Computer Networks HW1

B03902042 宋子維

Description

In this homework, we are asked to write a IRC robot program. We need to connect to socket and send/receive message to/from socket. I use python3 to implement this programming assignment.

How to Execute

There are one main program, namely **main.py** and two auxiliary programs, that is, **evalExp.py** and **parser.py** which contain some functions of robot. The directory structure is shown as follows. To execute my programs, just type `python3 main.py`.

```
    [b03902042]
        - config
        - main.py
        - evalExp.py
        - parser.py
```

Program Structure

In main.py, it first parse the config file to get the channel name and its key.

```
Config = {}
with open('config', 'r') as f:
    for line in f:
        tmp = line.split('=')
        Config[tmp[0]] = tmp[1].strip('\n\'')
Channel = Config['CHAN']
Key = Config['CHAN_KEY']
```

Then the robot create a socket and connect to port 6667 of *irc.freenode.net*. After connect to server, it must login with IRC command `USER`. Besides, I also use `NICK` command to change robot's nickname. Then, we can join the channel with `JOIN` command.

```
IRCSocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
# connect
IRCSocket.connect((IRCServer, IRCPort))
# login
sendMsg('USER {} {} {} {}'.format('robot', IRCServer, 'QAQ', 'WindRobot'))
sendMsg('NICK {}'.format(NickName))
# join
sendMsg('JOIN {} {}'.format(Channel, Key))
```

Note: sendMsg() is a function that can send message with IRC message format.

After joining the channel, it needs to keep receiving data from the socket. Then all the robot needs to do is parse the data it receives which will follow IRC message format. If robot gets PING message from server, it should send PONG back to that server.

```
IRCMsg = IRCSocket.recv(Bufsize).decode().strip('\r\n')
    if not IRCMsg:
        continue
msg = IRCMsg.split()
# PING-PONG response to server
if msg[0] == 'PING':
    sendMsg('PONG ' + msg[1])
    continue
```

The remaining part of main.py is to deal with the commands, such us **@repeat**, **@help**, **@play** and **@guess** from other users.

evalExp.py is a calculator (**@cal**) I implement which **can support not only the binary operators in the spec but also unary positive and unary negative!**

parser.py implement two HTML parsers, which are **@youtube** and **@books**(search items in 博客來). In this program, I use **BeautifulSoup** toolkit to analyze HTML file.

Challenge and Solution

The greatest challenge I encounter is to read IRC message format and IRC configuration. Because when we use `irssi` command to login, we don't need to use `USER` command to configure our identities, then I think robot can do the same thing with `irssi` command. Hence, in the beginning, my robot didn't use `USER` command to login and it direct sent `JOIN` command to try to join the channel. Then, it failed to join it.

The solution is that the robot must use `USER` command to login after connecting to the server. Otherwise, it can not take any action!

Reflections about this homework

It's a nice homework! It is the first time that I use IRC to chat with others, and so does my robot! It's fortunate to use python to do this assignment since many things can be easy in python programming (like HTML parser).