

Challenge 1 - Anonymous Poll

Your university wants you to fill an “anonymous” poll. There are many questions but some fields are mandatory:

- Gender
- Age
- Studies
- Academic year

Are you sure this is really anonymous? If you have the [list of all students](#) and you take a random poll, maybe you can identify the student that could have filled it.

The [list of students](#) is a text file. Each line has the information of one student in comma separated values:

```
name1,gender1,age1,studies1,academic_year1
name2,gender2,age2,studies2,academic_year2
...
```

The gender can be “M” (male) or “F” (female).

Input

Input starts with a number “T” ($T \leq 100$) followed by “T” cases. Each case is a list of comma separated values which are:

```
gender,age,studies,academic_year
```

Output

For each case you must print the list of students that match the input. If there

are more than one, print them in lexicographical order and comma separated. If there is no match, print "NONE".

Use the following format:

Case #Ti: Ri

Where "Ti" is the case number and "Ri" the result.

Sample input

```
5
M,21,Human Resources Management,3
F,20,Systems Engineering,2
M,20,Manufacturing Engineering,3
M,18,Electrical Engineering,4
F,25,Construction Engineering,4
```

Sample output

```
Case #1: NONE
Case #2: Morgan Martinez Moore
Case #3: Alfie Hernandez Diaz
Case #4: Mohammad Green Morales,Oliver Carter
Rivera
Case #5: Ellie Brown Reed,Laura Stewart
Foster,Nicole Peterson Torres
```

Submit & test your code

To test and submit code we provide a set of tools to help you. Download [contest tools](#) if you haven't already done that. You will then be able to test and submit your solution to this challenge with the challenge token.

Challenge token: aQmt1qCpU0u1-pDLRbws

To test your program

```
./test_challenge aQmt1qCpU0u1-pDLRbws path/program
```

A nice output will tell you if your program got the right solution or not. You can try as many times as you need.

To submit your program to the challenge

```
./submit_challenge aQmt1qCpU0u1-pDLRbws  
path/source_pkg.tgz path/program
```

Note that you first need to solve the test phase before submitting the code. During the submit phase, in some problems, we might give your program harder questions, so try to make your program failsafe.

Important: In this phase, you must provide the source code used to solve the challenge and, if necessary, a brief explanation of how you solved it.

Remember **you can only submit once!** Once your solution is submitted you won't be able to amend it to fix issues or make it faster, so please be sure your solution is finished before submitting it.

If you have any doubts, please check the [info section](#).

Go ahead

I'm done! :)

Once you have submitted your code, hit refresh and continue to next challenge.

I'm stuck! :(

Be sure you follow the [Tuenti Engineering](#) twitter for updates and possible hints during the contest.

If this challenge is too hard and you are blocked, you will be able to skip it after two hours. Note that **you won't be able to complete it later**, and you

have a limited number of challenges to skip.

Finally, if you run out of skips but are still really stuck with one problem, you will be able to skip it after 24 hours.

Challenge status:

Test case	Not done
Solution submitted	Not done
Skip	You still have to wait 0h, 30m and 0s to be able to skip this challenge

Refresh status

Tweet about this! [#TuentiChallenge4](#)



Follow [@Tuentieng](#)