**HW – 9**

**Source Code –**

*"""  
This class creates a coffee vending machine which is responsible for displaying the  
 menu of the coffee available,allowing the user to insert the money,selecting the  
 coffee ased on the number of cups of coffee available and the total cost inserted  
 by the user and refunding the money back to the user if he cancels the coffee/  
 if the user has not inserted the correct change.  
"""***class** CoffeeVendingMachine:  
 no\_of\_cups\_of\_coffee\_available = 0  
 cost\_of\_one\_coffeecup = 0.0  
 quarters = 0  
 dimes = 0  
 nickels = 0  
  
 *# Define a constructor for the quantity and the price of the coffee* **def** \_\_init\_\_(self, no\_of\_cups\_coffee, cost\_one\_coffeecup):  
 self.no\_of\_cups\_of\_coffee\_available = no\_of\_cups\_coffee  
 self.cost\_of\_one\_coffeecup = cost\_one\_coffeecup  
  
 *# Menu method displays the quantity and the price of the coffee* **def** menu(self):  
 **print**(**"The Coffee Menu is as follows : "**)  
 **print "{:>3}{:>30}{:>30}"**.format(**"Options"**,**"Coffee Name "**,**"Coffee Price Per Cup "**)  
 **print "{:>3}{:>32}{:>25}"**.format(**"1"**,**"Cappuccino"**,**"$4.25"**)  
 **print "{:>3}{:>37}{:>20}"**.format(**"2"**,**"Caffe Macchiato"**,**"$4.10"**)  
 **print "{:>3}{:>33}{:>23}"**.format(**"3"**,**"Caffe Mocha"**,**"$4.0"**)  
 **print "{:>3}{:>33}{:>24}"**.format(**"4"**,**"Caffe Latte"** ,**"$4.35"**)  
 **print "{:>3}{:>34}{:>23}"**.format(**"5"**,**"Irish Coffee"** ,**"$4.05"**)  
  
 **while** True:  
 **try**:  
 choice = int(input(**"Enter the choice of your coffee : "**))  
 **assert** (choice > 0.0), **"Choice should be greater than zero!!!"  
 break  
 except**:  
 **print "Enter a valid value for the choice!!!"  
 return** choice  
  
 *# It returns the cost of the coffee based on the user input* **def** retrieveCost(self, choice):  
 **if** (choice == 1):  
 **return** 4.25  
 **if** (choice == 2):  
 **return** 4.10  
 **if** (choice == 3):  
 **return** 4.0  
 **if** (choice == 4):  
 **return** 4.35  
 **if** (choice == 5):  
 **return** 4.05  
 **else**:  
 **return** 0  
  
 *# It allows the user to enter the given amount* **def** insert(self, quarters, dimes, nickels):  
 total\_amount = quarters \* 0.25 + dimes \* 0.1 + nickels \* 0.05  
 **return** total\_amount  
  
 *# Selects the coffee based on the number of cups of coffee available and the total money inserted* **def** select(self, cost, total\_amount, no\_of\_cups\_of\_coffee\_available):  
 **if** (total\_amount == cost **and** no\_of\_cups\_of\_coffee\_available > 0):  
 no\_of\_cups\_of\_coffee\_available -= 1  
 **print "Coffee is available"  
 print "Number of cups now available is:"**,no\_of\_cups\_of\_coffee\_available  
 **return** True  
 **if** (total\_amount < cost):  
 **print "Insufficient Funds"** obj.refund(total\_amount)  
 **return** False  
  
 *# It refunds the money back to the user if the coffee is not available* **def** refund(self, total\_amount):  
 **return** total\_amount  
  
*#Object Creation and calling the methods defined in the class*obj = CoffeeVendingMachine(5, 2.5)  
  
choice = obj.menu()  
**print "Choice is: "**, choice  
**while** True:  
 **try**:  
 quarters = int(input(**"Enter the quarters for the coffee:"**))  
 **assert** (quarters >= 0.0) ,**"Quarters should be greater than zero!!!"  
 break  
 except**:  
 **print "Enter a valid value for the quarters!!!"  
  
while** True:  
 **try**:  
 dimes = int(input(**"Enter the dimes for the coffee:"**))  
 **assert** (dimes >= 0.0) ,**"Dimes should be greater than zero!!!"  
 break  
 except**:  
 **print "Enter a valid value for the dimes!!!"  
  
while** True:  
 **try**:  
 nickels = int(input(**"Enter the nickels for the coffee:"**))  
 **assert** (nickels >= 0.0) ,**"Nickels should be greater than zero!!!"  
 break  
 except**:  
 **print "Enter a valid value for the nickels!!!"**total\_amount = obj.insert(quarters, dimes, nickels)  
**print "Total Amount entered by the user is :"**, total\_amount  
  
cost = obj.retrieveCost(choice)  
**print "Cost is: "**, cost  
  
**while** True:  
 **try**:  
 cancel\_coffee = raw\_input(**"Do you want to cancel the coffee(Y/N)?"**)  
 **assert** ((cancel\_coffee == **"Y"**) **or**(cancel\_coffee == **"N"**)) ,**"The input should be Y/N!!!"  
 break  
 except**:  
 **print "Enter a valid value Y/N!!!"  
  
if** (cancel\_coffee == **"Y"**):  
 refund\_amount = obj.refund(total\_amount)  
 **print "Amount available for refund:"**, total\_amount  
  
**if** (cancel\_coffee == **"N"**):  
 decision = obj.select(cost, total\_amount, 5)

**Output –**

1./usr/bin/python2.7 /Users/apple/PycharmProjects/Labs/CoffeeVendingMachine.py

The Coffee Menu is as follows :

Options Coffee Name Coffee Price Per Cup

1 Cappuccino $4.25

2 Caffe Macchiato $4.10

3 Caffe Mocha $4.0

4 Caffe Latte $4.35

5 Irish Coffee $4.05

Enter the choice of your coffee : 2

Choice is: 2

Enter the quarters for the coffee:16

Enter the dimes for the coffee:1

Enter the nickels for the coffee:0

Total Amount entered by the user is : 4.1

Cost is: 4.1

Do you want to cancel the coffee(Y/N)?Y

Amount available for refund: 4.1

Process finished with exit code 0

2. /usr/bin/python2.7 /Users/apple/PycharmProjects/Labs/CoffeeVendingMachine.py

The Coffee Menu is as follows :

Options Coffee Name Coffee Price Per Cup

1 Cappuccino $4.25

2 Caffe Macchiato $4.10

3 Caffe Mocha $4.0

4 Caffe Latte $4.35

5 Irish Coffee $4.05

Enter the choice of your coffee : 2

Choice is: 2

Enter the quarters for the coffee:16

Enter the dimes for the coffee:1

Enter the nickels for the coffee:0

Total Amount entered by the user is : 4.1

Cost is: 4.1

Do you want to cancel the coffee(Y/N)?N

Coffee is available

Number of cups now available is: 4

Process finished with exit code 0

3. /usr/bin/python2.7 /Users/apple/PycharmProjects/Labs/CoffeeVendingMachine.py

The Coffee Menu is as follows :

Options Coffee Name Coffee Price Per Cup

1 Cappuccino $4.25

2 Caffe Macchiato $4.10

3 Caffe Mocha $4.0

4 Caffe Latte $4.35

5 Irish Coffee $4.05

Enter the choice of your coffee : 5

Choice is: 5

Enter the quarters for the coffee:16

Enter the dimes for the coffee:0

Enter the nickels for the coffee:1

Total Amount entered by the user is : 4.05

Cost is: 4.05

Do you want to cancel the coffee(Y/N)?Y

Amount available for refund: 4.05

Process finished with exit code 0

4. /usr/bin/python2.7 /Users/apple/PycharmProjects/Labs/CoffeeVendingMachine.py

The Coffee Menu is as follows :

Options Coffee Name Coffee Price Per Cup

1 Cappuccino $4.25

2 Caffe Macchiato $4.10

3 Caffe Mocha $4.0

4 Caffe Latte $4.35

5 Irish Coffee $4.05

Enter the choice of your coffee : 3

Choice is: 3

Enter the quarters for the coffee:16

Enter the dimes for the coffee:0

Enter the nickels for the coffee:7

Total Amount entered by the user is : 4.35

Cost is: 4.0

Do you want to cancel the coffee(Y/N)?N

Process finished with exit code 0

5. /usr/bin/python2.7 /Users/apple/PycharmProjects/Labs/CoffeeVendingMachine.py

The Coffee Menu is as follows :

Options Coffee Name Coffee Price Per Cup

1 Cappuccino $4.25

2 Caffe Macchiato $4.10

3 Caffe Mocha $4.0

4 Caffe Latte $4.35

5 Irish Coffee $4.05

Enter the choice of your coffee : 4

Choice is: 4

Enter the quarters for the coffee:16

Enter the dimes for the coffee:2

Enter the nickels for the coffee:1

Total Amount entered by the user is : 4.25

Cost is: 4.35

Do you want to cancel the coffee(Y/N)?N

Insufficient Funds

Process finished with exit code 0

6. /usr/bin/python2.7 /Users/apple/PycharmProjects/Labs/CoffeeVendingMachine.py

The Coffee Menu is as follows :

Options Coffee Name Coffee Price Per Cup

1 Cappuccino $4.25

2 Caffe Macchiato $4.10

3 Caffe Mocha $4.0

4 Caffe Latte $4.35

5 Irish Coffee $4.05

Enter the choice of your coffee : 5

Choice is: 5

Enter the quarters for the coffee:16

Enter the dimes for the coffee:0

Enter the nickels for the coffee:1

Total Amount entered by the user is : 4.05

Cost is: 4.05

Do you want to cancel the coffee(Y/N)?N

Coffee is available

Number of cups now available is: 4

Process finished with exit code 0