Overview

Simulation of a Pizza - Station

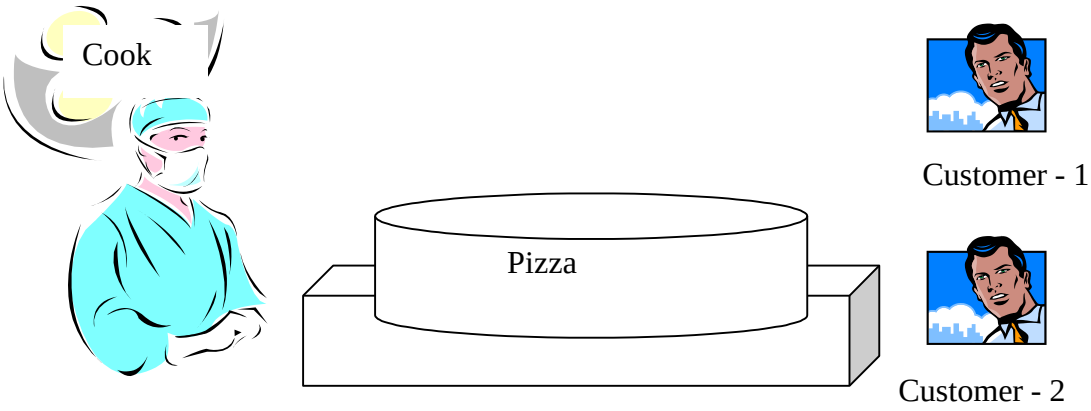
Station – Workflow

Pizza: has a name
Bar: has all pizzas on its plate; FIFO; with a capacity of ‚n' pizze
Cook:

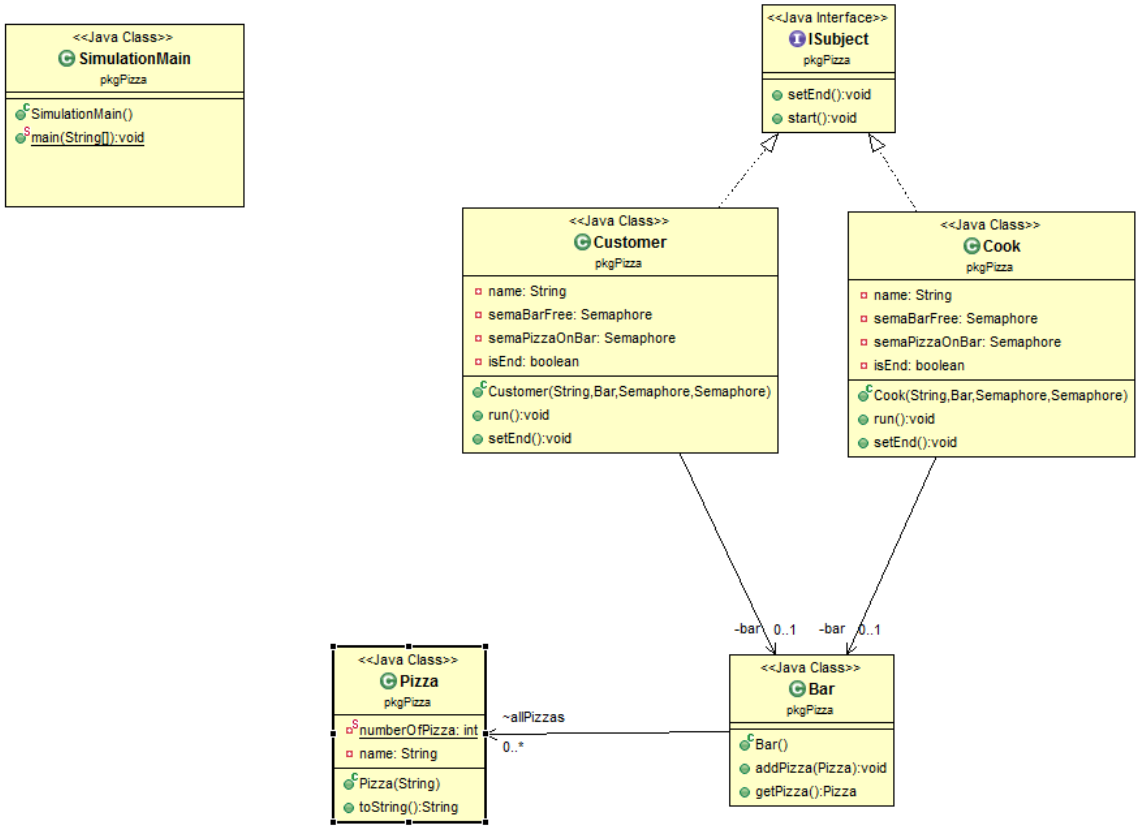
- has his name and access to the bar
- creates a pizza which lasts a random time
- waits, till bar has free space for pizza
- creates next pizza, whether there is an order of an customer or not

Customer

- has his name and access to the bar
- starts waiting for finished pizza
- takes pizza from bar
- starts eating pizza which lasts a random time
- as the pizza is very small he starts waiting for next finished pizza



Class Diagram



Protocol – Example

```
customer Bmeise: starts waiting for pizza
cook Adam: starts creating pizza 1
customer Ameise: starts waiting for pizza
cook Adam: finished pizza pizza 1
cook Adam: starts waiting for free bar
cook Adam: laying on bar pizza 1
cook Adam: starts creating pizza 2
customer Ameise: got pizza  pizza 1
customer Ameise: starts eating pizza pizza 1
customer Ameise: finished eating pizza 1
customer Ameise: starts waiting for pizza
cook Adam: finished pizza pizza 2
cook Adam: starts waiting for free bar
cook Adam: laying on bar pizza 2
cook Adam: starts creating pizza 3
customer Bmeise: got pizza  pizza 2
customer Bmeise: starts eating pizza pizza 2
cook Adam: finished pizza pizza 3
cook Adam: starts waiting for free bar
cook Adam: laying on bar pizza 3
cook Adam: starts creating pizza 4 *************** no customer order
customer Ameise: got pizza  pizza 3
customer Ameise: starts eating pizza pizza 3
cook Adam: finished pizza pizza 4
cook Adam: starts waiting for free bar
cook Adam: laying on bar pizza 4
cook Adam: starts creating pizza 5
cook Adam: finished pizza pizza 5 *************** also no ordering
cook Adam: starts waiting for free bar ********** so waiting because bar not empty
```

```
customer Bmeise: finished eating pizza 2
customer Bmeise: starts waiting for pizza
customer Bmeise: got pizza  pizza 4 ************ now customer gets pizza
customer Bmeise: starts eating pizza pizza 4
cook Adam: laying on bar pizza 5
cook Adam: starts creating pizza 6
cook Adam: finished pizza pizza 6
cook Adam: starts waiting for free bar
customer Bmeise: finished eating pizza 4
customer Bmeise: starts waiting for pizza
customer Bmeise: got pizza  pizza 5
customer Bmeise: starts eating pizza pizza 5
cook Adam: laying on bar pizza 6
cook Adam: starts creating pizza 7
customer Ameise: finished eating pizza 3
customer Ameise: starts waiting for pizza
customer Ameise: got pizza  pizza 6
customer Ameise: starts eating pizza pizza 6
cook Adam: finished pizza pizza 7
cook Adam: starts waiting for free bar
cook Adam: laying on bar pizza 7
cook Adam: starts creating pizza 8
customer Bmeise: finished eating pizza 5
customer Bmeise: starts waiting for pizza
customer Bmeise: got pizza  pizza 7
customer Bmeise: starts eating pizza pizza 7
cook Adam: finished pizza pizza 8
cook Adam: starts waiting for free bar
cook Adam: laying on bar pizza 8
cook Adam: starts creating pizza 9
cook Adam: finished pizza pizza 9
cook Adam: starts waiting for free bar
********* end of simulation **********************
cook Adam: laying on bar pizza 9
cook Adam finished
customer Ameise: finished eating pizza 6
customer Ameise finished
customer Bmeise: finished eating pizza 7
customer Bmeise finished
```

Hint for Developers

Implement the synchronization between threads (cook, customer) with the help of semaphores, which are created by “SimulationMain” and transferred to the threads.