RestaurantV2

Roel Janssen

March 7, 2017

1 RestaurantV1

The first version of the restaurant was a large improvement over the old, singlethreaded one. However, when multithreading occurs certain dangers become quite real. Therefore, the restaurantV1 will be checked for possible flaws and made threadsafe.

1.1 What Could Go Wrong

Three threads (waiter, server and cook) make use of the same two queues in the main thread. The waiter tries to add orders to the orderQueue, the cook tries to get them and add them to the readyQueue, and the server tries to get those.

This should be no problem, except for the fact that Orders have lists in them too. These lists can be added to and removed from too. If an order is being processed by the cook if the waiter is still adding meals to it, things will go wrong.

1.2 How to Fix That

If an order is being edited (invalid), it should be unavailable until it is valid again. This can be achieved with synchronization. This means it can be locked while invalid and unlocked while valid. This is done by adding the synchronized keyword. If this is used in a method, the instance will be locked until the method is done executing.

To be entirely sure the Order isn't edited between method calls, we will make the Queue synchronized too. This will make sure the orders are locked when they're invalid.