Briq Automation Engineer Coding Challenge

Thank you for your interest in working at Briq!

While leveraging technology to serve the construction industry and make efficient use of their data, we face a variety of technical challenges every day. As is often the case in software, solutions to such problems can come in all shapes and sizes.

As part of evaluating your technical aptitude as a automation engineer, we would also like to analyze your ability to formulate and execute on a solution to a prompt that is open to some interpretation.

The Challenge

This challenge has been designed to touch on the fundamental skills a Briq automation engineer should possess in order to perform in the role day-to-day. There is also plenty of room for exceptional candidates to exceed expectations.

Components

1. Create an Excel Writer
2. Create a PDF Reader
3. Build a web scraper using selenium -
    a. Use the provided url to capture all the information for all the cards / pins
    b. Output in an excel with correct headers
4. Build a REST api automation connector
    a. Use the provided endpoint to capture all the information and save it in an excel with correct headers
    b. Automation should be able to append new information for each run
5. Create a PDF data extraction automation script that extracts information from the PDF
    a. Extract project name (first line of bids for transport.pdf)
    b. Extract the table information and output as excel with correct headers
    c. The output excel should have an additional column for the project name / number (12 columns in total)
6. Create an excel reader to read Leads.xlsx and output it as a JSON

Note: See Appendix for additional information

Please use only java as the implementation language. The scripts should be able to run independently of each other and simultaneously.

Please include enough documentation to get your automations running locally. When submitting your solution, push all relevant files to a github repository and provide the link.

Partial submission is allowed if you are unable to code for a module. In case of partial submission mention in the documentation which components are missing.

Evaluation Criteria

Depending on your solutions, after completing the challenge you will have an opportunity to discuss your solution during a follow-up call with member(s) of our engineering team. Your solution will be evaluated on:

· Do all the solutions appear well structured and does it run locally?

· Is the code organized, clear, modularised and performant?

· Are the primary requirements of the challenge met?

· How have you solved for creating the output

· Ease of changing / adding implementation without impacting other modules

· Scalability and maintenance debt of the code

Beyond the evaluation criteria, feel free to showcase any specific skills, interests, or technologies you are fond of and would like to display.

Appendix

Web Scraper:
Design and create a repeatable automation script

1. Consider the script is supposed to run each week and extract only new records.
2. The script should be scalable and run for different similar cities

Sample url:
https://www.bizjournals.com/milwaukee/feature/crane-watch
https://www.bizjournals.com/seattle/feature/crane-watch
There are a lot of different cities available.

1. In the above url, open the map.
2. Extract all the information from all the pins. You have to click on each pin to display it's info.
3. Save the extracted info in an excel sheet

REST connector:
Link: https://data.sfgov.org/resource/p4e4-a5a7.json

1. Automate gathering data from this REST endpoint

2. Output all values to an excel

PDF Parser:
Bids for Transport.pdf
Active Licences.pdf
Statement.pdf

1. Bonus points if you can output Active Licences.pdf as a proper excel table
2. Bonus points if you can extract the following from Statement.pdf
    1. Bank Address
    2. Customer Name
    3. Customer Address
    4. Account Number
    5. Statement Date
    6. Ending Balance
    7. Total Withdrawals
    8. Total Deposits
    9. Total Checks

Excel Reader:
Leads.xlsx