Day 28: RegEx, Patterns, and Intro to Databases

by AvmnuSng

Objective

Today, we're working with regular expressions. Check out the Tutorial tab for learning materials and an instructional video!

Task

Consider a database table, *Emails*, which has the attributes *First Name* and *Email ID*. Given N rows of data simulating the *Emails* table, print an alphabetically-ordered list of people whose email address ends in *@gmail.com*.

Input Format

The first line contains an integer, N, total number of rows in the table.
Each of the N subsequent lines contains 2 space-separated strings denoting a person's first name and email ID, respectively.

Constraints

- $2 \leq N \leq 30$
- Each of the first names consists of lower case letters $[a - z]$ only.
- Each of the email IDs consists of lower case letters $[a - z]$, @ and . only.
- The length of the first name is no longer than 20.
- The length of the email ID is no longer than 50.

Output Format

Print an alphabetically-ordered list of first names for every user with a gmail account. Each name must be printed on a new line.

Sample Input

```
6
riya riya@gmail.com
julia julia@julia.me
julia sjulia@gmail.com
julia julia@gmail.com
samantha samantha@gmail.com
tanya tanya@gmail.com
```

Sample Output

```
julia
julia
riya
samantha
tanya
```

Medium Submitted 17695 times
Max Score 30

Need Help?

View Tutorial

View Discussions

View Editorial Solution

View Top Submissions

Rate This Challenge:

Download problem statement

Download sample test cases

Suggest Edits

Current Buffer (saved locally, editable) Java 8

```
import java.io.*;
import java.util.*;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

public class Solution {

    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN.
        Print output to STDOUT. Your class should be named Solution. */

        /*
        1. scan N = number of entries
        2. store entries intp a string array
        3. print entries that has @gmail.com in them
         */

        Scanner sc = new Scanner(System.in);

        Pattern p = Pattern.compile("\\p{Alpha}+");

        int N = sc.nextInt();
        sc.nextLine();

        String[] ar = new String[N];
        for (int i = 0; i < N; i++) {
            ar[i] = sc.nextLine();
        }

        String myString;
        ArrayList<String> L = new ArrayList<>();
        for (int i = 0; i < N; i++) {
            myString = ar[i];
            Matcher m = p.matcher(myString);

            if (m.find() && myString.contains("@gmail.com")) {
                L.add(m.group());
            }
```

```
        }

        L.sort(String::compareToIgnoreCase);

        for (String s :
                L) {
            System.out.println(s);
        }

    }
}
```

Line: 20 Col: 1

Upload Code as File | Test against custom input | Run Code | Submit Code

Testcase 0 ✔

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
6
riya riya@gmail.com
julia julia@julia.me
julia sjulia@gmail.com
julia julia@gmail.com
samantha samantha@gmail.com
tanya tanya@gmail.com
```

Your Output (stdout)

```
julia
julia
riya
samantha
tanya
```

Expected Output

```
julia
julia
riya
samantha
tanya
```