//Kelvin Kellner

//Mrs. Cooper

//May 3rd, 2019

//Unit Five Objects Coding Test

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

public class Garden {

// FILE I/O VARIABLES

private static String fileName = "plantinfo.txt";

// MAIN METHOD

public static void main(String[] args) throws IOException

{

// Text File Information Processing

FileReader file = new FileReader(fileName);

BufferedReader read = new BufferedReader(file);

int count = Integer.parseInt(read.readLine());

Plant[] plant = new Plant[count]; // Initialize our Plant array

for(int i=0;i<count;i++)

{

String type = read.readLine().trim();

int height = Integer.parseInt(read.readLine().trim());

String colour = read.readLine().trim();

String e = read.readLine();

boolean edible = stringToBoolean(e);

String p = read.readLine();

boolean perennial = stringToBoolean(p);

plant[i] = new Plant(type, height, colour, edible, perennial);

}

read.close();

// Printing Plant Information

findFood(plant);

System.out.println("\nThere are " + countPerennials(plant) + " perennial plants in the garden.");

} // End Main

// Find Food Method

// -> This method will go through an array of plants and print which one's are and are not edible.

public static void findFood(Plant[] plant)

{

for(int i=0;i<plant.length;i++) // For every plant in the array...

plant[i].eat(); // Call the "eat" method, which will print whether or not the plant is edible.

}

// Count Perennials Method

// -> This method will count, then return, the number of plants that are perennials within an array of plants.

public static int countPerennials(Plant[] plant)

{

int count=0;

for(int i=0;i<plant.length;i++) // For every plant in the array...

{

if(plant[i].isPerennial()) // If the plant is perennial...

count++; // Add one to the counter.

}

return count; // Return the final result.

}

// UTILITY METHODS

public static boolean stringToBoolean(String text)

{

if(text.equalsIgnoreCase("y") || text.equalsIgnoreCase("yes") || text.equalsIgnoreCase("true") || text.equalsIgnoreCase("t"))

return true;

else if(text.equalsIgnoreCase("n") || text.equalsIgnoreCase("no") || text.equalsIgnoreCase("false") || text.equalsIgnoreCase("f"))

return false;

else

{

System.out.println("Error: Invalid boolean input.\nFalse was returned by default.");

return false;

}

}

} // End Class