Deborah Barndt

ITMD 513 Open Source Programming

Professor Dr. Sam

Hw9

4-3-19

'''

Deborah Barndt

4-3-19

VendingMachine.py

hw9: Coffee Vending Machine

This program will create a vending machine that will tell the user how many

cups of coffee are available, the cost of one cup of coffee, and the total

amount of money that the user inserted into the machine. The machine will

require exact change.

Written by Deborah Barndt.

'''

class CoffeeVendingMachine:

numCups = 0

costCup = 0.00

totalMoneyInsert = 0.00

# Constructor for coffee vending machine.

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

# Quantity of the coffee available in the machine.

self.numCups = quantity

# Cost of the coffe per cup.

self.costCup = price

# Total amount of money inserted by the user.

self.totalMoneyInsert = 0

# Function that displays the vending machine menu, the quantity available, and cost per cup.

def menu(self):

print('\n--------------------------------------')

print(' Coffee Vending Machine\n')

print('Quantity of coffee available: {0}'.format(self.numCups, '%d'))

print('Price per cup:', format(self.costCup, '.2f'))

print('--------------------------------------\n\n')

# Function that inserts the given amount of money inserted by the user.

def insert(self, quarters, dimes, nickels):

# Calculate the money inserted by the user in the vending machine.

self.totalMoneyInsert += (quarters \* 0.25) + (dimes \* 0.10) + (nickels \* 0.05)

print('\n----------------------')

print(format(self.totalMoneyInsert, '.2f'), 'inserted')

print('----------------------\n')

# Function that dispenses the cup of coffee if the user has inserted enough money and

# coffee is available, otherwise displays a message.

def select(self):

print('\n------------------------------------------------------')

# If the amount inserted is less than the cost of one cup.

if (self.totalMoneyInsert < self.costCup):

print('Not enough funds. Please insert more money.')

# If the required cups of coffee is less than or equal to the quantity available.

elif ((self.totalMoneyInsert // self.costCup) <= self.numCups):

print('Thank you. Please collect your {0} cups of coffee.'.format(self.totalMoneyInsert // self.costCup))

# Calculate the total cups of coffee left available in the machine after request from user.

self.numCups = self.numCups - (self.totalMoneyInsert // self.costCup)

self.totalMoneyInsert = 0.00

# If the cups of coffee ordered is not available.

else:

print('\n------------------------------------------------------')

print('Not enough coffee available. Please come back later.')

print('------------------------------------------------------\n')

# Function that returns the money that was inserted to the user.

def refund(self):

print('\n------------------------------------------------------')

if (self.totalMoneyInsert == 0.00):

print('Money was not inserted. Cannot give refund.')

else:

print('Refund issued. Please collect {0:.2f} from return slot.'.format(self.totalMoneyInsert, '.2f'))

self.totalMoneyInsert = 0.00

print('------------------------------------------------------\n')

'''

Deborah Barndt

4-3-19

TestMain.py

hw9: Coffee Vending Machine

This program will create a vending machine that will tell the user how many

cups of coffee are available, the cost of one cup of coffee, and the total

amount of money that the user inserted into the machine. The machine will

require exact change.

Written by Deborah Barndt.

'''

from VendingMachine import CoffeeVendingMachine

def testmain():

option = ''

quantity = 10

cost = 0.50

vendingMachine = CoffeeVendingMachine(quantity, cost)

print('\n\nWelcome to the Coffee Vending Machine.')

while option.lower() != 'exit':

option = input('Please select an option from the menu by number or name:\n'

'1. Menu\n'

'2. Insert Money\n'

'3. Select Coffee\n'

'4. Refund Money\n'

'5. Exit\n\n'

'Select Option: ')

# If statement for each of the options on the menu.

if (option.lower() in '1. menu'):

vendingMachine.menu()

elif (option.lower() in '2. insert money'):

# Ask user to enter the number of quarters, dimes, and nickels they are inserting.

quarters = int(input('Enter the number of quarters inserted: '))

dimes = int(input('Enter the number of dimes inserted: '))

nickels = int(input('Enter the number of nickels inserted: '))

vendingMachine.insert(quarters, dimes, nickels)

elif (option.lower() in '3. select coffee'):

vendingMachine.select()

elif (option.lower() in '4. refund money'):

vendingMachine.refund()

elif (option.lower() in '5. exit'):

break

else:

print('Your input is invalid. Please enter a number or name from the menu.\n')

print('Thank you for using the Coffee Vending Machine.')

# Call testmain function to begin program.

testmain()

Output Result:















