Dishanth Iyer

Mar 8, 2022

Foundations of Programming: Python

Assignment 08

https://github.com/dishiyer/ITFdnd100-Mod08/

Assignment 08: Objects and Classes

The seventh module focused on classes and objects.

Preparing for Assignment

In preparation for completing the assignment, I followed the Assignment04.docx step by step to gather an understanding of the following:

a. Classes vs Objects
b. Standard Class patterns
c. Fields
d. Constructors and Destructors
e. Attributes
f. Properties
g. Type Hints
h. Doc Strings
i. GiHub Desktop

In order to do so, I reviewed the module video, the references and websites. I also reviewed the textbook material.

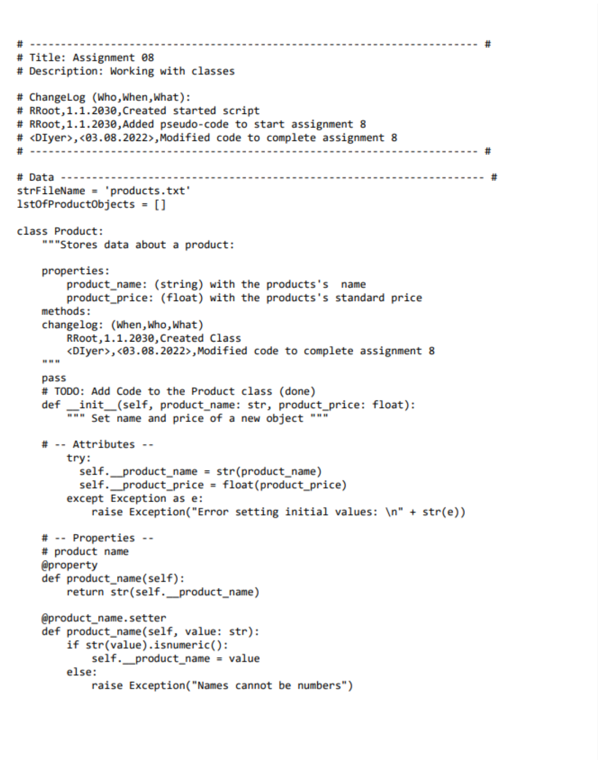
Performing the Assignment

I followed the instructions to create the Folders in the C: drive and creating the Assignment starter script. In addition, I installed PyCharm and followed the directions to setup the environment and directory. I tried to use comments in script to remind myself about the organization and purpose of the code. I also watched the videos for the class to get an understanding of what is expected. I finally finished the assignment by uploading to GitHub.

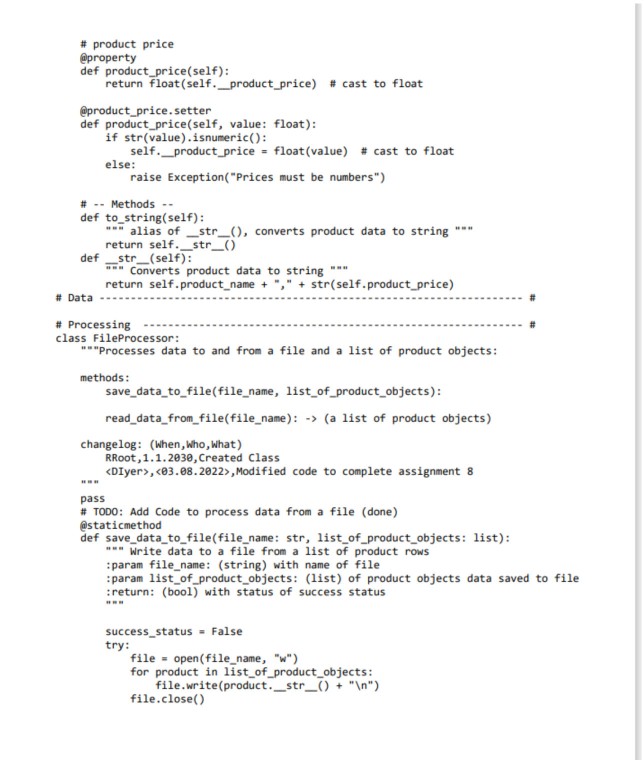
I updated my assignment after reviewing the Assignment module.

Create Python script

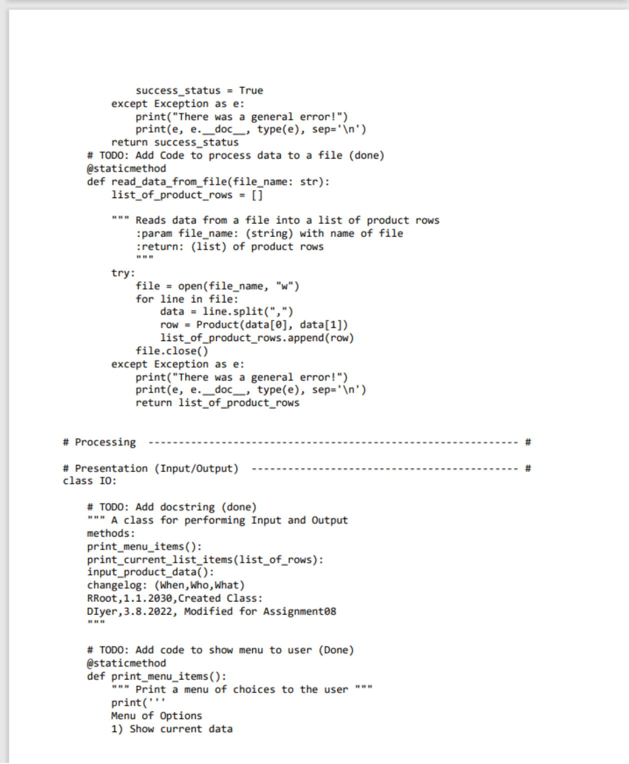
I created the Python script using the PyCharm IDE. I tried to add comments to explain the code as well.



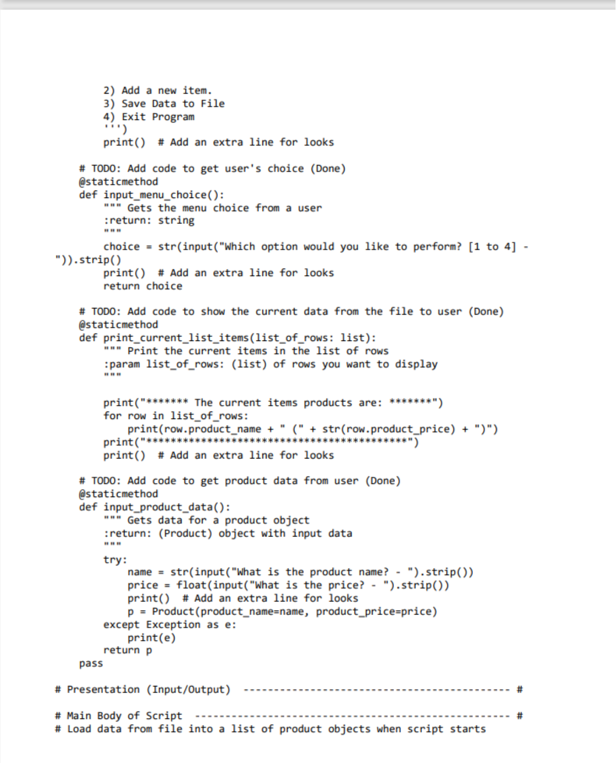
```
# ------------------------------------------------------------------------ #
# Title: Assignment 08
# Description: Working with classes

# ChangeLog (Who,When,What):
# RRoot,1.1.2030,Created started script
# RRoot,1.1.2030,Added pseudo-code to start assignment 8
# <DIyer>,<03.08.2022>,Modified code to complete assignment 8
# ------------------------------------------------------------------------ #

# Data -------------------------------------------------------------------- #
strFileName = 'products.txt'
lstOfProductObjects = []

class Product:
    """Stores data about a product:

    properties:
        product_name: (string) with the products's  name
        product_price: (float) with the products's standard price
    methods:
    changelog: (When,Who,What)
        RRoot,1.1.2030,Created Class
        <DIyer>,<03.08.2022>,Modified code to complete assignment 8
    """
    pass
    # TODO: Add Code to the Product class (done)
    def __init__(self, product_name: str, product_price: float):
        """ Set name and price of a new object """

    # -- Attributes --
        try:
          self.__product_name = str(product_name)
          self.__product_price = float(product_price)
        except Exception as e:
            raise Exception("Error setting initial values: \n" + str(e))

    # -- Properties --
    # product name
    @property
    def product_name(self):
        return str(self.__product_name)

    @product_name.setter
    def product_name(self, value: str):
        if str(value).isnumeric():
            self.__product_name = value
        else:
            raise Exception("Names cannot be numbers")
```



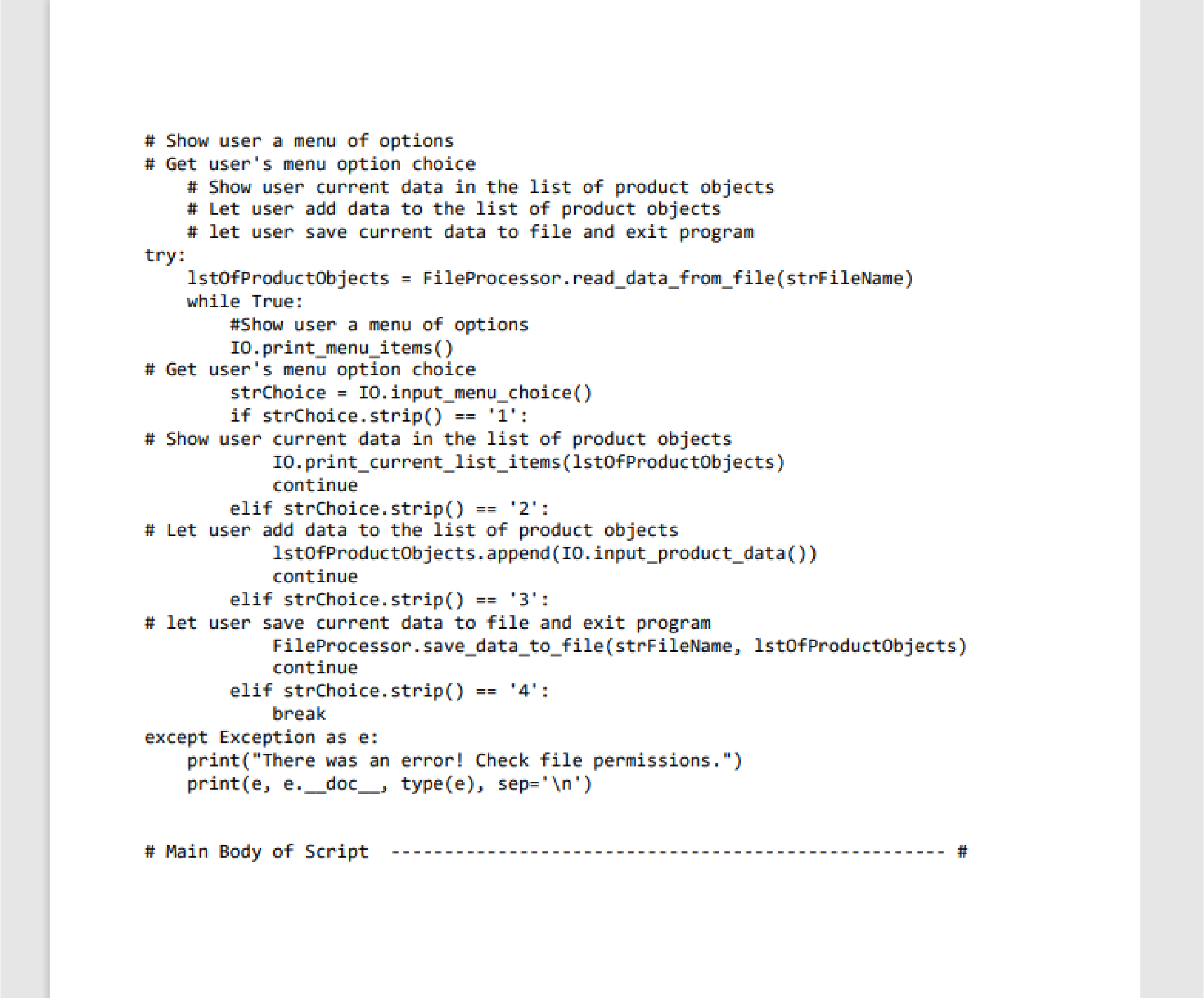
```
    # product price
    @property
    def product_price(self):
        return float(self.__product_price)  # cast to float

    @product_price.setter
    def product_price(self, value: float):
        if str(value).isnumeric():
            self.__product_price = float(value)  # cast to float
        else:
            raise Exception("Prices must be numbers")

    # -- Methods --
    def to_string(self):
        """ alias of __str__(), converts product data to string """
        return self.__str__()
    def __str__(self):
        """ Converts product data to string """
        return self.product_name + "," + str(self.product_price)
# Data -------------------------------------------------------------------- #

# Processing  ------------------------------------------------------------- #
class FileProcessor:
    """Processes data to and from a file and a list of product objects:

    methods:
        save_data_to_file(file_name, list_of_product_objects):

        read_data_from_file(file_name): -> (a list of product objects)

    changelog: (When,Who,What)
        RRoot,1.1.2030,Created Class
        <DIyer>,<03.08.2022>,Modified code to complete assignment 8
    """
    pass
    # TODO: Add Code to process data from a file (done)
    @staticmethod
    def save_data_to_file(file_name: str, list_of_product_objects: list):
        """ Write data to a file from a list of product rows
        :param file_name: (string) with name of file
        :param list_of_product_objects: (list) of product objects data saved to file
        :return: (bool) with status of success status
        """

        success_status = False
        try:
            file = open(file_name, "w")
            for product in list_of_product_objects:
                file.write(product.__str__() + "\n")
            file.close()
```



```
            success_status = True
        except Exception as e:
            print("There was a general error!")
            print(e, e.__doc__, type(e), sep='\n')
        return success_status
    # TODO: Add Code to process data to a file (done)
    @staticmethod
    def read_data_from_file(file_name: str):
        list_of_product_rows = []

        """ Reads data from a file into a list of product rows
            :param file_name: (string) with name of file
            :return: (list) of product rows
            """
        try:
            file = open(file_name, "w")
            for line in file:
                data = line.split(",")
                row = Product(data[0], data[1])
                list_of_product_rows.append(row)
            file.close()
        except Exception as e:
            print("There was a general error!")
            print(e, e.__doc__, type(e), sep='\n')
            return list_of_product_rows


# Processing  ------------------------------------------------------------- #

# Presentation (Input/Output)  -------------------------------------------- #
class IO:

    # TODO: Add docstring (done)
    """ A class for performing Input and Output
    methods:
    print_menu_items():
    print_current_list_items(list_of_rows):
    input_product_data():
    changelog: (When,Who,What)
    RRoot,1.1.2030,Created Class:
    DIyer,3.8.2022, Modified for Assignment08
    """

    # TODO: Add code to show menu to user (Done)
    @staticmethod
    def print_menu_items():
        """ Print a menu of choices to the user """
        print('''
        Menu of Options
        1) Show current data
```



```
        2) Add a new item.
        3) Save Data to File
        4) Exit Program
        ''')
        print()  # Add an extra line for looks

    # TODO: Add code to get user's choice (Done)
    @staticmethod
    def input_menu_choice():
        """ Gets the menu choice from a user
        :return: string
        """
        choice = str(input("Which option would you like to perform? [1 to 4] -
")).strip()
        print()  # Add an extra line for looks
        return choice

    # TODO: Add code to show the current data from the file to user (Done)
    @staticmethod
    def print_current_list_items(list_of_rows: list):
        """ Print the current items in the list of rows
        :param list_of_rows: (list) of rows you want to display
        """

        print("******* The current items products are: *******")
        for row in list_of_rows:
            print(row.product_name + " (" + str(row.product_price) + ")")
        print("*******************************************")
        print()  # Add an extra line for looks

    # TODO: Add code to get product data from user (Done)
    @staticmethod
    def input_product_data():
        """ Gets data for a product object
        :return: (Product) object with input data
        """
        try:
            name = str(input("What is the product name? - ").strip())
            price = float(input("What is the price? - ").strip())
            print()  # Add an extra line for looks
            p = Product(product_name=name, product_price=price)
        except Exception as e:
            print(e)
        return p
    pass

# Presentation (Input/Output)  -------------------------------------------- #

# Main Body of Script  ---------------------------------------------------- #
# Load data from file into a list of product objects when script starts
```



```
# Show user a menu of options
# Get user's menu option choice
    # Show user current data in the list of product objects
    # Let user add data to the list of product objects
    # let user save current data to file and exit program
try:
    lstOfProductObjects = FileProcessor.read_data_from_file(strFileName)
    while True:
        #Show user a menu of options
        IO.print_menu_items()
# Get user's menu option choice
        strChoice = IO.input_menu_choice()
        if strChoice.strip() == '1':
# Show user current data in the list of product objects
            IO.print_current_list_items(lstOfProductObjects)
            continue
        elif strChoice.strip() == '2':
# Let user add data to the list of product objects
            lstOfProductObjects.append(IO.input_product_data())
            continue
        elif strChoice.strip() == '3':
# let user save current data to file and exit program
            FileProcessor.save_data_to_file(strFileName, lstOfProductObjects)
            continue
        elif strChoice.strip() == '4':
            break
except Exception as e:
    print("There was an error! Check file permissions.")
    print(e, e.__doc__, type(e), sep='\n')


# Main Body of Script  ---------------------------------------------------- #
```

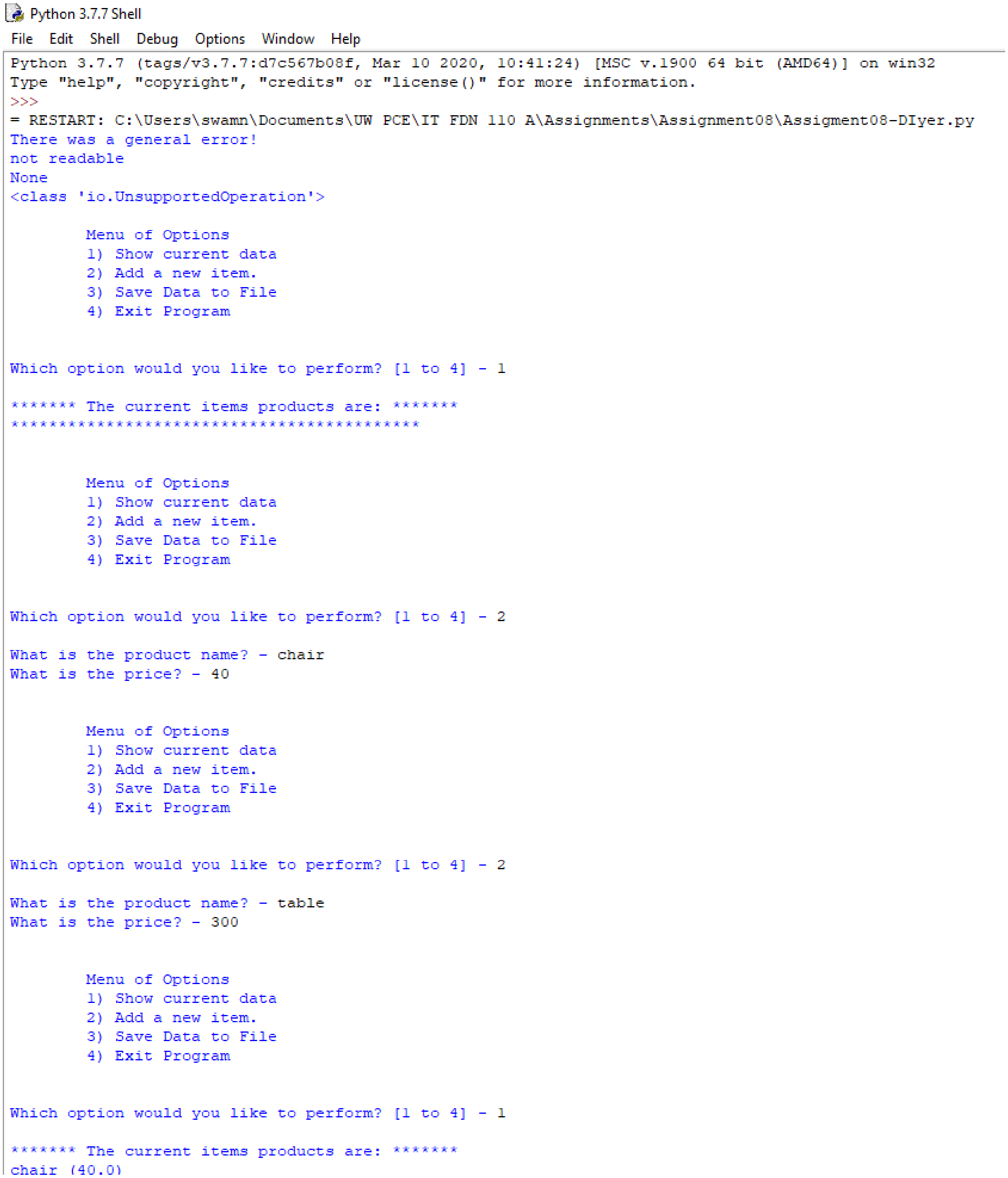
Utilizing the notes from the TA session and understanding how use Classes and Functions, I was able to complete the code.

I had to also review the Assignment answer to update my work.

Test Script

I tested the script within the IDE using the ‘Run’ function. The script executed successfully. The test also output the text file.

Below is a PDF output of the “Run” window results from PyCharm. I ran it twice to check the error handling as well as a good input.



```
Python 3.7.7 (tags/v3.7.7:d7c567b08f, Mar 10 2020, 10:41:24) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
= RESTART: C:\Users\swamn\Documents\UW PCE\IT FDN 110 A\Assignments\Assignment08\Assigment08-DIyer.py
There was a general error!
not readable
None
<class 'io.UnsupportedOperation'>

        Menu of Options
        1) Show current data
        2) Add a new item.
        3) Save Data to File
        4) Exit Program


Which option would you like to perform? [1 to 4] - 1

******* The current items products are: *******
*******************************************


        Menu of Options
        1) Show current data
        2) Add a new item.
        3) Save Data to File
        4) Exit Program


Which option would you like to perform? [1 to 4] - 2

What is the product name? - chair
What is the price? - 40


        Menu of Options
        1) Show current data
        2) Add a new item.
        3) Save Data to File
        4) Exit Program


Which option would you like to perform? [1 to 4] - 2

What is the product name? - table
What is the price? - 300


        Menu of Options
        1) Show current data
        2) Add a new item.
        3) Save Data to File
        4) Exit Program


Which option would you like to perform? [1 to 4] - 1

******* The current items products are: *******
chair (40.0)
```

Run Script

Once I was confident enough, I opened the Command Prompt in Windows and ran the script.

My code works as intended.