Java-排序-小组中每位都有一张卡片
题目描述：
小组中每位都有一张卡片，卡片上是 6 位内的正整数，将卡片连起来可以组成多种数字，计算组成的最大数字。

输入描述：
“,”号分割的多个正整数字符串，不需要考虑非数字异常情况，小组最多 25 个人
输出描述：
最大的数字字符串
补充说明：
示例 1
输入：
22,221
输出：
22221
说明：
示例 2
输入：
4589,101,41425,9999
输出：
9999458941425101
说明：

```
import java.util.*;
import java.util.stream.Collectors;

// 注意类名必须为 Main, 不要有任何 package xxx 信息
public class Main {
    private static String max = "";

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        while (scanner.hasNextLine()) {

            String[] split = scanner.nextLine().split(",");
            List<String> collect = Arrays.stream(split).sorted().collect(Collectors.toList());
            List<Num> comStrList = new ArrayList<>();
            int max = collect.stream().map(String::length).max(Integer::compareTo).get();
            for (int i = 0; i < collect.size(); i++) {
                String s = collect.get(i);
                String compareStr = s;
                if (compareStr.length() < max) {
                    compareStr += "0";
                }
                comStrList.add(new Num(s, compareStr));
```

```
            }

            List<Num> collect1 = comStrList.stream().sorted((s1,s2)->{
                String string = s1.string;
                String string1 = s2.string;
                if (string1.length() > string.length()&&string1.startsWith(string)) {
                    string1 = string1.substring(string.length());
                } else if (string.length() > string1.length()&&string.startsWith(string1)) {
                    string = string.substring(string1.length());
                }
                return string.compareTo(string1);
            }).collect(Collectors.toList());
            for (int i = collect1.size()-1; i >=0; i--) {
                System.out.print(collect1.get(i).string);

            }
            System.out.println();
        }
    }

    static class Num {

        String string;
        String compareStr;

        Num(String string, String compareStr) {
            this.string = string;
            this.compareStr = compareStr;
        }

        public String getCompareStr(){
            return compareStr;
        }
    }


}
```