./

GENESIS – ADVANCE-PYTHON

PROJECT REPORT

CAFETERIA BILLING SYSTEM



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver. Rel. No.** | **Release Date** | **Prepared. By** | **Reviewed By** | **To be Approved** | **Remarks/Revision Details** |
| 1.0 | 12/12/2020 | Parepalli Sai Kumar |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

**Details**

Contents

[Contents 3](#_Toc58780302)

[Activity 1 4](#_Toc58780303)

[1.1 Intial Code 4](#_Toc58780304)

[1.2 Pep8Online: 6](#_Toc58780305)

[Figure 1: Pep8 Online Snapshot 6](#_Toc58780306)

[1.3 Jupiter Code Output: 7](#_Toc58780307)

[Figure 2: Jupiter Output for activity 1 7](#_Toc58780308)

[Activity 2 8](#_Toc58780309)

[2.1 Code with OOPS concepts: 8](#_Toc58780310)

[2.2 Pepe8 Online: 12](#_Toc58780311)

[Figure 3: Pep8 Output Snapshot Using OOPS 13](#_Toc58780312)

[2.3 Output Code: 13](#_Toc58780313)

[Figure 4: Jupiter Output Using OOPS 13](#_Toc58780314)

[3 Github link: 14](#_Toc58780315)

**Table of Figures**

[Figure 1: Pep8 Online Snapshot 6](#_Toc58780010)

[Figure 2: Jupiter Output for activity 1 7](#_Toc58780011)

[Figure 3: Pep8 Output Snapshot Using OOPS 12](#_Toc58780012)

[Figure 4: Jupiter Output Using OOPS 13](#_Toc58780013)

# Activity 1

Cafeteria Billing System

## 1.1 Intial Code

class Cafeteria:

def meals(self):

print("Enter the Quantity ")

m = int(input())

c = m\*100

return c

def chapathi(self):

print("Enter the Quantity ")

m = int(input())

c = m\*30

return c

def noodles(self):

print("Enter the Quantity ")

m = int(input())

c = m\*40

return c

def coffee(self):

print("Enter the Quantity ")

m = int(input())

c = m\*15

return c

def Tea(self):

print("Enter the Quantity ")

m = int(input())

c = m\*12

return c

def Pizza(self):

print("Enter the Quantity ")

m = int(input())

c = m\*30

return c

def Burger(self):

print("Enter the Quantity ")

m = int(input())

c = m\*35

return c

def French\_Fries(self):

print("Enter the Quantity ")

m = int(input())

c = m\*25

return c

def Dosa(self):

print("Enter the Quantity ")

m = int(input())

c = m\*20

return c

obj = Cafeteria()

b = 0

print("Welcome to Cafeteria")

print("Please select the necessities that are available")

print("1 for Meals the cost is 50/-")

print("2 for Chapthi the cost is 30/-")

print("3 for noodles the cost is 40/-")

print("4 for coffee the cost is 15/-")

print("5 for Tea the cost is 12/-")

print("6 for Pizza the cost is 30/-")

print("7 for Burger the cost is 35/-")

print("8 for French\_Fries the cost is 25/-")

print("9 for Dosa the cost is 20/-")

while 1:

N = int(input("Enter the choice:"))

if (N == 1):

b = b+obj.meals()

elif (N == 2):

b = b+obj.chapathi()

elif (N == 3):

b = b+obj.noodles()

elif (N == 4):

b = b+obj.coffee()

elif (N == 5):

b = b+obj.Tea()

elif (N == 6):

b = b+obj.Pizza()

elif (N == 7):

b = b+obj.Burger()

elif (N == 8):

b = b+obj.French\_Fries()

elif (N == 9):

b = b+obj.Dosa()

else:

break

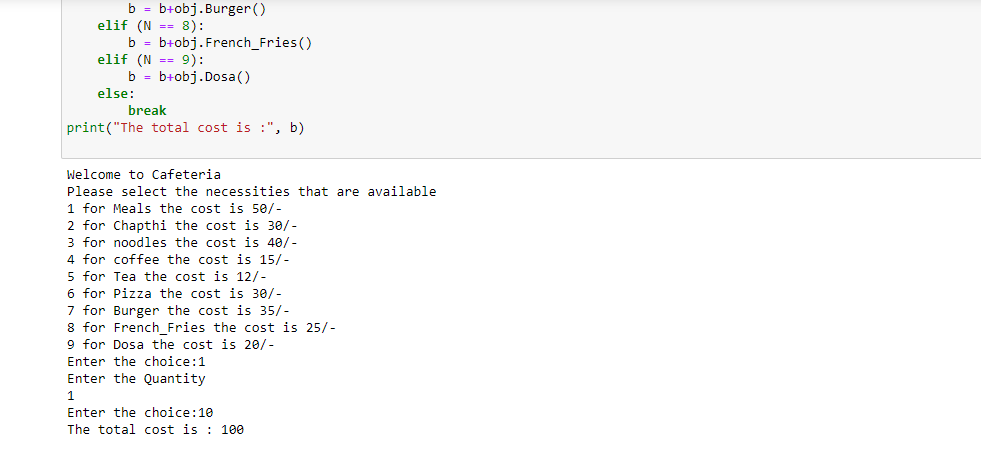
print("The total cost is :", b)

## 1.2 Pep8Online:



### Figure 1: Pep8 Online Snapshot

## 1.3 Jupiter Code Output:



### Figure 2: Jupiter Output for activity 1

# Activity 2

## 2.1 Code with OOPS concepts:

import re

class drinks():

# new class 'drinks' is created

def pepsi(self):

# new method pepsi is created

print("Enter the number of bottles required ")

# quantity entered by user

self.c = (int(input())).\_\_mul\_\_(15)

return self.c

# returns the total cost of pepsi only

def sprite(self):

# new method sprite is created

print("Enter the number of bottles required ")

self.c = (int(input())).\_\_mul\_\_(15)

return self.c

# returns the total cost of sprite only

class lunch(drinks):

# new sub class 'lunch' is created from super class 'drinks'

def meals(self):

# new method meals is created

print("Enter the number of plates ")

self.c = (int(input())).\_\_mul\_\_(50)

return self.c

# returns the total cost of meals only

class break\_fast(drinks):

# new sub class 'break\_fast' is created from super class 'drinks'

def chapathi(self):

# new method chapathi is created

print("Enter the number of plates ")

self.c = (int(input())).\_\_mul\_\_(30)

return self.c

# returns the total cost of chapathi only

def Dosa(self):

# new method dosa is created

print("Enter the number of plates ")

self.c = (int(input())).\_\_mul\_\_(20)

return self.c

# returns the total cost of dosa only

def Idli(self):

# new method idli is created

print("Enter the number of plates ")

self.c = (int(input())).\_\_mul\_\_(10)

return self.c

# returns the total cost idli only

class snacks(drinks):

# new sub class 'snacks' is created from super class 'drinks'

def coffee(self):

# new method coffee is created

print("Enter the number of cups ")

self.c = (int(input())).\_\_mul\_\_(15)

return self.c

# returns the total cost coffee only

def Tea(self):

# new method Tea is created

print("Enter the number of cups ")

self.c = (int(input())).\_\_mul\_\_(12)

return self.c

# returns the total cost Tea only

def noodles(self):

# new method noodels is created

print("Enter the number of plates ")

self.c = (int(input())).\_\_mul\_\_(40)

return self.c

# returns the total cost noodels only

def Pizza(self):

# new method Pizza is created

print("Enter the number pizza's required ")

self.c = (int(input())).\_\_mul\_\_(30)

return self.c

# returns the total cost Pizza only

def Burger(self):

# new method Burger is created

print("Enter the number of burgers ")

self.c = (int(input())).\_\_mul\_\_(35)

return self.c

# returns the total cost Burger only

def French\_Fries(self):

# new method French\_Fries is created

print("Enter the number of plates ")

self.c = (int(input())).\_\_mul\_\_(25)

return self.c

# returns the total cost French Fries only

class menu:

# new class created to display the menu items

def call(self):

# new method call is created

print("Welcome to Cafeteria")

print("Please select the necessities that are available")

print("1 for Meals the cost is 50/-")

print("2 for Chapthi the cost is 30/-")

print("3 for noodles the cost is 40/-")

print("4 for coffee the cost is 15/-")

print("5 for Tea the cost is 12/-")

print("6 for Pizza the cost is 30/-")

print("7 for Burger the cost is 35/-")

print("8 for French\_Fries the cost is 25/-")

print("9 for Dosa the cost is 20/-")

print("10 for Idli the cost is 10/-")

print("11 for pepsi the cost is 15")

print("12 for sprite the cost is 15")

class choice(lunch, break\_fast, snacks):

# new class choice acts as child class for lunch,breakfast and snacks

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

# constructor to intialize values

self.b = 0

def select(self):

# to select the particulat item in the menu

while 1:

print("Please Select the Appropriate Number for the item")

while 1:

try:

N = int(input("Enter the choice"))

break

except:

print("Please select the valid number")

if (N == 1):

self.b = (self.b).\_\_add\_\_(obj.meals())

elif (N == 2):

self.b = (self.b).\_\_add\_\_(obj.chapathi())

elif (N == 3):

self.b = (self.b).\_\_add\_\_(obj.noodles())

elif (N == 4):

self.b = (self.b).\_\_add\_\_(obj.coffee())

elif (N == 5):

self.b = (self.b).\_\_add\_\_(obj.Tea())

elif (N == 6):

self.b = (self.b).\_\_add\_\_(obj.Pizza())

elif (N == 7):

self.b = (self.b).\_\_add\_\_(obj.Burger())

elif (N == 8):

self.b = (self.b).\_\_add\_\_(obj.French\_Fries())

elif (N == 9):

self.b = (self.b).\_\_add\_\_(obj.Dosa())

elif (N == 10):

self.b = (self.b).\_\_add\_\_(obj.Idli())

elif (N == 11):

self.b = (self.b).\_\_add\_\_(obj.pepsi())

elif (N == 12):

self.b = (self.b).\_\_add\_\_(obj.sprite())

else:

break

print("The total Cost:", self.b)

class Final:

# Final class is created

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

self.lg = self.Visting()

class Visiting:

def display(self):

print("Thank you for Visting")

print(re.match("[A-Z a-z]+", "Cafeteria"))

print(re.match("[A-Z a-z]+", "Billing"))

print(re.match("[A-z a-z]+", "System"))

object1 = menu()

object1.call()

obj = choice()

obj.select()

g = Final.Visiting()

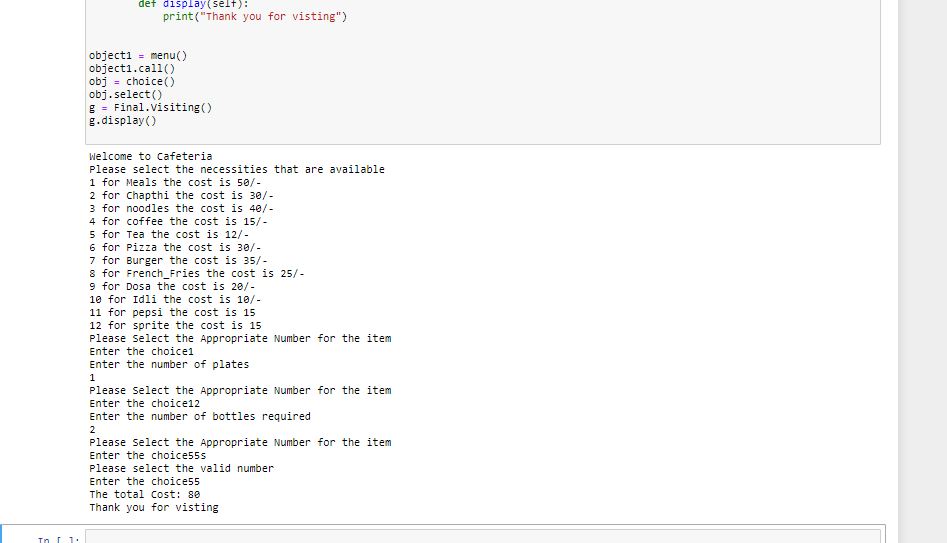
g.display()

## 2.2 Pepe8 Online:



### Figure 3: Pep8 Output Snapshot Using OOPS

## 2.3 Output Code:



### Figure 4: Jupiter Output Using OOPS

3 Github link:

<https://github.com/99003189/Advance_Python_Project.git>