

# Tiffin / Food Ordering system

Using python language



python

**Candidate Name:** Shrawan Limbu

**Candidate Email ID:** [50284@itcinfotech.com](mailto:50284@itcinfotech.com)

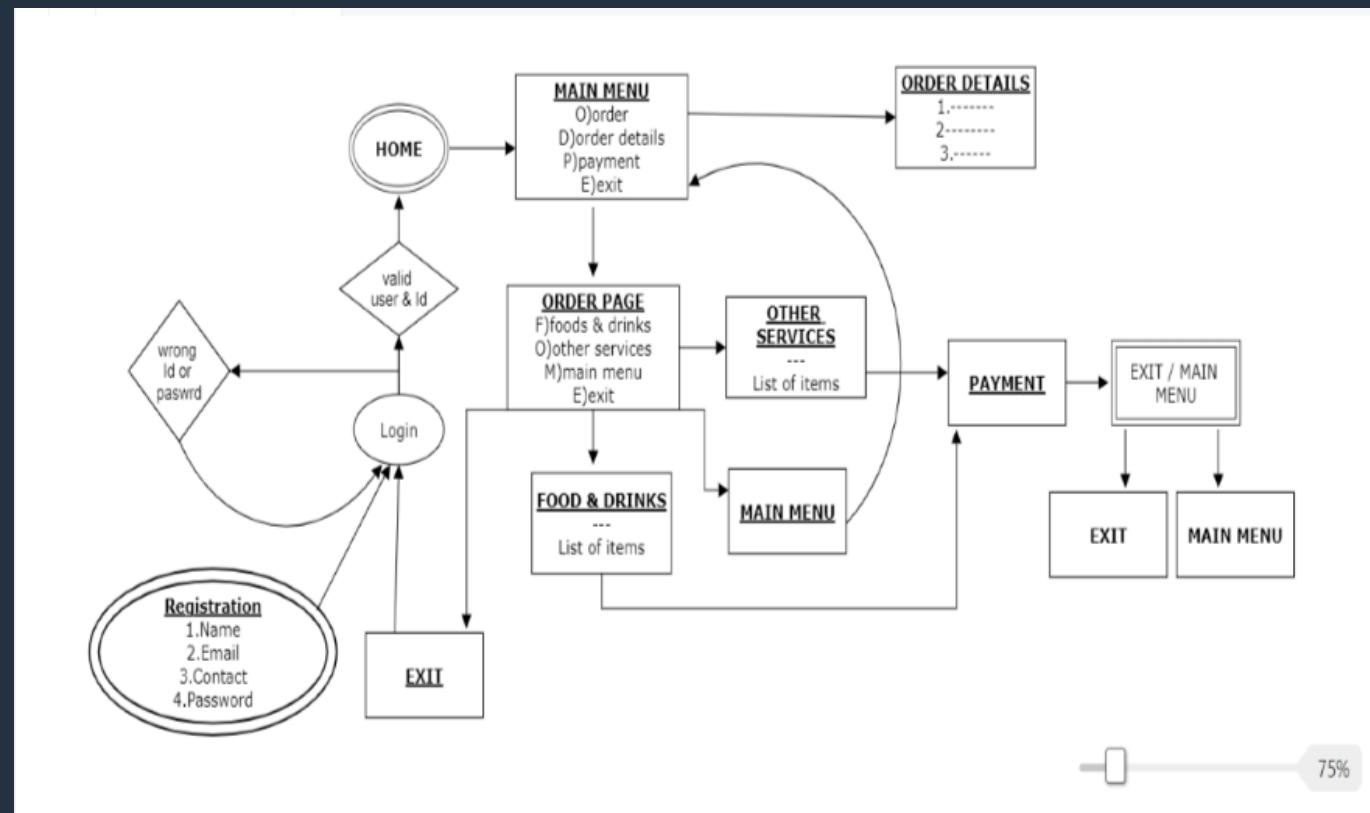
**Batch Name:** Digital-Python-DevOps

**Project Title:** Tiffin ordering system using python

**Date:** 10/10/2022 - 14/10/2022

**Project Manager (s):** Ramapavan.S

# ER-DIAGRAM:



## **a. Business Specification/Requirements:**

- **1.User able to register him/her details.**
- **2.After successfully registration he/she be able to login, using saved credentials.**
- **3.Home should contain 5 options: 1)order food. 2)other services. 3)order details. 4)contact direct. 5)exit.**
- **4.Order details should show bill and discount applied.**

## **b. Functional Output/Requirements:**

- **1. It should be User friendly.**
- **2. Compatibility Mac OS / window.**
- **3. With each step user will be given option to select, based on which user will instruct program to work.**

## c. Tool Chain/Technical Coverage:

- **1.python for coding language.**
- **2.VS code for coding.**
- **3.Notepad as to save details of items.**
- **4.Git/github.**
- **5.AWS to deploy.**



Visual Studio Code



File Edit Selection View Go Run Terminal Help index.py - Food Ordering System - Visual Studio Code

EXPLORER OPEN EDITORS 1 unsaved index.py User\_Data.txt

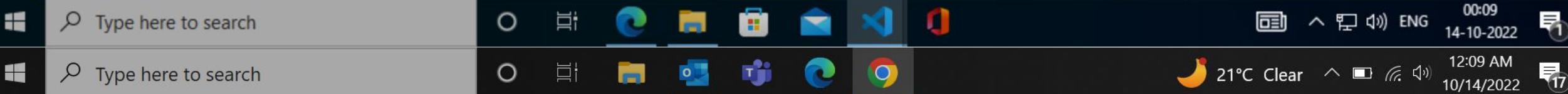
FOOD ORDERING SYST... files list\_drinks.fsd list\_foods.fsd list\_services.fsd report.fsd

index.py

```
1 print("\n" * 5)
2
3 from dataclasses import dataclass
4 import datetime
5 import os
6
7
8
9 list_foods = []
10 list_drinks = []
11 list_services = []
12
13 list_item_price = [0] * 100
14
15 var_discount_1 = 200
16 var_discount_2 = 1000
17 var_discount_3 = 5000
18 var_discount_1_rate = 0.05
19 var_discount_2_rate = 0.10
20 var_discount_3_rate = 0.15
21
22
23
```

Activate Windows  
Go to Settings to activate Windows.

Blackbox Found Code



# FUNCTION #01

**(def def\_default()):**

**It has been created for the global list of item which will be called when user asks for the list of food/services items.**

```
OnePOINT Sign In ERD -- SmartDraw
File Help • index.py - Food Ordering System - Visual Studio Code
index.py User_Data.txt
index.py > def_default
25     navigator_symbol = "\\\""
26
27
28     def def_default():
29         global list_drinks, list_foods, list_services, list_item_order, l
30         list_item_order= [0] * 100
31     def_default()
32
33     def def_main():
34
35
36         print("*" * 28 + "TIFFIN ORDERING SYSTEM" + "*" * 24 + "\n")
37         print("*" * 31 + "MAIN MENU" + "*" * 32 + "\n")
38             "\t(O) ORDER\n"
39             "\t(D) ORDER DETAILS\n"
40             "\t(P) PAYMENT\n"
41             "\t(C) CONTACT DIRECTLY\n"
42             "\t(E) EXIT\n"+
43             " _ " * 72)
44
45
46         input_1 = str(input("Please Select Your Operation: ")).upper()
47         if (len(input_1) == 1):
Ln 30, Col 20 Spaces:
```

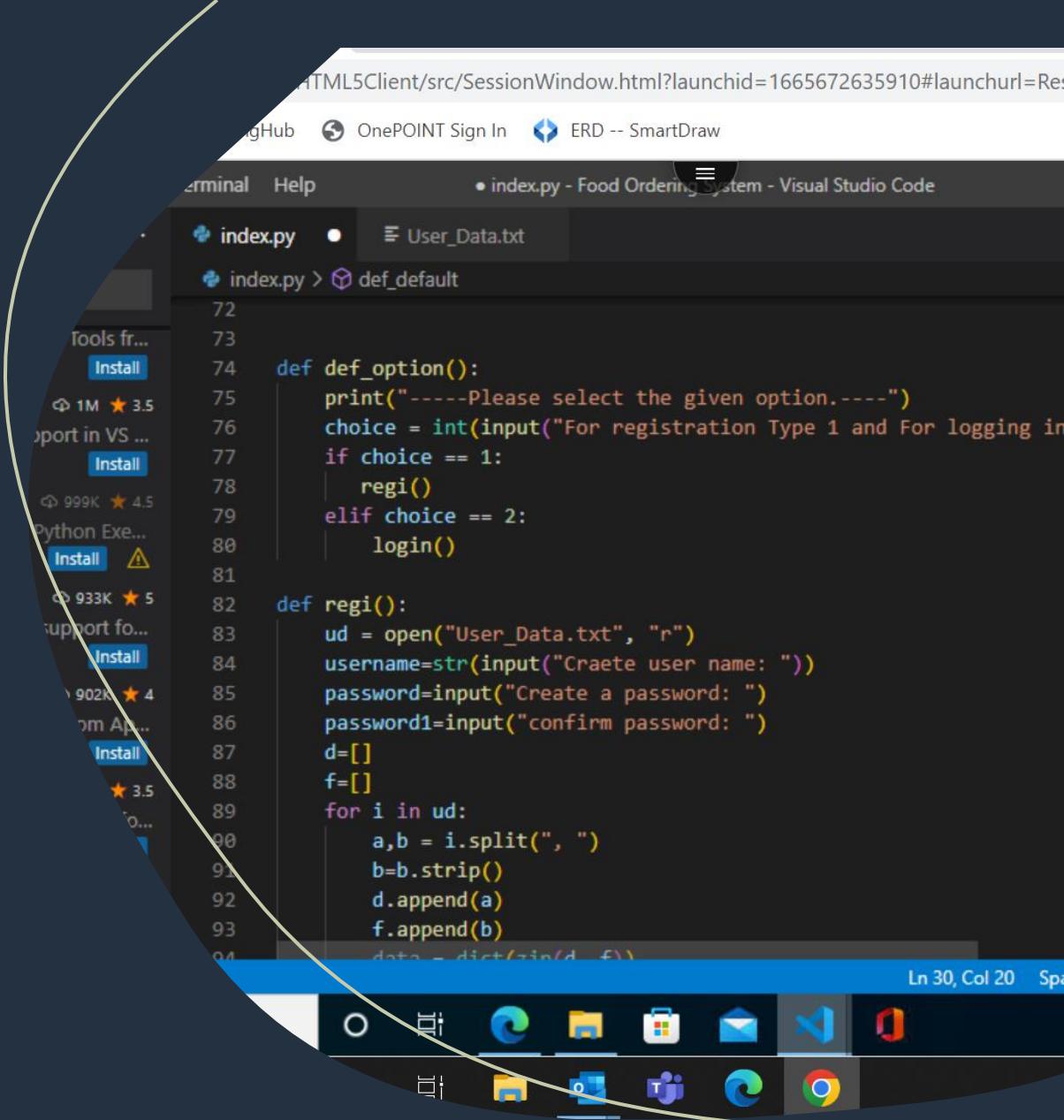
## Function #02

**(def def\_option()):**

**It given option to the user for  
REGISTER or LOGIN.**

**If the user press '1' he/she will  
reach to registration part.**

**If the user press '2' he/she will  
reach to login part.**

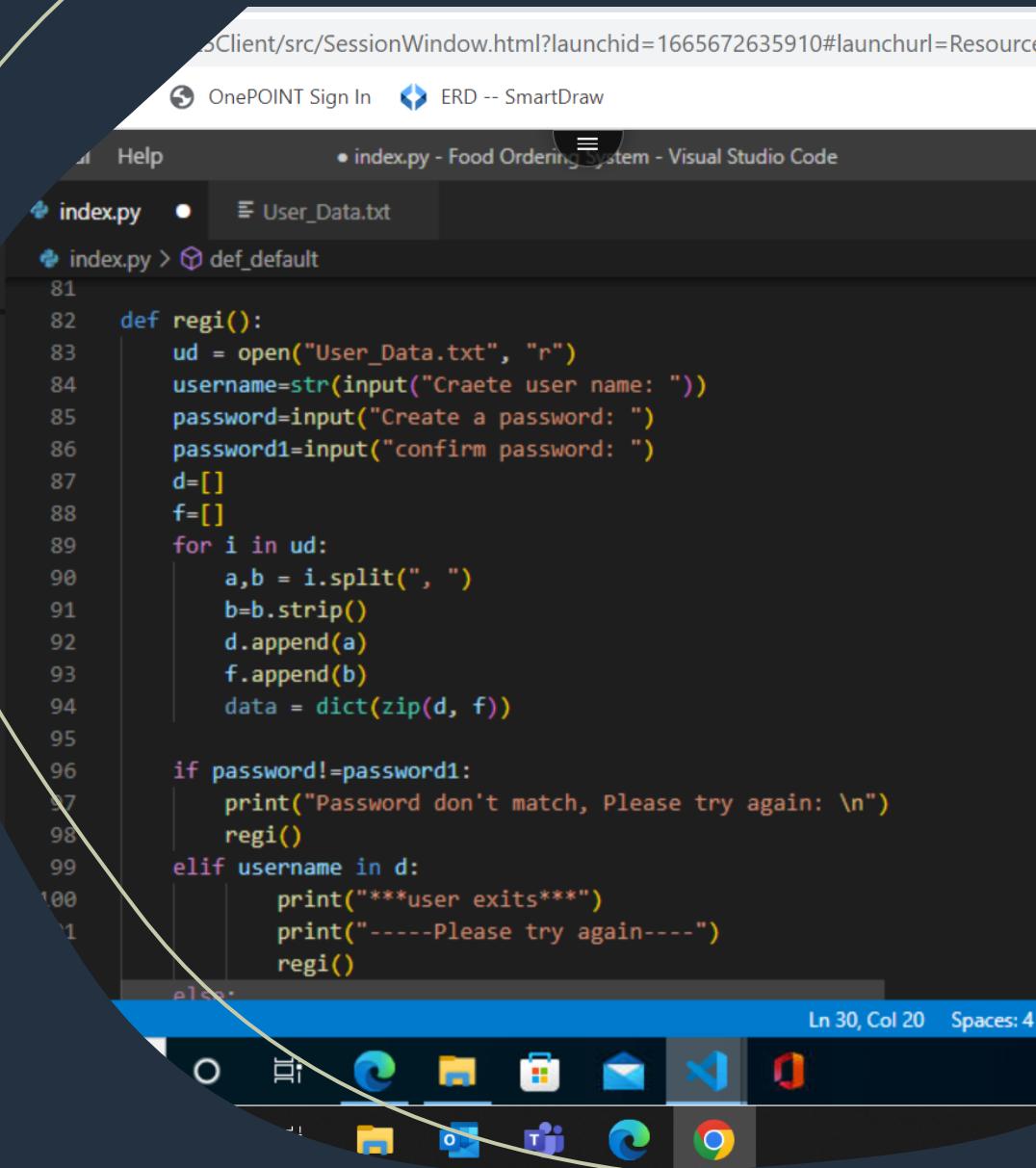


```
index.py - Food Ordering System - Visual Studio Code
index.py User_Data.txt
index.py > def_default
72
73
74 def def_option():
75     print("----Please select the given option.----")
76     choice = int(input("For registration Type 1 and For logging in"))
77     if choice == 1:
78         regi()
79     elif choice == 2:
80         login()
81
82 def regi():
83     ud = open("User_Data.txt", "r")
84     username=str(input("Create user name: "))
85     password=input("Create a password: ")
86     password1=input("confirm password: ")
87     d=[]
88     f=[]
89     for i in ud:
90         a,b = i.split(", ")
91         b=b.strip()
92         d.append(a)
93         f.append(b)
94     data = dict(zip(d,f))
Ln 30, Col 20  Sp
```

# Function #03

**(def regi()):**

**This function allows user to register him/her to access the inside functions.**



The screenshot shows a Microsoft Visual Studio Code window with the following details:

- Title Bar:** OnePOINT Sign In, ERD -- SmartDraw, index.py - Food Ordering System - Visual Studio Code
- File Explorer:** Shows various files and extensions installed (e.g., Pylance, GitLens, Python, Prettier, Eslint).
- Code Editor:** The file "index.py" is open, containing the following code:

```
def regi():
    ud = open("User_Data.txt", "r")
    username=str(input("Create user name: "))
    password=input("Create a password: ")
    password1=input("confirm password: ")
    d=[]
    f=[]
    for i in ud:
        a,b = i.split(", ")
        b=b.strip()
        d.append(a)
        f.append(b)
    data = dict(zip(d, f))

    if password!=password1:
        print("Password don't match, Please try again: \n")
        regi()
    elif username in d:
        print("***user exists***")
        print("-----Please try again----")
        regi()
    else:
```
- Status Bar:** Ln 30, Col 20 Spaces: 4

# Function #04

**(def\_login()):**

**Inside this function the user will use the credentials:**

## 1. Username

## 2. Password

## To enter the main menu.

# Function #05

**(def def\_main()):**

**In this function there will be 5 options available for the user:**

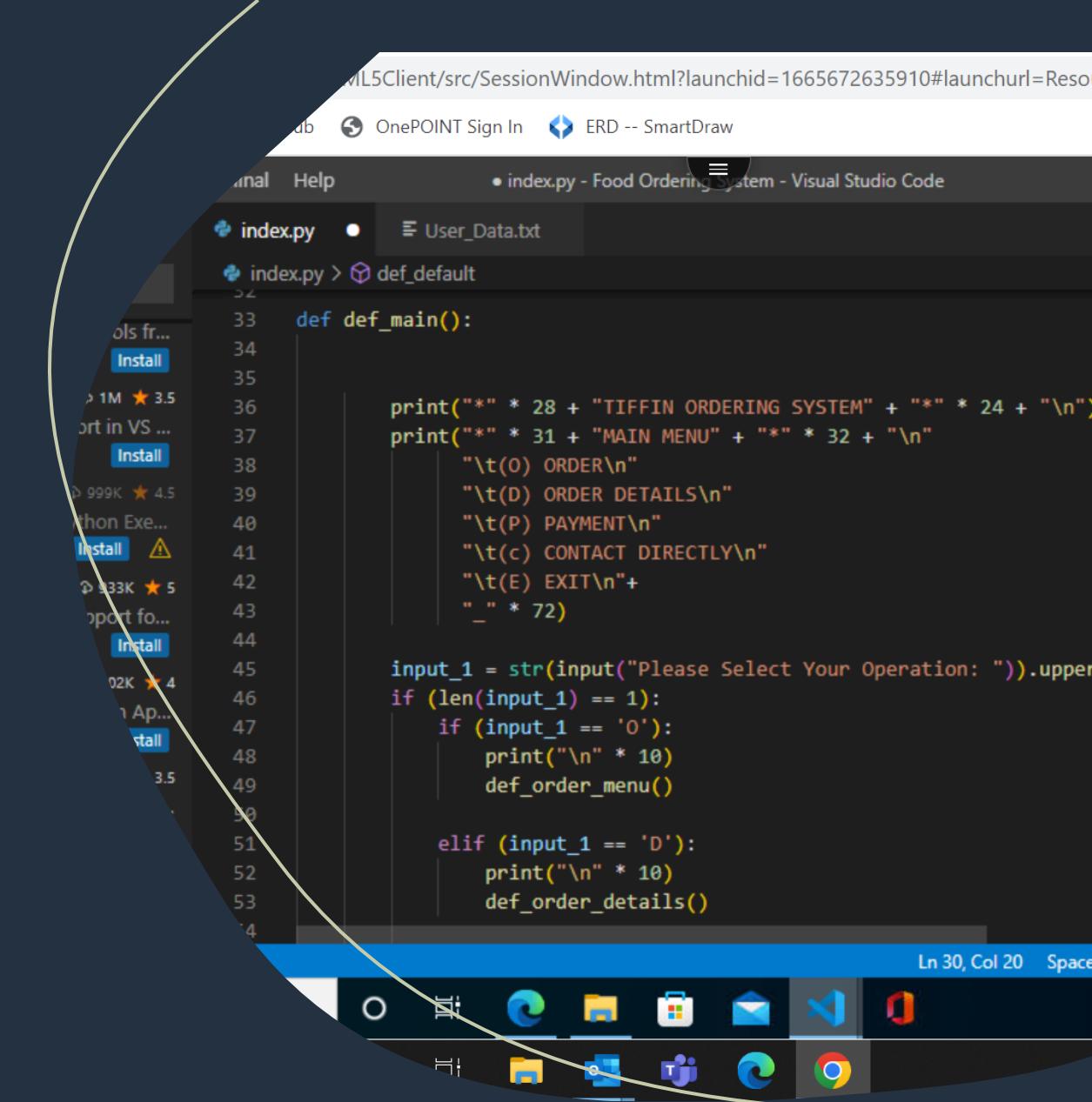
**1.Order:-inside this user can order foods/drinks or can go for other services like extra spoon,plate,glass etc.**

**2.Oder details:-inside this user can see the details of order or previous order.**

**3.Contact directly:-inside this, contact of tiffin provider will be given.**

**4.Payment:-allow user to pay the bills**

**5.Exit:-exiting user will be given "thank you message"**



```
index.py - Food Ordering System - Visual Studio Code
index.py User_Data.txt
index.py > def_default

def def_main():

    print("*" * 28 + "TIFFIN ORDERING SYSTEM" + "*" * 24 + "\n")
    print("*" * 31 + "MAIN MENU" + "*" * 32 + "\n")
    "\t(O) ORDER\n"
    "\t(D) ORDER DETAILS\n"
    "\t(P) PAYMENT\n"
    "\t(C) CONTACT DIRECTLY\n"
    "\t(E) EXIT\n" +
    "_" * 72)

    input_1 = str(input("Please Select Your Operation: ")).upper()
    if (len(input_1) == 1):
        if (input_1 == 'O'):
            print("\n" * 10)
            def_order_menu()

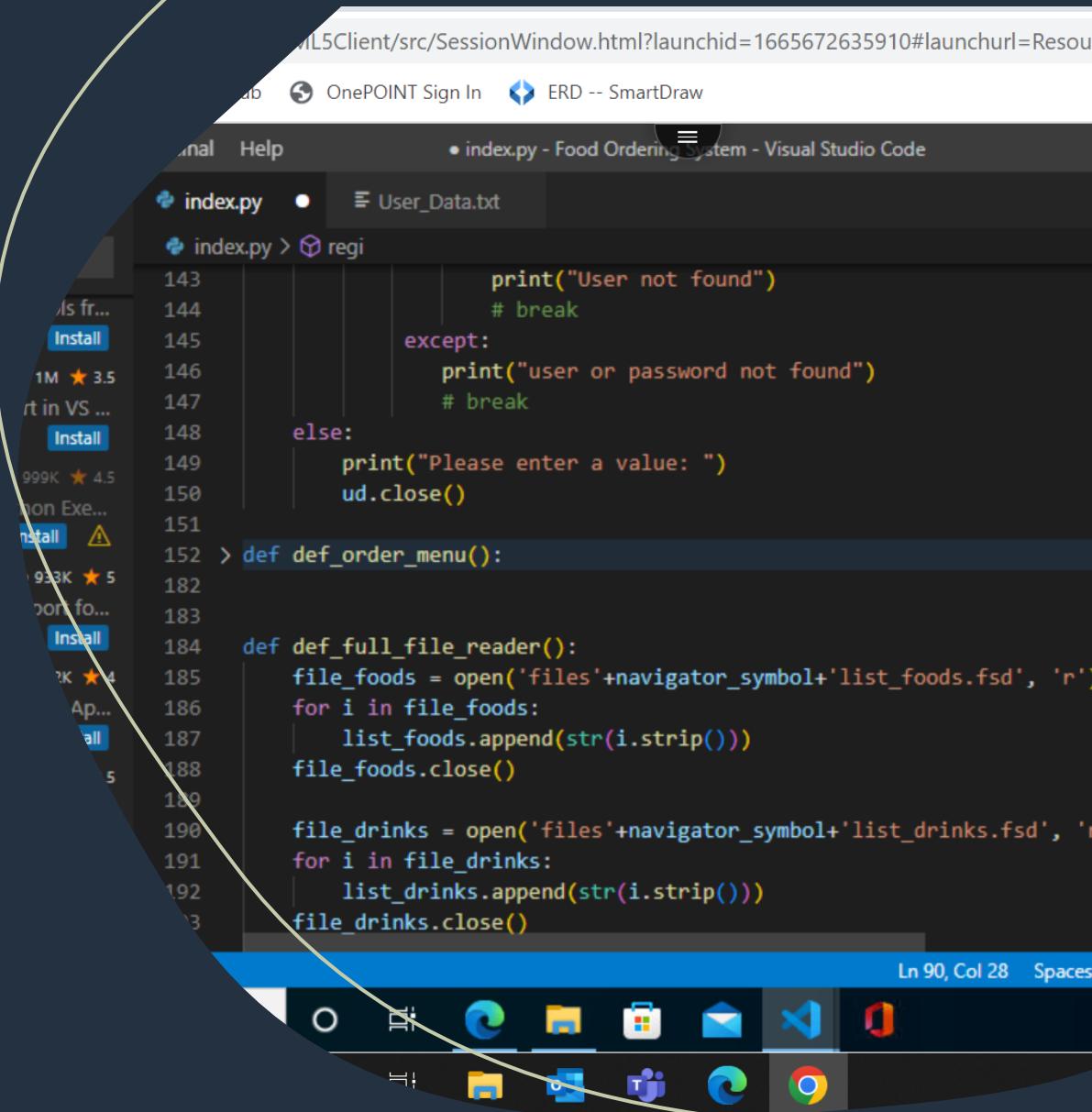
        elif (input_1 == 'D'):
            print("\n" * 10)
            def_order_details()

Ln 30, Col 20 Space
```

# Function #06

**(def def\_order\_menu()):**

**This function will call other  
function, which contains the list of  
foods , drinks other services.**



The screenshot shows a Visual Studio Code window with the following code:

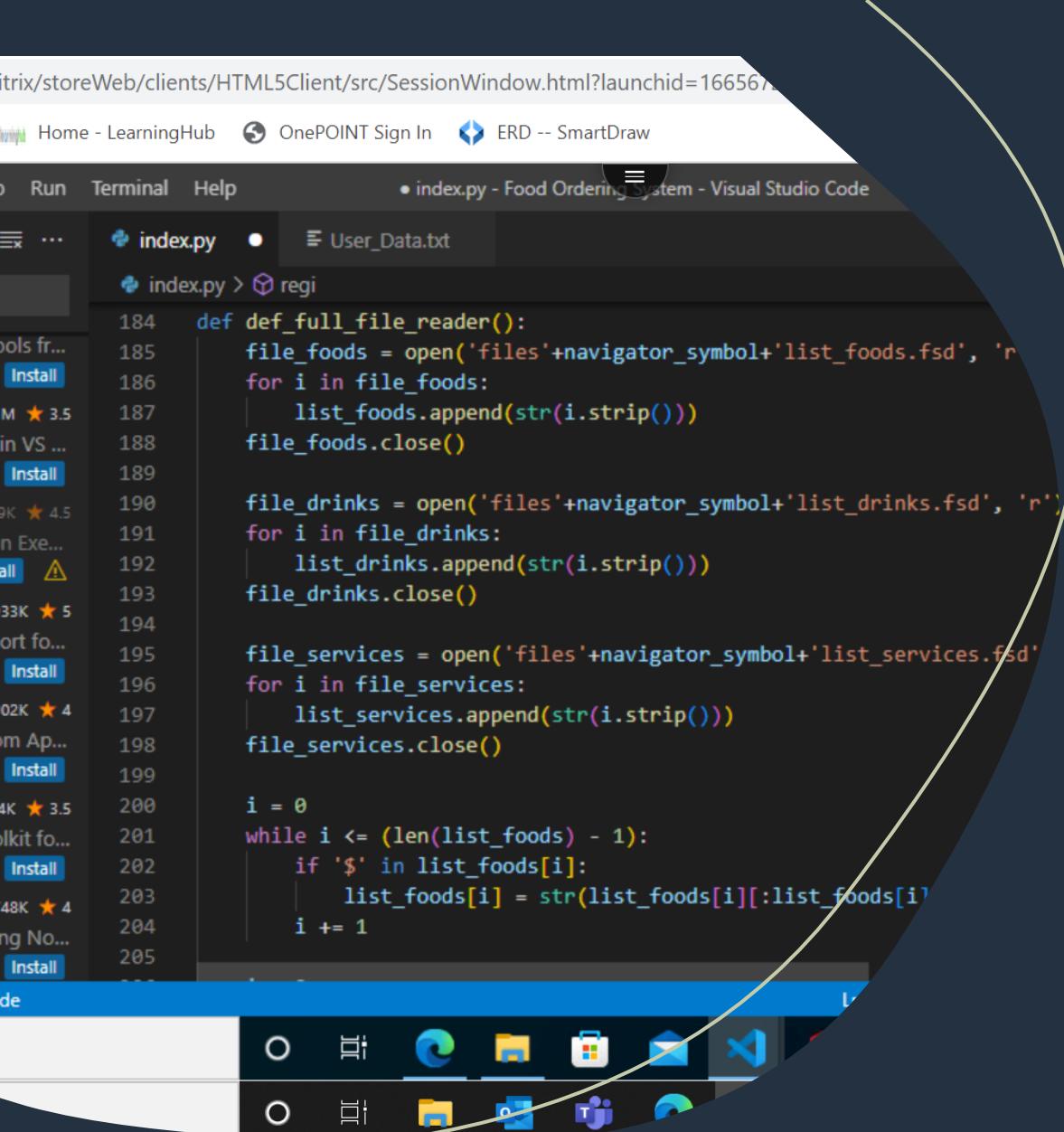
```
index.py - Food Ordering System - Visual Studio Code

index.py User_Data.txt

143     print("User not found")
144     # break
145 except:
146     print("user or password not found")
147     # break
148 else:
149     print("Please enter a value: ")
150     ud.close()
151
152 > def def_order_menu():
153
154     def def_full_file_reader():
155         file_foods = open('files'+navigator_symbol+'list_foods.fsd', 'r')
156         for i in file_foods:
157             list_foods.append(str(i.strip()))
158         file_foods.close()
159
160         file_drinks = open('files'+navigator_symbol+'list_drinks.fsd', 'r')
161         for i in file_drinks:
162             list_drinks.append(str(i.strip()))
163         file_drinks.close()

Ln 90, Col 28 Spaces
```

The code defines a function `def_order_menu()` which calls `def_full_file_reader()`. The `def_full_file_reader()` function reads files named `list_foods.fsd` and `list_drinks.fsd`, stripping each line and appending it to lists `list_foods` and `list_drinks` respectively.



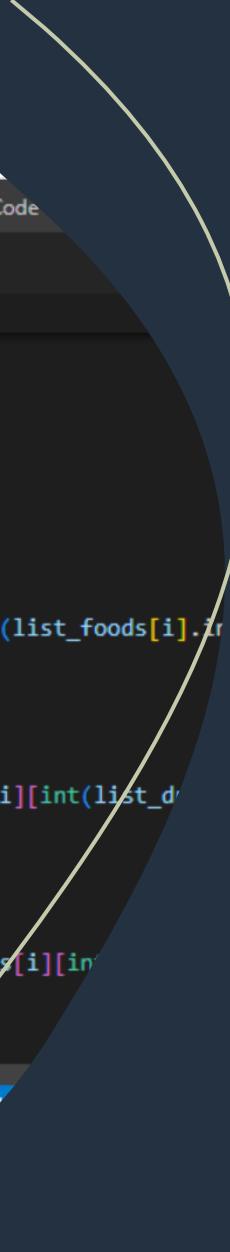
A screenshot of the Visual Studio Code interface. The title bar shows "index.py - Food Ordering System - Visual Studio Code". The left sidebar lists files: "index.py" (marked with a green dot), "User\_Data.txt" (marked with a black dot), and "index.py > regi" (marked with a purple dot). The main editor area contains Python code for a food ordering system. The code defines a function `def\_full\_file\_reader()` which reads three files: `list\_foods.fsd`, `list\_drinks.fsd`, and `list\_services.fsd`. It appends the contents of each file to lists `list\_foods`, `list\_drinks`, and `list\_services` respectively. The code then iterates through the `list\_foods` list, removing any entries that contain a '\$' character and adjusting the indices of the remaining elements.

```
184     def def_full_file_reader():
185         file_foods = open('files'+navigator_symbol+'list_foods.fsd', 'r')
186         for i in file_foods:
187             list_foods.append(str(i.strip()))
188         file_foods.close()
189
190         file_drinks = open('files'+navigator_symbol+'list_drinks.fsd', 'r')
191         for i in file_drinks:
192             list_drinks.append(str(i.strip()))
193         file_drinks.close()
194
195         file_services = open('files'+navigator_symbol+'list_services.fsd')
196         for i in file_services:
197             list_services.append(str(i.strip()))
198         file_services.close()
199
200         i = 0
201         while i <= (len(list_foods) - 1):
202             if '$' in list_foods[i]:
203                 list_foods[i] = str(list_foods[i][:list_foods[i].index('$')])
204             i += 1
```

# Function #07

**(Def def\_file\_reader()):**

**This function will contain all the list of foods , drinks and other services available.**



Home - LearningHub OnePOINT Sign In ERD -- SmartDraw

Go Run Terminal Help

index.py - Food Ordering System - Visual Studio Code

index.py User\_Data.txt

Tools fr...  
Install  
1M ★ 3.5  
rt in VS ...  
Install  
hon Exe...  
Install △  
933K ★ 5  
port fo...  
Install  
902K ★ 4  
from Ap...  
Install  
814K ★ 3.5  
oolkit fo...  
Install  
748K ★ 4  
ing No...  
Install

Code

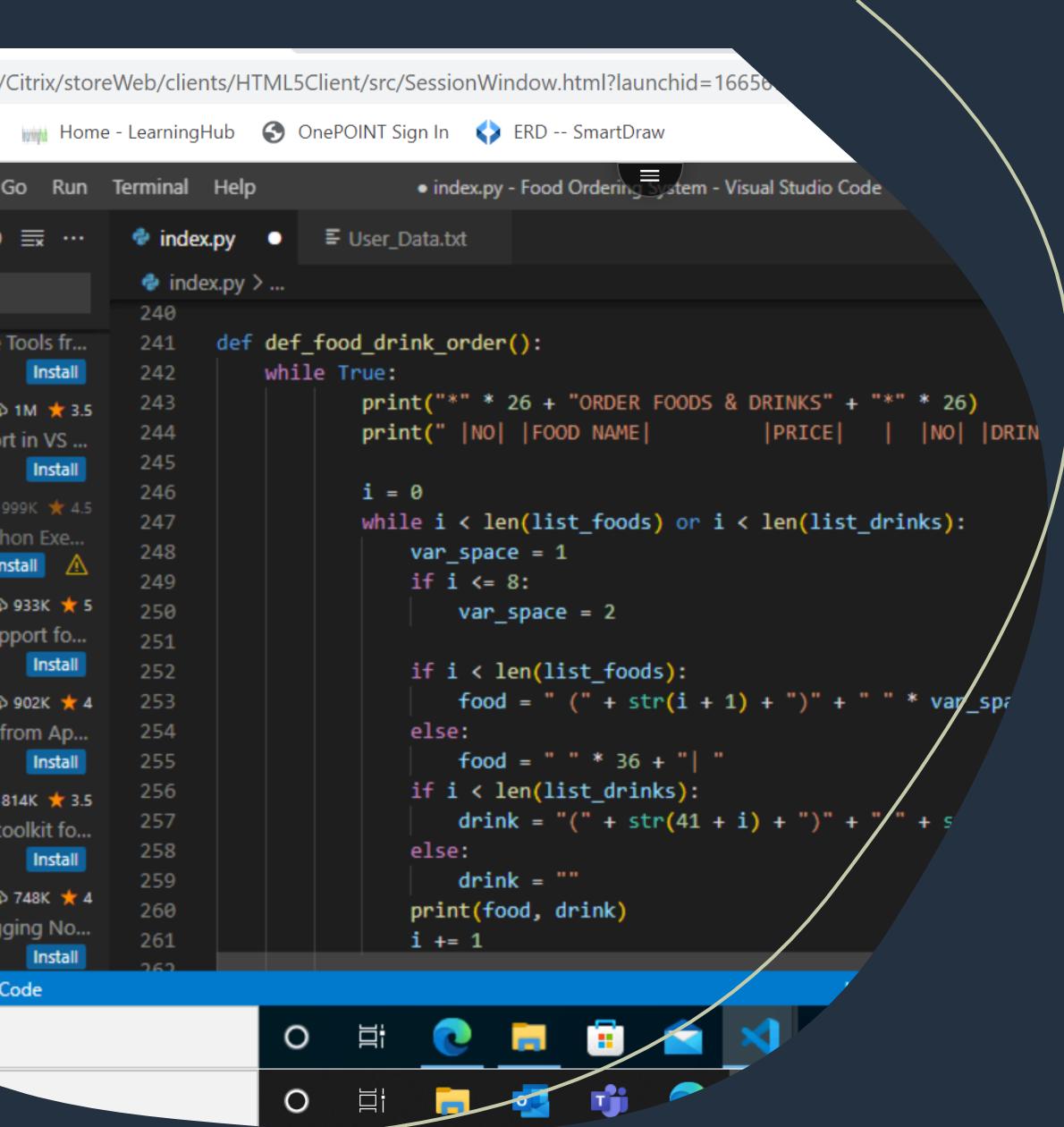
```
219 def def_file_sorter():
220     global list_foods, list_drinks, list_services
221     list_foods = sorted(list_foods)
222     list_drinks = sorted(list_drinks)
223     list_services = sorted(list_services)
224
225     i = 0
226     while i < len(list_foods):
227         list_item_price[i] = float(list_foods[i][int(list_foods[i].i.
228         i += 1
229
230         i = 0
231         while i < len(list_drinks):
232             list_item_price[40 + i] = float(list_drinks[i][int(list_drinks[i].
233             i += 1
234
235         i = 0
236         while i < len(list_services):
237             list_item_price[80 + i] = float(list_services[i][int(list_services[i].
238             i += 1
239 def_file_sorter()
```

File Explorer View Taskbar

# Function #08

**(def def\_file\_sorter()):**

**This function will sort all the items inside the list of foods , drinks and other services.**



## Function #09

```
(def def_food_drink_order()):
```

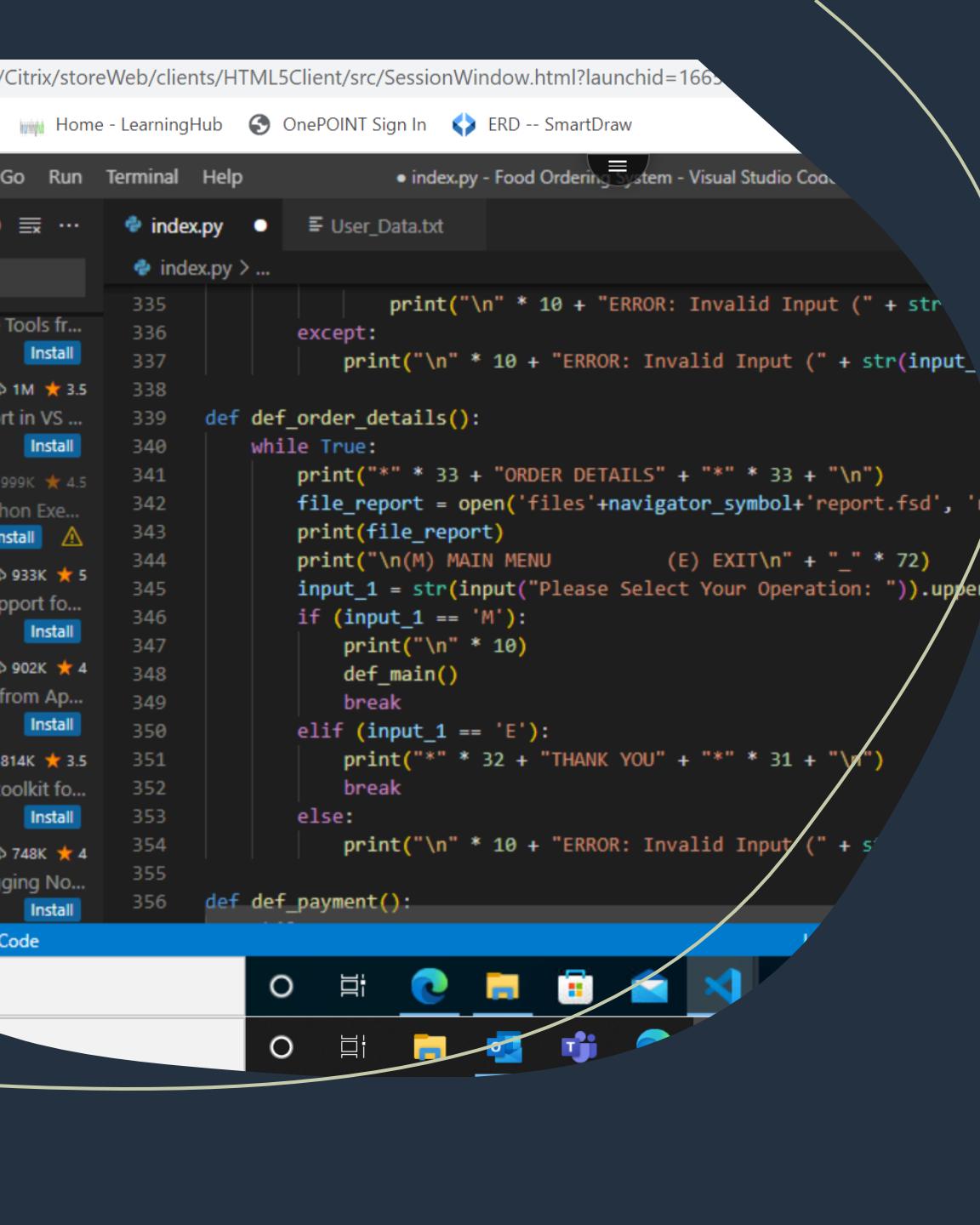
**This function will call all the listed items within it.**

```
  index.py  ●  User_Data.txt
+ index.py > ...
299     print("\n" * 10 + "ERROR: Invalid Input (" + s
300
301 def def_other_services():
302     while True:
303         print("*" * 29 + "OTHER SERVICES" + "*" * 29)
304         print(" |NO| |SERVICE NAME|      |PRICE| ")
305
306         i = 0
307         while i < len(list_services):
308             print(" (" + str(81+ i) + ")" + " " + str(list_services[i]))
309
310             i += 1
311
312         print("\n (M) MAIN MENU          (P) PAYMENT")
313
314         input_1 = input("Please Select Your Operation: ").upper()
315         if (input_1 == 'M'):
316             print("\n" * 10)
317             def_main()
318             break
319         if (input_1 == 'E'):
320             print("*" * 32 + "THANK YOU" + "*" * 31 + "\n")
321             break
```

# Function #10

**(def def\_other\_services()):**

**This function will contain all the list of the other services within it.**



The screenshot shows a Visual Studio Code interface with the following details:

- Top Bar:** Home - LearningHub, OnePOINT Sign In, ERD -- SmartDraw.
- Menu Bar:** Go, Run, Terminal, Help.
- Code Editor:** File: index.py, User\_Data.txt. The code editor displays the following Python script:

```
335     print("\n" * 10 + "ERROR: Invalid Input (" + str_
336 except:
337     print("\n" * 10 + "ERROR: Invalid Input (" + str(input_
338
339 def def_order_details():
340     while True:
341         print("*" * 33 + "ORDER DETAILS" + "*" * 33 + "\n")
342         file_report = open('files'+navigator_symbol+'report.fsd', 'r')
343         print(file_report)
344         print("\n(M) MAIN MENU          (E) EXIT\n" + "_" * 72)
345         input_1 = str(input("Please Select Your Operation: ")).upper()
346         if (input_1 == 'M'):
347             print("\n" * 10)
348             def_main()
349             break
350         elif (input_1 == 'E'):
351             print("*" * 32 + "THANK YOU" + "*" * 31 + "\n")
352             break
353         else:
354             print("\n" * 10 + "ERROR: Invalid Input (" + str_
355
356 def def_payment():
```
- Sidebar:** Shows various extensions and their status (Install or Not Installed).
- Bottom Bar:** Taskbar icons for browser, file explorer, terminal, etc.

# Function #11

**(def def\_order\_details()):**

**This function will contain the order details of the item user ordered.**

Terminal Help

index.py - Food Ordering System - Visual Studio Code

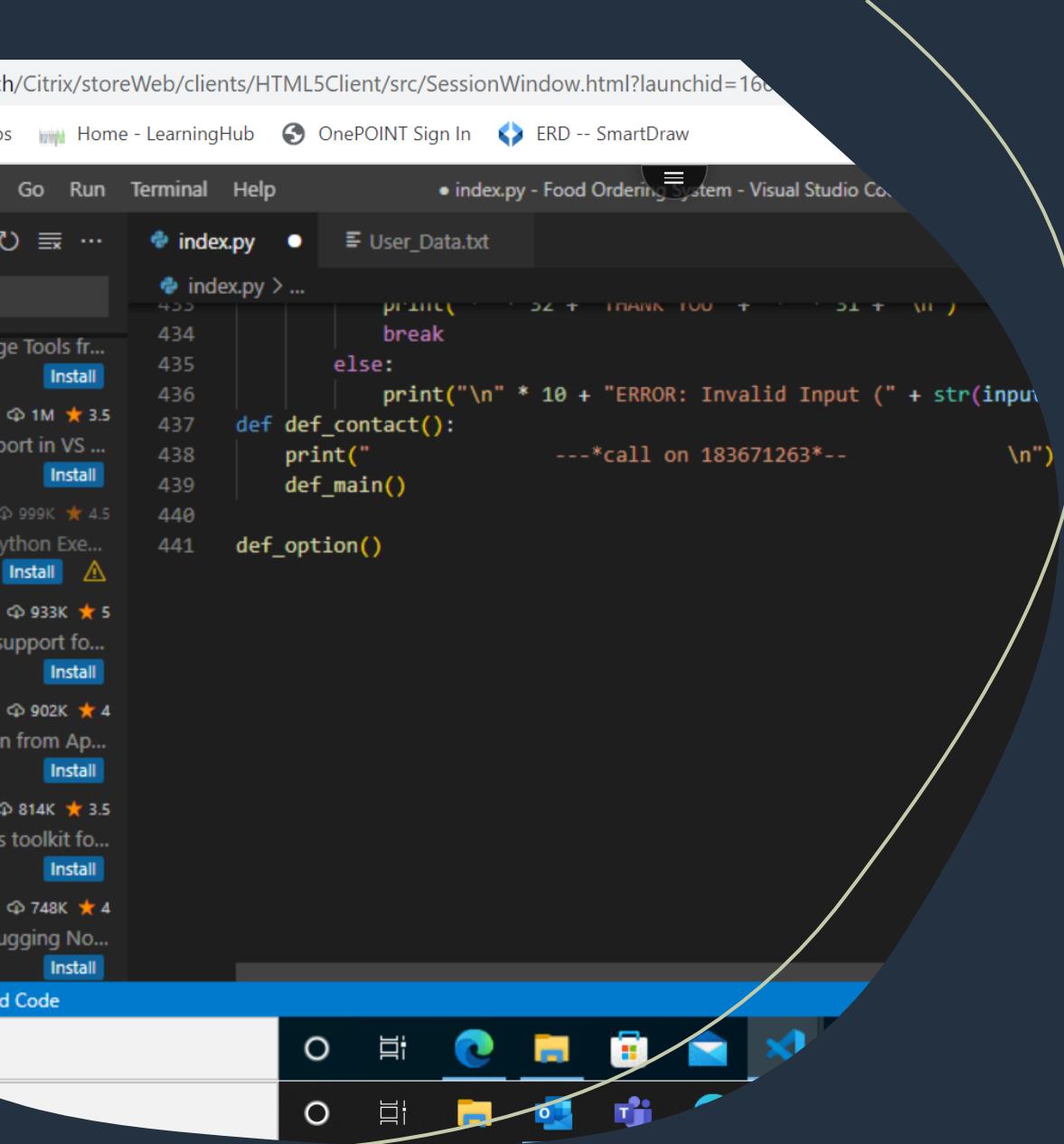
index.py User\_Data.txt

```
355
356     def def_payment():
357         while True:
358             print("*" * 32 + "PAYMENT" + "*" * 33 + "\n")
359             total_price = 0
360
361             report_new = "\n\n\n" + " " * 17 + "*" * 35 + "\n" + " " * 17 + "DA"
362             i = 0
363             while i < len(list_item_order):
364                 if(list_item_order[i] != 0):
365                     if (i >= 0) and (i < 40):
366                         report_new += "\n" + " " * 17 + str(list_foods[i]) +
367                         print(" " * 17 + str(list_foods[i]) + " x " + str()
368                         total_price += list_item_price[i] * list_item_order[i]
369
370                     if (i >= 40) and (i < 80):
371                         report_new += "\n" + " " * 17 + str(list_drinks[i])
372                         print(" " * 17 + str(list_drinks[i - 40]) + " "
373                         total_price += list_item_price[i] * list_item_order[i]
374
375                     if (i >= 80) and (i < 100):
376                         report_new += "\n" + " " * 17 + str(list_ser
377                         print(" " * 17 + str(list_services[i - 80]))
378                         total_price += list_item_price[i] * list_item_order[i]
379
380             print(report_new)
381             print("Total Price: " + str(total_price))
382             print(" ")
383
384             user_input = input("Enter 'Y' to continue or 'N' to exit: ")
385             if user_input == 'N':
386                 break
387
388         print("Thank you for using our food ordering system!")
389         break
390
391     print("ERROR: Invalid Input") + str(input_1)
```

# Function #12

**(def def\_payment()):**

**This function will allow to pay the bill for the order.**



A screenshot of the Visual Studio Code interface. The title bar shows "index.py - Food Ordering System - Visual Studio Code". The left sidebar lists various Python files and their details, such as "index.py" (434 lines, 1M size), "User\_Data.txt" (441 lines, 999K size), and "def\_contact.py" (441 lines, 814K size). The main editor area displays the following Python code:

```
    print("THANK YOU")
    break
else:
    print("\n" * 10 + "ERROR: Invalid Input (" + str(input))
def def_contact():
    print("----*call on 183671263*--\n")
def_main()
def_option()
```

# Function #13

**(def def\_contact()):**

**This function will show the contact number of tiffin distributer to the user.**

File handling.

```
index.py - Food Ordering System - Visual Studio Code
```

Gmail YouTube Maps Home - LearningHub OnePOINT Sign In ERD -- SmartDraw

File Edit Selection View Go Run Terminal Help

EXPLORER OPEN EDITORS 1 unsaved index.py User\_Data.txt

FOOD ORDERING SYSTEM files list\_drinks.fsd list\_foods.fsd list\_services.fsd report.fsd index.py User\_Data.txt

index.py > ...

```
112     print( )
113     print("*****enter details to LOGIN*****")
114     ud=open("User_Data.txt", "r")
115     username=str(input("Please enter username: "))
116     password=input("Please enter password: ")
117
118     if not len(username or password)<1 :
119
120         d=[]
121         f=[]
122         for i in ud:
123             a,b = i.split(", ")
124             b=b.strip()
125             d.append(a)
126             f.append(b)
127             data = dict(zip(d, f))
128
129         try:
130             if data[username]:
131                 try:
132                     if password == data[username]:
133                         print("Welcome back "+ username)
134                         def main()
```

Activate Windows  
Go to Settings to activate Windows

Ln 217, Col 23 Spaces: 4 UTF-8 LF Python 3.10.7 64-bit Go Live 00:05 14-10-2022 1

Type here to search

Windows Taskbar icons: File Explorer, Microsoft Edge, Mail, Microsoft Teams, Microsoft Word, Google Chrome

System tray icons: Battery, Signal, Wi-Fi, Volume, Language (ENG), Date (14-10-2022), Notification (1)

System tray icons: Battery, Signal, Wi-Fi, Volume, Language (ENG), Date (14-10-2022), Notification (1)

System tray icons: Battery, Signal, Wi-Fi, Volume, Language (ENG), Date (10/14/2022), Notification (17)

vlabs.iiht.tech/Citrix/storeWeb/clients/HTML5Client/src/SessionWindow.html?launchid=1665686019871#launchurl=Resources%2FLaunchIca%2FQ29...

# Exception handling.

```
• index.py - Food Ordering System - Visual Studio Code
```

File Edit Selection View Go Run Terminal Help

EXPLORER OPEN EDITORS 1 unsaved

- index.py
- User\_Data.txt

FOOD ORDERING SYSTEM files

- list\_drinks.fsd
- list\_foods.fsd
- list\_services.fsd
- report.fsd

index.py User\_Data.txt

```
try:  
    if data[username]:  
        try:  
            if password == data[username]:  
                print("Welcome back " + username)  
                def_main()  
  
            else:  
                print("Username or Password incorrect")  
                # break  
            except:  
                print("Password or Username incorrect")  
                # break  
        else:  
            print("User not found")  
            # break  
        except:  
            print("user or password not found")  
            # break  
    else:  
        print("Please enter a value: ")  
        uid.close()
```

Activate Windows Go to Settings to activate Windows.

Ln 217, Col 23 Spaces: 4 UTF-8 LF Python 3.10.7 64-bit Go Live



# Uploading in Github

**In this section we will upload  
the program ,list and files which  
we used for project.**



ITC Infotech Digital-Python-DevC x +

vlabs.iiht.tech/Citrix/storeWeb/clients/HTML5Client/src/SessionWindow.html?launchid=1665719005218#launchurl=R

Gmail YouTube Maps Home - LearningHub OnePOINT Sign In ERD -- SmartDraw

MINGW64:/c/Users/itcdigpy06/Desktop/tiffinFoodOrderingSystem

```
t config --global --edit
doing this, you may fix the identity used for this commit with:
t commit --amend --reset-author

es changed, 507 insertions(+)
e mode 100644 User_Data.txt
e mode 100644 files/list_drinks.fsd
e mode 100644 files/list_foods.fsd
e mode 100644 files/list_services.fsd
e mode 100644 files/report.fsd
e mode 100644 index.py

py06@itcdigpy06 MINGW64 ~/Desktop/tiffinFoodOrderingSystem (master)
branch -m main

py06@itcdigpy06 MINGW64 ~/Desktop/tiffinFoodOrderingSystem (main)
remote add origin https://github.com/ShrawanLimbu/tiffin-foodorderingsyste

py06@itcdigpy06 MINGW64 ~/Desktop/tiffinFoodOrderingSystem (main)
push -u origin main
ating objects: 9, done.
ng objects: 100% (9/9), done.
compression using up to 4 threads
ssing objects: 100% (8/8), done.
g objects: 100% (9/9), 4.11 KiB | 601.00 KiB/s, done.
9 (delta 0), reused 0 (delta 0), pack-reused 0
ps://github.com/ShrawanLimbu/tiffin-foodorderingsystem
w branch]      main -> main
'main' set up to track 'origin/main'.

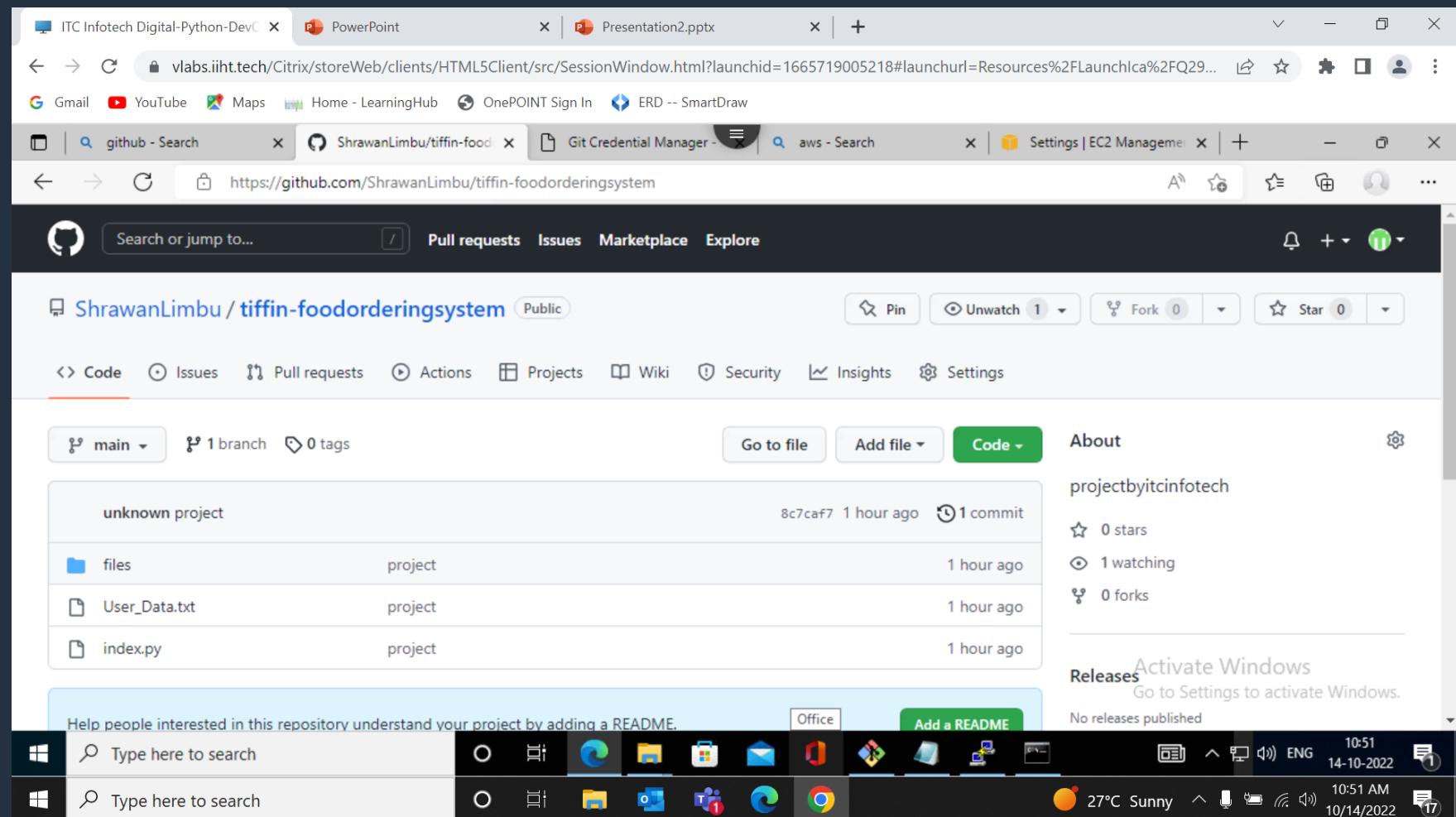
py06@itcdigpy06 MINGW64 ~/Desktop/tiffinFoodOrderingSystem (main)
```

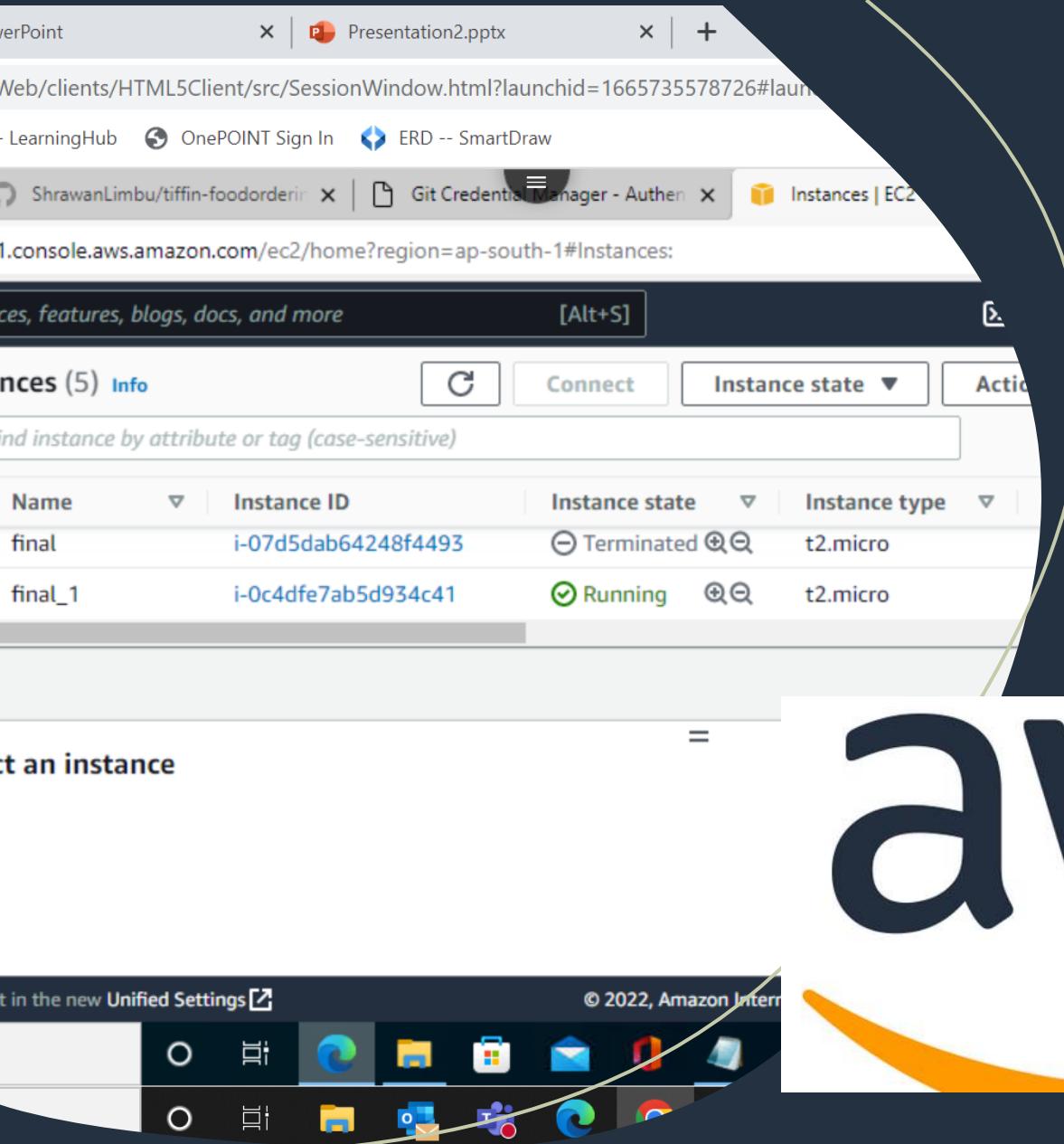
Type here to search

Type here to search

A screenshot of a Windows taskbar showing several pinned icons: File Explorer, Microsoft Edge, Mail, OneDrive, Microsoft Word, Microsoft Excel, Microsoft Powerpoint, Microsoft Teams (with a red notification badge), Microsoft Word, Microsoft Excel, Microsoft Powerpoint, and Microsoft Edge.

# This is how it looks after successful upload:

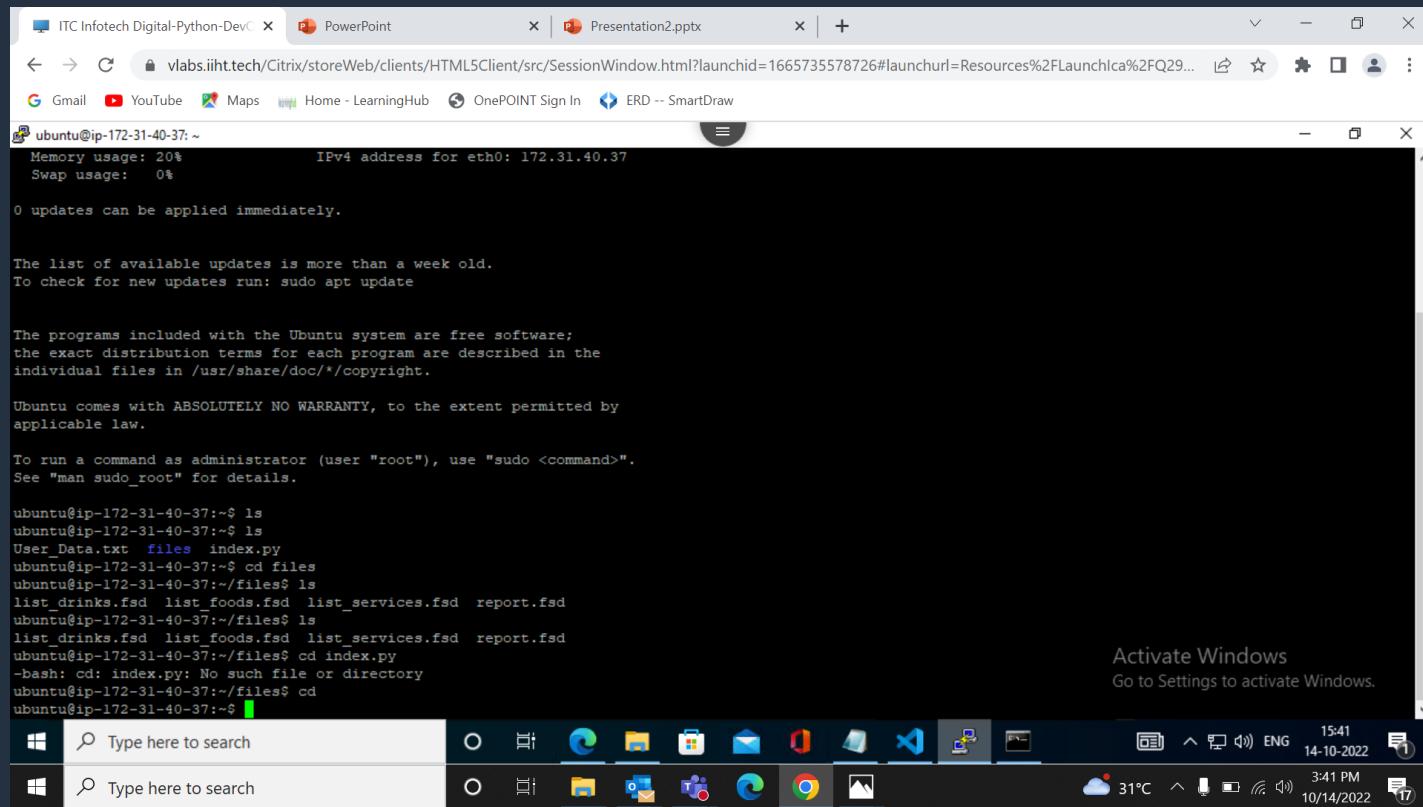




# Deploying project in AWS: **created using instance in EC2.**



# Project deployed.



The screenshot shows a Windows desktop environment with a terminal window open. The terminal window displays the following text:

```
Memory usage: 20%           IPv4 address for eth0: 172.31.40.37
Swap usage:   0%
0 updates can be applied immediately.

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-40-37:~$ ls
ubuntu@ip-172-31-40-37:~$ ls
User_Data.txt  files  index.py
ubuntu@ip-172-31-40-37:~/files$ ls
list_drinks.fsd  list_foods.fsd  list_services.fsd  report.fsd
ubuntu@ip-172-31-40-37:~/files$ ls
list_drinks.fsd  list_foods.fsd  list_services.fsd  report.fsd
ubuntu@ip-172-31-40-37:~/files$ cd index.py
-bash: cd: index.py: No such file or directory
ubuntu@ip-172-31-40-37:~/files$ cd
ubuntu@ip-172-31-40-37:~$
```

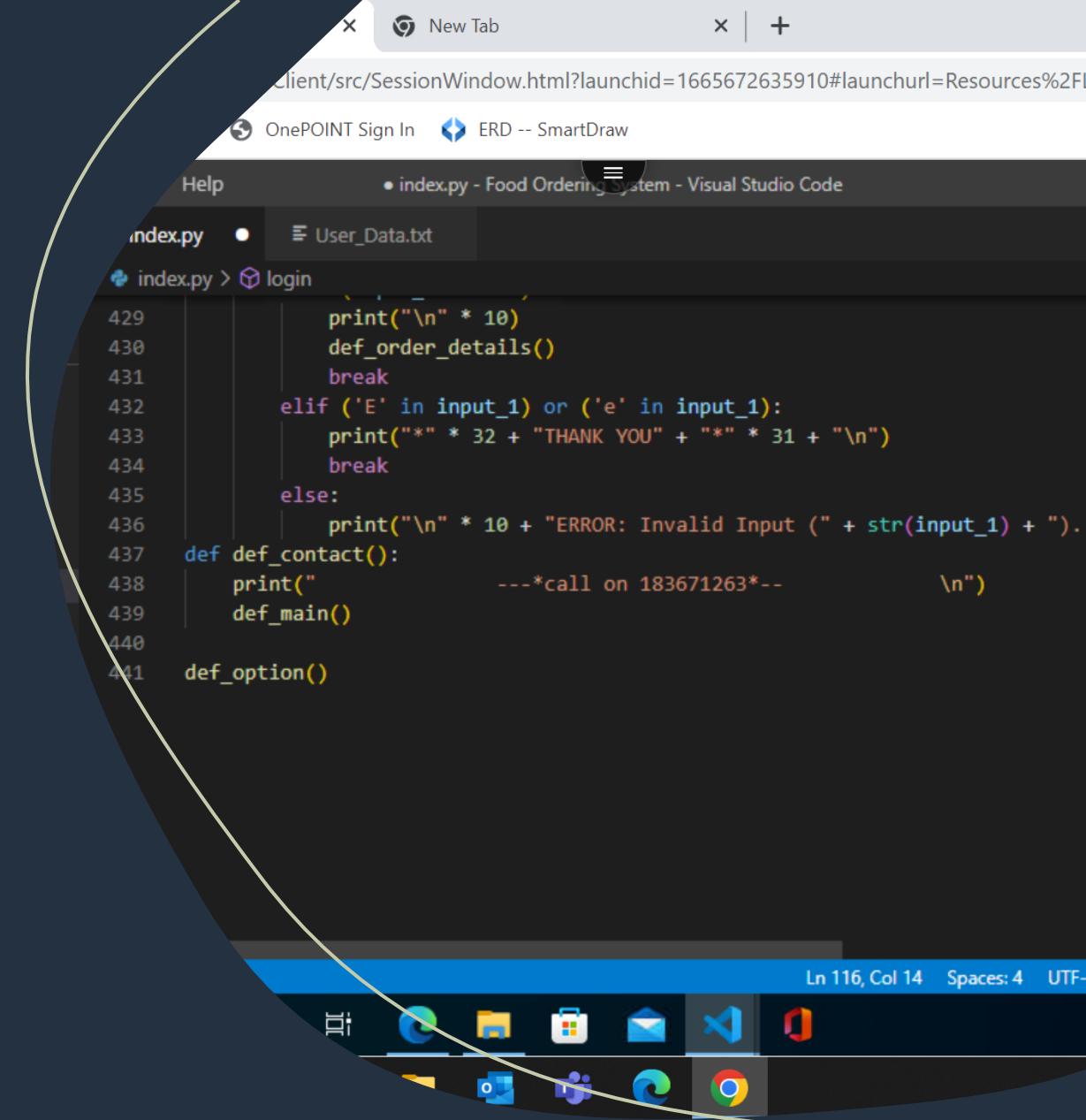
The desktop taskbar at the bottom shows several pinned icons, including File Explorer, Mail, Edge, and File History. The system tray indicates a battery level of 31%, a temperature of 31°C, and a date/time of 10/14/2022 3:41 PM. A watermark for "Activate Windows" is visible in the bottom right corner of the terminal window.

## d. Risk & issues:

- **1.Logout/Exit must be done manually.**
- **2.Single profile user interface only.**
- **3.Order cannot be undone.**
- **4.User Cannot jump directly to different section without completing the present operation.**

## e. Conclusion:

- **User can order the tiffin with menu or he/she can order by provided number. With every bill there will be the proper indication of order value and discount applied(if applicable). User will get other services which will allow them to order addition items like: extra spoon,glass,plate etc.**



The screenshot shows a Microsoft Visual Studio Code interface with the following details:

- Title Bar:** Client/src/SessionWindow.html?launchid=1665672635910#launchurl=Resources%2F
- Activity Bar:** OnePOINT Sign In, ERD -- SmartDraw
- File Explorer:** index.py, User\_Data.txt
- Code Editor:** The code is written in Python and defines several functions:
  - `def_order_details()`: Prints a separator of 10 newlines and then prints "THANK YOU" surrounded by asterisks.
  - `def_contact()`: Prints a call instruction: "...\*call on 183671263\*--".
  - `def_main()`: A placeholder function.
  - `def_option()`: A placeholder function.
- Status Bar:** Ln 116, Col 14, Spaces: 4, UTF-8
- Taskbar:** Shows icons for various Microsoft applications: File Explorer, Task View, File, Settings, Mail, Microsoft Edge, and Microsoft Word.



Thank You.

Have any

