"GROUP 4 | Khalifa, Hamad, Tim | GCIS123.604 | PROF. D. KOVACEVIC"

import csv

"""

Here we import the CSV module to operate with CSV files.

"""

class Article:

"""

Here we represent an item in the inventory and its attributes for name, price, and quantity.

"""

def \_\_init\_\_(self, name, price, quantity):

self.name = name

self.price = price

self.quantity = quantity

def getname(self):

"""

Here we return the name of the item.

"""

return self.name

def getprice(self):

"""

Here we return the price of the item.

"""

return self.price

def getquan(self):

"""

Here we return the current quantity of the item available in stock.

"""

return self.quantity

def changequan(self, newquan):

"""

Here we update the quantity of the item.

"""

self.quantity = newquan

class Cart:

"""

Here we manage the shopping cart containing a list of items. It has functions to add, remove, and process items.

"""

def \_\_init\_\_(self):

self.listproducts = [] #This line is done to show an empty cart.

def addproducts(self, name, quantity):

"""

Here we add products to the cart or change and update their quantity if they are already present in cart.

"""

existing = next((item for item in self.listproducts if item.getname() == name.getname()), None)

if existing:

existing.changequan(existing.getquan() + quantity)

else:

self.listproducts.append(Article(name.getname(), name.getprice(), quantity))

def remove(self, name, quantity):

"""

Here the line removes a specific quantity of an item from the cart

It also deletes the item if the quantity is 0.

"""

for item in self.listproducts:

if item.getname() == name.getname():

new\_quantity = item.getquan() - quantity

if new\_quantity > 0:

item.changequan(new\_quantity)

elif new\_quantity == 0:

self.listproducts.remove(item)

else:

return "Not enough products in stock"

def checkout(self):

"""

Here we count the total price for all items,

apply discount of 10% discount for more than 3 similar items in cart,

including a 7% VAT.

"""

total = 0

for item in self.listproducts:

total\_cost\_item = item.getprice() \* item.getquan()

if item.getquan() >= 3:

total\_cost\_item \*= 0.9 # Apply discount

total += total\_cost\_item

cost\_VAT = total \* 1.07 # Add VAT to the total cost.

return cost\_VAT

def getlistproducts(self):

"""

Here we showcase all of the products which are currently included in the cart and their quantities.

"""

if not self.listproducts:

return "Your cart is empty"

for element in self.listproducts:

print(f"{element.getname()} {element.getquan()}")

def read\_data(path):

"""

Reads product data from a CSV file and returns it as a dictionary with product names as keys.

"""

try:

with open(path, mode="r", encoding="utf-8") as file:

text = csv.reader(file)

data = {}

headers = next(text) # Skip headers

for row in text:

if row:

data[row[0].lower()] = [float(row[2]), int(row[1])] # This line is used to store data in a dictionary with the product name as key and price and stock as values.

return data

except Exception as e:

return str(e)

def listproducts():

"""

Here we make a list of all products from the inventory, their prices and amounts in stock.

"""

for name, details in INVENTORY.items():

print(f"{name.title()}: Price - {details[0]}, Stock - {details[1]}")

def menu(cart):

"""

Here we make a user menu whihc allows for interactions.

"""

global INVENTORY

while True:

print("\n1. List all inventory items with prices")

print("2. List cart shopping items")

print("3. Add an item to the shopping cart")

print("4. Remove an item from the shopping cart")

print("5. Checkout")

print("6. Exit")

choice = input("Enter your choice by number: ")

try:

choice = int(choice)

except ValueError:

print("Please enter a valid number.")

continue

if choice == 1:

listproducts()

elif choice == 2:

cart.getlistproducts()

elif choice == 3:

item\_name = input("Enter the item name to add to the cart: ").lower()

if item\_name in INVENTORY:

quantity = int(input("Enter the quantity: "))

item\_details = INVENTORY[item\_name]

item = Article(item\_name, item\_details[0], quantity)

cart.addproducts(item, quantity)

print("Item added successfully.")

else:

print("Item not found in inventory.")

elif choice == 4:

item\_name = input("Enter the item name to remove from the cart: ").lower()

if item\_name in INVENTORY:

quantity = int(input("Enter the quantity: "))

cart.remove(Article(item\_name, INVENTORY[item\_name][0], 0), quantity)

else:

print("Item not found in cart.")

elif choice == 5:

print(f"Total amount due (including VAT): AED {cart.checkout():.2f}")

elif choice == 6:

print("Exiting program.")

break

else:

print("Invalid choice. Please enter a number between 1 and 6.")

if \_\_name\_\_ == "\_\_main\_\_":

INVENTORY = read\_data("products.csv")

shopping\_cart = Cart()

print("Welcome to our electronic store!")

menu(shopping\_cart)