.displayobj].itemno)+"    "+s.detailslist[s.displayobj].itemname

s.buttonslist[s.displayobj]=Button(s.admdash,text=txt,fg="#1536CF",bg="#030C34",font="Narkisim 17")
                            s.buttonslist[s.displayobj].place(x=225,y=(185+40*(s.displayobj-1)),width=400)
            except EOFError:
                    f.close()
                    s.cflag="sers"
    def serstock(s):
        if s.serptr==1:
            s.serbyname.state="DISABLE"
            s.serbycost.state="DISABLE"
            s.serbysold.state="DISABLE"
            s.serbynamee.state="DISABLE"
            s.serbycoste.state="DISABLE"
            s.serbysolde.state="DISABLE"
            s.status=Label(s.admdash,text="Press Again to confirm Search",bg="#010F16",font="Narkisim
11",fg="#2ECC71")
            s.status.place(x=225,y=25)
            s.serptr*=-1
        elif s.serptr==-1:
            if s.select==0:
                f=open("Snacks.dat","rb")
            elif s.select==1:
                f=open("Juices.dat","rb")
            elif s.select==2:
                f=open("Etools.dat","rb")
            serindex=int(s.serbystocke.get())
            s.detailslist=[]
            s.buttonslist=[]
            didyoumean=""
            try:
                while True:
                    x=pickle.load(f)
                    if x.itemstock != serindex:
                        pass
                    elif x.itemstock == serindex:
```

```
                        s.detailslist.append(x)
                        s.buttonslist.append(random.random())
                        for s.displayobj in range(len(s.buttonslist)):
                            txt=str(s.detailslist[s.displayobj].itemno)+"    "+s.detailslist[s.displayobj].itemname

s.buttonslist[s.displayobj]=Button(s.admdash,text=txt,fg="#1536CF",bg="#030C34",font="Narkisim 17")
                            s.buttonslist[s.displayobj].place(x=225,y=(185+40*(s.displayobj-1)),width=400)
            except EOFError:
                        f.close()
                        s.cflag="sers"
    def sersold(s):
        if s.serptr==1:
            s.serbyname.state="DISABLE"
            s.serbystock.state="DISABLE"
            s.serbysold.state="DISABLE"
            s.serbynamee.state="DISABLE"
            s.serbystocke.state="DISABLE"
            s.serbysolde.state="DISABLE"
            s.status=Label(s.admdash,text="Press Again to confirm Search",bg="#010F16",font="Narkisim
11",fg="#2ECC71")
            s.status.place(x=225,y=25)
            s.serptr*=-1
        elif s.serptr==-1:
            if s.select==0:
                f=open("Snacks.dat","rb")
            elif s.select==1:
                f=open("Juices.dat","rb")
            elif s.select==2:
                f=open("Etools.dat","rb")
            serindex=int(s.serbysolde.get())
            s.detailslist=[]
            s.buttonslist=[]
            didyoumean=""
            try:
                while True:
                        x=pickle.load(f)
                        if x.itemsold != serindex:
                            pass
                        elif x.itemsold == serindex:
                            s.detailslist.append(x)
```

```
                        s.buttonslist.append(random.random())
                        for s.displayobj in range(len(s.buttonslist)):
                            txt=str(s.detailslist[s.displayobj].itemno)+"    "+s.detailslist[s.displayobj].itemname

s.buttonslist[s.displayobj]=Button(s.admdash,text=txt,fg="#1536CF",bg="#030C34",font="Narkisim 17")
                            s.buttonslist[s.displayobj].place(x=225,y=(185+40*(s.displayobj-1)),width=400)
            except EOFError:
                        f.close()
                        s.cflag="sers"
#Update Options
#itemno,itemname,itemcost,itemstock,itemimage
    def updateitems(s):
        s.cleardash()
        s.cflag="updf"
        upframe=PhotoImage(file='upframe.gif')
        s.uptaskbar=Frame(s.admdash,bg="black")
        label=Label(s.uptaskbar,image=upframe)
        label.image = upframe
        label.pack()
        s.uptaskbar.place(x=240,y=73,width=780)
        s.uphead=Label(s.admdash,text="Update",font="Narkisim 36",bg="#0B3652",fg="white")
        s.updateitemnamee=Entry(s.admdash,font="Narkisim 18",bg="Black",fg="white")
        s.updateitemnume=Entry(s.admdash,font="Narkisim 20",bg="Black",fg="white")
        s.updateser=Button(s.admdash,command=s.serupdate,font="Narkisim 18",text="Search",fg="White",bg="red")
        s.uphead.place(x=540,y=25,height=38)
        s.updateitemnamee.place(x=350,y=82,width=225,height=35)
        s.updateitemnume.place(x=690,y=82,width=60,height=35)
        s.updateser.place(x=830,y=82,height=35,width=175)
    def serupdate(s):
            s.cflag="upds"
            if s.select==0:
                f=open("Snacks.dat","rb")
            elif s.select==1:
                f=open("Juices.dat","rb")
            elif s.select==2:
                f=open("Etools.dat","rb")
            s.temp=open("T.dat","ab")
            flag=0
            serindex=s.updateitemnamee.get()
            serindex1=s.updateitemnume.get()
```

```
            try:
                while True:
                        s.x=pickle.load(f)
                        if s.x.itemname.upper() == serindex.upper():
                            s.updateitemname=Label(s.admdash,text="Name:
"+s.x.itemname,bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemcost=Label(s.admdash,text="Cost:
"+str(s.x.itemcost),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemstock=Label(s.admdash,text="Stock:
"+str(s.x.itemstock),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemsold=Label(s.admdash,text="Sold:
"+str(s.x.itemsold),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemgain=Label(s.admdash,text="Gain:
"+str(s.x.itemgain),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            itemdisplay=PhotoImage(file=s.x.itemimage)
                            s.displayimagelble=Label(s.admdash,image=itemdisplay,bg="#050518",font="Narkisim
36",fg="#22D8C0")
                            s.displayimagelble.image=itemdisplay
                            s.updateitemnamee=Entry(s.admdash,bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemcoste=Entry(s.admdash,bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemstocke=Entry(s.admdash,bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemsolde=Entry(s.admdash,bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemgaine=Entry(s.admdash,bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.imglinke=Entry(s.admdash,fg="#050518",bg="#3292A2",font="Narkisim 11")

s.checkimg=Button(s.admdash,command=s.upimagecheck,bg="blue",fg="yellow",font="Narkisim 20",text="Check")
                            s.updateitemnamen=Label(s.admdash,text="Name: ",bg="#000415",font="Narkisim
16",fg="#22D8C0")
                            s.updateitemcostn=Label(s.admdash,text="Cost: ",bg="#000415",font="Narkisim
16",fg="#22D8C0")
                            s.updateitemstockn=Label(s.admdash,text="Stock: ",bg="#000415",font="Narkisim
16",fg="#22D8C0")
                            s.updateitemsoldn=Label(s.admdash,text="Sold: ",bg="#000415",font="Narkisim
16",fg="#22D8C0")
                            s.updateitemgainn=Label(s.admdash,text="Gain: ",bg="#000415",font="Narkisim
16",fg="#22D8C0")
                            s.imglinke.place(x=725,y=130,height=25,width=200)
                            s.checkimg.place(x=925,y=130,height=23)
                            s.updatebutton=Button(s.admdash,command=s.updateinput,text="Update",font="Narkisim
20",bg="black",fg="Green")
```

```
                    s.displayimagelble.place(x=325,y=135,height=200,width=200)
                    s.updateitemname.place(x=325,y=345,height=30,width=150)
                    s.updateitemcost.place(x=325,y=385,height=30,width=150)
                    s.updateitemstock.place(x=325,y=425,height=30,width=150)
                    s.updateitemsold.place(x=325,y=465,height=30,width=150)
                    s.updateitemgain.place(x=325,y=505,height=30,width=150)
                    s.updateitemnamen.place(x=625,y=365,height=30,width=200)
                    s.updateitemcostn.place(x=625,y=405,height=30,width=200)
                    s.updateitemstockn.place(x=625,y=445,height=30,width=200)
                    s.updateitemsoldn.place(x=625,y=485,height=30,width=200)
                    s.updateitemgainn.place(x=625,y=525,height=30,width=200)
                    s.updateitemnamee.place(x=825,y=365,height=30,width=200)
                    s.updateitemcoste.place(x=825,y=405,height=30,width=200)
                    s.updateitemstocke.place(x=825,y=445,height=30,width=200)
                    s.updateitemsolde.place(x=825,y=485,height=30,width=200)
                    s.updateitemgaine.place(x=825,y=525,height=30,width=200)
                    s.updatebutton.place(x=745,y=565,height=30,width=200)
                    s.no=s.x.itemno
                    flag+=1
                elif s.x.itemno != int(serindex1) or s.x.itemname.upper()!=serindex.upper():
                    if flag==1:
                        s.no+=1
                        s.x.itemno-=1
                        pickle.dump(s.x,s.temp)
        except EOFError:
                f.close()
                s.temp.close()
        if s.select==0:
            os.remove("Snacks.dat")
            os.rename("T.dat","Snacks.dat")
        elif s.select==1:
            os.remove("Juices.dat")
            os.rename("T.dat","Juices.dat")
        elif s.select==2:
            os.remove("Etools.dat")
            os.rename("T.dat","Etools.dat")
        s.ccount=0
def updateinput(s):
    if s.select==0:
            f=open("Snacks.dat","ab")
```

```
        elif s.select==1:
                f=open("Juices.dat","ab")
        elif s.select==2:
                f=open("Etools.dat","ab")
        x=Item()
        itemno=s.no
        itemname=s.updateitemnamee.get()
        itemcost=int(s.updateitemcoste.get())
        itemstock=int(s.updateitemstocke.get())
        itemimage=s.imglinke.get()
        x.itfn(itemno,itemname,itemcost,itemstock,itemimage)
        pickle.dump(x,f)
        s.status=Label(s.admdash,text="Item Updated successfully",bg="#010F16",font="Narkisim
11",fg="#2ECC71")
        s.status.place(x=540,y=580,width=250,height=10)
        s.ccount+=1
        f.close()
    def upimagecheck(s):
                ilink=s.imglinke.get()
                y='\ '
                x=y.strip(" ")
                imagelink=ilink.replace (x,"\\" )
                itemimage=PhotoImage(file=imagelink)
                s.imagelblee=Label(s.admdash,image=itemimage)
                s.imagelblee.image=itemimage
                s.imagelblee.place(x=725,y=150,height=200,width=200)
#Delete Options
    def deleteitem(s):
        s.cleardash()
        s.cflag="delf"
        delframe=PhotoImage(file='delframe.gif')
        s.deltaskbar=Frame(s.admdash,bg="black")
        label=Label(s.deltaskbar,image=delframe)
        label.image = delframe
        label.pack()
        s.deltaskbar.place(x=240,y=73,width=780)
        s.delhead=Label(s.admdash,text="Delete",font="Narkisim 36",bg="#0B3652",fg="white")
        s.delitemnamee=Entry(s.admdash,font="Narkisim 18",bg="Black",fg="white")
        s.delitemnume=Entry(s.admdash,font="Narkisim 20",bg="Black",fg="white")
        s.delser=Button(s.admdash,command=s.serdelete,font="Narkisim 18",text="Delete",fg="Yellow",bg="red")
```

```
        s.delhead.place(x=540,y=25,height=38)
        s.delitemnamee.place(x=350,y=82,width=225,height=35)
        s.delitemnume.place(x=690,y=82,width=60,height=35)
        s.delser.place(x=830,y=84,height=35,width=175)
    def serdelete(s):
            s.cflag="dels"
            if s.select==0:
                f=open("Snacks.dat","rb")
            elif s.select==1:
                f=open("Juices.dat","rb")
            elif s.select==2:
                f=open("Etools.dat","rb")
            s.temp=open("T.dat","ab")
            flag=0
            serindex=s.delitemnamee.get()
            serindex1=s.delitemnume.get()
            try:
                while True:
                        s.x=pickle.load(f)
                        if s.x.itemname.upper() == serindex.upper():
                            s.updateitemname=Label(s.admdash,text="Name:
"+s.x.itemname,bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemcost=Label(s.admdash,text="Cost:
"+str(s.x.itemcost),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemstock=Label(s.admdash,text="Stock:
"+str(s.x.itemstock),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemsold=Label(s.admdash,text="Sold:
"+str(s.x.itemsold),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            s.updateitemgain=Label(s.admdash,text="Gain:
"+str(s.x.itemgain),bg="#000415",font="Narkisim 16",fg="#22D8C0")
                            itemdisplay=PhotoImage(file=s.x.itemimage)
                            s.displayimagelble=Label(s.admdash,image=itemdisplay,bg="#050518",font="Narkisim
36",fg="#22D8C0")
                            s.displayimagelble.image=itemdisplay
                            s.deletebutton=Label(s.admdash,text="Item deleted Successfully",font="Narkisim
24",bg="black",fg="Green")
                            s.displayimagelble.place(x=480,y=135,height=200,width=200)
                            s.updateitemname.place(x=485,y=345,height=30,width=150)
                            s.updateitemcost.place(x=485,y=385,height=30,width=150)
                            s.updateitemstock.place(x=485,y=425,height=30,width=150)
```

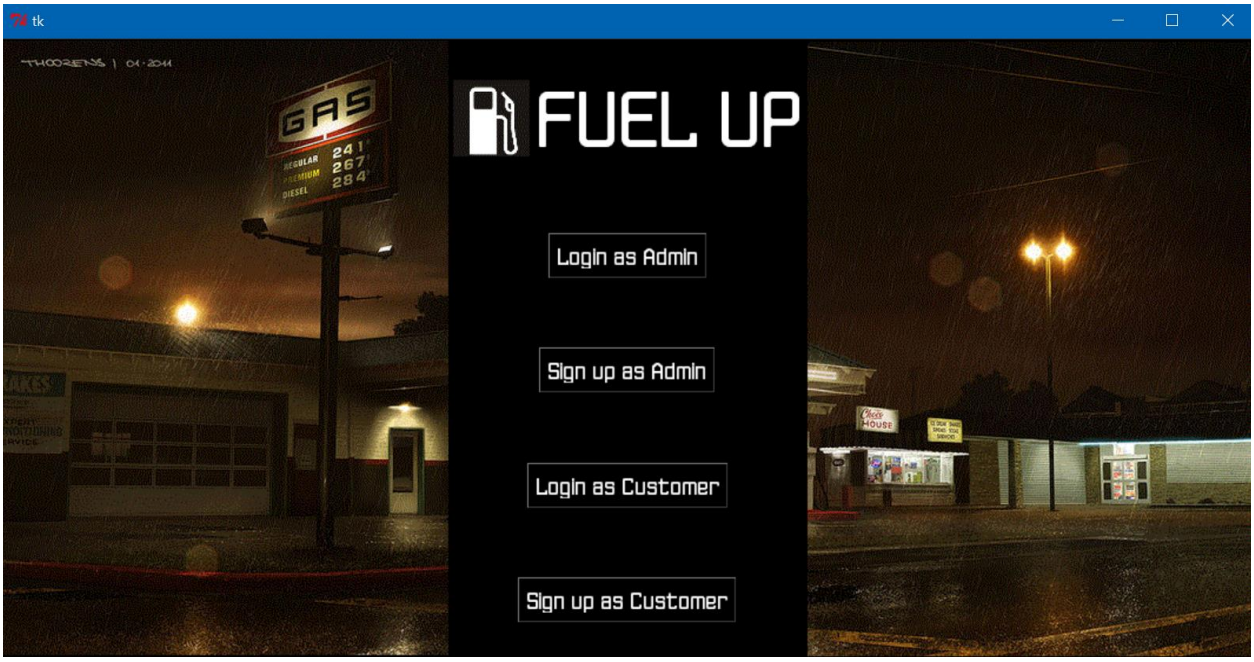
```
                        s.updateitemsold.place(x=485,y=465,height=30,width=150)
                        s.updateitemgain.place(x=485,y=505,height=30,width=150)
                        s.deletebutton.place(x=325,y=565,height=40,width=400)
                        flag+=1
                    elif s.x.itemno != int(serindex1) or s.x.itemname.upper()!=serindex.upper():
                        if flag==1:
                            s.x.itemno-=1
                        pickle.dump(s.x,s.temp)
            except EOFError:
                    f.close()
                    s.temp.close()
            if s.select==0:
                os.remove("Snacks.dat")
                os.rename("T.dat","Snacks.dat")
            elif s.select==1:
                os.remove("Juices.dat")
                os.rename("T.dat","Juices.dat")
            elif s.select==2:
                os.remove("Etools.dat")
                os.rename("T.dat","Etools.dat")
    def Login(s):
            f=open("Admin.dat","rb")
            n=s.LEmailE.get()
            x=s.LPassE.get()
            try:
                while True:
                        obj=pickle.load(f)
                        if obj.ademail!= n:
                            pass
                        elif obj.ademail==n and obj.adpass==x:
                            s.currentadmin=obj
                            s.admindash()
                else:
                    Error=Label(s.admw,text="Invalid Entry",bg="#010F16",font="Narkisim
11",fg="#E74C3C").place(x=620,y=140,width=250,height=10)
            except EOFError:
                    f.close()
class MainWindow:
        def MW(s):
            s.mw=Toplevel()
```

```
            s.mw.title("Welcome to C-Store")
            dash=PhotoImage(file='WelcomeWindow.gif')
            s.f3=Frame(s.mw)
            label1=Label(s.f3,image=dash)
            label1.image=dash
            label1.pack()
            s.f3.pack()
            admlg=PhotoImage(file='admin logo.gif')
            s.adminlogo=Frame(s.mw,bg="black")
            label1=Label(s.adminlogo,image=admlg)
            label1.image = admlg
            label1.pack()
            s.adminlogo.place(x=730,y=210,width=200,height=200)
            custlg=PhotoImage(file='cust logo.gif')
            s.custogo=Frame(s.mw,bg="black")
            label2=Label(s.custogo,image=custlg)
            label2.image = custlg
            label2.pack()
            s.custogo.place(x=120,y=210,width=200,height=180)
            storelg=PhotoImage(file='Logo cart.gif')
            s.storelogo=Frame(s.mw,bg="black")
            label3=Label(s.storelogo,image=storelg)
            label3.image = storelg
            label3.pack()
            s.storelogo.place(x=475,y=380,width=100,height=150)
            #120 400
            adminbutton=Button(s.mw,text="Admin",command=s.adm,relief="groove",bg="White",font="Narkisim
36",fg="Black")
            custbutton=Button(s.mw,text="Customer",command=s.cust,relief="groove",bg="Black",font="Narkisim
36",fg="White")
            adminbutton.place(x=730,y=400,width=202,height=50)
            custbutton.place(x=120,y=400,width=200,height=50)
            s.mw.mainloop()
        def cust(s):
            s.mw.destroy()
            X=customers()
            X.customermain()
        def adm(s):
            s.mw.destroy()
            x=Admin()
```

```
                x.adminmain()
root=Toplevel()
G=MainWindow()
customfonts123= tkFont.Font(family="JLS Data GothicR  NC", size=15)
customfontl23= tkFont.Font(family="JLS Data GothicR  NC", size=50)
labelmain=Label(root,text='THE FUEL APP', font=customfontl23, fg='white', bg='black')
img1001=PhotoImage(file='fuelapp.gif')
labelpic=Label(root,image=img1001, font=customfonts123, fg='white', bg='black')
fuelupbutton=Button(root,text='Fuel Up', bg='black',fg='white', font=customfonts123,command=lambda:mainfuelapp())
cstorebutton=Button(root,text='C-Store',bg='black',fg='white',font=customfonts123,command=lambda:G.MW())
labelmain.grid(row=0,column=1,columnspan=5)
labelpic.grid(column=0,row=0)
fuelupbutton.grid(row=1,column=1, rowspan=6)
cstorebutton.grid(row=1,column=4, rowspan=6)
root.configure(bg='black')
root.mainloop()
```

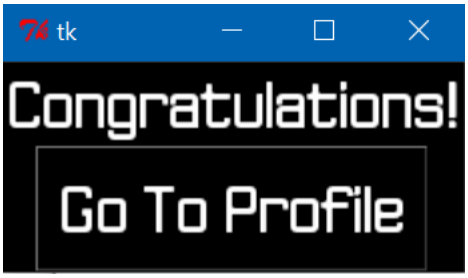
# Output Screenshots

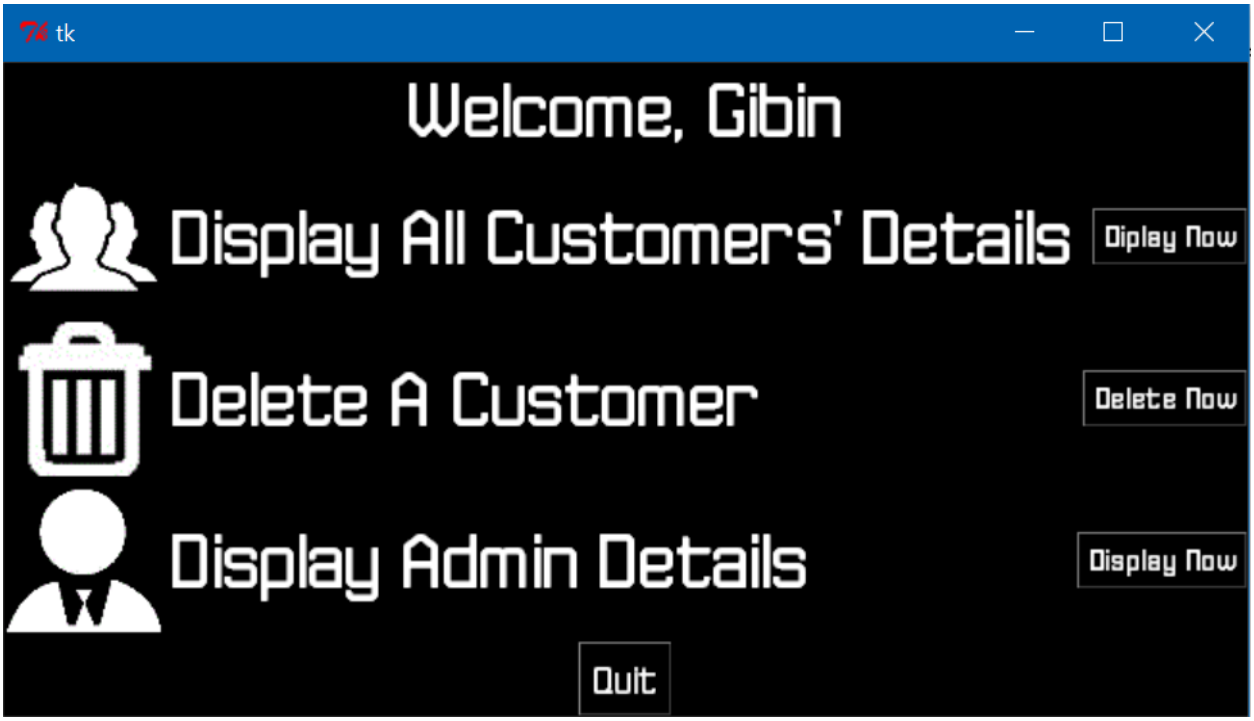

tk
THE FUEL APP
Fuel Up
C-Store


tk
GAS
FUEL UP
Login as Admin
Sign up as Admin
Login as Customer
Sign up as Customer


tk
Admin Login
Username:
Password:
Login

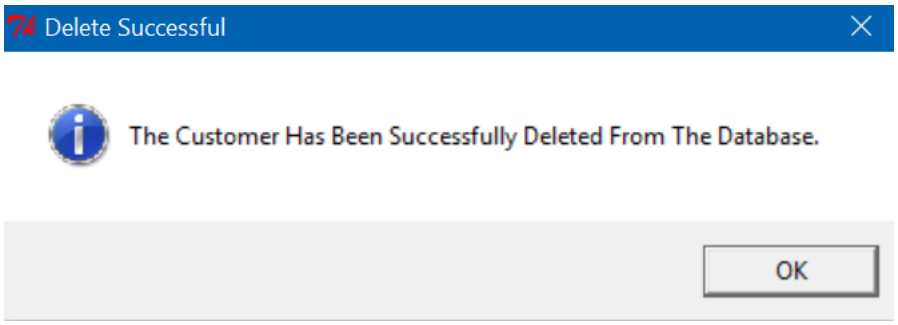

tk
Admin Sign Up
Enter Your Username:
Enter Your Password:
Enter Your Designation:
Sign up

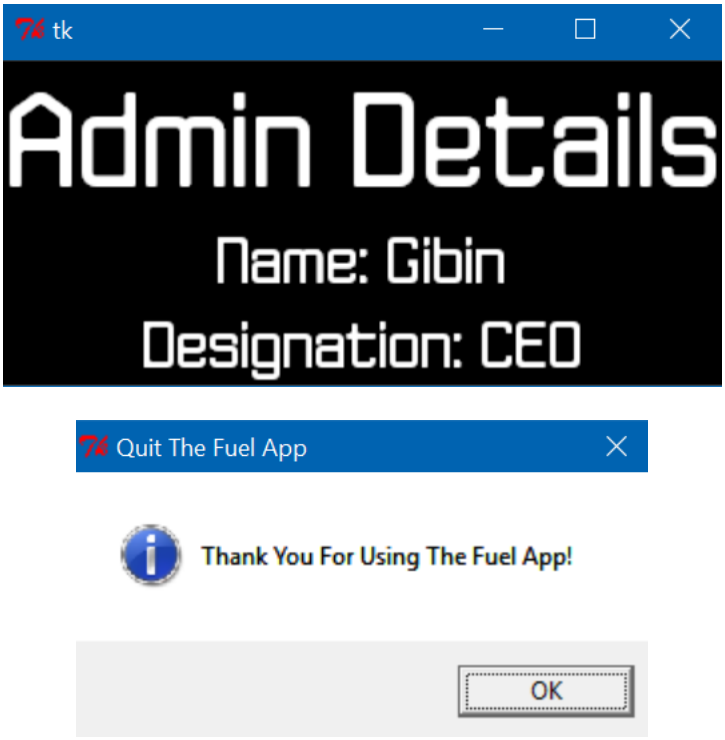

tk
Congratulations!
Go To Profile


tk
Welcome, Gibin
Display All Customers' Details
Diplay Now
Delete A Customer
Delete Now
Display Admin Details
Display Now
Quit

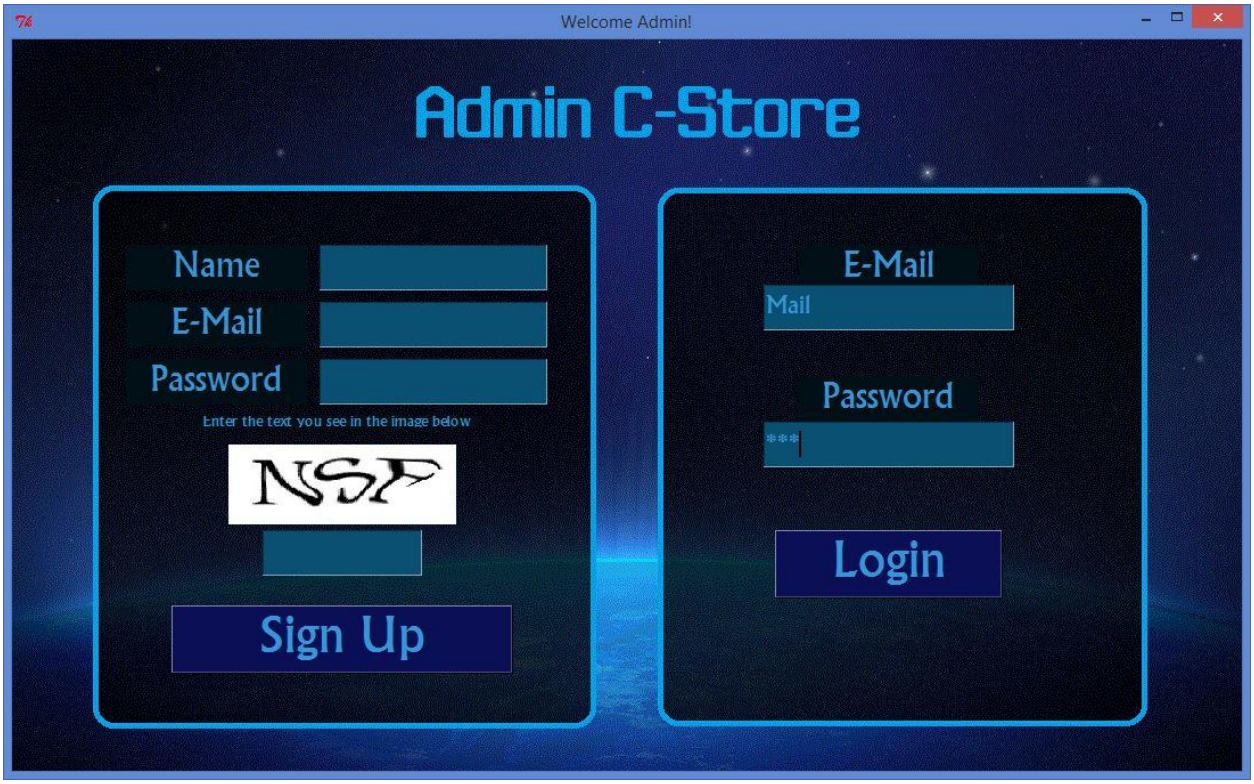

tk
Customer Details
Name: Gibin
Total CO2 Emissions Throughout The Year: 774.0
Total Expenditure For The Year: 3300

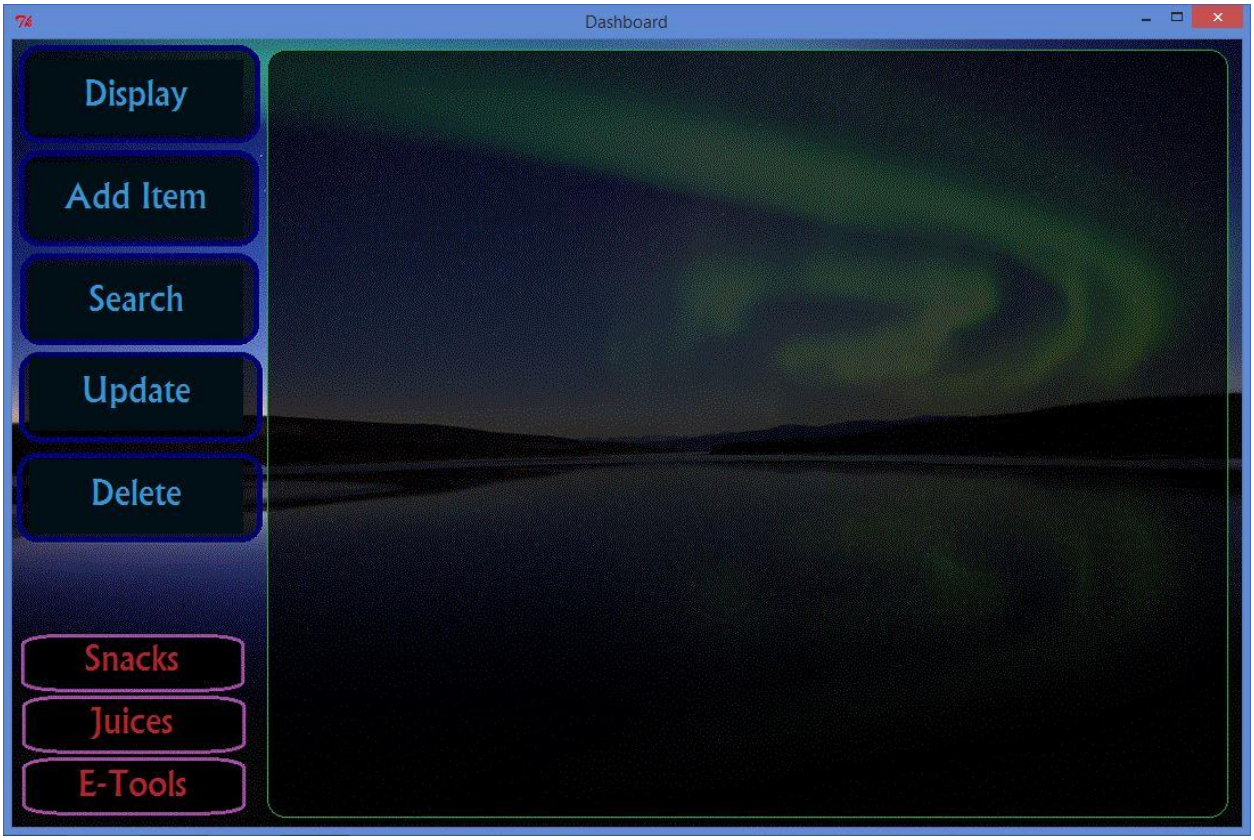

tk
Delete Customer
Enter The Customer You Wish To Delete:
Delete


Delete Successful
The Customer Has Been Successfully Deleted From The Database.
OK

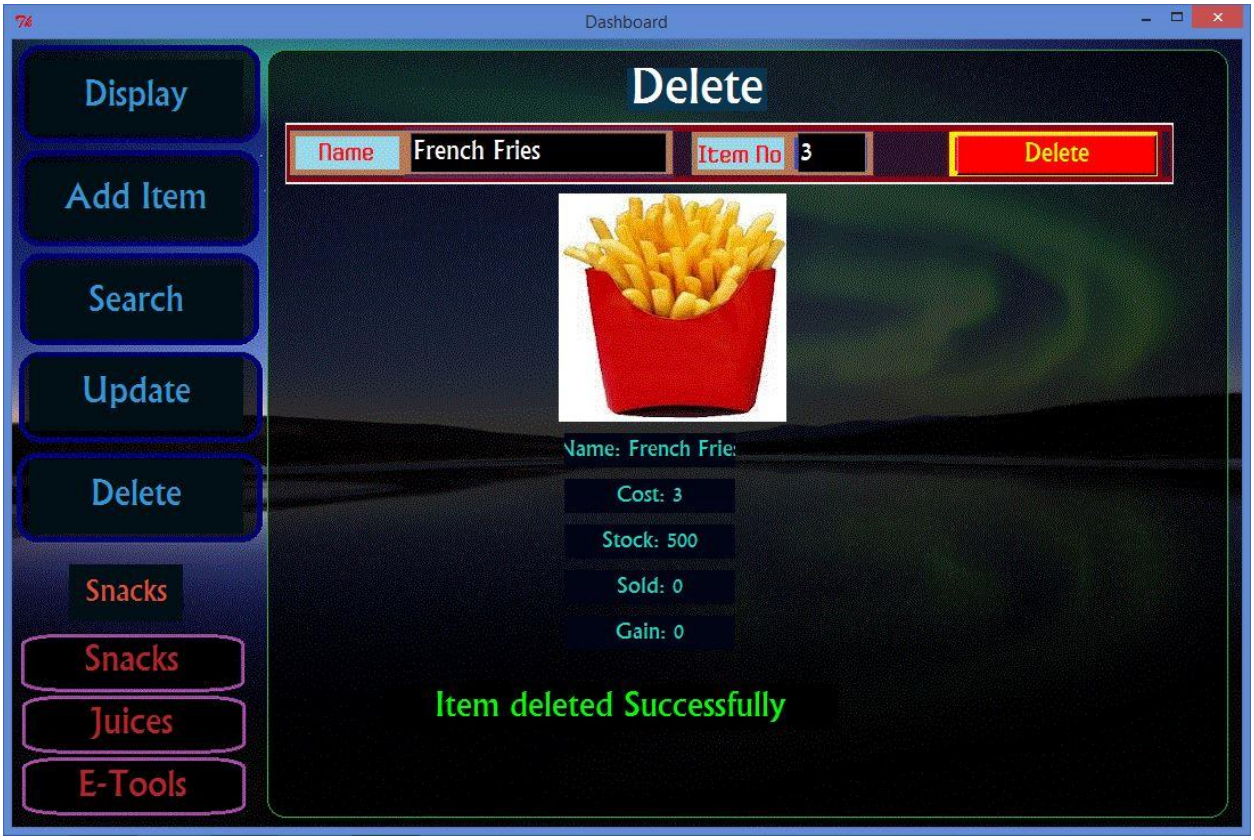

tk
Admin Details
Name: Gibin
Designation: CEO
Quit The Fuel App
Thank You For Using The Fuel App!
OK


Welcome to C-Store
C-Store
Customer
Admin
C-Store


Welcome Admin!
Admin C-Store
Name
E-Mail
Password
Enter the text you see in the image below
NSF
Sign Up
E-Mail
Mail
Password
Login


Dashboard
Display
Add Item
Search
Update
Delete
Snacks
Juices
E-Tools

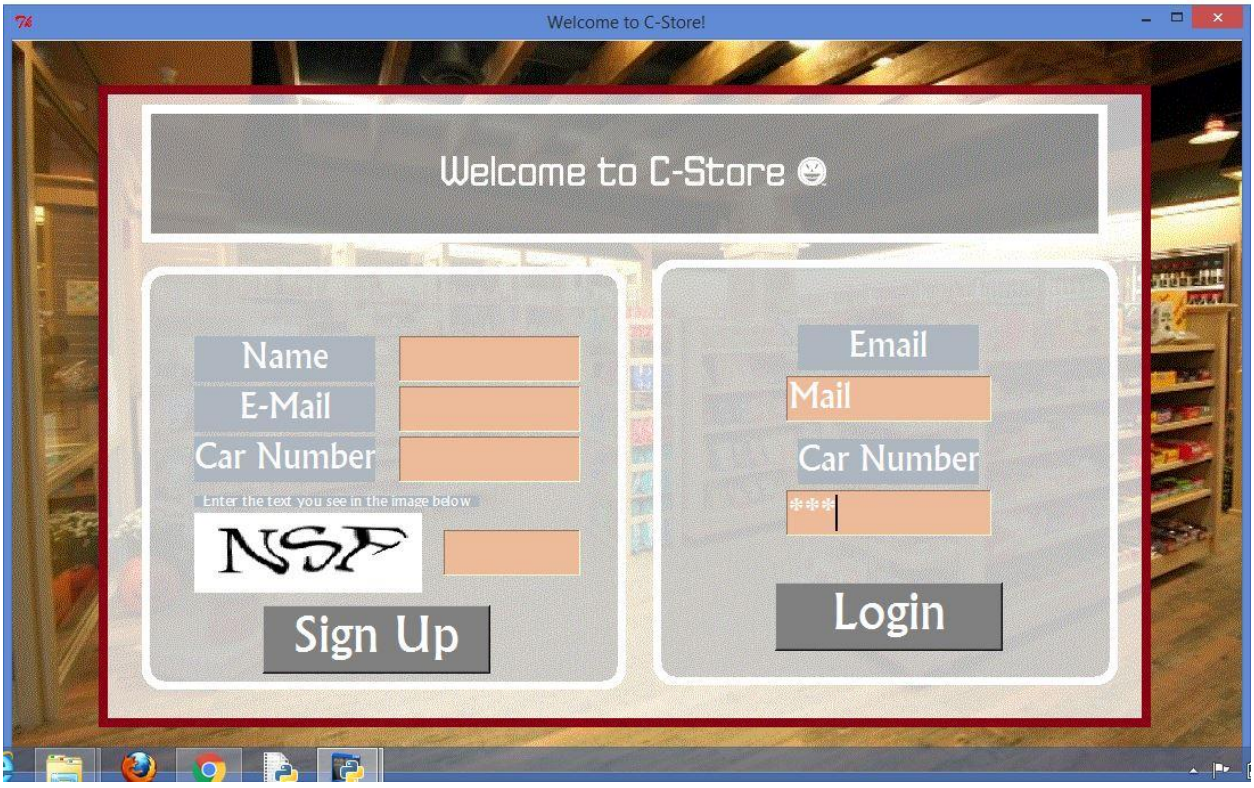

Dashboard
Display
Add Item
Search
Update
Delete
Snacks
Juices
E-Tools
Add New Item
Item Number
3
Item Name
French Fries
Cost
3
Stock left
200
Image Link
French Fries.gif
Check Link
Item Added successfully
Add Item

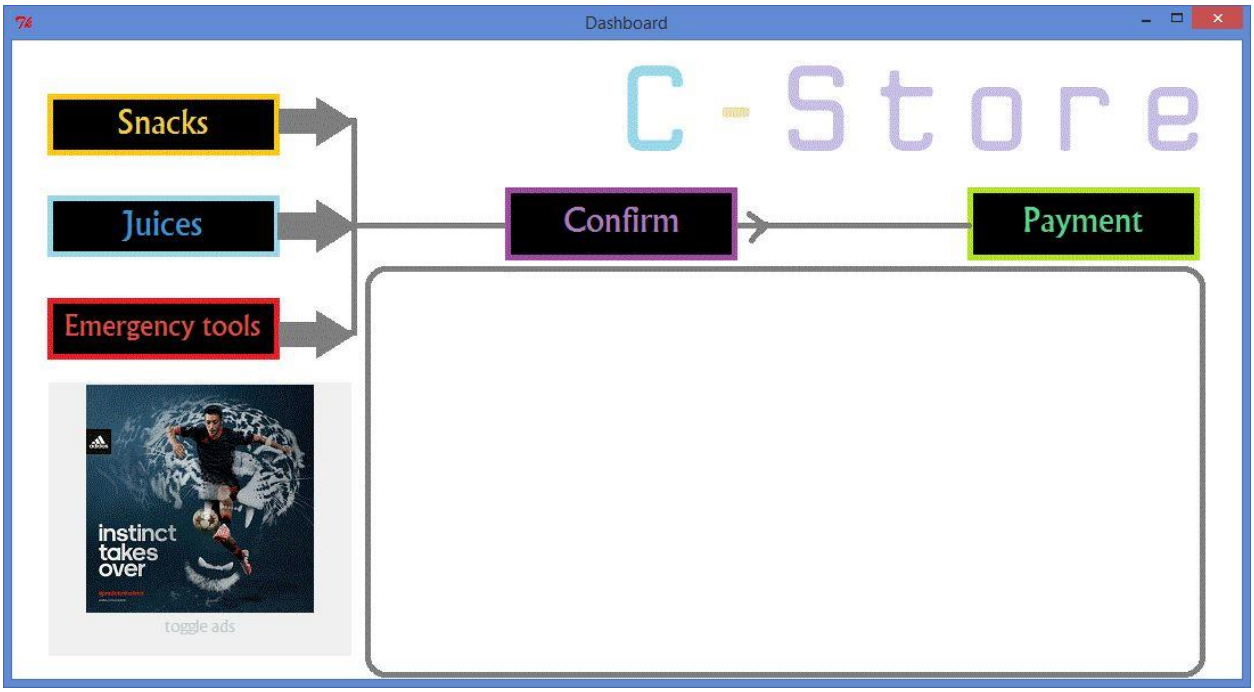

Dashboard
Display
Add Item
Search
Update
Delete
Snacks
Snacks
Juices
E-Tools
Delete
Name
French Fries
Item No
3
Delete
Name: French Frie
Cost: 3
Stock: 500
Sold: 0
Gain: 0
Item deleted Successfully

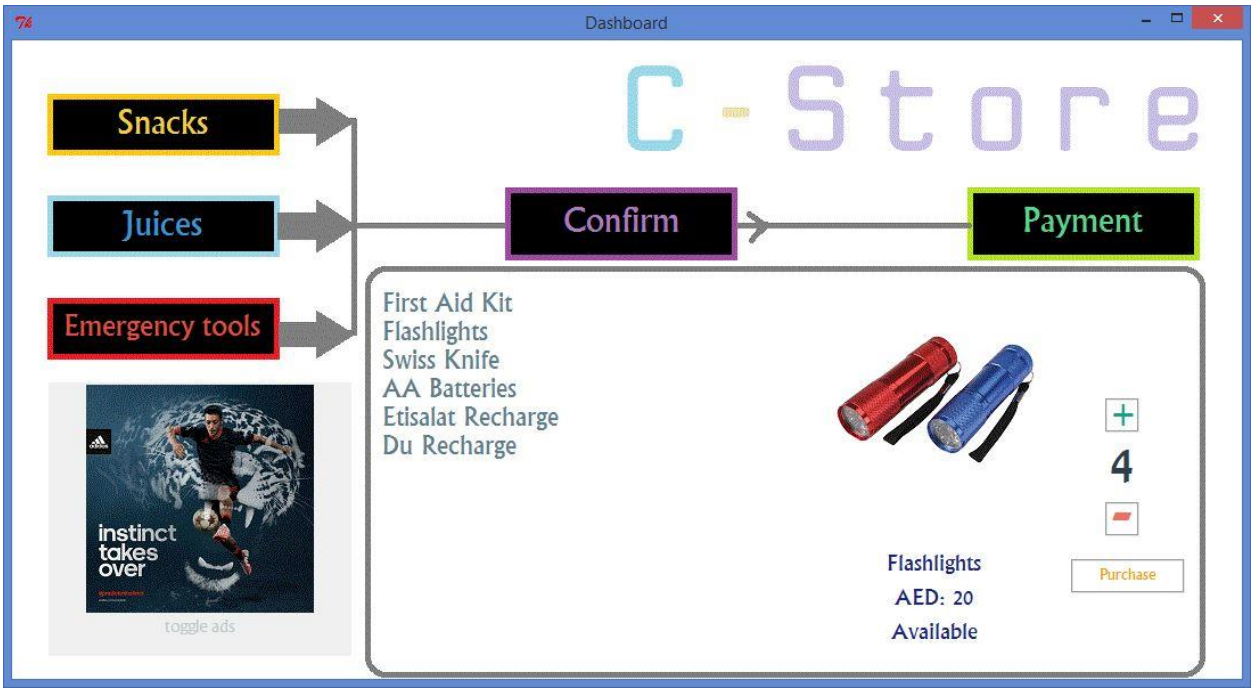

Dashboard
Display
Add Item
Search
Update
Delete
Snacks
Juices
E-Tools
1 Veg Sandwich
2 Veg Burger
3 French Fries
Name: French Fries
Cost: 3
Stock: 200
Sold: 0
Gain: 0

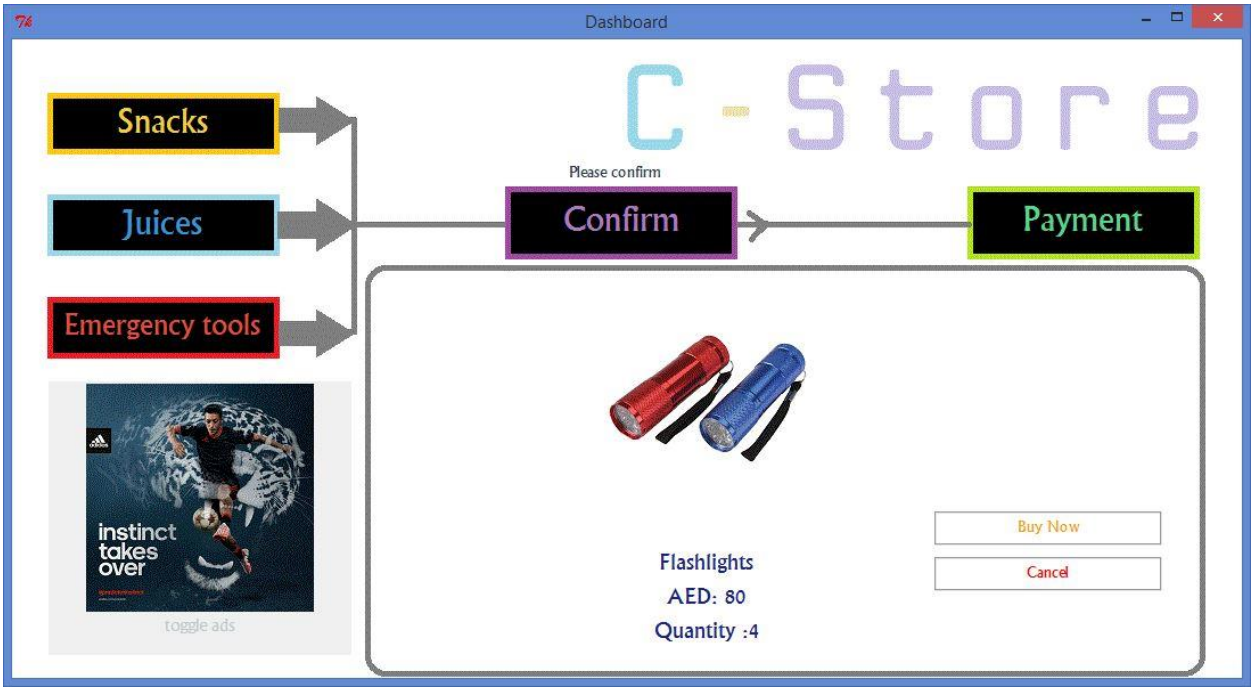

Dashboard
Display
Add Item
Search
Update
Delete
Snacks
Juices
E-Tools
Press Again to confirm Search
Name
French Fries
Cost
3
Stock Left
200
Sold
0
Name: French Fries
Cost: 3
Stock: 200
Sold: 0
Gain: 0

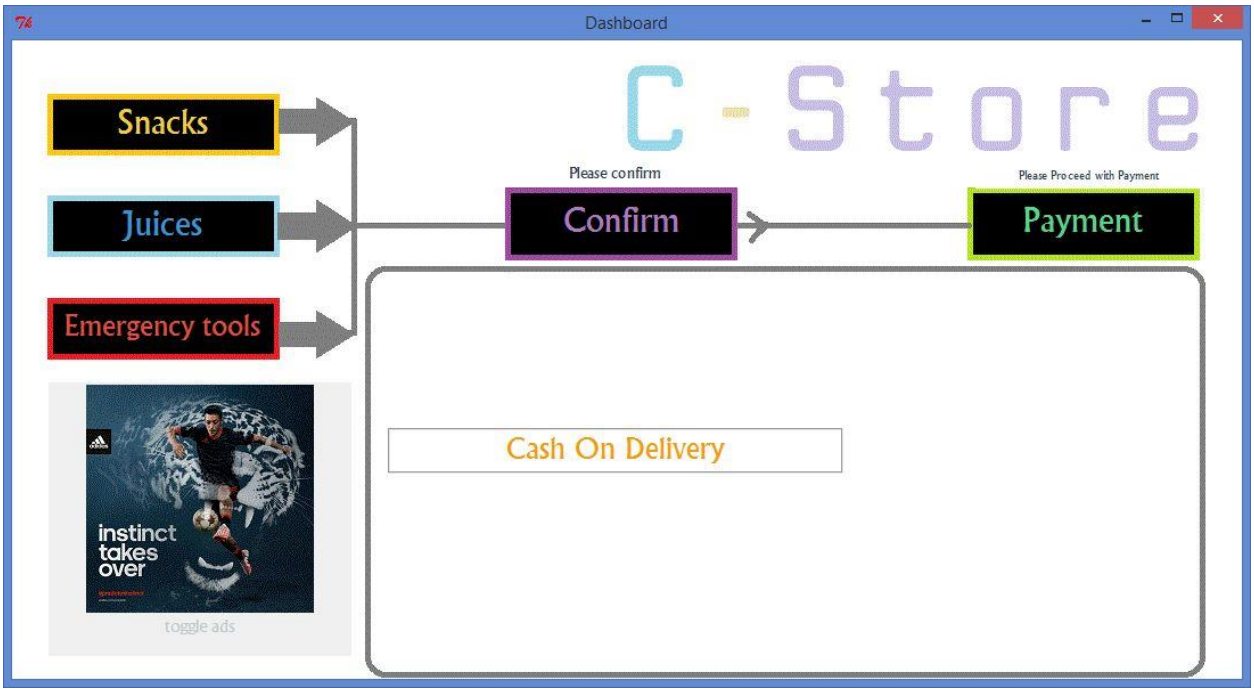

Dashboard
Update
Display
Add Item
Search
Update
Delete
Snacks
Juices
E-Tools
Name
French Fries
Item No.
3
Search
French Fries.gif
Check
Name: French Fries
Cost: 3
Stock: 200
Sold: 0
Gain: 0
Name:
French Fries
Cost:
3
Stock:
500
Sold:
0
Gain:
0
Item Updated successfully
Update


Welcome to C-Store!
Welcome to C-Store
Name
E-Mail
Car Number
Enter the text you see in the image below
NSP
Sign Up
Email
Mail
Car Number
***
Login


Dashboard
C-Store
Snacks
Juices
Emergency tools
Confirm
Payment
instinct takes over
toggle ads


Dashboard
C-Store
Snacks
Juices
Emergency tools
Confirm
Payment
First Aid Kit
Flashlights
Swiss Knife
AA Batteries
Etisalat Recharge
Du Recharge
4
Flashlights
AED: 20
Available
Purchase
instinct takes over
toggle ads


Dashboard
C-Store
Snacks
Juices
Emergency tools
Please confirm
Confirm
Payment
Flashlights
AED: 80
Quantity :4
Buy Now
Cancel
instinct takes over
toggle ads


Dashboard
C-Store
Snacks
Juices
Emergency tools
Please confirm
Confirm
Please Proceed with Payment
Payment
Cash On Delivery
instinct takes over
toggle ads


Dashboard
C-Store
Snacks
Juices
Emergency tools
Please confirm
Confirm
Please Proceed with Payment
Payment
Thank You For Using C-Store
Token
Car number: 101
Flashlights 20AED
Quantity: 4
11:22
22-10-2016
instinct takes over
toggle ads

# BIBLIOGRAPHY


- https://thenewboston.com/videos.php?cat=99
- http://stackoverflow.com/questions/17466561/best-way-to-structure-a-tkinter-application
- https://wiki.python.org/moin/TkInter
- https://docs.python.org/2/library/tkinter.html
- http://effbot.org/tkinterbook/