

```

1  public class Customer {
2      Waiter waiter;
3
4      public Customer(Waiter waiter){
5          this.waiter = waiter;
6      }
7      public void placeSoupOrder(){
8          System.out.println("want to eat (drink actually) soup");
9          SoupOrder so = new SoupOrder();
10         waiter.placeOrder(so); // order soup
11
12     }
13     public void placeKottuOrder(){
14         System.out.println("want to eat kottu");
15         KottuOrder ko = new KottuOrder();
16         waiter.placeOrder(ko); // order kottu
17     }
18 }
19
20 public class Waiter {
21     private Queue<Order> order_queue;
22     Cook cook;
23
24     public Waiter(Cook cook) {
25         order_queue = new LinkedList<>();
26         this.cook = cook;
27     }
28     void placeOrder(Order order) {
29         order.setCook(cook);
30         order_queue.add(order);
31         order_queue.remove().execute();
32     }
33
34 }
35
36 public class Cook {
37
38     public Cook() {
39         // TODO Auto-generated constructor stub
40     }
41     public void cookSoup(){
42         System.out.println("cooking soup \n");
43     }
44     public void cookKottu(){
45         System.out.println("cooking kottu \n");
46     }
47
48 }
49
50 public interface Order {
51     public void execute ( );
52     public void setCook(Cook cook);
53 }
54
55 public class KottuOrder implements Order{
56
57     private Cook cook;
58
59     public KottuOrder() {
60     }
61     public void setCook(Cook cook){
62         this.cook = cook;
63     }
64
65     public void execute(){
66         cook.cookKottu();
67     }
68
69 }
70
71 public class SoupOrder implements Order{
72     private Cook cook;
73
74     public SoupOrder() {
75
76     }
77

```

```
78     public void setCook(Cook cook) {
79         this.cook = cook;
80     }
81     public void execute() {
82         cook.cookSoup();
83     }
84
85 }
86
87 public class Main {
88
89     public static void main(String[] args) {
90         Cook cook = new Cook();
91         Waiter waiter = new Waiter(cook);
92         Customer customer = new Customer(waiter);
93         customer.placeSoupOrder();
94         customer.placeKottuOrder();
95     }
96
97 }
98
```