

This is my git hub

[ChaiyawutTar/big-brain-delivery-food: Nope \(github.com\)](#)

Big Brain Delivery App

By Chaiyawut SKE20

Project Log

Date	Work
27 Nov	Admin Console for Stock add and Store Open and close
28 Nov	User Class and Admin Class
29 Nov	Console for login register and new order
30 Nov	Finish up a login, View, New, Cancel order
1 Dec	Linking between stock and menu
10 Dec	Day 1 of adding GUI and moving functions and method to the place where it should be
11 Dec	Making UI deliver an order to the user

Overview

This is a food ordering program. where customers can order food You can check your order and you can cancel your order. Start by logging in. When running the program, you will be prompted to enter your username and password. If you have already registered, you will be able to choose to order food. If you still need to register, will report it. I will give the user. Enter username, first name, last name, password, address, district, province, postal code, telephone, and email. Next, let log in Enter your username and password if you are a user, you have 3 choices.

first -> 1.View order

If you haven't ordered food yet(not yet New order), you must order food first.

I will show your order number and you orders that you have ordered.

second -> 2.New order

I will show you order in my stocks and give you the price of any dish.

choose you dish -> Enter number of that dish.

choose you drink -> Enter number of that drink.

and you confirm you order.

third -> 3.Cancel order

If you haven't ordered food yet(not yet New order), you must order food first.

I will show your order number and you orders that you have ordered.

choose the orders that you want to cancel.

if you are admin, you have 5 choices but you can not choose choices 1 and 2 because you are admin.

fourth -> 4. Shop management

this have any command. This command is like git hub. Ex.git add -A , git push

.....

1.exit -> break your program.

2.stock -> command about the stock.

stock dish adds menu number price -> add stock dish with the name of menu number and price.

Ex. stock dish add KitKat 5 10

stock dish add menu number price -> add stock drink with name of menu number and price.

Ex. stock drink add NamDang 10 10

stock view -> view admin stock.

stock remove -> decreasing the number of the food to 0.

Ex. stock remove "kit kat"

3. shop -> This common indicates the status of the program.

The program is set to run from 8am to 8pm.

shop auto -> setting this program to open on time.

shop on -> setting this program to open and change to manual.

shop off -> setting this program to close and change to manual.

fifth -> 5.Order management

this have display on turtle. I will show all of user who have orders and If I click to Deliver, this program will sent food to user who orders.

Required Library and Tools

- Turtle
- json
- os
- datetime
- Colors

Program Design and Code Structure

class stock

this class is about the order for subclass.
update_stock() -> update stock dump files to stock.json.
update_order_stock(user, ordersMenu) -> update order of each user.
cancel_order(user) -> cancel the order of each user.
deliver_order(user) -> in order management, if click to Deliver, this method will remove the order in the system and send to user.
new_order(user) -> add a new order for each user.
viewOrder(user) -> show order.
getNumOrders(user) -> get a number of order.

class Display(Stock)

justifyRight(text1,text2,length) -> setting order format.
greenText(text) -> change colors in the terminal.
coloredText(text,color) -> Displays text that changes color.
run() -> running the program
landingPage(user,userDict) - > setting terminal condition of each choices.
choiceSelector(user) -> raise error choices.
n_click(x,y) -> bottom in turtle.
graphic_interface(admin) -> Display python graphics.

class User(Stock)

```
login(userDict) -> login console.  
register() -> user register.  
new_user_add(new_data) -> keep new user data to file user.json
```

class Admin(User)

```
justifyRight(text1,text2,length) -> setting order format.  
autoShopStatus(self) -> setting date time to open or close the program.  
get_user_order(user) -> get an order for each user in the list.  
isAdmin(user) -> check the user who login to the customer or admin.  
adminConsole() -> console of admin.
```

class Color

```
greenText(text) -> change colors in terminal.  
coloredText(text,color) -> Displays text that changes color.
```