Assignment #05: IRC Bot

Background

We’ve covered a lot of Ruby and by this point you are expected to be familiar with the language. Now we’re going to explore a real world use case of Ruby.

IRC bots are simple scripts (in our case written in Ruby) that connect to IRC and parse the chat for commands. When an IRC bot sees a command it is programmed to respond to, it executes the command using preprogrammed logic to respond in the IRC channel.

Your assignment is to apply your newfound knowledge of Ruby to build an awesome IRC bot that can respond to at least one command. What command the IRC bot listens for and what it does in response is completely up to you.

The basic flow of an IRC bot is:

1. The bot connects to the IRC server
2. The bot assigns itself a nickname usually ending in the word “bot”
3. The bot joins a channel (in this case #bitmaker)
4. The bot goes into a loop where it listens for commands and responds until it is shut down

Learning Goals:

Adv Ruby

- TCPSocket
    - server, port
- File IO (.eof?)
- Gems (socket)

Advanced OOP

- Modules
- Multiple Dependencies
- Many Classes
- Getters and setters (attr_reader, attr_writer, attr_accessor)
- Good code design
- public
- private

IRC and IRC Protocol

- What is IRC?
- PRIVMSG, USER, NICK, JOIN

External Resources:

- Stackoverflow: My Ruby IRC bot doesn't connect to the IRC server. What am I doing wrong?
- Wikipedia: Internet Relay Chat bot

Suggested Process:

1. Begin with just trying to open a connection to the IRC Server.
2. Then start looking at reading/sending messages from/to the server.
3. Once you can connect and read/send messages, then you should focus on your functionality. For example, open a connection to another service, read it's data and then tell the channel about it.

Tips and Tricks

- Make sure to look over the IRC Protocol documentation. It will give you some good insight into the way IRC works, and might give you an idea for a cool, helpful IRC Bot.
- Try to make a class for each thing that could be acted upon, e.g. you could make a class for the Bot, a User (someone who is logged into the channel), a Message, etc.
- Be careful about infinite loops, they can nuke (read: make unavailable, take down) the IRC server and/or get you banned from it for a while.