The `log_scraper.py` script is my solution to Arrcus' IP scraping coding challenge.

As required, this script takes in a list of device log files and a valid IPv4 address, and outputs the path of the IP address across the devices. I have tried to handle several cases where the input provided is illegal, such that the script fails gracefully after returning a message pointing to the error's root cause.

Files provided

This solution folder contains the following files:

- `log_scraper.py`: My solution
- This `README.pdf` file with building and running instructions.
- A `logs/` folder containing several device logs. This is my test input.

Requirements

This script requires a minimum Python version of 3.4. Since it uses only native Python modules, no additional packages are required.

Installing Python 3

If you do not have Python 3 on your machine, you can install an appropriate version with the following commands:

On OS X

`brew install python3` (Note, you must have the `brew` installer for this command to work)

On Ubuntu

`sudo apt-get install python3.6`

Testing the Script

This script provides a command line interface to the IP scraping functionality. When calling the script, you must provide the following as input:

1. A valid IPv4 address, which you would like to search for in the log files
2. Either a folder that contains valid log files, or a list of paths to log files, but not both.

The script will open and read all provided log files, and compile an output message that contains the path of the IP address in the format specified in the problem statement.

For directions on using the script, please use the help flag:

```
python log_scraper.py -h
```

Calling the Script with One or More Log File Paths

```
python log_scraper.py -i <a valid IPv4 address> -f <file 1> <file 2> <file 3>
<etc...>
```

For example:

```
python log_scraper.py -i 18.44.45.86 -f logs/device1.txt logs/device3.txt
logs/device4.txt
```

Calling the Script with a Directory Containing Device Log Files

```
python log_scraper.py -i <a valid IPv4 address> -d <path to a folder that
contains log files>
```

For example:

```
python log_scraper.py -i 18.44.45.86 -d logs/
```