capbay (1189545-D)

CapBay Technical Test

Fictional situation (as we're not in the mamak business). You can send us the code via a zip of the folder with instructions of how to run it in the README.

Include any additional safeguards/features that you think will be beneficial in the scalability/configurability/testability of the API. We recommend implementing the solution in Laravel PHP.

Our CapBay Mamak sells the following products:

Code	Name	Price
B001	Kopi	RM2.5
F001	Roti Kosong	RM1.5
B002	Teh Tarik	RM2

We understand that the mamak is a place of gathering and good food is best enjoyed together. Hence, drinks such as Kopi and Teh Tarik have a buy 1 free 1 package (on the same drink only, e.g. 2 Kopi then 1 will be free).

Also, our Roti Kosong is famous and we love it when people eat a lot of them. Hence, when people buy 2 or more Roti Kosong, the price should drop to RM1.2 each.

The system that we have in stores can scan items in any order and because our food/beverage of the month changes often, we need to be flexible regarding our pricing rules.

The interface to our checkout looks something like this (shown in PHP):

```
$checkout = new Checkout($pricingRules);
$checkout->scan($item);
$checkout->scan($item);
$price = $checkout->total;
```

Implement a checkout system that fulfils these requirements.

```
Test Data
List: B001,F001,B002,B001,F001
Total price expected: RM6.9

List: B002,B002,F001
Total price expected: RM3.5

List: B001,B001,B002
Total price expected: RM4.5
```