```java
//to check whether an array is wonderous or not
import java.util.Scanner;
class wonderous_square //class starts
{
    public static void main()                          //main starts
    {
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the number of rows and columns");
        int n=sc.nextInt();
        if (n < 2 || n > 10) {
            System.out.println("Invalid value of n!");
            return;
        }
        int a[][] = new int[n][n];
        System.out.println("Enter elements of the matrix: ");
        for (int i = 0; i < n; i++) {
            System.out.println("Enter Row "+ (i+1) + " :");
            for (int j = 0; j < n; j++) {
                a[i][j] = sc.nextInt();
            }
        }System.out.println("The Matrix is:");
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                System.out.print(a[i][j] + "\t");
            }
            System.out.println();
        }
        //Check Wondrous
        int nSq = n * n;
        double validSum = 0.5 * n * (nSq + 1);
        boolean wondrous = isWondrous(a);
        if (wondrous) {
            System.out.println("Yes, it represents a wondrous square");
        } else {
            System.out.println("Not a wondrous square");
        }
        //Print Prime Numbers
        printPrime(a);
    }
    public static boolean isWondrous(int arr[][]) {
        int n = arr.length;
        int nSq = n * n;
        double validSum = 0.5 * n * (nSq + 1);
        /*
         * seenArr is used to check that
         * numbers are not repeated
         */
        boolean seenArr[] = new boolean[nSq];
        for (int i = 0; i < n; i++) {
            int rSum = 0, cSum = 0;
            for (int j = 0; j < n; j++) {
                if (arr[i][j] < 1 || arr[i][j] > nSq) {
                    return false;
                }
                //Number is not distinct
                if (seenArr[arr[i][j] - 1]) {
                    return false;
                }
                seenArr[arr[i][j] - 1] = true;
                rSum += arr[i][j];
                cSum += arr[j][i];
            }
            if (rSum != validSum || cSum != validSum) {
                return false;
            }
        }
        return true;
    }
    public static void printPrime(int arr[][]) {
        int n = arr.length;
        System.out.println("Prime\tRow Index\tColumn Index");
        for (int i = 0; i < n; i++) {
            for (int j = 0; j < n; j++) {
                if (isPrime(arr[i][j])) {
                    System.out.println(arr[i][j] + "\t" + i + "\t\t" + j);
                }
            }
        }
    }
    public static boolean isPrime(int num) {
        int c = 0;
        for (int i = 1; i <= num; i++) {
            if (num % i == 0) {
                c++;
            }
        }
        return c == 2;

}
 //main ends
}
    //class ends
```