spec.md

Introduction to Programming

Assignment: Task 3

The Problem

The University of Poppleton is seeking to develop a new technical support system, based on an interactive online chat. The budget is tight, but a project has been started to develop a prototype. You will be working on this development.

Preamble

It is *strongly recommended* that before starting the main exercise below you complete these small tasks. They will give you useful code that will help in the assignment and, as they are not part of the submission, your friendly tutor will be able to offer assistance.

Choosing a Random String

Write a function that when passed a non-empty list of strings returns a random string from the list. Test it from the Python Console using some sample data. When it works it will look something like this:

```
>>> ANSWERS = ['Yes', 'No', 'Maybe', 'Ni!']
>>> random_string(ANSWERS)
'Yes'
>>> random_string(ANSWERS)
'Ni!'
```

Checking an Email

Write a function that checks if a given string could plausibly be an email address at a particular domain. There will be two parameters: the email address and the domain. The rules to be applied are:

- The address must contain exactly one @ character.
- The @ character must have at least two characters before it.
- The characters after the @ character must be the domain (the other parameter), and nothing else.

Write a short program to test the function. Its output might resemble:

```
fred@pop.ac.uk is valid at pop.ac.uk.
fred@pop.ac.uk is invalid at unipop.ac.uk.
@pop.ac.uk is invalid at pop.ac.uk.
f@pop.ac.uk is invalid at pop.ac.uk.
fred.at.pop.ac.uk is invalid at pop.ac.uk.
```

Extracting the User

Write a function that takes an email address and returns the left-hand part of it (that is, the part before the @ character). This is often the real name of the user, so can be useful.

Test the function in the normal way. The output could be:

```
fred@pop.ac.uk --> fred
fred.smith@pop.ac.uk --> fred.smith
```

Waiting for Something

Write a short program that prompts the user to enter a string. If the string is not "Ni!" (ignore the case of the letters), the program should prompt the user to re-enter. If the string is "Ni!" (or "NI!", or "ni!", or even "nI!") the program should exit.

Spotting a Word

Write a function that takes a string representing a question, along with another string containing a single word. The function should return True if the word is in the question, or False if not. Ignore word boundaries (so the word you are seeking could be inside another word), and ignore the case of the words.

The System

Now develop the following Chat System. You will find that the programs and functions you have from the exercises above will come in very useful.

The program should work as follows.

- The program should prompt the user to enter their University of Poppleton email address. This should be checked for validity (the domain must be *pop.ac.uk*) and the user should be informed whether or not it is valid. If the email is invalid the program should exit, otherwise it will continue.
- The chat system should introduce the operator by name. The name should be selected randomly from a long list so as to give the impression that there are a lot of operators whereas, in reality, there is only one.
- The chat system should greet the user by name. The user's name is the left-hand part of their email (before the @).
- The chat system should then prompt the user to enter their question, and the system should respond appropriately. The system's response should be as follows:
  - If the question contains the word "library", the response should be "The library is closed today.".
  - If the question contains "WiFi", the system responds "WiFi is excellent across the campus.".
  - If the question contains "deadline", the system responds "Your deadline has been extended by two working days."
  - *You should add at least three more responses.*
  - If no matching word is found, the chat system should respond randomly with phrases such as "Hmmm", "Oh, yes, I see", "Tell me more" and the like.
- This continues until the user enters "bye", "exit" or any other word that indicates they have had enough.
- To simulate the robustness of the Poppleton WiFi network, after each question there is a possibility that the system should randomly disconnect. This should happen 10% of the time, but should be configurable (so that at busy times a random disconnect is more likely).

This program seems complex, but if built up gradually it is actually not very long. Pay attention to opportunities for code reuse, and structure the code well, and all will be fine.

Examples

Some sample runs of the final program are below.

Here Bob has a WiFi problem. "Janice" is a randomly chosen name.

```
Welcome to Pop Chat
One of our operators will be pleased to help you today.

Please enter your Poppleton email address: bob@pop.ac.uk
Hi, Bob! Thank you, and Welcome to PopChat!
My name is Janice, and it will be my pleasure to help you.
---> WiFi is down.
Wifi is excellent across campus.
---> No. It's down.
You should try working on this system.
---> Really. It's down.
Mmmm.
---> exit

Thanks, Bob, for using PopChat. See you again soon!
```

Jim has a good experience here:

```
Welcome to Pop Chat
One of our operators will be pleased to help you today.

Please enter your Poppleton email address: jim@pop.ac.uk
```

```
Hi, Jim! Thank you, and Welcome to PopChat!
My name is Fiona, and it will be my pleasure to help you.
---> I need coffee.
Teekee is open until 9pm this evening.
---> Goodbye.

Thanks, Jim, for using PopChat. See you again soon!
```

Sadly, Jane was the victim of a network error:

```
Welcome to Pop Chat
One of our operators will be pleased to help you today.

Please enter your Poppleton email address: jane@pop.ac.uk
Hi, Jane! Thank you, and Welcome to PopChat!
My name is Fiona, and it will be my pleasure to help you.
*** NETWORK ERROR ***

Thanks, Jane, for using PopChat. See you again soon!
```

Bob was having a long chat before he discovered that accessing the help system terminates the chat (implenting this is optional):

```
Welcome to Pop Chat
One of our operators will be pleased to help you today.

Please enter your Poppleton email address: bob@pop.ac.uk
Hi, Bob! Thank you, and Welcome to PopChat!
My name is Arthur, and it will be my pleasure to help you.
---> I need to go to the library.
The library is closed today.
---> I need a book.
Oh, my.
---> Or I will miss my deadline.
Your deadline has been extended by two working days.
---> Great. Thanks.
Yes.
---> Yes?
You should try working on this system.
---> Is this a real person?
That is interesting.
---> help

Thanks, Bob, for using PopChat. See you again soon!
```