Job Hunt Project Report

Jimmy Walsh, Connor Rubenstein, Liz Stevens

At the end of the week, your team will submit a Final Report that describes the following:

SOURCE : Data scraped from builtin.com (and associated websites. Example: Austin TechHub featured on builtin.com has url https://www.builtinaustin.com/.

* **E**xtract: your original data sources and how the data was formatted (CSV, JSON, pgAdmin 4, etc).

- Extracted TechHub Name, Average Cost of Living, Average Salary per Job Category (giving 8 fields), Popular Industries found in each TechHub via website builtin.com scraping with BeautifulSoup and Selenium.
- Data was extracted as a Beautiful Soup or Selenium object.
- Extracted job listings in each of our chosen 8 Job Categories from each TechHub on the site. In order to do so we created a loop utilizing Splinter and BeautifulSoup. The loop does the following:
    1. Navigated to each TechHub's url by appending the TechHub url description
    2. Utilized splinter to select the 7 different job categories we wanted to collect jobs for
    3. On each job category page we extracted the html and used BeautifulSoup to find chunks of code that contained job information
    4. Utilized a while loop to collect 5 jobs from each page that had job information
        - Utilized a try, except in order to bypass ads and jobs not including job_title and/or company
    5. Each time we went through a job we extracted a job title and company and if available a city location

- Utilized the try, except in order to still grab jobs without a city listed and those without a city assigned none

* **T**ransform: what data cleaning or transformation was required.

- Transformed collected data into Pandas Data Frames to be easily visualized.
- Renamed certain column headers to merge properly across multiple Data Frames.
- Converted foreign key objects to integers for uploading into PostgreSQL.
- Reorganized Columns to match database.
- Merged data frames with Names and Primary Keys in order to put foreign keys into job_posting, salary, and tech_hub tables.
- Saved finalized Dataframes as CSV files to be loaded into our SQl Database.

* **L**oad: the final database, tables/collections, and why this was chosen.

- Wanted a relational database, because our data is interconnected.
- Chose to upload CSV’s so that we maintained our primary and foreign key relationships which went away when we utilized SQLAlchemy.
- Upload final CSV files into Job_Hunt Database.

Please upload the report to Github and submit a link to Bootcampspot.