Each day, you will complete this page (the assignment will be posted daily)

1. **Paste your code for the summative assignment below.**

```
# Omaima Gohir
# 5/31/2023
# This program will demonstrate an online store which the user can pick from the many options and get the total amount of money back in tax.

"""
This function will take the user's input and determine if it is a valid item in the store menu. If the item is not correctly spelled, or incorrect in any way, the code will continue asking for a valid answer.

  Preconditions:
  item != integer or negative integer
  items must be spelled correctly
  done = code stopping

  Parameter:
  total = total money

  Returns:
  the total price of the shopping cart items in the costumer's basket
"""


#item names stored with keys and values in a dictionary, which has a nested dictionary inside it
item_prices = {
  "Milk": {"2%": 4.32, "3.25%": 4.32},
  "Bread": {"Regular": 2.50},
  "Water": {"Regular": 30.00},
  "Eggs": {"A Dozen": 3.87},
  "Butter": {"Regular": 3.42},
  "Oil": {"Olive oil": 5.60, "Avocado oil": 5.60},
  "Yogurt": {"Regular": 3.43},
  "Coke": {"20 cans": 23.00, "Jug": 5.00}
}

#total amount of money starting cart with $0.00
total = 0.00

#Display available items/introduction
print("Welcome to MooMoo's Magic Mart! This is my Online Grocery Store made for you!")
print("Here are the Available Items:")
print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")

#item used for the dictionary, stored as the key and values
for key, values in item_prices.items():
  print(key + " ---> "),
  for type, price in values.items(): #values is looking at the nested dictionaries
    print(type + ": $" + str(price)),
  print() #gives space between items/prices listed

print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
print("What would you like to add to your cart?")
print("Enter 'done' to proceed to check-out!")
```

```
#When the done_flag is not false (true), continue.
done_flag = False
while not done_flag:
    print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
    cart = input("Add to Cart: ")
#when done_flag is true (not true/false), skip code and go to outer print statement
    if cart.lower() == "done":
        done_flag = True
    else:
        found_item = False
        for item in item_prices: #item stored in the prices
            if cart.capitalize() == item:
                found_item = True #when an item is in the item_prices, code will continue.

        if not found_item:
            #if item is not in item_prices, code will print invalid item.
            print("Invalid Item")

        if cart.capitalize() == "Milk": #capitalize first letter of key
            print("Which milk?")
            print("2% or 3.25%?")
            print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
            milk_type = input("Select the milk type: ")
            #milk type (2% or 3.25%) not in item prices, jumps to outer print statement.
            while milk_type not in item_prices["Milk"]:
                print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
                print("Invalid milk type")
                #user has to re-write, till correct
                milk_type = input("Select the milk type: ")

            print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
            quantity = float(input("How much Milk do you need?: "))
            #item prices in milk for the milk type chosen
            milk_price = item_prices["Milk"][milk_type]
            #stored in the total variable with the milk price * customer amount * tax
            total += (milk_price  * 1.13) * quantity

        elif cart.capitalize() == "Bread": #capitalize first letter of key
            print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
            bread_type = "Regular" #type in the variable
            quantity = float(input("How much bread do you need?: "))
            #in item prices, pick the bread and bread types price
            bread_price = item_prices["Bread"][bread_type]
            #stored in the total variable with the bread price * customer amount * tax
            total += (bread_price * 1.13) * quantity

        elif cart.capitalize() == "Water": #capitalize first letter of key
            print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
            water_type = "Regular" #type in the variable
            quantity = float(input("How much water do you need?: "))
            # in item prices, picks the water and water types price; stored in regular
            water_price = item_prices["Water"][water_type]
            #stored in the total variable with the water price * customer amount * tax
    total += (water_price  * 1.13) * quantity

elif cart.capitalize() == "Eggs": #capitalize first letter of key
    print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
    egg_type = "A Dozen" #type in the variable
    quantity = float(input("How much Eggs (A dozen per box) do you need?: "))
    #in item prices, picks the egg and egg type
    egg_price = item_prices["Eggs"][egg_type]
    #stored in the total variable with the egg price * customer amount * tax
    total += (egg_price  * 1.13) * quantity

elif cart.capitalize() == "Butter": #capitalize first letter of key
    print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
    butter_type = "Regular" #type in the variable
    quantity = float(input("How much Butter do you need?: "))
    #in item prices, picks the butter and butter type
    butter_price = item_prices["Butter"][butter_type]
    #stored in the total variable with the butter price * customer amount * tax
    total += (butter_price  * 1.13) * quantity

elif cart.capitalize() == "Oil": #capitalize first letter of key
    print("Which oil?")
    print("Avocado oil or Olive oil")
    print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
    oil_type = input("Select the oil type: ")
    #when oil type (Avocado oil or Olive oil) not in item prices, print statement.
    while oil_type not in item_prices["Oil"]:
        print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
        print("Invalid oil type")
        #user has to re-write, till correct
        oil_type = input("Select the oil type: ")

    print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
    quantity = float(input("How much oil do you need?: "))
    #item prices in oil for the oil type.
    oil_price = item_prices["Oil"][oil_type]
    #stored in the total variable with the oil price * customer amount * tax
    total += (oil_price * 1.13) * quantity

elif cart.capitalize() == "Yogurt": #capitalize first letter of key
    print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
    yogurt_type = "Regular" #type in the variable
    quantity = float(input("How much Yogurt do you need?: "))
    #in item prices, picks the yogurt and yogurt type
    yogurt_price = item_prices["Yogurt"][yogurt_type]
    #stored in the total variable with the yogurt price * customer amount * tax
    total += (yogurt_price  * 1.13) * quantity

elif cart.capitalize() == "Coke": #capitalize first letter of key
    print("Which coke?")
    print("20 cans or Jug?")
    print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
    coke_type = input("Select the coke type: ")
    #when coke type(20 cans or jug) not in item prices, print statement.
```

```
        while coke_type not in item_prices["Coke"]:
            print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
            print("Invalid coke type")
            #user has to re-write, till correct
            coke_type = input("Select the coke type: ")

        print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
        quantity = float(input("How much coke do you need?: "))
        #item prices in coke for the coke type.
        coke_price = item_prices["Coke"][coke_type]
        #stored in the total variable with the coke price * customer amount * tax
        total += (coke_price * 1.13) * quantity

print("- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -")
#prints total and the exact amount of money rounded to the nearest 2nd digit. Total has all values in the nested
dictionary that are picked.
print("Total: $", round(total, 2))
print("Thanks for stopping by! :D")
```

2. **Detail what you added today.**

   Today in my code I added the for loop in order to be able to show the prices in the output of the program. The program will now show the user the many different prices of the items. I learned how to do for loops for the past week, so in order to use it, I started off by making a dictionary, and by using the key and values of that dictionary, I was able to easily show the [pices, without having to do much work. I had a few minutes after so i just tried modifying the docstring and other information making it look more clean, but i still need to finish it.

3. **Provide details for at least two coding items you plan to work on next class.**

   - Next class, i need to add more documentation
   - Want to see if there are any mistakes
   - If im done this, i want to start test cases