2. Data Scrape and Clean:

**Section 1. Scrape websites and save HTML in csv file**

**Output:** data/merged\_searched\_job\_html.csv

**Step1. Get searched job list HTML**

Run python3 code/get\_searched\_job\_html.py and enter position, location, date range, sort to scrape the job list html in the indeed website (https://www.indeed.com/jobs?q=data%20analyst&l=Austin). In this project, here are all the files that need to be generated in this step:

|  |  |
| --- | --- |
| **Generate Files** | **Parameters Entered** |
| data/Data Analyst\_Texas\_30.csv | Data Analyst, Texas, 30, date |
| data/Data Analyst\_ California \_30.csv | Data Analyst, California, 30, date |
| data/Data Analyst\_ New York State\_30.csv | Data Analyst, New York State, 30, date |
| data/Data Engineer\_ Texas \_30.csv | Data Engineer, Texas, 30, date |
| data/Data Engineer\_California\_30.csv | Data Engineer, California, 30, date |
| data/Data Engineer\_ New York State \_30.csv | Data Engineer, New York State, 30, date |
| data/Data Scientist\_ Texas \_30.csv | Data Scientist, Texas, 30, date |
| data/Data Scientist\_ California \_30.csv | Data Scientist, California, 30, date |
| data/Data Scientist\_New York State\_30.csv | Data Scientist, New York State, 30, date |

Run python3 code/merge\_html\_csv.py, which merges csv files in step 1 and add Title and Location columns. The output file is data/merged\_searched\_job\_html.csv.

*Note: Since the generated file is larger than 100M which cannot be uploaded in GitHub, please run the python script and generate it locally.*

**Step 2. Randomly select 100 jobs in each job titles (Data Analyst, Data Engineer, Data Scientist) and get their job description HTML**

Runpython3 code/get\_job\_des.py and enter a job title (Data Analyst/ Data Scientist/ Data Engineer) to get the selected 300 job descriptions. Here are all the files that need to be generated:

|  |  |
| --- | --- |
| **Generate Files** | **Parameters Entered** |
| data/job\_des\_html\_Data Analyst.csv | Data Analyst |
| data/ job\_des\_html\_Data Engineer.csv | Data Engineer |
| data/job\_des\_html\_Data Scientist.csv | Data Scientist |

**Section 2.** **Clean HTML files that generated in section 1-1 and get basic information for each job**

**Output: job\_basic\_info.csv**

**Section 3. Clean HTML files that generated in section 1-2 and get detailed information for each job**

**Output: job\_desc\_info.csv**

**Step 1. Get industry information based on the files that generated in step 2**

Run python3 code/get\_industry.py and enter a job title (Data Analyst/ Data Scientist/ Data Engineer) to fetch industry information for all the jobs in step 2. Here are the output files:

|  |  |
| --- | --- |
| **Generate Files** | **Parameters Entered** |
| data/job\_industry\_Data Analyst.csv | Data Analyst |
| data/ job\_industry\_Data Engineer.csv | Data Engineer |
| data/ job\_industry\_Data Scientist.csv | Data Scientist |

**Step 2. Get skill keywords counts based on the files that generated in step 2**

Run python3 code/get\_skills\_list.py and enter a job title (Data Analyst/ Data Scientist/ Data Engineer) to get skill keywords counts in the job description for all the jobs in step 2. Here are the output files:

|  |  |
| --- | --- |
| **Generate Files** | **Parameters Entered** |
| data/job\_skills\_counts\_Data Analyst.csv | Data Analyst |
| data/ job\_skills\_counts\_Data Engineer.csv | Data Engineer |
| data/ job\_skills\_counts\_Data Scientist.csv | Data Scientist |

**Section 4. Analyze and visualize cleaned data**