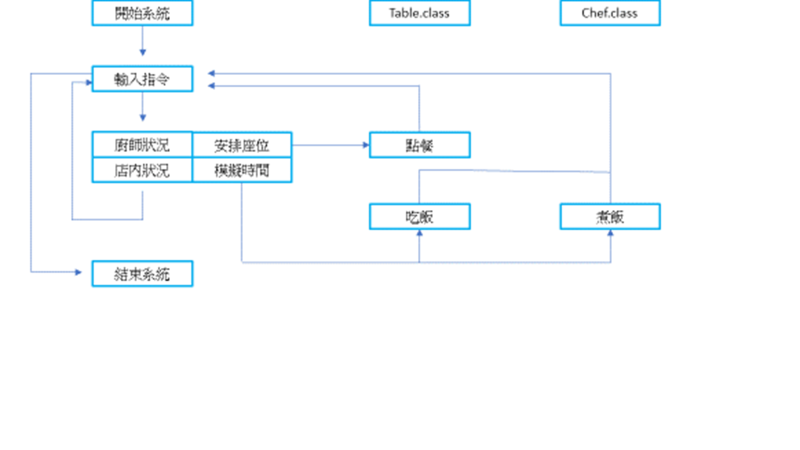
1. 需求描述:

使用者可能會在輸入編號時輸入非系統所需的字串,針對這點做了辨識錯誤的功能,可以在輸入錯誤時提醒使用者再次輸入。

1. 程式流程:



1. 功能/邏輯說明:

判斷座位是否有人:

判斷該座位是否還有餐點未上或是餐點已上桌尚未食用完

1. 確認2個廚師餐點做好的優先順序:

原先製作餐點時是廚師1做完才換廚師2,會造成廚師1的餐點無論如何都先上菜,因此設了一個變數eattime2為(餐點做好時還有剩下多少時間),eattime2較大的餐點為較早準備好的餐點,再將上菜順序重新做排序

1. 確認餐點準備好時還有多久可以吃:

比較eattime2與預計的開始食用時間,利用較小的那個當作實際開始食用時間

1. 判斷餐點是誰點的:

新增了一個class名稱為Food,內有一個arraylist名稱為who,在每次點餐時會將點餐者輸入至該arraylist中,就可以知道該送餐給誰

再來,食物new在2個不同的廚師中,所以2個廚師不會送餐給錯誤的人

1. 時間改變:

將時間設為變數time作為參數輸入進cook()與eat()這2個method中

1. 廚師煮飯:

令一個變數cooktime為(廚師正在準備餐點的剩餘準備時間),若每執行一次迴圈(也就是過了一分鐘),eattime與time都減1,直到eattime==1時,就將當時正在製作的餐點送上桌,並且迴圈執行完該次後eattime就會變成0,也就代表著餐點已經製作完成,然後在eattime==0時,將下一項該準備的餐點製作時間存入eattime中,並開始製作下一項餐點,直到time=0時結束迴圈

1. 吃東西:

與煮飯同概念

1. 使用說明:

進入系統後,根據系統提示輸入與欲執行的功能相對應的指令,

(5)其他

class中的變數除了arraylist與部分array外皆使用private

3個class的功能分別為

1. Table.class:

內有arraylist儲存該桌的人正在等待的餐點與正在食用的餐點,可以執行點餐與吃飯2個動作

2.Chef.class:

內有arraylist儲存尚未準備的餐點,可以跟廚師點餐,廚師也可以做飯跟送餐

3.Food.class:

內有一個arraylist可用來分辨該食物做好後該送至哪一桌