

## แบบฝึกหัดชุดที่ 6

### 1. จงเติม Class ให้สมบูรณ์ พร้อมแสดงการเรียกใช้งาน

```
class Car:
    def __init__(self, license, brand, color):
        # c = Car('AA1234', 'Honda', 'White')
        # มีตัวแปร report สำหรับเก็บข้อมูลประวัติการซ่อมบำรุง โดยกำหนดค่าเริ่มต้นเป็นลิสต์ว่าง

    def __str__(self):
        # คืนสตริงของรถยนต์ เช่น 'AA1234 - White Honda'

    def __lt__(self, rhs):
        # เรียงลำดับรถยนต์โดยเปรียบเทียบป้ายทะเบียนรถแบบสตริง

    def add_report(self, new_report):
        # เพิ่มประวัติการซ่อมบำรุง โดยไม่ต้องคืนค่า
        # ตัวแปร new_report เก็บ tuple (วันที่, คำอธิบาย, ราคา)
        # เช่น c.add_report( ('25 May 2017', 'change tires', 1500) )

    def total_payment(self):
        # คืนค่าใช้จ่ายทั้งหมดที่ใช้ในการซ่อมบำรุงที่ผ่านมา

    def max_payment(self):
        # คืนลิสต์ของประวัติการซ่อมบำรุง (วันที่, คำอธิบาย, ราคา) ทุกรายการ ที่มีค่าใช้จ่ายมากที่สุด
        # กรณีที่รถยนต์ไม่มีประวัติการซ่อมบำรุงเลย ให้คืนค่าลิสต์ว่าง
```

แบบฝึกหัดที่ 6 > 6.1.py > ...

```
1 class Car:
2
3     def __init__(self, license, brand, color):
4         self.license = license
5         self.brand = brand
6         self.color = color
7         self.report = []
8
9     def __str__(self):
10         return "{} - {}".format(self.license, self.color, self.brand)
11
12     def __lt__(self, rhs):
13         return self.license < rhs.license
14
15     def add_report(self, new_report):
16         self.report.append(new_report)
17
18     def total_payment(self):
19         pay = 0
20         for i in self.report:
21             pay += i[2]
22         return pay
23
24     def max_payment(self):
25         if len(self.report) <= 0:
26             return self.report
27         else:
28             max = 0
29             for payment in self.report:
30                 max = payment[2]
31                 max_pay = payment
32             return max_pay
```

```
34 c1 = Car('AA1234', 'Honda', 'While')
35 print(c1)
36 c1.add_report(('25 May 2017', 'change tires', 1500))
37 c1.add_report(('25 May 2017', 'change tires', 2000))
38 c1.add_report(('25 May 2017', 'change tires', 2500))
39 print(c1.total_payment())
40 print(c1.max_payment())
41 c2 = Car('AA1111', 'Honda', 'Black')
42 print(c2)
43 c2.add_report(('25 May 2017', 'change tires', 1500))
44 print(c2.total_payment())
45 print(c2.max_payment())
46 print(c1 < c2)
47
48
```

```
'AA1234 - WhileHonda'
6000
('25 May 2017', 'change tires', 2500)
'AA1111 - BlackHonda'
1500
('25 May 2017', 'change tires', 1500)
False
```

## 2. จงเติม Class ให้สมบูรณ์ พร้อมแสดงการเรียกใช้งาน

```
class ShoppingCart:
    def __init__(self, id):
        self.id = id
        self.books = []
        # books เก็บลิสต์ของหนังสือในตะกร้าพร้อมจำนวน เช่น [[b1,2],[b3,7]]

    def add_book(self, book, n):
        # เพิ่มข้อมูลการซื้อหนังสือ book เพิ่มอีก n เล่ม โดยไม่ต้องคืนค่า
        # หากไม่มีหนังสือเล่มนี้ในตะกร้า ให้เพิ่มลิสต์ [book, n] ต่อท้าย books
        # หากเคยมีข้อมูลหนังสือเล่มนี้ในตะกร้าแล้ว ให้เพิ่มจำนวนที่ซื้ออีก n เล่ม
        # เช่น ถ้า books = [[b1,2]] และเราสั่ง add_book(b1,3) จะได้ books = [[b1,5]]

    def delete_book(self, book):
        # ลบข้อมูลการซื้อหนังสือ book ออกจากตะกร้า โดยไม่ต้องคืนค่า
        # ถ้าในตะกร้าไม่มีหนังสือ book ไม่ต้องทำอะไร

    def get_total(self):
        # คืนค่าราคารวมของหนังสือทั้งหมดในตะกร้า

    def __lt__(self, rhs):
        # ตะกร้าที่มีราคารวมของหนังสือน้อยกว่า จะเป็นตะกร้าที่น้อยกว่า
```

```

1 class ShoppingCart:
2     def __init__(self, id):
3         self.id = id
4         self.books = []
5
6     def add_book(self, book, n, total):
7         for name_book in self.books:
8             if book == name_book[0]:
9                 n += name_book[1]
10                self.books.remove(name_book)
11            self.books.append([book, n, total])
12
13    def delete_book(self, book):
14        for name_book in self.books:
15            if book == name_book[0]:
16                self.books.remove(name_book)
17
18    def get_total(self):
19        total = 0
20        for cost in self.books:
21            total += int(cost[2]) * int(cost[1])
22        return total
23
24    def __lt__(self, rhs):
25        return self.get_total() < rhs.get_total()
26

```

```

27 book1 = ShoppingCart(4567)
28 book2 = ShoppingCart(88)
29 book1.add_book('b1', 1, 15)
30 book1.add_book('b2', 3, 20)
31 book1.add_book('b3', 5, 25)
32 book1.add_book('b4', 7, 30)
33 print(book1>book2)
34 book1.delete_book('b1')
35 print(book1.books)
36 print(f'รวมรวม = {book1.get_total()} บาท')
37

```

True

[['b2', 3, 20], ['b3', 5, 25], ['b4', 7, 30]]  
รวมรวม = 395 บาท