Validating Email Addresses With a Filter

15 more points to get your next star!

Rank: **680689** | Points: **95/110**

Problem | Submissions | Leaderboard | Editorial

You are given an integer N followed by N email addresses. Your task is to print a list containing only valid email addresses in lexicographical order.

Valid email addresses must follow these rules:

- It must have the username@websitename.extension format type.
- The username can only contain letters, digits, dashes and underscores $[a-z], [A-Z], [0-9], [\_-]$.
- The website name can only have letters and digits $[a-z], [A-Z], [0-9]$.
- The extension can only contain letters $[a-z], [A-Z]$.
- The maximum length of the extension is 3.

Concept

A filter takes a function returning True or False and applies it to a sequence, returning a list of only those members of the sequence where the function returned True. A Lambda function can be used with filters.

Let's say you have to make a list of the squares of integers from 0 to 9 (both included).

```
>> l = list(range(10))
>> l = list(map(lambda x:x*x, l))
```

Now, you only require those elements that are greater than 10 but less than 80.

```
>> l = list(filter(lambda x: x > 10 and x < 80, l))
```

Easy, isn't it?

Example

Complete the function fun in the editor below.

fun has the following paramters:

- string s: the string to test

Returns

- boolean: whether the string is a valid email or not

Input Format

The first line of input is the integer N, the number of email addresses.

N lines follow, each containing a string.

Constraints

Each line is a non-empty string.

Sample Input

```
3
lara@hackerrank.com
brian-23@hackerrank.com
britts_54@hackerrank.com
```

Sample Output

```
['brian-23@hackerrank.com', 'britts_54@hackerrank.com', 'lara@hackerrank.com']
```

Change Theme Language Python 3

```python
import re
def fun(s):
    # return True if s is a valid email, else return False
    patten = "^[a-zA-Z0-9\-\_]+@[A-Za-z0-9]+[\.][a-zA-Z]{1,3}$"
    return re.search(patten,s)!=None

def filter_mail(emails):
    return list(filter(fun, emails))

if __name__ == '__main__':
    n = int(input())
    emails = []
    for _ in range(n):
        emails.append(input())

filtered_emails = filter_mail(emails)
filtered_emails.sort()
print(filtered_emails)
```

Line: 12 C

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

Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Sample Test case 0

Sample Test case 1

Input (stdin) Download

```
3
lara@hackerrank.com
brian-23@hackerrank.com
britts_54@hackerrank.com
```

Your Output (stdout)

```
['brian-23@hackerrank.com', 'britts_54@hackerrank.com',
'lara@hackerrank.com']
```

Expected Output Download

```
['brian-23@hackerrank.com', 'britts_54@hackerrank.com',
'lara@hackerrank.com']
```