web_scraper.py

1.0 Introduction

Web_scraper.py is the primary file used for command line interface with back end components, Selenium_Navigation.py and Scraper.py.

2.0 Architectural and component-level design

1. main(args)
    a. Main processes command line arguments given upon starting the script. The valid arguments are as follows:

        -p --pages: scrapes the number of pages equal to the proceeding number. Defaults to 0.
        -j --jobs: scrapes UP TO the given number of jobs per page, defaults to 25.
        -o --output: appends formatted data, as formatted by data_formatter.py, to an output file called 'formatted.txt.'
        -u --upload: disables or enables data upload to the database. Defaults to true. Argument is <true> or <false>.
        -h --head: If entered, will NOT run the browser in headless mode, making the window visible for the user.
        --help: Prints a usage message.

    b. Defines a list of job titles to search and scrape data from.
        i. This is intentionally NOT a command line argument, to help prevent the database from being flooded with erroneous tables and job names.

    c. Uses getopt and sys python libraries in order to parse command line arguments.
        i. If improper commands are given, returns an error message prompting you to use the --help argument

d. All arguments are assigned to proper values to be passed into the initialization function of the Crawler class, as well as the Crawler's scrape_jobs function

e. Prints a message indicating the number of pages, the number of jobs per page, and the list of job titles being scraped before instantiating a Crawler object.

f. Finally, the scrape_jobs function is called with the given argument, for every job title.
   i. It should be noted that the job scraping process is incredibly lengthy, taking up to 24 hours for a full suite of job titles to be scraped to completion.

g. After the execution of the crawler finishes, the web driver is closed and execution ends.

3.0 Testing

1. Primary Testing Method: Full System Testing. Because it is a simple command line interface script, unit testing was not necessary.

4.0 Known Bugs

1. --hel will print the help message