

ปัญหา ที่เก็บผลิตภัณฑ์ 2 [ProductStore2] (6 คะแนน)

[ทรัพยากรการคำนวณ: 1 วินาที, 32 MB]

ในตอนนี้ ร้านขายของออนไลน์มีคลาสของผลิตภัณฑ์แบบพื้นฐานแล้ว เราจะลองสร้างคลาสเพิ่มเติมอีกสามคลาสที่เจาะจงไปกลุ่มสินค้าพิเศษอีกสองกลุ่มคือ รองเท้า (Shoe) และ หนังสือ (Book) ซึ่งทั้งสองกลุ่มนี้จะรับทอดคลาส **Product** ที่เราสร้างไว้ในข้อที่แล้วแต่มีข้อกำหนดเพิ่มเติมขึ้นมา ซึ่งในข้อนี้จะจัดการเฉพาะคลาส **Shoe** ก่อนตามรายละเอียดดังนี้

ตัวสร้างของคลาส **Shoe** จะรับพารามิเตอร์เพิ่มเติมจากตัวสร้างของ **Product** อีกสื่ออย่างคือ ก. รองเท้าสำหรับเพศใด ถ้าเป็นเลข 1 หมายถึงรองเท้าผู้หญิง (female) เลข 2 หมายถึงรองเท้าผู้ชาย (male), ข. ขนาดรองเท้า เป็นจำนวนเต็ม, ค. ชนิดรองเท้า ซึ่งเลข 1 หมายถึงรองเท้ากีฬา (sport), 2 หมายถึงรองเท้าทำงาน (formal) และ 3 หมายถึงรองเท้าแตะ (casual) และ ง. ยี่ห้อ เป็นสตริง ซึ่งตัวสร้างจะต้องเก็บข้อมูลที่เพิ่มเติมสื่ออย่างนี้ลงไปในตัวแปรสมาชิกคลาส

ในคลาสนี้เมธอด **printInfo** จะ override เมธอด **printInfo** อันเดิมที่มีอยู่ในคลาส **Product** และจะเริ่มต้นการทำงานด้วยการเรียก **printInfo** ของคลาส **Product** และพิมพ์ข้อมูลเพิ่มเติมมาในรูปแบบข้างล่างนี้

รองเท้าสำหรับเพศใด, ขนาด, ชนิด, ยี่ห้อ

เช่น

male, 12, sport, Niko

female, 8, formal, Munchen

จงสร้างคลาส **Shoe** เสริมขึ้นมาจากคลาส **Product** เดิมที่มีอยู่ ทั้งนี้ข้อมูลเข้า และการตรวจสอบต่าง ๆ ตลอดจนการเรียกใช้ **printInfo** จะเป็นไปในลักษณะเดียวกันกับข้อที่แล้ว ต่างกันเพียงข้อมูลรองเท้าบางอย่างที่เพิ่มขึ้นมา

รูปแบบข้อมูลเข้า

บรรทัดแรก	เป็นค่า N ซึ่งแทนจำนวนผลิตภัณฑ์ที่ต้องการเก็บ โดยที่ N <= 10,000
บรรทัดที่สอง	ชื่อของผลิตภัณฑ์ตัวแรก เป็นสตริงยาวไม่เกิน 120 อักขรและ อาจจะมีช่องว่างอยู่ภายใน (ในกรณีที่ เป็นข้อมูลชื่อที่ผิด จะเป็น บรรทัดเปล่า)
บรรทัดที่สาม	ค่า sku ของผลิตภัณฑ์ตัวแรก ไม่มีช่องว่างอยู่ภายใน (ในกรณีที่ เป็น ข้อมูลชื่อที่ผิด จะเป็นบรรทัดเปล่า)
บรรทัดที่สี่	ราคาของผลิตภัณฑ์ตัวแรก เป็นจำนวนจริง
บรรทัดที่ห้า	น้ำหนักของผลิตภัณฑ์ตัวแรก เป็นจำนวนเต็ม
บรรทัดที่หก	เป็นข้อมูลที่เพิ่มเติมขึ้นมาของคลาส Shoe แต่ละส่วนคั่นด้วย ช่องว่างหนึ่งช่องและรับประกันว่าชื่อยี่ห้อจะไม่มีช่องว่าง และข้อมูล ทั้ง 4 ส่วนในนี้จะไม่มีความผิดพลาดแต่อย่างใด
บรรทัดที่เหลือ	เป็นข้อมูลของผลิตภัณฑ์ตัวต่อ ๆ มา รูปแบบอยู่ในลักษณะเดียวกับ ผลิตภัณฑ์ตัวแรก

รูปแบบผลลัพธ์

กลุ่มแรก	เป็นผลจากการเรียกใช้เมธอด checkValidity ผ่านตัวแปร อ้างอิงวัตถุ Product ที่เก็บไว้ในอาร์เรย์หรือ ArrayList
กลุ่มที่สอง	เป็นผลจากการเรียกใช้เมธอด printInfo ผ่านตัวแปรอ้างอิง วัตถุ Product ที่เก็บไว้ในอาร์เรย์หรือ ArrayList

ตัวอย่าง

ข้อมูลเข้า	ผลลัพธ์
3 Toon BSlam AERAS 720 0 2 12 1 Niko Working 1SFJURF 0 -100 1 8 2 Munchen RongTuaTae 65.75 100 1 6 3 Satellite	true weight is invalid false sku is invalid false Toon BSlam, AERAS, 720.0, 0 male, 12, sport, Niko Working, 1SFJURF, 0.0, -100, Invalid female, 8, formal, Munchen RongTuaTae, , 65.75, 100, Invalid female, 6, casual, Satellite

โครงสร้างโค้ดและเกณฑ์การให้คะแนน

โปรแกรมจะต้องทำงานถูกต้องอย่างน้อย 20% ของชุดทดสอบ จึงจะได้คะแนน และห้ามเปลี่ยนโค้ดในส่วนที่ไม่ได้บอกให้เปลี่ยนได้ ถ้าใครเปลี่ยนแล้วได้คะแนน จะโดนเปลี่ยนเป็นติดลบเพราะถือเป็นการจงใจเลี่ยงการตอบในประเด็นที่โจทย์ต้องการทดสอบผู้เรียนและทำให้ได้คะแนนอย่างผิด ๆ (ให้แก้ไขได้เฉพาะตรงที่มีคำว่า Your Code)

```
import java.util.Scanner;
class Product {
    // Enter Your Code Here (from previous problem).
}

class Shoe /* Your Code */ {
    // Enter Your Code Here
}

public class ProductStore2 {
    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        int N = scan.nextInt();
        /* Your Code */
        for(int i = 0; i < N; ++i) {
            scan.nextLine(); // throw away new-line char
            String name = scan.nextLine();
            String sku = scan.nextLine();
            double prices = scan.nextDouble();
            int weight = scan.nextInt();
            int gender = scan.nextInt();
            int size = scan.nextInt();
            int type = scan.nextInt();
            String brand = scan.next();
            /* Your Code */
        }

        // Call checkVaidity of each product object.
        for(int i = 0; i < N; ++i) {
            /* Your Code */
            System.out.println(valid);
        }
    }
}
```

```
// Call printInfo of each product object.  
for(int i = 0; i < N; ++i) {  
    /* Your Code */  
}  
}  
}
```