Restaurant Ordering/Rating System

Fast Eats

Done by:

A Pranav & Ashwin Reddy
Class 12 F1 February 2023

BONAFIDE CERTIFICATE

Certified to be the bonafide project work done by

Master / Miss ____

of Class____________in PADMA SESHADRI BALA BHAVAN SR. SEC. SCHOOL, CHENNAI.

During the year ________________________

Date____________________ P.G.T. in _________________________

Chennai

Submitted for All – India Senior Secondary Practical held in

____at

___________________________________Chennai.

Date _______________________ Examiner

Seal

Acknowledgements

This project would not have been possible without the help of my computer science teacher, Mr. Praveen Venkatachari, who guided us at every step of our project. I would like to thank my teammate who worked tirelessly with me to bring this project to life. I would like to thank the school management who gave us the wonderful opportunity to work on this project and provided us with the necessary resources to complete this project.

Finally, I would also like to thank my family and friends for their support without which the project wouldn’t be what it is today.

Index

Introduction

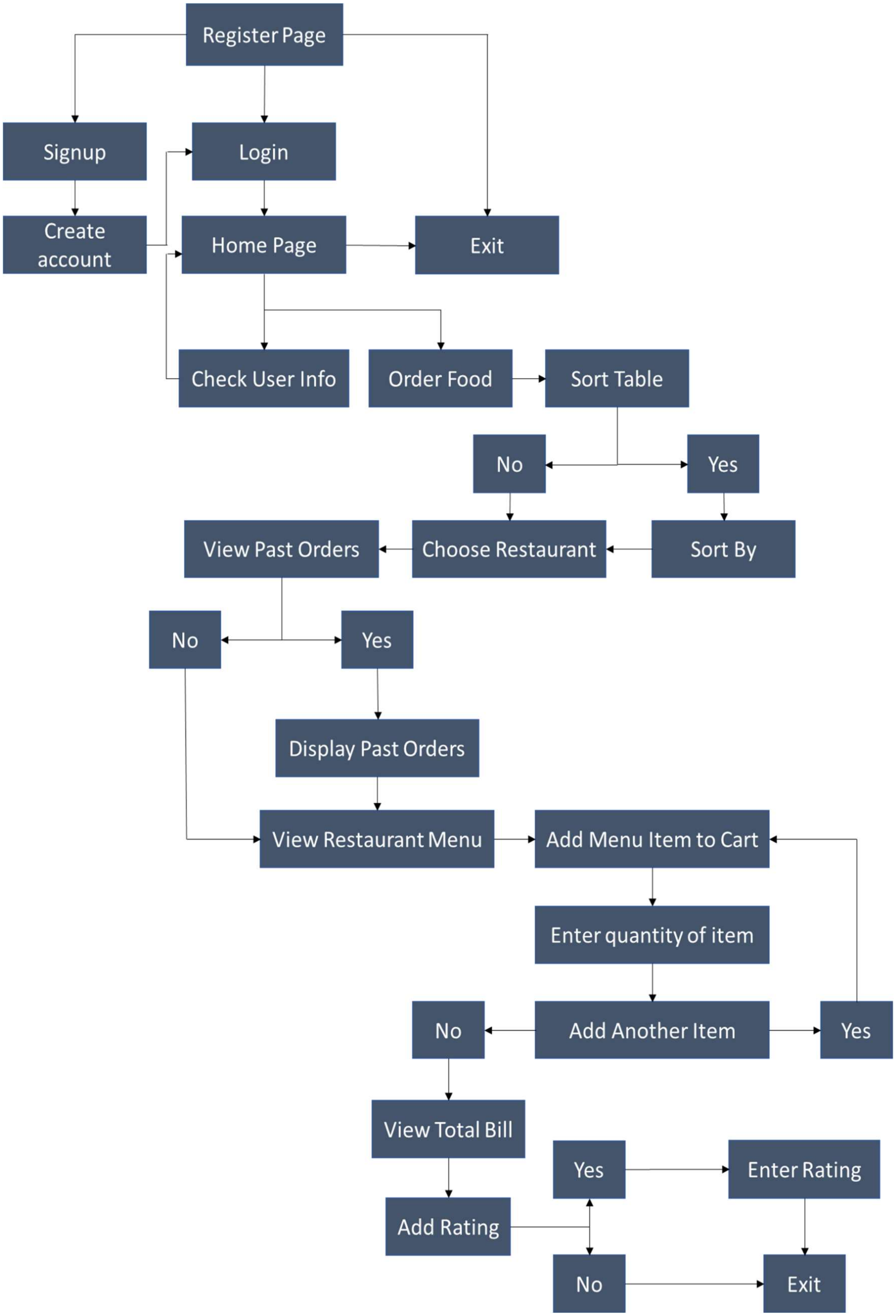
What is one common thing every human shares? Food. Food is a universal language and necessity for the human race to survive. An average adult consumes about 2 kilograms of food and if we multiply that by 8 billion, we can say that around 16 billion kilograms of food are consumed each and every day. However, preparing and cooking food is not an easy task. It takes long hours and tedious preparation for someone to make a meal that will get consumed in under half an hour. People saw this task of cooking a meal as a nuisance. This however gave birth to the restaurant and food service industry. People started paying someone else to cook food which would be consumed by them shortly.

Up till the 20th century this was mainly done in restaurants. A person had to go to a restaurant and physically order the food for consumption. After analyzing the problem further, they came up with the idea of delivery systems. A person could just send a message or call the restaurant and place the order which would then be prepared and delivered right to their doorstep. This was huge as someone who has just come home from a long day of work, or someone who feels tired or unmotivated would prefer to relax at home and enjoy a cooked meal delivered right to them rather than making their own food. This system was a huge success. But as time passed we could see some of the problems with the same. This process of calling and placing your orders had a few downsides. Firstly, multiple people are not able to call the restaurant and place an order at the same time. Therefore a person has to wait until the receiver is free to place their order. Secondly, this system requires a human interface at the receivers end to receive and note these orders down and pass it on to the chefs. This system hence allowed for human error to take place. A busy day might entail hundreds of orders, each one different from the former. It is a tedious job to keep track of these orders and note each one of them correctly. In an age of technological revelation and digitalization, the people get busier and these new technological devices and applications make lives easier. The On set of the 21th century has brought about many changes in our lifestyle and most of these have been for the better, making our lives much easier. After analyzing the problem with the current delivery system, people started to make online websites and applications where a person could easily choose a restaurant, pick their favorite dishes and these will be delivered to their doorstep within minutes. Everyone nowadays has phones and computers to access the internet and place an order. This was the coming of a new age of digital food service. This system was perfect, It had no human interaction so the room was error was very minimal. A person is able to clearly see what their options are and choose from a wide variety of them. And best of all, it requires absolutely minimal effort, just 3 clicks and their food is on the way. We saw the advantages that this system has and hence this was the inspiration behind our application. By removing unnecessary human intervention at every turn. We have made the process of having food delivered from restaurants much easier and a stress free task that anyone can do anywhere with the press of a few buttons.

Objective and Scope

The main objective of our product was to improve quality of life. We want to make the task of enjoying an exquisite meal at home something one can do with little to no effort. Our project gives users the freedom of choice. They are allowed to choose from a wide variety of their favorite restaurants on the go. Our secondary objective was to reduce the number of mediators between the person ordering the food and the chef that prepares it. This allows for minimal error and loss of information compared to when there are multiple mediators like the waiter, call receiver etc. Using our application, the orders would ideally appear on a small screen that was set-up in the restaurant kitchen to which the order was placed. This would then be noted by the chef who prepares the meal and sends it out for dispatch, where one of our own certified delivery men would pick up the order and deliver it to the users location. Our project has a lot of potential to be developed into a massive system like an online retailer. Currently we are partnered with restaurants and they are our main partners. However we would like to partner with ideally every restaurant in our city and slowly expand into other cities too . We can also start an online grocery retailing section of our app by partnering with local supermarkets and stores.

System Design



List of Datasets and Storage Units

Files

Name	Type	Purpose
Userdata.csv	CSV File	Used to store user info (Username and password)
Data.csv	CSV File	Contains restaurant data (Restaurant Name,Menu)
Rating.csv	CSV File	Contains user input Restaurant rating
Ratingavg.csv	CSV File	Contains average rating for each restaurant
Restloc.csv	CSV File	Contains the restaurant name and their respective locations in Chennai
Phoneno.dat	Binary File	Contains the previous orders of every user

Contents of Datasets

Userdata.csv

Encrypted Username	Encrypted Password
gAAAAABjj_X_3VL3OYlwfsGAfRBFXRBvDVkDPQtg1CZmoH8cw6n064Vz5Ou6GKHPrM_zsl7zVc4oOyC2c9QqUhNs9s2C-k1aig==	gAAAAABjj_X_zxvORaHUmX2QXRc2-w3aPBd7UZa4JuBHzbSh22w6Hs1edcXujU-c809kFhxaamwciN_h23VvidU37VCPm1sb0g==
gAAAAABjj_ZL-aYT3w6S0OUfZ-R7mEgFmRwhfQJPBeqkT8cd_Pf___3m9Ef7GVZVh8pPlUuUh7EJdht-EiB9fJ0j_xv8vxzMo6Q==	gAAAAABjj_ZLvfqdGuQw7_OHaMgmprO59ZoNgy3DkE6MXPKZF1iTzGyxoQFFm_nFTPPqBwjAZA_vBvc8-pqnp7r6kwhTj_mUuA==
gAAAAABjj_bfdbd_LUHePced3pUYrxRJJj_WXNKp5k0NE6OWUt6bN03gmit743QLkouGco9bGi_In4yM2NzFCI93INLOLn1oRQ==	gAAAAABjj_bf_3CS1pXH4rcCR2kaXAKoohx7yQmrNCHZHeZcOBiNew2JD79b98_wJMTHFMba1DIwNPNBOgwicXYdaPKL0s7rDQ==
gAAAAABjj_saltifOO1u7teezEd8y68JGjegWONPJLkkqpxCM-v8N6Hg4Pt63k0HeTH3amMqQuIgMat_5rG2P5z9qG2TPp-2hQ==	gAAAAABjj_saGbEv9AowWG9GBM4XFXUGENyCxW1OWtLCoq-Y8ehkPxeVLsTSJ8495AmtHxvC6RH70UzkAhfd6tPskKhErqjxCw==

Data.csv

Format for the following CSV File:

Menu →

Restaurant Name		
Item	**Veg/Non-Veg**	**Price**

Geetham Veg Restaurant		
Idli	Veg	50
Dosa	Veg	70
Ghee Roast	Veg	164
Butter Roast	Veg	164
Plain Uthappam	Veg	119
Onion Uthappam	Veg	138
Rice	Veg	100
Chapathi 2nos	Veg	106
Coffee	Veg	45
Horlicks	Veg	90
Tea	Veg	50
Milk	Veg	50

Saravana Bhavan		
Sambar Vadai 2nos	Veg	90
Curd Vadai 2nos	Veg	104.76
Chapati 2nos	Veg	76.19
Rice	Veg	80
Chapati	Veg	90
Idli	Veg	45
Poori	Veg	60
Dosai	Veg	75
Masala Dosai	Veg	90
Coffee	Veg	60
Dosa	Veg	70
A2B Veg		
Mini Idly	Veg	120
Sambar Idly	Veg	90
Masala Dosai	Veg	130
Onion Dosai	Veg	130

Poori Masala	Veg	105
Rava Kitchadi	Veg	90
Medhu Vadai	Veg	60
Sambar Vadai-1 Pc	Veg	80
Shree Mithai		
Samosa	Veg	58
Butter Muruku	Veg	130
Salted Potato Chips	Veg	68
Dal Kachori	Veg	58
Ribbon Muruku	Veg	90
Salted Triangle Puff	Veg	59
Vegetable Spring Roll	Veg	54
Pani Poori	Veg	61
Pav Bhaji	Veg	125
Bhel Poori	Veg	80

Rating.csv

Name of the Restaurant	Ratings
Geetham Veg Restaurant	4.1;4.1;4.0;4.0;4.1;4.1;3.8;4.1;1.0;3.0;
Saravana Bhavan	4.0;3.2;4.1;2.0;3.2;4.1;4.3;5.0;
Shree Mithai	1.0;2.3;4.4;3.2;4.6;5.0;4.5;3.2;4.3;4.4;
A2B Veg	4.3;4.3;3.2;5.0;4.3;4.8;4.9;4.2;3.2;
Manoj Bhavan Veg Restaurant	3.9;4.1;4.2;4.8;4.2;3.2;3.6;2.0;1.0;
Burger King	5.0;4.2;3.8;4.2;3.9;4.3;3.4;4.3;4.6;3.8;
KFC	3.8;4.1;3.2;5.0;4.6;4.1;3.9;4.8;5.0;3.8;4.1;
Subway	4.0;4.3;2.5;3.2;4.3;3.8;4.2;4.6;4.3;5.0;
Domino's Pizza	2.0;4.0;3.2;4.2;4.6;4.0;4.8;5.0;4.8;4.6;
Oven Story Pizza	3.9;4.1;0.5;4.1;3.8;4.1;4.8;5.0;4.8;5.0;4.8;
Pizza Hut	4.0;4.1;3.8;3.8;4.2;3.9;4.3;4.1;4.1;4.8;5.0;4.3;2.1;4.1;
The Bowl Company	3.8;4.1;4.1;3.8;4.1;4.0;3.8;4.1;4.1;4.2;4.1;
Cafe De Paris	3.9;4.2;2.0;4.3;1.0;4.3;5.0;4.8;4.3;3.6;4.2;
Krispy Kreme	4.1;3.8;4.3;3.0;3.0;2.8;4.1;3.2;5.0;
Writer's Cafe	4.1;3.8;4.2;3.9;4.2;3.8;4.3;3.9;3.2;3.9;4.1;
Roll Baby Roll	3.2;4.1;4.5;4.7;3.2;4.9;4.3;4.5;5.0;4.8;4.6;4.3;

The Sandwich Shop	4.2;3.2;3.8;2.5;5.0;4.2;3.8;2.9;5.0;4.6;5.0;
Sigree	4.3;3.6;3.9;2.6;4.0;4.6;1.0;0.5;4.5;4.6;4.8;4.3;
Chai Kings	4.2;3.2;4.9;5.0;3.2;2.9;3.8;2.0;5.0;3.8;4.3;4.2;3.8;5.0
Cake Works	4.6;1.0;0.5;3.2;5.0;4.5;3.9;4.6;5.0;2.8;4.9;4.2;

Ratingavg.csv

Name of the Restaurant	Average Rating	Number Of Ratings
Geetham Veg Restaurant	3.63	10
Saravana Bhavan	3.73	8
Shree Mithai	3.69	10
A2B Veg	4.24	9
Manoj Bhavan Veg Restaurant	3.44	9
Burger King	4.15	10
KFC	4.02	4
Subway	4.02	10
Domino's Pizza	4.1	6
Oven Story Pizza	4.08	11
Pizza Hut	3.89	14
The Bowl Company	4.01	11
Cafe De Paris	3.78	11
Krispy Kreme	3.7	9
Writer's Cafe	3.94	11
Roll Baby Roll	4.34	11

The Sandwich Shop	4.01	11
Sigree	3.49	11
Chai Kings	3.84	11
Cake Works	3.63	11

Restloc.csv

Name of the Restaurant	Location
Geetham Veg Restaurant	T Nagar
Saravana Bhavan	KK Nagar
A2B Veg	Ashok Nagar
Shree Mithai	Ashok Nagar
Manoj Bhavan Veg Restaurant	Ashok Nagar
Burger King	Ashok Nagar
KFC	Ashok Nagar
Subway	Valasaravakkam
Domino's Pizza	K.K Nagar
Oven Story Pizza	Vadapalani
Pizza Hut	Ashok Nagar
The Bowl Company	Vadapalani
Cafe De Paris	Alwarpet
Krispy Kreme	Thousand Lights
Writer's Cafe	Egmore
Roll Baby Roll	Nungambakkam

The Sandwich Shop	Kodambakkam
Sigree	Anna Nagar
Chai Kings	Egmore
Cake Works	Choolaimedu

Phoneno.dat

```
[['8432504059', datetime.datetime(2022, 12, 10, 18, 40, 0, 183847), 'Shree Mithai', [['S.No',
'Item', 'Quantity', 'Price'], [1, 'Bhel Poori', 3, 240], [2, 'Pav Bhaji', 4, 500], [3, 'Samosa Chaat', 1,
95]]], ['8432504059', datetime.datetime(2022, 12, 10, 18, 42, 57, 701400), 'Geetham Veg
Restaurant', [['S.No', 'Item', 'Quantity', 'Price'], [1, 'Idli', 5, 250], [2, 'Coffee', 2, 90], [3,
'Chapati', 3, 240]]], ['8432504059', datetime.datetime(2022, 12, 10, 18, 44, 25, 821757),
'Shree Mithai', [['S.No', 'Item', 'Quantity', 'Price'], [1, 'Samosa Chaat', 4, 380], [2, 'Bhel Poori',
2, 160]]], ['8432504059', datetime.datetime(2022, 12, 10, 18, 51, 42, 912191), 'Geetham Veg
Restaurant', [['S.No', 'Item', 'Quantity', 'Price'], [1, 'Idli', 5, 250], [2, 'Dosa', 3, 210], [3, 'Coffee',
3, 135]]], ['8432504059', datetime.datetime(2022, 12, 11, 14, 29, 57, 104080), 'Geetham Veg
Restaurant', [['S.No', 'Item', 'Quantity', 'Price'], [1, 'Idli', 3, 150]]]]
```

List of Global Variables and Functions

Global Variables

Global Variables	Purpose
restdict	Dictionary containing data of all restaurants
phoneno	Contains the phone number of the user
ratingdict	Dictionary containing the rating of respective restaurants
restchoice	Contains the name of the restaurant chosen by the user
averrest	Contains the average price of each restaurant
cart	Contains the list of food ordered by the user
ratelist	Contains the ratings of each restaurant
Key	A common to key encrypt all user data

User Defined Functions

User Defined Functions	Purpose
entersite()	Function for login/sign up
getdata()	Function to get data from data.csv file
getrestloc()	Function to get the restaurant location
averrestau(restdict)	Function to get the average price of each restaurant
dispavg(averrest, restdict, getrestloc())	Function to get the data for the restaurant which the user chose.
addtocart(restdict)	Function to get the order of the user
viewcart(cart)	Function to give the bill based on the order of the user
ratingscreate()	Function to create a file called ratefile which contains empty ratings in a particular format and provides rating in list format
ratingsavg()	To write the rating provided by the user to ratings.csv

Module Functions

Module	Function	Purpose
CSV	reader()	To read the contents of a csv file
CSV	writer()	To write content to a csv file
Cryptography	fernet()	To encrypt user data using a particular key
Cryptography	key.encrypt()	To encrypt user data using the key provided
Cryptography	key.decrypt()	To decrypt user data based on the key provided
Ast	literal_eval()	To evaluate expression to its literal value
Prettytable	prettytable()	To display the data in an aesthetically pleasing manner to the user
Decimal	decimal()	To convert a floating number to decimal for operations
Random	randint()	To make the login page eye catching
Pickle	load()	To load data from a binary file
Pickle	dump()	To dump data into a binary file
Time	sleep()	To add aesthetics to login page
Datetime	datetime.datetime.now()	To record the time at which an order is placed

Source Code

```
import csv
from prettytable import PrettyTable
from decimal import *
import random
import pickle
import time
import datetime as dt
from cryptography.fernet import Fernet


# ASKS THE USER WHETHER THEY WANT TO LOGIN OR SIGNUP
def entersite():
    print('''Welcome to Fast Eats!
    1.Sign up
    2.Login
    3.Exit''')
    choice = int(input("What would you like to do: "))
    if choice == 1:
        signup()
    elif choice == 2:
        login()
    elif choice == 3:
        quit()


key = Fernet(b'7FXASAwFtL74HPsAtwXMjTrmyAQM3-pUF_C6dpsGeF4=')


# ALLOWS THE USER TO SIGNUP
def signup():
    f = open('UserData.csv', 'a', newline='')
    w = csv.writer(f)
    while True:
        phoneno = input("Enter Phone number: ")
        if len(phoneno) == 10:
            break
        else:
            print("Enter Valid Phone Number!")
            continue
    while True:
        password = input("Enter password(Include an 
uppercase,lowercase,number and special character): ")
        conditions = [0, 0, 0, 0]
        for i in password:
            if i.isupper():
                conditions[0] = 1
            elif 33 <= ord(i) <= 47:
                conditions[3] = 1
            elif 58 <= ord(i) <= 64:
                conditions[3] = 1
            elif i.islower():
                conditions[1] = 1
            elif i in '0123456789':
                conditions[2] = 1
        for i in range(len(conditions)):
            if conditions[i] != 1 and i == 0:
                print("Please include an uppercase character!")
                i = 1
                break
            elif conditions[i] != 1 and i == 1:
                print("Please include a lowercase character!")
                i = 1
                break
            elif conditions[i] != 1 and i == 2:
                print("Please include a number!")
                i = 1
                break
            elif conditions[i] != 1 and i == 3:
                print("Please include a special character!")
                i = 1
                break
        if i == 1:
            continue
        else:
            break
    while True:
        repass = input("Please Re-Enter your password: ")
        if repass == password:
            all_u_data = []
            while True:
                try:
                    chck_data = pickle.load(f)
                    all_u_data.append(chck_data)
                except:
                    break
            for i in all_u_data:
                if all_u_data[i][0] == phoneno:
                    print("Account with given phone number
already exists!")
                    entersite()
                else:
                    break
                break
        else:
            print("Passwords do not match")
            continue
        break
    bytephonenum = bytes(phoneno, 'utf-8')
    encrypphonenum = key.encrypt(bytephonenum)
    encrypphonenum = str(encrypphonenum, 'utf-8')
    bytepassw = bytes(password, 'utf-8')
    encryppassw = key.encrypt(bytepassw)
    encryppassw = str(encryppassw, 'utf-8')
    w.writerow([encrypphonenum, encryppassw])
    f.close()
    print("Account has been created, Login to continue")
    login()


# ALLOWS THE USER TO LOGIN BASED ON PREVIOUSLY STORED USER
DETAILS
def login():
    f = open('UserData.csv', 'r')
    global phoneno
    global password
    phoneno = input("Enter Phone Number: ")
    password = input("Enter Password: ")
    r = csv.reader(f)
    all_u_data = list(r)
    loginorno = 0
    for i, j in all_u_data:
        i = i.lstrip("b'")
        i = i.rstrip("'")
        j = j.lstrip("b'")
        j = j.rstrip("'")
        if str(key.decrypt(bytes(i, 'utf-8')), 'utf-8') ==
phoneno and str(key.decrypt(bytes(j, 'utf-8')),

'utf-8') == password:
            print("Signing In", end='')
            y = random.randint(2,6)
            for i in range(y):
                time.sleep(0.5)
                print('.', end='')
                f.close()
            print("Successfully logged In!")
            loginorno = 1
    if loginorno == 0:
        time.sleep(1.5)
        print("Invalid Credentials!")
        entersite()


# VIEW USER INFO
def viewinfo():
    print("Phone Number:", phoneno)
    print("Password:", password)


# VIEW PREVIOUS ORDERS BASED ON RESTAURANT CHOSEN
def viewords(phoneno, restchoice):
    yorn = input("Would you like to view your past orders from
this restaurant?(Y/N)")
    if yorn.lower() == 'y':
        phstr = str(phoneno) + '.dat'
        try:
            f = open(phstr, 'rb')
            pastords = []
            while True:
                try:
                    data = pickle.load(f)
                    pastords.append(data)
                except:
                    break
            f.close()
            sno = 1
            checknum = 0
            for i in range(len(pastords)):
                if pastords[i][2] == restchoice:
                    print("Order Placed on", pastords[i][1],
"from", pastords[i][2])
                    checknum = 1
                    order = PrettyTable(pastords[i][3][0])
                    for j in pastords[i][3]:
                        if type(j[0]) == int:
                            order.add_row(j)
                    print(order)
            if checknum == 0:
                print("You have not placed any orders from this
restaurant!")
                time.sleep(2)
                return 0
            return 1
        except:
            print("You have not placed any orders from this 
restaurant!")
            time.sleep(2)
            return 0


# TO GET DATA FOR EACH RESTAURANT FROM A CSV FILE
def getdata():
    f = open('data.csv', 'r')
    r = csv.reader(f)
    data = list(r)
    d1 = {}
    global ratingdict
    ratingdict = {}
    f.close()
    f = open("rateavg.csv", "r")
    rateavg = csv.reader(f)
    rateavg = list(rateavg)
    for i in rateavg:
        if i != []:
            if i[1] != '':
                roundavg = round(Decimal(i[1]), 1)
                ratingdict[i[0]] = [roundavg, i[-1]]
            elif i[1] == '':
                roundavg = 0
                ratingdict[i[0]] = [roundavg, i[-1]]
    for i in data:
        if i == []:
            data.remove(i)
    for i in data:  # To create a dictionary----> {'Restaurant 
Name1':[[Food Name1,Price1],[Food Name2,Price2]],'Restaurant 
Name2':[[Food Name1,Price1],[Food Name2,Price2]]}
        if len(i) == 1:
            l1 = []
            restau = i[0]
            d1[restau] = l1
        elif len(i) != 1:
            l1.append(i)
    for i in d1:
        if i not in ratingdict:
            ratingdict[i] = [0, '0']
    return d1



# TO GET THE LOCATION OF EACH RESTAURANT
def getrestloc():
    f = open("restloc.csv", newline='')
    r = csv.reader(f)
    data = list(r)
    restlocdata = {}
    for i in data:
        restlocdata[(i[0])] = i[-1]
    return restlocdata


# TO FIND THE AVERAGE PRICE OF EACH RESTAURANT BASED ON THEIR
MENU
def averrestau(restdict):
    lowprice = ('a', 1000000000000000000000000000000000000000)
    averres = []
    for i in restdict:
        menu = restdict[i]
        price = []
        for items in menu:
            itemprice = float(items[-1])
            price.append(itemprice)
        sumprices = sum(price)
        average = round(Decimal(sumprices / len(price)), 1)
        averres.append((i, average))
    return averres


# TO DISPLAY THE DETAILS OF EACH RESTAURANT
def dispavg(averrest, restdict, locdata):
    from math import ceil
    myTable = PrettyTable(["Number", "Restaurant Name", "Average
Price", "Rating", "Number of Ratings", "Location"])
    print("Choose a restaurant using the numbers to order from:")
    locations = list(locdata.keys())
    restlist = []
    for i in range(len(averrest)):
        if ratingdict[averrest[i][0]][-1] == '':
            myTable.add_row(
                [i + 1, averrest[i][0], averrest[i][1],
ratingdict[averrest[i][0]][0], 0, locdata[locations[i]]])
            restlist.append(averrest[i][0])
        else:
            myTable.add_row(
                [i + 1, averrest[i][0], averrest[i][1],
ratingdict[averrest[i][0]][0], ratingdict[averrest[i][0]][-1],
                 locdata[locations[i]]])
            restlist.append(averrest[i][0])
    print(myTable)
    usersort = input("Would you like to sort this table(Y/N):")
    if usersort.lower() == "y":
        print('''Sort by:
        1.Restaurant Name
        2.Average Price
        3.Rating
        4.Location:'''))
        Typesort = int(input(“How Would you like to sort the
table(Enter Number): “))
        if typesort == 1:
            print(myTable.get_string(sortby="Restaurant Name"))
        elif typesort == 2:
            print(myTable.get_string(sortby="Average Price"))
        elif typesort == 3:
            print(myTable.get_string(sortby="Rating",
reversesort=True))
        elif typesort == 4:
            print(myTable.get_string(sortby="Location"))
    while True:
        restnum = int(input("Enter which restaurant you would
like to choose:"))
        if (restnum > 0 and restnum <= len(restlist)) and
type(restnum) == int:
            menu = restdict[restlist[restnum - 1]]
            break
        else:
            print("Enter Valid Restaurant Number!")
            continue
    restchoice = restlist[restnum - 1]
    if viewords(phoneno, restchoice):
        print("Continuing in 10 seconds!")
        time.sleep(10)
    myTable2 = PrettyTable(["Number", "Dishes", "Veg/Non Veg",
"Price"])
    n = 1
    for i in menu:
        myTable2.add_row([n, i[0], i[1], i[-1]])
        n += 1
    print(myTable2)
    return restlist[restnum - 1]


# TO CREATE THE CART OF THE USER
def addtocart(restdict):
    cart = {}
    i = 0
    while True:
        if i == 0:
            global restchoice
            restchoice = dispavg(averrest, restdict,
getrestloc())
            menu = restdict[restchoice]
            while True:
                foodchoice = int(
                    input("Enter Item Number of food item you
would like to add: "))
                if foodchoice > 0 and foodchoice <= len(menu):
                    break
                else:
                    print("Enter Valid Food Item Number!")
            while True:
                quantity = int(input("Enter quantity you would
like to order: "))
                if quantity > 0 and quantity < 50:
                    break
                elif quantity > 50:
                    print("The required quantity of food is not
available")
                else:
                    print("Enter a Valid Amount!")
            i += 1
        else:
            foodchoice = int(
                input("Enter Item Number of food item you would
like to add: "))
            quantity = int(input("Enter quantity you would like
to order: "))
            menu = restdict[restchoice]
        for items in menu:
            if items[0] == restdict[restchoice][foodchoice -
1][0]:
                itemprice = int(items[-1])
                price = itemprice
        for i in restdict:
            for j in range(len(restdict[restchoice])):
                if i == restchoice and
restdict[restchoice][foodchoice - 1][0] not in list(cart.keys()):
                    cart[restdict[i][foodchoice - 1][0]] =
(restchoice, price, quantity)
                    break
                elif i == restchoice and
restdict[restchoice][j][0] in list(cart.keys()):
                    quan = cart[restdict[restchoice][foodchoice -
1][0]][-1] + quantity
```

```
                    cart[restdict[restchoice][foodchoice - 1][0]]
= (restchoice, price, quan)
                    break
        yorn = input("Would you like to add another item(y/n)? ")
        if yorn.lower() == 'n':
            break
    return cart


# TO DISPLAY THE BILL BASED ON THE CART OF THE USER
def viewcart(cart):
    from math import ceil
    order = [["S.No", "Item", "Quantity", "Price"]]
    bill = PrettyTable(["S.No", "Item", "Quantity", "Price"])
    total = 0
    serialno = 1
    for i in cart:
        bill.add_row([serialno, i, cart[i][-1], (cart[i][1] *
cart[i][-1])])
        order.append([serialno, i, cart[i][-1], (cart[i][1] *
cart[i][-1])])
        total += ((cart[i][1]) * (cart[i][-1]))
        serialno += 1
    print(bill)
    print("Total = Rs.", total)
    print("GST = 18%")
    print("Grand Total = Rs.", ceil(total + total * 0.18), )
    phstr = str(phoneno) + '.dat'
    time = dt.datetime.now()
    f = open(phstr, 'ab')
    pickle.dump([phoneno, time, restchoice, order], f)
    f.close()


#FUNCTION TO FETCH NUMBER OF RATINGS A RESTAURANT HAS
def ratingscreate():
    ratingslist = []
    for i in restdict:
        ratelisele = [i, []]
        ratingslist.append(ratelisele)
        ratefile = open("rating.csv", "w")
        w = csv.writer(ratefile)
    for i in ratingslist:
        w.writerow(i)
    ratefile.close()
    with open("rating.csv") as f:
        r = csv.reader(f)
        l = list(r)
        no_ratings = []
        for i in l:
            no_ratings.append(i[-1])
    return no_ratings


# TO CREATE A FILE WITH THE AVERAGE RATING OF EACH RESTAURANT
def ratingavgcreate():
    ratingavglist = []
    for i in restdict:
        ratelisele = [i, '']
        ratingavglist.append(ratelisele)
        rateavgfile = open("rateavg.csv", "w")
        w = csv.writer(rateavgfile)
    for i in ratingavglist:
        w.writerow(i)
    rateavgfile.close()


# TO CREATE A FILE WITH THE RATING OF EACH RESTAURANT
def rating():
    l1 = list(cart.values())
    restname = l1[0][0]
    print("Thank You for making a purchase from", restname)
    yorn = input("Would you like to add a rating for the 
following restaurant(Y/N)?")
    while True:
        if yorn.lower() == "y":
            rating = input("Enter your rating for the following 
restaurant(_/5):")
            if float(rating) >= 0 and float(rating) <= 5:
                print("Your Feedback has been recorded!")
                ratefile = open("rating.csv",
                                "r")
                r = csv.reader(ratefile)
                ratings = list(r)
                ratefile.close()
                for i in ratings:
                    if i != []:
                        if i[0] == restname:
                            oldratings = i[1]
                            i[1] = oldratings + rating + ';'
                ratefile = open("rating.csv", "w")
                w1 = csv.writer(ratefile)
                for i in ratings:
                    if i != []:
                        w1.writerow(i)
                ratefile.close()
                break
            else:
                print("Please Enter Valid Rating!")
                continue
        else:
            print("Enjoy your food!")
            break


# TO UPDATE THE FILE BASED ON THE RATING PROVIDED BY THE USER
def ratingsavg():
    ratefile = open("rating.csv", "r")
    allrates = csv.reader(ratefile)
    allrates = list(allrates)
    ratefile.close()
    rateavgfile = open("rateavg.csv",
                       "r")
    r = csv.reader(rateavgfile)
    ratings = list(r)
    ratefile.close()
    rateavgfile = open("rateavg.csv", "w")
    w2 = csv.writer(rateavgfile)
    for i in allrates:
        if i != []:
            try:
                values = i[1].split(';')
                for j in range(len(values)):
                    if values[j] != '':
                        values[j] = float(values[j])
                    elif values[j] == '':
                        values.remove('')
                rateavg = sum(values) / len(values)
                l1 = [i[0], rateavg, len(values)]
                w2.writerow(l1)
            except:
                w2.writerow([i[0], 0, 0])


# TO DISPLAY A MENU TO ASK THE USER WHAT THEY WANT TO DO
def menu():
    print('''What would you like to do today?
        1.Check user info
        2.Order food
        3.Exit:''')
    userchoice = int(input(“Enter what you would like to do : “))
    if userchoice == 1:
        viewinfo()
        menu()
    elif userchoice == 2:
        global restdict
        restdict = getdata()
        global averrest
        averrest = averrestau(restdict)
        global cart
        cart = addtocart(restdict)
        viewcart(cart)
        ratefile = open("rating.csv", "r")
        ratelist = csv.reader(ratefile)
        ratelist = list(ratelist)
        if ratelist == []:
            ratingscreate()
        rating()
        ratingavgcreate()
        ratingsavg()
    elif choice == 3:
        print("Thank You, Have a nice day!")


entersite()
menu()
```

Sample Output

Logging in to the application

```
Welcome to Fast Eats!
    1.Sign up
    2.Login
    3.Exit
What would you like to do: 1
Enter Phone number: 8432504059
Enter password(Include an uppercase,lowercase,number and special
character): P@ss123
Please Re-Enter your password: P@ss123
Account has been created, Login to continue
Enter Phone Number: 8432504059
Enter Password: P@ss123
Signing In……Successfully logged In!
```

Checking Account Details

```
What would you like to do today?
        1.Check user info
        2.Order food
        3.Exit
Enter what you would like to do :1
Phone Number: 8432504059
Password: P@ss123
```

Ordering Food

```
What would you like to do today?
        1.Check user info
        2.Order food
        3.Exit
Enter what you would like to do :2
Choose a restaurant using the numbers to order from:
```

```
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
| Number |       Restaurant Name       | Average Price | Rating | Number of Ratings |    Location     |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
|   1    |   Geetham Veg Restaurant    |     95.5      |  4.1   |         6         |     T Nagar     |
|   2    |       Saravana Bhavan       |     76.5      |  4.0   |         9         |    KK Nagar     |
|   3    |           A2B Veg           |    100.6      |  4.2   |         9         |   Ashok Nagar   |
|   4    |        Shree Mithai         |     78.3      |  4.4   |         9         |   Ashok Nagar   |
|   5    | Manoj Bhavan Veg Restaurant |    157.3      |  3.4   |         9         |   Ashok Nagar   |
|   6    |         Burger King         |    248.5      |  4.1   |         10        |   Ashok Nagar   |
|   7    |             KFC             |    333.9      |  4.0   |         4         |   Ashok Nagar   |
|   8    |            Subway           |    225.6      |  4.0   |         10        | Valasaravakkam  |
|   9    |        Domino's Pizza       |    599.0      |  4.1   |         10        |    K.K Nagar    |
|   10   |      Oven Story Pizza       |    378.5      |  4.1   |         11        |    Vadapalani   |
|   11   |          Pizza Hut          |    345.6      |  4.0   |         13        |   Ashok Nagar   |
|   12   |      The Bowl Company       |    206.5      |  4.0   |         9         |    Vadapalani   |
|   13   |        Cafe De Paris        |    394.4      |  3.8   |         11        |    Alwarpet     |
|   14   |         Krispy Kreme        |    156.5      |  3.7   |         9         | Thousand Lights |
|   15   |        Writer's Cafe        |    205.0      |  3.9   |         11        |      Egmore     |
|   16   |        Roll Baby Roll       |    131.2      |  4.3   |         11        |   Nungambakkam  |
|   17   |      The Sandwich Shop      |    165.0      |  4.0   |         11        |   Kodambakkam   |
|   18   |            Sigree           |    387.5      |  3.5   |         11        |    Anna Nagar   |
|   19   |          Chai Kings         |    214.9      |  3.8   |         11        |      Egmore     |
|   20   |          Cake Works         |    538.5      |  3.6   |         11        |   Choolaimedu   |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
```

Sorting the Restaurants

```
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
| Number |       Restaurant Name       | Average Price | Rating | Number of Ratings |    Location     |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
|   1    |   Geetham Veg Restaurant    |     95.5      |  4.1   |         6         |     T Nagar     |
|   2    |       Saravana Bhavan       |     76.5      |  4.0   |         9         |    KK Nagar     |
|   3    |           A2B Veg           |    100.6      |  4.2   |         9         |   Ashok Nagar   |
|   4    |        Shree Mithai         |     78.3      |  4.4   |         9         |   Ashok Nagar   |
|   5    | Manoj Bhavan Veg Restaurant |    157.3      |  3.4   |         9         |   Ashok Nagar   |
|   6    |         Burger King         |    248.5      |  4.1   |         10        |   Ashok Nagar   |
|   7    |             KFC             |    333.9      |  4.0   |         4         |   Ashok Nagar   |
|   8    |            Subway           |    225.6      |  4.0   |         10        | Valasaravakkam  |
|   9    |        Domino's Pizza       |    599.0      |  4.1   |         10        |    K.K Nagar    |
|   10   |      Oven Story Pizza       |    378.5      |  4.1   |         11        |    Vadapalani   |
|   11   |          Pizza Hut          |    345.6      |  4.0   |         13        |   Ashok Nagar   |
|   12   |      The Bowl Company       |    206.5      |  4.0   |         9         |    Vadapalani   |
|   13   |        Cafe De Paris        |    394.4      |  3.8   |         11        |    Alwarpet     |
|   14   |         Krispy Kreme        |    156.5      |  3.7   |         9         | Thousand Lights |
|   15   |        Writer's Cafe        |    205.0      |  3.9   |         11        |      Egmore     |
|   16   |        Roll Baby Roll       |    131.2      |  4.3   |         11        |   Nungambakkam  |
|   17   |      The Sandwich Shop      |    165.0      |  4.0   |         11        |   Kodambakkam   |
|   18   |            Sigree           |    387.5      |  3.5   |         11        |    Anna Nagar   |
|   19   |          Chai Kings         |    214.9      |  3.8   |         11        |      Egmore     |
|   20   |          Cake Works         |    538.5      |  3.6   |         11        |   Choolaimedu   |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
Would you like to sort this table(Y/N):Y
```

Sorting By Name:

```
Sort by:
1.Restaurant Name
2.Average Price
3.Rating
4.Location
How Would you like to sort the table(Enter Number): 1
```

```
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
| Number |       Restaurant Name       | Average Price | Rating | Number of Ratings |     Location    |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
|   3    |           A2B Veg           |     100.6     |  4.2   |         9         |   Ashok Nagar   |
|   6    |         Burger King         |     248.5     |  4.1   |        10         |   Ashok Nagar   |
|   13   |        Cafe De Paris        |     394.4     |  3.8   |        11         |     Alwarpet    |
|   20   |          Cake Works         |     538.5     |  3.6   |        11         |   Choolaimedu   |
|   19   |          Chai Kings         |     214.9     |  3.8   |        11         |      Egmore     |
|   9    |        Domino's Pizza       |     599.0     |  4.1   |        10         |    K.K Nagar    |
|   1    |    Geetham Veg Restaurant   |      95.5     |  4.1   |         6         |     T Nagar     |
|   7    |             KFC             |     333.9     |  4.0   |         4         |   Ashok Nagar   |
|   14   |         Krispy Kreme        |     156.5     |  3.7   |         9         | Thousand Lights |
|   5    | Manoj Bhavan Veg Restaurant |     157.3     |  3.4   |         9         |   Ashok Nagar   |
|   10   |       Oven Story Pizza      |     378.5     |  4.1   |        11         |    Vadapalani   |
|   11   |          Pizza Hut          |     345.6     |  4.0   |        13         |   Ashok Nagar   |
|   16   |        Roll Baby Roll       |     131.2     |  4.3   |        11         |   Nungambakkam  |
|   2    |       Saravana Bhavan       |      76.5     |  4.0   |         9         |     KK Nagar    |
|   4    |         Shree Mithai        |      78.3     |  4.4   |         9         |   Ashok Nagar   |
|   18   |            Sigree           |     387.5     |  3.5   |        11         |    Anna Nagar   |
|   8    |            Subway           |     225.6     |  4.0   |        10         |  Valasaravakkam |
|   12   |       The Bowl Company      |     206.5     |  4.0   |         9         |    Vadapalani   |
|   17   |      The Sandwich Shop      |     165.0     |  4.0   |        11         |   Kodambakkam   |
|   15   |        Writer's Cafe        |     205.0     |  3.9   |        11         |      Egmore     |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
```

Sorting by Average Price:

```
Sort by:
1.Restaurant Name
2.Average Price
3.Rating
4.Location
How Would you like to sort the table(Enter Number): 2
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
| Number |       Restaurant Name       | Average Price | Rating | Number of Ratings |     Location    |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
|   2    |       Saravana Bhavan       |      76.5     |  4.0   |         9         |     KK Nagar    |
|   4    |         Shree Mithai        |      78.3     |  4.4   |         9         |   Ashok Nagar   |
|   1    |    Geetham Veg Restaurant   |      95.5     |  4.1   |         6         |     T Nagar     |
|   3    |           A2B Veg           |     100.6     |  4.2   |         9         |   Ashok Nagar   |
|   16   |        Roll Baby Roll       |     131.2     |  4.3   |        11         |   Nungambakkam  |
|   14   |         Krispy Kreme        |     156.5     |  3.7   |         9         | Thousand Lights |
|   5    | Manoj Bhavan Veg Restaurant |     157.3     |  3.4   |         9         |   Ashok Nagar   |
|   17   |      The Sandwich Shop      |     165.0     |  4.0   |        11         |   Kodambakkam   |
|   15   |        Writer's Cafe        |     205.0     |  3.9   |        11         |      Egmore     |
|   12   |       The Bowl Company      |     206.5     |  4.0   |         9         |    Vadapalani   |
|   19   |          Chai Kings         |     214.9     |  3.8   |        11         |      Egmore     |
|   8    |            Subway           |     225.6     |  4.0   |        10         |  Valasaravakkam |
|   6    |         Burger King         |     248.5     |  4.1   |        10         |   Ashok Nagar   |
|   7    |             KFC             |     333.9     |  4.0   |         4         |   Ashok Nagar   |
|   11   |          Pizza Hut          |     345.6     |  4.0   |        13         |   Ashok Nagar   |
|   10   |       Oven Story Pizza      |     378.5     |  4.1   |        11         |    Vadapalani   |
|   18   |            Sigree           |     387.5     |  3.5   |        11         |    Anna Nagar   |
|   13   |        Cafe De Paris        |     394.4     |  3.8   |        11         |     Alwarpet    |
|   20   |          Cake Works         |     538.5     |  3.6   |        11         |   Choolaimedu   |
|   9    |        Domino's Pizza       |     599.0     |  4.1   |        10         |    K.K Nagar    |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
```

Sorting By Rating:

```
Sort by:
1.Restaurant Name
2.Average Price
3.Rating
4.Location
How Would you like to sort the table(Enter Number): 3
```

```
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
| Number |       Restaurant Name       | Average Price | Rating | Number of Ratings |     Location    |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
|   4    |         Shree Mithai        |      78.3     |  4.4   |         9         |   Ashok Nagar   |
|   16   |        Roll Baby Roll       |     131.2     |  4.3   |         11        |   Nungambakkam  |
|   3    |           A2B Veg           |     100.6     |  4.2   |         9         |   Ashok Nagar   |
|   10   |       Oven Story Pizza      |     378.5     |  4.1   |         11        |    Vadapalani   |
|   9    |        Domino's Pizza       |     599.0     |  4.1   |         10        |    K.K Nagar    |
|   6    |         Burger King         |     248.5     |  4.1   |         10        |   Ashok Nagar   |
|   1    |    Geetham Veg Restaurant   |      95.5     |  4.1   |         6         |     T Nagar     |
|   17   |      The Sandwich Shop      |     165.0     |  4.0   |         11        |   Kodambakkam   |
|   12   |       The Bowl Company      |     206.5     |  4.0   |         9         |    Vadapalani   |
|   11   |          Pizza Hut          |     345.6     |  4.0   |         13        |   Ashok Nagar   |
|   8    |            Subway           |     225.6     |  4.0   |         10        |  Valasaravakkam |
|   7    |             KFC             |     333.9     |  4.0   |         4         |   Ashok Nagar   |
|   2    |       Saravana Bhavan       |      76.5     |  4.0   |         9         |     KK Nagar    |
|   15   |        Writer's Cafe        |     205.0     |  3.9   |         11        |      Egmore     |
|   19   |          Chai Kings         |     214.9     |  3.8   |         11        |      Egmore     |
|   13   |        Cafe De Paris        |     394.4     |  3.8   |         11        |     Alwarpet    |
|   14   |         Krispy Kreme        |     156.5     |  3.7   |         9         | Thousand Lights |
|   20   |          Cake Works         |     538.5     |  3.6   |         11        |   Choolaimedu   |
|   18   |            Sigree           |     387.5     |  3.5   |         11        |    Anna Nagar   |
|   5    | Manoj Bhavan Veg Restaurant |     157.3     |  3.4   |         9         |   Ashok Nagar   |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
```

Sorting By Location:

```
Sort by:
1.Restaurant Name
2.Average Price
3.Rating
4.Location
How Would you like to sort the table(Enter Number): 4
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
| Number |       Restaurant Name       | Average Price | Rating | Number of Ratings |     Location    |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
|   13   |        Cafe De Paris        |     394.4     |  3.8   |         11        |     Alwarpet    |
|   18   |            Sigree           |     387.5     |  3.5   |         11        |    Anna Nagar   |
|   3    |           A2B Veg           |     100.6     |  4.2   |         9         |   Ashok Nagar   |
|   4    |         Shree Mithai        |      78.3     |  4.4   |         9         |   Ashok Nagar   |
|   5    | Manoj Bhavan Veg Restaurant |     157.3     |  3.4   |         9         |   Ashok Nagar   |
|   6    |         Burger King         |     248.5     |  4.1   |         10        |   Ashok Nagar   |
|   7    |             KFC             |     333.9     |  4.0   |         4         |   Ashok Nagar   |
|   11   |          Pizza Hut          |     345.6     |  4.0   |         13        |   Ashok Nagar   |
|   20   |          Cake Works         |     538.5     |  3.6   |         11        |   Choolaimedu   |
|   15   |        Writer's Cafe        |     205.0     |  3.9   |         11        |      Egmore     |
|   19   |          Chai Kings         |     214.9     |  3.8   |         11        |      Egmore     |
|   9    |        Domino's Pizza       |     599.0     |  4.1   |         10        |    K.K Nagar    |
|   2    |       Saravana Bhavan       |      76.5     |  4.0   |         9         |     KK Nagar    |
|   17   |      The Sandwich Shop      |     165.0     |  4.0   |         11        |   Kodambakkam   |
|   16   |        Roll Baby Roll       |     131.2     |  4.3   |         11        |   Nungambakkam  |
|   1    |    Geetham Veg Restaurant   |      95.5     |  4.1   |         6         |     T Nagar     |
|   14   |         Krispy Kreme        |     156.5     |  3.7   |         9         | Thousand Lights |
|   10   |       Oven Story Pizza      |     378.5     |  4.1   |         11        |    Vadapalani   |
|   12   |       The Bowl Company      |     206.5     |  4.0   |         9         |    Vadapalani   |
|   8    |            Subway           |     225.6     |  4.0   |         10        |  Valasaravakkam |
+--------+-----------------------------+---------------+--------+-------------------+-----------------+
```

Viewing Past Orders:

```
Enter which restaurant you would like to choose:3
Would you like to view your past orders from this
restaurant?(Y/N)y
Order Placed on 2022-12-10 18:40:00.183847 from Shree Mithai
```

```
+------+--------------+----------+-------+
| S.No |     Item     | Quantity | Price |
+------+--------------+----------+-------+
|  1   |  Bhel Poori  |    3     |  240  |
|  2   |  Pav Bhaji   |    4     |  500  |
|  3   | Samosa Chaat |    1     |   95  |
+------+--------------+----------+-------+
Order Placed on 2022-12-10 18:44:25.821757 from Shree Mithai
+------+--------------+----------+-------+
| S.No |     Item     | Quantity | Price |
+------+--------------+----------+-------+
|  1   | Samosa Chaat |    4     |  380  |
|  2   |  Bhel Poori  |    2     |  160  |
+------+--------------+----------+-------+
Continuing in 10 seconds!
```

Ordering from a restaurant

```
Enter which restaurant you would like to choose:1
+--------+---------+------------+-------+
| Number |  Dishes | Veg/Non Veg | Price |
+--------+---------+------------+-------+
|   1    |   Idli  |     50     |   50  |
|   2    |   Dosa  |     70     |   70  |
|   3    |  Coffee |     45     |   45  |
|   4    |   Rice  |    100     |  100  |
|   5    | Chapati |     80     |   80  |
+--------+---------+------------+-------+
Enter Item Number of food item you would like to add: 1
Enter quantity you would like to order: 5
Would you like to add another item(y/n)? y
Enter Item Number of food item you would like to add: 2
Enter quantity you would like to order: 3
Would you like to add another item(y/n)? y
Enter Item Number of food item you would like to add: 3
Enter quantity you would like to order: 3
Would you like to add another item(y/n)? n
+------+--------+----------+-------+
| S.No |  Item  | Quantity | Price |
+------+--------+----------+-------+
|  1   |  Idli  |    5     |  250  |
|  2   |  Dosa  |    3     |  210  |
|  3   | Coffee |    3     |  135  |
+------+--------+----------+-------+
Total = Rs. 595
GST = 18%
Grand Total = Rs. 703
Thank You for making a purchase from Geetham Veg Restaurant
```

Rating A Restaurant:

```
Would you like to add a rating for the following
restaurant(Y/N)?y
Enter your rating for the following restaurant(_/5):3
Your Feedback has been recorded
```

Challenges, Limitations and the Future

Challenges:

- Hard to Work with data stored in CSV as it is retrieved as string.
- Difficulty fetching user data for login function.
- Tough to handle lots of data sets.

Limitations:

- No limit to quantity of food that the user can purchase.
- Unable to predict delivery time as that would require gps integration.
- Limited Menu items and same menu throughout the day [menu doesn't change for breakfast, lunch or dinner].
- No special offers or deals currently.
- Datasets must be available on every device.
- Code cannot run every possible action user can take at once.
- Unable to simulate real time situations .
- Limited Knowledge about python-cloud integration.

Future Scope:

To overcome these limitations we would ideally like to implement a cloud based server user interface where the restaurateur can update their menus and items based on real time. We would also like to add an option where someone could duplicate their previous order and then make changes to that. Partnering with more up and coming restaurants and local cafes to increase the number of options that the user has. We would also like to add a maximum deliverable distance threshold as some restaurants will not be able to deliver high quality food past a specific distance. Adding city based restaurant list since different cities and localities has different restaurants.

Bibliography

- https://www.geeksforgeeks.org/fernet-symmetric-encryption-using-cryptography-module-in-python/

- https://www.geeksforgeeks.org/creating-tables-with-prettytable-library-python/

- https://stackoverflow.com/questions/606191/convert-bytes-to-a-string

- CBSE Class 12 NCERT Textbook

- Computer Science with Python : Textbook for CBSE Class 12, Preeti Arora, Sultan Chand & Sons (P) Ltd, 2022.