Description

Let's add new functionality to our program by making it read the user credentials and verify whether they are correct or not. Every user must provide the following credentials: their first and last name and a valid email address.

The email address format is defined in RFC standards and is very complicated. In reality, email addresses are usually verified by sending a verification letter to which the user is required to reply and confirm the validity of the address. So in general, it is feasible just to check whether the provided string looks like an email address. It should contain the name part, the `@` symbol, and the domain part.

Checking names is even more intricate. We are not going to require users' photo ID and similar stuff, but take into your account that a learning platform may issue personal certificates of accomplishment, so it would not make any sense to issue a certificate with a name like ~D0MInAt0R~.

We need to put several limitations on the personal name format. First, users should state their full names that include the first and the last name. Accept only ASCII characters, from `A` to `Z` and from `a` to `z` as well as hyphens `-` and apostrophes `'`. For example, **Jean-Claude** and **O'Neill** are valid names, but **Stanisław Oğuz** is not. We respect every student, but we kindly request them to write their names using English-alphabet letters only.

In addition to the above, some students may have really long names like Robert Jemison Van de Graaff or John Ronald Reuel Tolkien. Do not restrict their right to indicate their full name during registration. In this case, use the following convention: the first part of the full name before the first blank space is the first name, and the rest of the full name should be treated as the last name.

We are not done yet! A name may contain one or more hyphens and/or apostrophes, but don't allow them as the first or the last character of any part of the name. Also, these characters cannot be adjacent to each other. The first name and the last name must be at least two characters long.

You may use unit tests to be sure you've implemented all name and email format requirements correctly.

Objectives

In addition to the features of the first stage, your program should:

1. Recognize a new command: `add students` and respond with the following message: `Enter student credentials or 'back' to return`.
2. Recognize a new `back` command and react as follows: if users want to finish adding new students, the program should print a message with the total number of students added during the session, for example: `Total 5 students have been added.` Otherwise, print a hint: `Enter 'exit' to exit the program`.
3. The program should read user credentials from the console and check whether they match the established patterns. If the credentials match all patterns, print `The student has been added.` Otherwise, it should print which part of the credentials is not acceptable: `Incorrect first name`, `Incorrect last name` and `Incorrect email`.
4. If the input cannot be interpreted as valid credentials, the program should print `Incorrect credentials`.

Examples

The greater-than symbol followed by a space (`> `) represents the user input. Note that it's not part of the input.

Example 1: *students with correct credentials*

```
Learning Progress Tracker
> add students
Enter student credentials or 'back' to return:
> John Doe jdoe@mail.net
The student has been added.
> Jane Doe jane.doe@yahoo.com
The student has been added.
> back
Total 2 students have been added.
> exit
Bye!
```

Example 2: *students with incorrect credentials*

```
Learning Progress Tracker
> back
Enter 'exit' to exit the program.
> add students
Enter student credentials or 'back' to return:
> help
Incorrect credentials.
> John Doe email
Incorrect email.
> J. Doe name@domain.com
Incorrect first name.
> John D. name@domain.com
Incorrect last name.
```

```
> back
Total 0 students have been added.
> exit
Bye!
```

Example 3: *students with correct and incorrect credentials*

```
Learning Progress Tracker
> add students
Enter student credentials or 'back' to return:
> Jean-Clause van Helsing jc@google.it
The student has been added.
> Mary Luise Johnson maryj@google.com
The student has been added.
> 陳 港 生
Incorrect first name.
> back
Total 2 students have been added.
> exit
Bye!
```