class MenuItem:  
 def \_\_init\_\_(self, name, water, milk, coffee, cost):  
 self.name = name  
 self.cost = cost  
 self.ingredients = {  
 "water": water,  
 "milk": milk,  
 "coffee": coffee,  
 }  
  
  
class Menu:  
 def \_\_init\_\_(self):  
 self.menu = [  
 MenuItem(name="latte", water=200, milk=150, coffee=24, cost=2.5),  
 MenuItem(name="espresso", water=50, milk=0, coffee=18, cost=1.5),  
 MenuItem(name="cappuccino", water=250, milk=100, coffee=24, cost=3.0),  
 ]  
  
 def get\_items(self):  
 return "/".join([item.name for item in self.menu])  
  
 def find\_drink(self, order\_name):  
 for item in self.menu:  
 if item.name == order\_name:  
 return item  
 print("Sorry, that item is not available.")  
 return None  
  
  
class CoffeeMaker:  
 def \_\_init\_\_(self):  
 self.resources = {  
 "water": 300,  
 "milk": 200,  
 "coffee": 100,  
 }  
  
 def report(self):  
 print(f"Water: {self.resources['water']}ml")  
 print(f"Milk: {self.resources['milk']}ml")  
 print(f"Coffee: {self.resources['coffee']}g")  
  
 def is\_resource\_sufficient(self, drink):  
 for item in drink.ingredients:  
 if drink.ingredients[item] > self.resources[item]:  
 print(f"Sorry, there is not enough {item}.")  
 return False  
 return True  
  
 def make\_coffee(self, drink):  
 for item in drink.ingredients:  
 self.resources[item] -= drink.ingredients[item]  
 print(f"Here is your {drink.name} ☕. Enjoy!")  
  
  
class MoneyMachine:  
 CURRENCY = "$"  
  
 COIN\_VALUES = {  
 "quarters": 0.25,  
 "dimes": 0.10,  
 "nickels": 0.05,  
 "pennies": 0.01,  
 }  
  
 def \_\_init\_\_(self):  
 self.profit = 0  
 self.money\_received = 0  
  
 def report(self):  
 print(f"Money: {self.CURRENCY}{self.profit}")  
  
 def process\_coins(self):  
 print("Please insert coins.")  
 total = 0  
 for coin, value in self.COIN\_VALUES.items():  
 count = int(input(f"How many {coin}?: "))  
 total += count \* value  
 return total  
  
 def make\_payment(self, cost):  
 self.money\_received = self.process\_coins()  
 if self.money\_received >= cost:  
 change = round(self.money\_received - cost, 2)  
 if change > 0:  
 print(f"Here is {self.CURRENCY}{change} in change.")  
 self.profit += cost  
 return True  
 else:  
 print("Sorry, that's not enough money. Money refunded.")  
 return False  
  
  
# ---------- Main Coffee Machine Program ----------  
  
money\_machine = MoneyMachine()  
coffee\_maker = CoffeeMaker()  
menu = Menu()  
  
is\_on = True  
  
while is\_on:  
 options = menu.get\_items()  
 choice = input(f"What would you like? ({options}): ").lower()  
 if choice == "off":  
 is\_on = False  
 elif choice == "report":  
 coffee\_maker.report()  
 money\_machine.report()  
 else:  
 drink = menu.find\_drink(choice)  
 if drink and coffee\_maker.is\_resource\_sufficient(drink):  
 if money\_machine.make\_payment(drink.cost):  
 coffee\_maker.make\_coffee(drink)