**Coffee Machine Program Requirements**

1. **Prompt user by asking “**What would you like? (espresso / latte / cappuccino)​​**”**

* Check the user’s input to decide what to do next.
* The prompt should show every time action has completed, e.g. once the drink is dispensed. The prompt should show again to serve the next customer.

1. **Turn off the Coffee Machine by entering “off**​​**” to the prompt.**

* For maintainers of the coffee machine, they can use “off” as the secret word to turn off the machine. Your code should end execution when this happens.

1. **Print report.**

* When the user enters “report” to the prompt, a report should be generated that shows the current resource values. e.g.

**Water: 100ml**

**Milk: 150ml**

**Coffee: 24g**

1. **Print Profit.**

* When the user enters “profit” to the prompt, a report should be generated that shows the current profit values. e.g.
* **Money: ₹25**

1. **Check resources sufficient?**

* When the user chooses a drink, the program should check if there are enough resources to make that drink.
* E.g. if Latte requires 150ml water but there is only 100ml left in the machine. It should not continue to make the drink but print: “**Sorry there is not enough water.​**”
* The same should happen if another resource is depleted, e.g. milk or coffee.

1. **Process Payments.**

* If there are sufficient resources to select the drink, then the program should prompt the user to select option 1 to insert coins or option 2 to scan and pay.
* Remember that ₹1, ₹2, ₹5 and ₹10 coins are acceptable in machine.
* Calculate the monetary value of the coins inserted. E.g. 1-(₹1), 2-(₹2), 2-(₹5) and 1-(₹10) = 1\*1 + 2 x 2 + 1\*5 + 1\*10 = ₹20
* Option 2 is for online payment like scanning QR and paying the amount. But instead, You have to Enter the Amount of the coffee price.

1. **Check transaction successful?**

* Check that the user has inserted enough money to purchase the drink they selected. E.g. Latte costs ₹25, but they only inserted ₹20 then after counting the coins the program should say “**Sorry that's not enough money. Money refunded.​​**”.
* But if the user has inserted enough money, then the cost of the drink gets added to the machine as the profit and this will be reflected the next time the “Profit Report” is triggered. E.g.

**Water: 100ml**

**Milk: 150ml**

**Coffee: 24g**

**Money: ₹25**

* If the user has inserted too much money, the machine should offer a change.

E.g. “Here is ₹5 Rupees in change.”

1. **Make Coffee.**

* If the transaction is successful and there are enough resources to make the drink the user selected, then the ingredients to make the drink should be deducted from the coffee machine resources, and the price of coffee added to the profit.

E.g.

report before purchasing a latte:

**Water: 300ml**

**Milk: 200ml**

**Coffee: 100g**

Profit before purchasing a latte:

**Money: ₹0**

Report after purchasing latte:

**Water: 200ml**

**Milk: 50ml**

**Coffee: 76g**

Profit after purchasing latte:

**Money: ₹25**

* Once all resources have been deducted and profit added, tell the user “Here is your latte. Enjoy! Have a nice day!”. If a latte was their choice of drink.