

Coffee Vending Machine

Packages & Classes

1. io.shubham.coffeemachine.Configurations

- a. **Configuration.java** – Contains logic for retrieving Initial Configuration like Ingredients Quantity available, Ingredients required for various drinks and their corresponding quantities and number of outlets from json file.

2. io.shubham.coffeemachine.Drinks –

- a. **Drinks.java** – A base class where in we have specified various member variables that are required to make a drink. It is an abstract class since no object is made up of this class.
- b. **BlackTea.java** – Extends base class Drinks.java
- c. **GreenTea.java** - Extends base class Drinks.java
- d. **HotTea.java** - Extends base class Drinks.java
- e. **HotCoffee.java** - Extends base class Drinks.java

3. io.shubham.coffeemachine.Ingredients

- a. **Ingredient.java** – Contains member variables indicating the quantity of various ingredients available, also the minimum number of required ingredients quantity. It has the logic to check if a request can be fulfilled at a time i.e. enough ingredients are available to create a specific type of drink. It also has the logic to check for low quantity of ingredients available at a time.

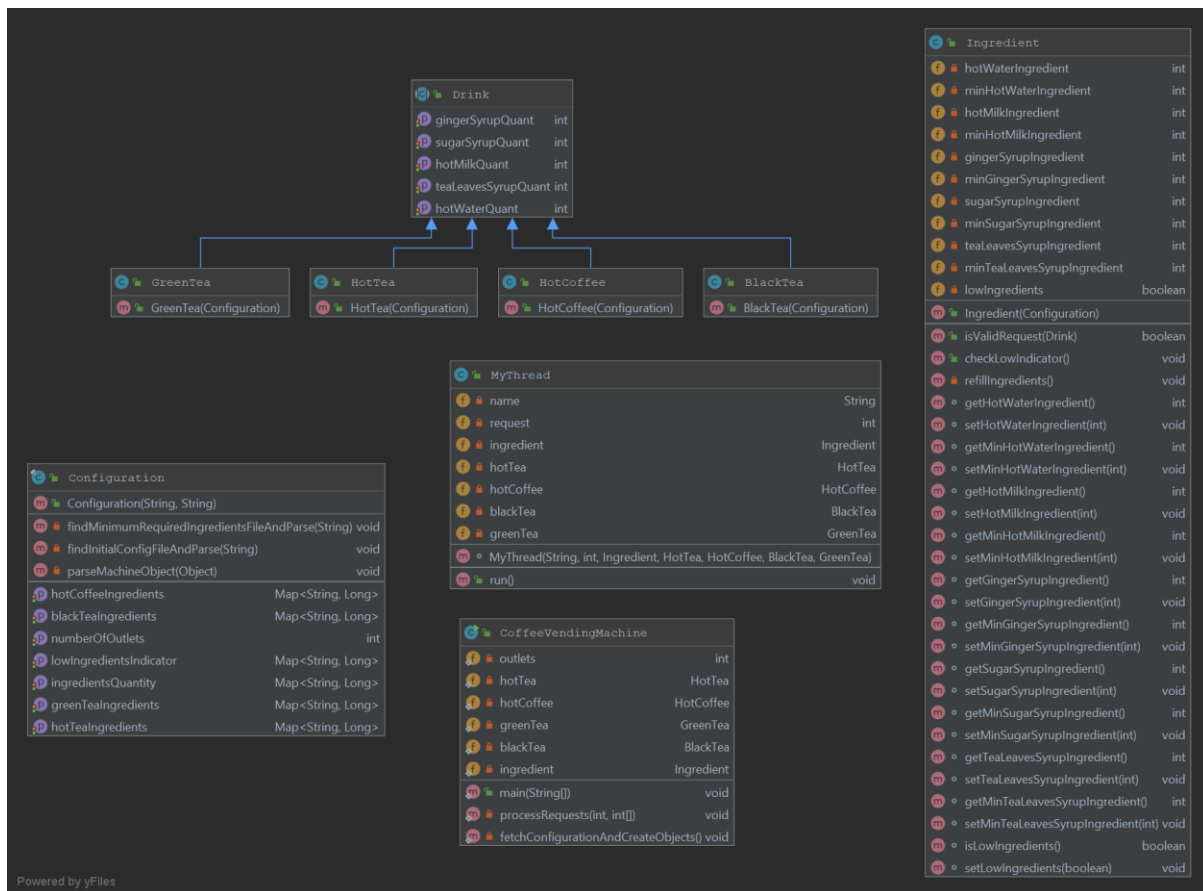
4. io.shubham.coffeemachine.VendingMachine

- a. **CoffeeVendingMachine.java** – Contains our main class which gets input from user. It contains logic to fetch initial configuration and set Drinks and Ingredients objects with the initial configuration. It contains logic to start multiple threads in case multiple requests are placed at a time.
- b. **MyThread.java** – It contains logic to prepare a particular type of drink that user has requested. Multiple threads can work on this part at the same time.

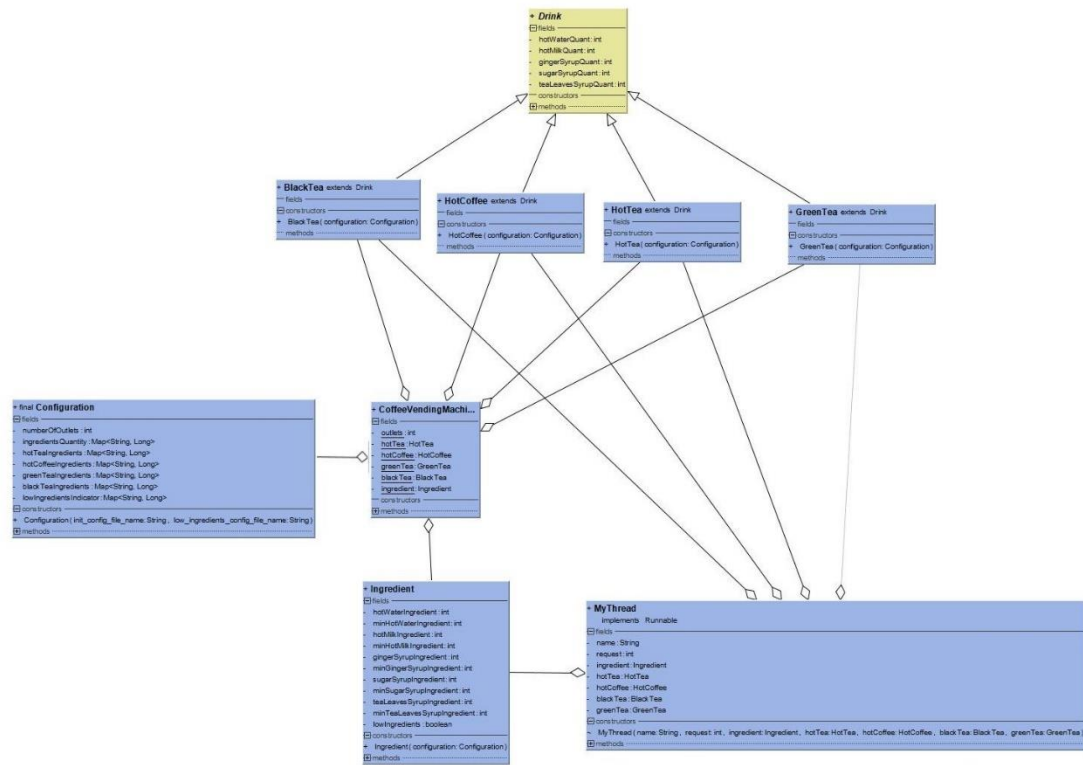
Configuration Files –

- 1. **Coffee-machine.json** – It has the initial configuration like number of outlets, total ingredients quantity, ingredients required for 4 types of drinks.
- 2. **Low-ingredients-indicator.json** – It has the configuration for the minimum quantity of ingredients to be present with our vending machine, otherwise it notifies about low ingredients and asks to refill.

Below is a diagram showing various classes, their member variables, constructors and member functions –



Class Diagram –



Outputs –

Below are some sample output snippets when given the same user input each time to check concurrency –

```
How many drinks do you want?
You can request for upto 3 drinks at a time

1
Enter 1 for Hot_Tea
Enter 2 for Hot_Coffee
Enter 3 for Black Tea
Enter 4 for Green Tea

To place order for multiple drinks, /please enter a space or line separated input
0 1 1
green_tea is prepared
black_tea is prepared
HotTea cannot be prepared because hot_water is not sufficient
All the current requests have been processed!!
Ingredients are low, do you want to refill now, press 1/0?
1
Displaying the present ingredients quantity
Hot Water = 100
Hot Milk = 500
Ginger Syrup = 40
Sugar Syrup = 0
Tea Leaves Syrup = 70
Please enter the quantity of ingredients you want to refill in the same order (0 for no refill)
50 40 40 40 40
Ingredients are not low
Enter 9 to exit
```

PFB some more snippets while rerunning the program and giving the same input –

```
How many drinks do you want?
You can request for upto 3 drinks at a time

1
Enter 1 for Hot_Tea
Enter 2 for Hot_Coffee
Enter 3 for Black Tea
Enter 4 for Green Tea

To place order for multiple drinks, /please enter a space or line separated input
1 1 1
green_tea is prepared
hot_tea is prepared
BlackTea cannot be prepared because hot_water is not sufficient
All the current requests have been processed!!
Ingredients are not low
Enter 9 to exit
1 to continue
|
```

Please note that the indicator for low ingredients has been configured using a json, a particular type of ingredient may be low, but some requests still might be fulfilled by the current ingredients as can be seen in the above snapshot.

```
"C:\Program Files\Java\jdk1.8.0_141\bin\java.exe" ...
How many drinks do you want?
You can request for upto 3 drinks at a time

3
Enter 1 for Hot_Tea
Enter 2 for Hot_Coffee
Enter 3 for Black Tea
Enter 4 for Green Tea

To place order for multiple drinks, /please enter a space or line separated input

4 1 3
hot_tea is prepared

black_tea is prepared

GreenTea cannot be prepared because hot_water is not sufficient

All the current requests have been processed!!

Ingredients are low, do you want to refill now, press 1/0?
|
```

```
"C:\Program Files\Java\jdk1.8.0_141\bin\java.exe" ...
How many drinks do you want?
You can request for upto 3 drinks at a time

3
Enter 1 for Hot_Tea
Enter 2 for Hot_Coffee
Enter 3 for Black Tea
Enter 4 for Green Tea

To place order for multiple drinks, /please enter a space or line separated input

4 1 3
green_tea is prepared

black_tea is prepared

HotTea cannot be prepared because hot_water is not sufficient

All the current requests have been processed!!

Ingredients are low, do you want to refill now, press 1/0?

5
Enter 9 to exit
1 to continue
|
```