*/\*  
กำหนดจำนวนเต็ม N ดำเนินการตามเงื่อนไขต่อไปนี้:  
- ถ้า N เป็นเลขคี่ให้พิมพ์ Weird  
- ถ้า N เป็นเลขคู่และอยู่ในช่วง 2 ถึง 5 ให้พิมพ์ Not Weird  
- ถ้า N เป็นเลขคู่และอยู่ในช่วง 6 ถึง 20 ให้พิมพ์ Weird  
- ถ้า N เป็นเลขคู่และมากกว่า 20 ให้พิมพ์ Not Weird  
 \*/  
  
// ตามที่โจทย์กำหนด เราสามารถเขียนโปรแกรมเพื่อตรวจสอบเงื่อนไขดังกล่าวใน Java 8 ได้ดังนี้:*import java.io.\*; *// นำเข้าคลาสและอินเทอร์เฟซที่เกี่ยวข้องกับการทำงานเกี่ยวกับไฟล์และการสื่อสารกับระบบ Input/Output (I/O)*import java.math.\*; *// นำเข้าคลาสและอินเทอร์เฟซที่เกี่ยวข้องกับการทำงานทางคณิตศาสตร์แบบกว้างขึ้น เช่น การทำงานกับจำนวนเต็มที่มีความยาวเท่านั้น*import java.security.\*; *// นำเข้าคลาสและอินเทอร์เฟซที่เกี่ยวข้องกับความปลอดภัยและการจัดการกับระบบรักษาความปลอดภัยใน Java*import java.text.\*; *// นำเข้าคลาสและอินเทอร์เฟซที่เกี่ยวข้องกับการจัดรูปแบบและการปรับแต่งข้อความเช่นการจัดรูปแบบวันที่และเวลา*import java.util.\*; *// นำเข้าคลาสและอินเทอร์เฟซที่เกี่ยวข้องกับโครงสร้างข้อมูลที่มีอยู่และการดำเนินการกับข้อมูล เช่น รายการ (List), ชุดข้อมูล (Set), แผนภูมิแฮชแมป (HashMap) เป็นต้น*import java.util.concurrent.\*; *// นำเข้าคลาสและอินเทอร์เฟซที่เกี่ยวข้องกับการทำงานแบบพร้อมกัน (concurrent programming) เช่น สร้างเธรด (Thread), ส่วนของการจัดการกับการทำงานพร้อมกัน (ConcurrentHashMap) เป็นต้น*import java.util.regex.\*; *// นำเข้าคลาสและอินเทอร์เฟซที่เกี่ยวข้องกับการทำงานกับ Regular Expression (regex) เพื่อค้นหาและจับคู่รูปแบบข้อความ*public class JavaIf\_Else {  
 private static final Scanner *scanner* = new Scanner(System.*in*);  
  
 public static void main(String[] args) {  
 System.*out*.print("ป้อนตัวเลข : ");  
 int N = *scanner*.nextInt(); *// ใช้ Scanner เพื่อรับค่าตัวเลขจากผู้ใช้ผ่านทางคีย์บอร์ด  
 //scanner.skip("(\r\n|[\n\r\u2028\u2029\u0085])?");* if (N % 2 != 0) { *// ถ้าตัวเลข N หารด้วย 2 ไม่ลงตัว (เป็นเลขคี่) เราพิมพ์ "Weird" ออกทางหน้าจอ* System.*out*.println("Weird");  
 } else {  
 if (N >= 2 && N <= 5) { *// ถ้าตัวเลข N เป็นเลขคู่และมีค่าระหว่าง 2 ถึง 5 เราพิมพ์ "Not Weird" ออกทางหน้าจอ* System.*out*.println("Not Weird");  
 } else if (N >= 6 && N <= 20) { *// ถ้าตัวเลข N เป็นเลขคู่และมีค่าระหว่าง 6 ถึง 20 เราพิมพ์ "Weird" ออกทางหน้าจอ* System.*out*.println("Weird");  
 } else { *// ถ้าตัวเลข N เป็นเลขคู่และมากกว่า 20 เราพิมพ์ "Not Weird" ออกทางหน้าจอ* System.*out*.println("Not Weird");  
 }  
  
 *scanner*.close();  
 }  
 }  
}  
  
*/\*  
ในโปรแกรมนี้ เราใช้ Scanner เพื่อรับค่าจำนวนเต็ม n จากผู้ใช้ แล้วตรวจสอบเงื่อนไขตามที่โจทย์กำหนดโดยใช้คำสั่ง if-else  
ในการตรวจสอบ หาก n เป็นเลขคี่ เราจะพิมพ์ "Weird" ออกทางหน้าจอ ถ้า n เป็นเลขคู่ เราจะตรวจสอบเงื่อนไขเพิ่มเติมว่า n  
อยู่ในช่วงใด และพิมพ์ผลลัพธ์ตามนั้นออกทางหน้าจอ ตามเงื่อนไขที่กำหนดในโจทย์  
 \*/*

*/\*  
ในความท้าทายนี้ คุณต้องอ่านจำนวนเต็ม จำนวนคู่ และสตริงจาก stdin จากนั้นพิมพ์ค่าตามคำแนะนำในส่วนรูปแบบผลลัพธ์ด้านล่าง  
เพื่อทำให้ปัญหาง่ายขึ้นเล็กน้อย โค้ดส่วนหนึ่งมีให้ในตัวแก้ไข  
หมายเหตุ: เราขอแนะนำให้กรอก Java Stdin และ Stdout I ก่อนดำเนินการนี้  
รูปแบบการป้อนข้อมูล  
อินพุตมีสามบรรทัด:  
1. บรรทัดแรกประกอบด้วยจำนวนเต็ม  
2. บรรทัดที่สองประกอบด้วยคู่  
3. บรรทัดที่สามประกอบด้วยสตริง  
รูปแบบเอาต์พุต  
เอาต์พุตมีสามบรรทัด:  
1. ในบรรทัดแรก พิมพ์ String: ตามด้วย String ที่ไม่เปลี่ยนแปลงที่อ่านจาก stdin  
2. ในบรรทัดที่สอง พิมพ์ Double: ตามด้วยการอ่านซ้ำสองครั้งที่ไม่เปลี่ยนแปลงจาก stdin  
3. ในบรรทัดที่สาม พิมพ์ Int: ตามด้วยจำนวนเต็มที่ไม่เปลี่ยนแปลงที่อ่านจาก stdin เพื่อทำให้ปัญหาง่ายขึ้น โค้ดส่วนหนึ่งมีอยู่ในตัวแก้ไขแล้ว  
หมายเหตุ: หากคุณใช้เมธอด nextLine() ต่อจากเมธอด nextInt() ในทันที โปรดจำไว้ว่า nextint() อ่านโทเค็นจำนวนเต็ม ด้วยเหตุนี้  
อักขระขึ้นบรรทัดใหม่สุดท้ายสำหรับบรรทัดของอินพุตจำนวนเต็มนั้นยังคงอยู่ในคิวในบัฟเฟอร์อินพุตและ ถัดไป nextLine() จะอ่านส่วนที่เหลือของบรรทัดจำนวนเต็ม (ซึ่งว่างเปล่า)  
 \*/  
  
/\*  
Input ที่พิมพ์เข้ามา  
42  
3.1415  
Welcome to HackerRank's Java tutorials!  
  
Output ที่แสดงผล  
String: Welcome to HackerRank's Java tutorials!  
Double: 3.1415  
Int: 42  
\*/*import java.util.Scanner;  
public class JavaStdinAndStdoutII {  
 public static void main(String[] args) {  
 Scanner scan = new Scanner(System.*in*);  
 int i = scan.nextInt();  
 Double d = scan.nextDouble();  
 scan.nextLine(); *// อ่านเอาชนิดข้อมูลที่เหลือหลังจาก nextDouble() ซึ่งเป็นเอนเทอร์รัดบรรทัดใหม่* String s = scan.nextLine();  
  
 *// Write your code here.* System.*out*.println("String: " + s);  
 System.*out*.println("Double: " + d);  
 System.*out*.println("Int: " + i);  
 }  
}

*/\*  
สามารถใช้ฟังก์ชัน System.out.printf ของ Java เพื่อพิมพ์เอาต์พุตที่จัดรูปแบบได้ จุดประสงค์ของแบบฝึกหัดนี้คือเพื่อทดสอบความเข้าใจของคุณเกี่ยวกับการจัดรูปแบบเอาต์พุตโดยใช้ printf  
เพื่อให้คุณเริ่มต้นได้ ส่วนหนึ่งของโซลูชันมีให้ในตัวแก้ไข คุณต้องจัดรูปแบบและพิมพ์อินพุตเพื่อทำโซลูชันให้เสร็จสมบูรณ์  
รูปแบบการป้อนข้อมูล  
ทุกบรรทัดที่ป้อนจะมีสตริงตามด้วยจำนวนเต็ม  
แต่ละสตริงจะมีอักขระที่เป็นตัวอักษรได้สูงสุด 10 และแต่ละจำนวนเต็มจะอยู่ในช่วงรวมตั้งแต่่ 0 ถึง 999  
รูปแบบเอาต์พุต  
ในแต่ละบรรทัดของเอาต์พุตควรมีสองคอลัมน์:  
คอลัมน์แรกมีสตริงและถูกเว้นไว้โดยใช้อักขระทั้งหมด  
คอลัมน์ที่สองมีจำนวนเต็มซึ่งแสดงเป็นตัวเลขทั้งหมด หากอินพุตดั้งเดิมมีตัวเลขน้อยกว่าสามหลัก คุณต้องรองเลขศูนย์นำหน้าเอาต์พุตด้วยศูนย์  
อินพุตตัวอย่าง  
java 100  
cpp 65  
python 50  
เอาต์พุตตัวอย่าง  
================================  
java 100  
cpp 065  
python 050  
================================  
 \*/*import java.util.Scanner;  
public class JavaOutputFormatting {  
 public static void main(String[] args) {  
 Scanner sc=new Scanner(System.*in*);  
 System.*out*.println("================================");  
 for(int i=0;i<3;i++)  
 {  
 String s1=sc.next();  
 int x=sc.nextInt();  
 *//Complete this line* System.*out*.printf("%-15s%03d%n", s1, x);  
 }  
 System.*out*.println("================================");  
  
 }  
}  
  
*/\*  
โค้ดด้านบนเป็นโปรแกรมที่ใช้ในการรับข้อมูลจากผู้ใช้และพิมพ์ผลลัพธ์ออกทางหน้าจอ โดยมีการใช้ลูป for เพื่อรับค่าสตริงและจำนวนเต็มจากผู้ใช้  
ซึ่งจำนวนรอบในการรับข้อมูลจะเป็น 3 รอบตามที่กำหนดในลูป for  
ในแต่ละรอบของลูป for เราใช้ Scanner เพื่อรับค่าสตริงจากผู้ใช้ผ่าน sc.next() และรับค่าจำนวนเต็มผ่าน sc.nextInt() จากนั้นเราใช้ System.out.printf()  
เพื่อพิมพ์ผลลัพธ์ออกทางหน้าจอ โดยใช้รูปแบบการจัดรูปแบบ (%) เพื่อกำหนดความกว้างและรูปแบบของข้อมูลที่จะแสดงผล โดยในที่นี้จะใช้ %-15s  
เพื่อกำหนดให้สตริงมีความกว้าง 15 ตัวอักษรและจัดชิดซ้าย และ %03d เพื่อกำหนดให้จำนวนเต็มมีความกว้าง 3 หลักและเติมเลขศูนย์ข้างหน้ากรณีที่มีน้อยกว่า 3 หลัก  
โดยสรุปแล้ว โปรแกรมนี้จะรับข้อมูลสตริงและจำนวนเต็มจากผู้ใช้ และพิมพ์ผลลัพธ์ออกทางหน้าจอในรูปแบบตามที่โจทย์กำหนด  
System.out.printf() เป็นเมธอดในคลาส PrintStream ที่ใช้ในการพิมพ์ผลลัพธ์ออกทางหน้าจอโดยกำหนดรูปแบบการจัดรูปแบบ (format)  
ของข้อมูลที่จะแสดงผล รูปแบบการจัดรูปแบบถูกกำหนดด้วยพารามิเตอร์ที่ตามหลังสตริงข้อความที่ต้องการพิมพ์ โดยใช้ตัวอักษรพิเศษ (%) เพื่อระบุประเภทข้อมูลและรูปแบบที่ต้องการ  
ในกรณีนี้ "%-15s%03d%n" คือรูปแบบการจัดรูปแบบที่กำหนดให้พิมพ์สตริงและจำนวนเต็มออกทางหน้าจอ โดยแต่ละส่วนของรูปแบบจะมีความหมายดังนี้:  
%-15s: จัดชิดซ้าย (-) และกำหนดความกว้างของสตริงที่จะแสดงผลให้มีความกว้าง 15 ตัวอักษร  
%03d: กำหนดความกว้างของจำนวนเต็มที่จะแสดงผลให้มีความกว้าง 3 หลักและเติมเลขศูนย์ข้างหน้ากรณีที่มีน้อยกว่า 3 หลัก  
%n: ใช้ในการขึ้นบรรทัดใหม่  
ดังนั้น คำสั่ง System.out.printf("%-15s%03d%n", s1, x); จะทำการพิมพ์สตริง s1 โดยจัดชิดซ้ายและกำหนดความกว้าง 15 ตัวอักษร ตามด้วยจำนวนเต็ม x  
ที่เติมเลขศูนย์ข้างหน้าและกำหนดความกว้าง 3 หลัก และสุดท้ายพิมพ์เครื่องหมายขึ้นบรรทัดใหม่ (%n)  
 \*/*

*/\*  
วัตถุประสงค์  
ในการท้าทายนี้ เราจะใช้การวนซ้ำเพื่อช่วยเราคำนวณเลขง่ายๆ  
กำหนดจำนวนเต็ม N พิมพ์ 10 ทวีคูณแรก. แต่ละรายการ N\*i (โดยที่ 1<= i <= 10 ) ควรพิมพ์บนบรรทัดใหม่ในรูปแบบ: N x i = ผลลัพธ์  
รูปแบบการป้อนข้อมูล  
จำนวนเต็มเดียว N.  
ข้อ จำกัด  
2 <= N <= 20  
  
รูปแบบเอาต์พุต  
พิมพ์เอาต์พุต 10 บรรทัด; แต่ละบรรทัด i (โดยที่ 1 <= i <= 10) มีผลลัพธ์ของ N\*i ในรูปแบบ:  
N x i = ผลลัพธ์  
อินพุตตัวอย่าง  
2  
เอาต์พุตตัวอย่าง  
2 x 1 = 2  
2 x 2 = 4  
2 x 3 = 6  
2 x 4 = 8  
2 x 5 = 10  
2 x 6 = 12  
2 x 7 = 14  
2 x 8 = 16  
2 x 9 = 18  
2 x 10 = 20  
 \*/*import java.io.\*;  
import java.math.\*;  
import java.security.\*;  
import java.text.\*;  
import java.util.\*;  
import java.util.concurrent.\*;  
import java.util.function.\*;  
import java.util.regex.\*;  
import java.util.stream.\*;  
import static java.util.stream.Collectors.*joining*;  
import static java.util.stream.Collectors.*toList*;  
public class J4\_JavaLoopsI {  
 public static void main(String[] args) throws IOException {  
 BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.*in*));  
 int N = Integer.*parseInt*(bufferedReader.readLine().trim());  
 bufferedReader.close();  
 for (int i = 1; i <= 10; i++){  
 int result = N \* i ;  
 System.*out*.printf("%d x %d = %d%n", N, i, result);  
 }  
 }  
}  
*/\*  
ในโค้ดด้านบน เราใช้ BufferedReader เพื่ออ่านข้อมูลจากผู้ใช้ผ่าน System.in และรับค่า N ด้วย Integer.parseInt(bufferedReader.readLine().trim()) เช่นเดียวกับเดิม  
จากนั้น เราปิดการใช้งาน bufferedReader ด้วย bufferedReader.close()  
ในลูป for เราจะทำการคำนวณและพิมพ์สูตรคูณของ N ด้วยตัวเลข 1 ถึง 10 โดยใช้ System.out.printf() เพื่อจัดรูปแบบข้อความที่จะแสดงผล โดยในที่นี้จะใช้ %d x %d = %d%n  
ซึ่งรับพารามิเตอร์ตามลำดับเพื่อกำหนดค่าที่จะแสดงผล  
สุดท้ายเราจะพิมพ์เครื่องหมายบรรทัดใหม่ (%n) หลังจากลูป for เพื่อคั้นแต่ละบรรทัด  
สรุปแล้ว โค้ดนี้จะรับจำนวนเต็ม N จากผู้ใช้และพิมพ์สูตรคูณของ N ด้วยตัวเลข 1 ถึง 10 ตามรูปแบบที่กำหนดในโจทย์  
 \*/*