

## ปัญหา แกร็บแพนด้า 2 [GrabPanda2] (5 คะแนน)

หลังจากที่ทางระบบแอปพลิเคชันสามารถจัดการเรื่องการคำนวณค่าใช้จ่ายของผู้ส่งอาหารแล้ว เราจะต้องคำนวณส่วนแบ่งรายได้ออกเป็นสามส่วนคือผู้ขายอาหาร ไรเดอร์ และเจ้าของระบบแอปพลิเคชัน **GrabPanda** ซึ่งมีข้อกำหนดเรื่องการคำนวณดังนี้

1. ถ้าค่าอาหารรวมแล้วไม่เกิน 200 บาท ไรเดอร์จะได้ 20 บาท ส่วนเจ้าของระบบจะได้ 30% ของค่าอาหาร<sup>1</sup> (เศษสตางค์ปัดทิ้ง) ซึ่งสามารถคำนวณได้จาก (ราคาอาหารรวม  $\times$  3) / 10 ส่วนผู้ขายอาหารจะได้รายได้ส่วนที่เหลือจากที่ไรเดอร์กับเจ้าของระบบได้ไป
2. ถ้าค่าอาหารรวมแล้วมากกว่า 200 บาท ไรเดอร์จะได้เพิ่มเป็น 30 บาท ส่วนเจ้าของระบบจะได้ส่วนแบ่ง 30% ของค่าอาหารในลักษณะเดิม และผู้ขายอาหารจะได้รายได้ส่วนที่เหลือจากการหักให้ไรเดอร์กับเจ้าของระบบ

เราจะเปลี่ยนแปลงคลาส **FoodOrder** ตามข้อกำหนดต่อไปนี้

1. คลาสจะมีฟิลด์ข้อมูลสมาชิกคลาสชื่อ **grandTotal** เป็นจำนวนเต็มซึ่งแทนค่าใช้จ่ายทั้งหมดที่ผู้ส่งอาหารต้องจ่าย ซึ่งค่านี้จะถูกบันทึกด้วยเมธอด **finalizeReceipt** (เป็นไปได้ว่าผู้เรียนอาจจะทำแบบนี้มาตั้งแต่ข้อที่แล้ว ซึ่งไม่ผิด เพียงแต่ในข้อนี้ จะบังคับให้ทำในลักษณะดังกล่าว)
2. มีเมธอดชื่อ **distribIncome** ซึ่งไม่รับพารามิเตอร์และไม่คืนค่าใด เมธอดนี้จะนำค่าจาก **grandTotal** และ **foodTotal** มาประมวลผลเพื่อรายงานว่าแต่ละฝ่ายจะได้ส่วนแบ่งรายได้เท่าใดบ้าง โดยรายงานในรูปแบบ

**Rider:** รายได้ที่ไรเดอร์จะได้รับ

**GrabPanda:** รายได้ที่เจ้าของระบบ **GrabPanda** จะได้รับ

**Food Seller:** รายได้ที่ผู้ขายอาหารจะได้รับ

---

<sup>1</sup> เกร็ดความรู้ (ไม่เกี่ยวกับการสอบ): ส่วนแบ่ง 30% นี้ (เรียกกันว่าค่า GP) ถือว่ามากเกินไปในหลายประเทศ เช่น อเมริกาและยุโรป โดยประเทศจำนวนมากจะออกกฎหมายกำหนดเพดานส่วนแบ่งไว้ เช่น กำหนดไว้ไม่ให้เกิน 16% ของค่าอาหาร เพื่อไม่ให้ผู้ขายอาหารและผู้บริโภคเดือดร้อน ยกตัวอย่าง เช่น หากผู้ขายอาหารต้องการรายได้เพิ่มขึ้น 10 บาท แต่ค่า GP คือ 30% แบบไทย ๆ ในลักษณะนี้ (ปีพ.ศ. 2565) ผู้ขายจะต้องเพิ่มราคา 14 บาท 30 สตางค์ ทำให้ผู้บริโภคจะต้องรับภาระมากเกินไป

จึงเขียนโปรแกรมที่รับข้อมูลเข้าเป็นชื่อผู้สั่งอาหารและรายการค่าอาหารที่สั่ง เพื่อสร้างและเรียกใช้เมธอดในคลาส **FoodOrder** จากนั้นให้พิมพ์ผลลัพธ์ออกมาเป็นใบเสร็จผ่านทางเมธอด **finalizeReceipt** และพิมพ์ผลลัพธ์เกี่ยวกับส่วนแบ่งรายได้ตามมาด้วยเมธอด **distribIncome**

### รูปแบบข้อมูลเข้า

[เหมือนกับข้อที่แล้ว]

หมายเหตุ เช่นเดิม ห้ามเมธอด **main** ทำการบวกราคา ทำการคำนวณส่วนแบ่งรายได้ หรือพิมพ์ผลลัพธ์ออกมาเอง ถ้าหาก **main** มีการทำเช่นนั้น คะแนนจะลดลง นอกจากนี้ เมธอด **main** จะต้องเป็นผู้เรียกเมธอด **distribIncome** (เมธอดในคลาส **FoodOrder** จะไม่ใช่ฝ่ายที่เรียกใช้ **distribIncome**)

### รูปแบบผลลัพธ์

เป็นผลลัพธ์ที่พิมพ์จากเมธอด **finalizeReceipt** และตามด้วยผลลัพธ์ที่พิมพ์จาก **distribIncome** ในรูปแบบที่ระบุไว้ข้างบน ส่วนรายละเอียดขอให้ดูจากตัวอย่างผลลัพธ์ทางด้านล่าง

### ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์	ข้อมูลเข้า	ผลลัพธ์
Messi 2 19 30	Messi 49 20 69 Rider: 20 GrabPanda: 14 Food Seller: 35	Mbappe 2 47 70	Mbappe 117 10 127 Rider: 20 GrabPanda: 35 Food Seller: 72
Ziyech 5 50 30 10 40 20	Ziyech 150 10 160 Rider: 20 GrabPanda: 45 Food Seller: 95	Fernandes 5 50 60 55 40 20	Fernandes 225 0 225 Rider: 30 GrabPanda: 67 Food Seller: 128

Modric	Modric		Maeda	Maeda
3	240		1	241
70	0		241	0
80	240			241
90	Rider: 30			Rider: 30
	GrabPanda: 72			GrabPanda: 72
	Food Seller: 138			Food Seller: 139

### เกณฑ์การให้คะแนน

โปรแกรมจะต้องทำงานถูกต้องอย่างน้อย **50%** ของชุดทดสอบ จึงจะได้คะแนน

**หมายเหตุ** ในระบบตรวจแบบออนไลน์อาจจะมีข้อจำกัดว่า **1.** ห้ามสร้างคลาสไว้ในแพ็คเกจ **2.** ชื่อคลาสจะต้องตรงกับชื่อโจทย์ที่อยู่ในระบบตรวจ และ **3.** ห้ามมีอักขระภาษาไทย (แม้แต่ในคอมเมนต์ของโค้ดก็ห้ามมี) ถ้าใส่มาจะถูกพิจารณาว่า **compile error**

### โครงสร้างโค้ดสำหรับการทำข้อสอบ

แนะนำให้เติมเต็มคลาส **FoodOrder** ตามข้อกำหนด ส่วนในเมธอด **main** ให้เน้นแก้โค้ดตรงที่เป็น **???**

```
import java.util.Scanner;
class FoodOrder {
    // Enter your code here
}

public class GrabPanda2 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        String name = scan.next();
        int N = scan.nextInt();

        ??? order = ???;
        for(int i = 0; i < N; ++i) {
            int price = scan.nextInt();
            ???
        }
        ???
        ???
    }
}
```