27.11.2023

CENG 113 – Programming Basics

Assignment 2

Due: 03.12.2023 - 23:59

Assignment Description

For this assignment, your objective is to create a Python program for restaurant staff to efficiently manage customer orders. The program should allow employees to collect order details gradually from customers, ultimately generating an invoice that comprehensively outlines the order specifics along with the total price upon completion. Essential product information is provided in the assignment through the "categories.txt," "products.txt," and "portions.txt" files.

Program Flow

Upon executing the program, it should display a list of all categories, prompting the user to select. Following the category selection, the program should present the products within the chosen category, asking the user to pick a specific product. Subsequently, the available portions for the selected product should be listed, and the user prompted to choose a portion.

Following the portion selection, the chosen product should be added to the order, and the user queried about adding another product. This entire sequence needs to be reiterated if the user desires to include new products. Once the product addition phase concludes, the program should output details for all added products and the overall order cost.

Formats of the Provided Documents

The materials provided in the assignment exhibit a consistent format, where each row is structured with columns separated by the ";" symbol. The symbols "#" and ";" are distinctive and they are not employed in names or prices. Leveraging these unique symbols during the filtering phases can be advantageous.

In the **"categories.txt"** file, the initial column holds the distinctive ID number corresponding to each category, while the second column is the name associated with the respective category.

#Category ID	Category Name

E.g.

#1;Sandwiches and Wraps

#2;Salads

…

In the **"products.txt"** file, the first column signifies the category ID to which the product belongs. Consequently, after selecting a category, the products within that chosen category can be identified using this ID. The second column holds the product's name, and the third column contains the unique code assigned to the product.

#Category ID	Product Name	Product Code

E.g.

#1;Patty with Cheese;PWC

#1;Patty with Beef;PWB

…

In the **"portions.txt"** file, the first column signifies the product code to which the portion belongs. Therefore, following each product selection, the portions associated with the chosen product can be retrieved using this code. The second column denotes the portion's name, and the third column displays the corresponding price of the portion. To ensure a correct program flow, each combination of product code and portion name must be unique.

#Product Code	Portion Name	Portion Price

E.g.

#PWC;1/4 lb;3.49

#PWC;1/2 lb;4.29

…

Requirements

The functions to be used in your assignment must include the following three functions:

- Given that all files share a common structure, the **prepareInfo()** function should be designed to accept user selections and file names as inputs. This function conducts the necessary file operations at the related stage and organizes the data for subsequent stages. It is expected that this function is implemented in a generic manner, ensuring applicability to all files.
- The **printMenu()** function should present the information to the user based on the related stage. It should print categories, products within the selected category, or portions of the chosen product, according to the ongoing stage of the order-taking process.
- The **getUserInput()** function should get the input from the user and format the input in a manner conducive to subsequent stages in the program.

Please read the assignment description and sample program output carefully and design your program in the required flow. **It is expected that the submitted program will work in accordance with the described flow and have the specified restrictions and features.**

Submission Rules

- Assignments must be done and submitted individually. Students found to have cheated will automatically receive 0 points.
- You must submit your assignment by the specified deadline.
- The submitted file must contain your source code and **code explanations (Comments)**. Your program should be able to operate correctly when placed in the same directory as the assignment files. **Please do not send the provided assignment (.txt) files.**
- The name of the submitted file must be your student number. **(Ex. 123456.py)**
- Only Python files with .py and .ipynb extensions will be evaluated. **Files with .txt, .pdf, .jpeg etc. extensions will not be accepted.**
- **Please send your file without compressing it.**

Sample Output in Expected Format

Welcome to the Store

---

1. Sandwiches and Wraps
2. Salads
3. Milkshake
4. Fries
5. Sauce
6. Side

Please select the category: 1

Sandwiches and Wraps

---

1. Patty with Cheese
2. Patty with Beef
3. Kids Cheeseburger
4. Chicken Wrap

Please select the product: 1

Patty with Cheese

---

1. 1/4 lb.
2. 1/2 lb.
3. 3/4 lb.

Please select the portion: 1

Would you like to complete the order (y, n)? n

---

1. Sandwiches and Wraps
2. Salads
3. Milkshake
4. Fries
5. Sauce
6. Side

Please select the category: 2

Salads

---

1. Humus & Chicken Salad
2. Classic Chicken Salad
3. Chicken Caesar

Please select the product: 1

Humus & Chicken Salad

---

1. Full Size
2. 1/2 Size

Please select the portion: 1

Would you like to complete the order (y, n)? n

---

1. Sandwiches and Wraps
2. Salads
3. Milkshake
4. Fries
5. Sauce
6. Side

Please select the category: 5

Sauce

---

1. Chili Sauce

Please select the product: 1

Chili Sauce

---

1. Small
2. Large

Please select the portion: 1

Would you like to complete the order (y, n)? y

Order Recipe

Sandwiches and Wraps	Patty with Cheese	1/4 lb.	3.49$
Salads	Humus & Chicken Salad	Full Size	5.89$
Sauce	Chili Sauce	Small	0.99$

Total: 10.37$