

QA Candidate Interview Project

Project Instructions

This project has three separate exercises. Please complete all exercises.

For the Manual QA challenge (#1), provide results as a word or text document in the form of Test Notes.

For the Automation QA challenge (#2), provide results as a file (or collection of files) in any programming language or automation framework you choose to use.

For the SQL skills (#3), provide results inline in your email response or as a separate document.

Solve this yourself without help from anyone else.

Feel free to reference online help - stackoverflow, google, etc.

#1 Manual - QA an Elevator

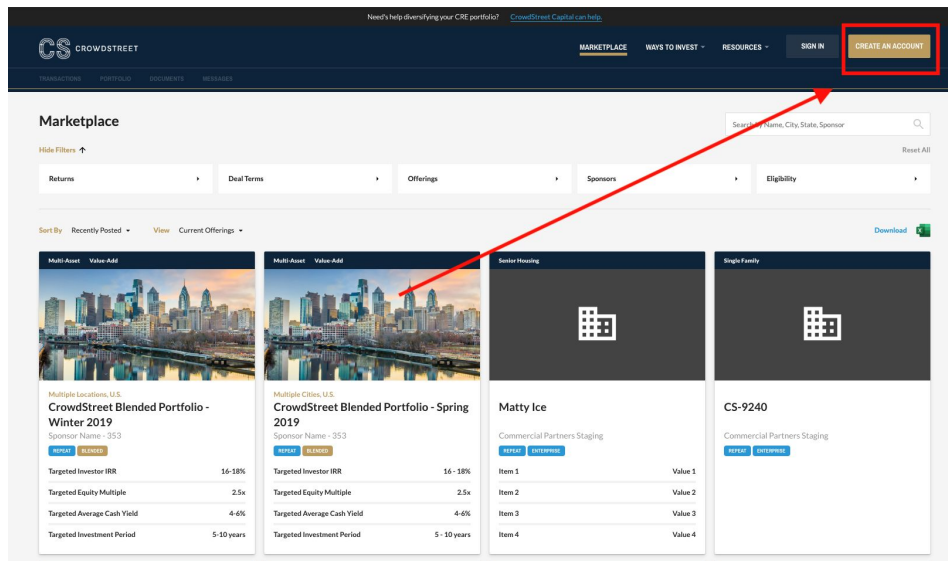
Scenario: You have to QA a 10 story new elevator. The elevator needs to be available on the opening day of the building in 1 week. Write tests scenarios that you would execute to ensure the elevator is ready to use in time for the release.



The 5th floor will have a broken elevator button.
What do you do?

#2 Automation Challenge - Register

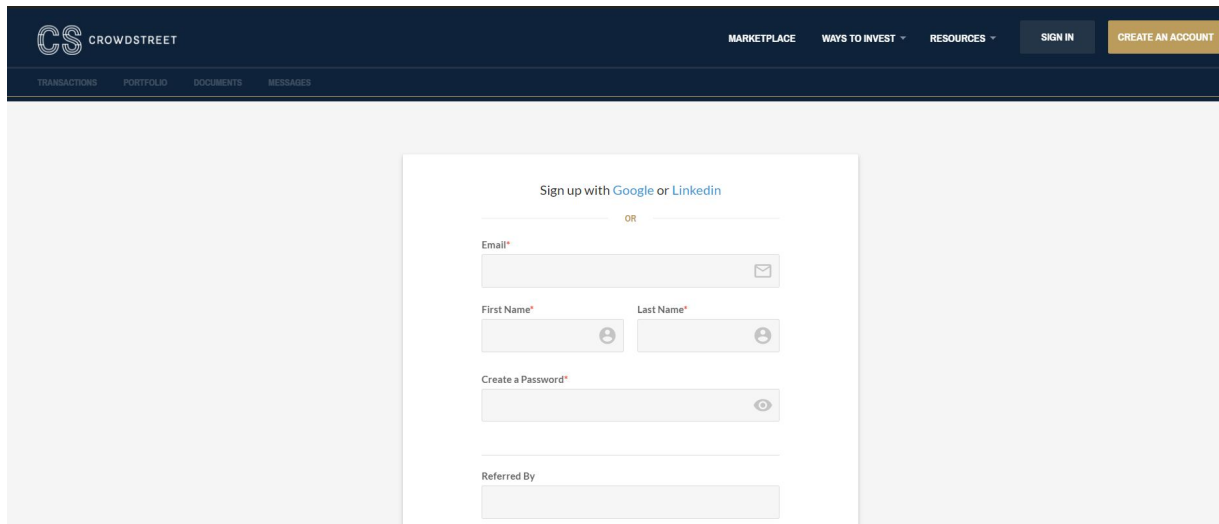
Write an automation test that automates the process of creating a **new account** and walking through the registration process at test.crowdstreet.com.



#2 Automation Challenge

Rules

- Write the test in any automation language or framework and export the steps.
- Bonus points if you can provide a github repo to fork from, including instructions to run the test locally.
- Don't test for negative test cases, only happy path testing.
- Run through the entire registration process and complete your info with faux data.



The screenshot shows the Crowdstreet registration page. The header is dark blue with the 'CS CROWDSTREET' logo on the left. On the right, there are links for 'MARKETPLACE', 'WAYS TO INVEST', 'RESOURCES', 'SIGN IN', and a yellow 'CREATE AN ACCOUNT' button. Below the header, a navigation bar contains links for 'TRANSACTIONS', 'PORTFOLIO', 'DOCUMENTS', and 'MESSAGES'. The main content area is light gray and features a white registration form. The form has the heading 'Sign up with Google or LinkedIn' and an 'OR' separator. Below this, there is an 'Email*' field with an envelope icon. Then, 'First Name*' and 'Last Name*' fields, each with a person icon. Next is a 'Create a Password*' field with an eye icon. Finally, there is a 'Referred By' field.

CS CROWDSTREET

MARKETPLACE WAYS TO INVEST RESOURCES SIGN IN CREATE AN ACCOUNT

TRANSACTIONS PORTFOLIO DOCUMENTS MESSAGES

Sign up with Google or LinkedIn

OR

Email*

First Name* Last Name*

Create a Password*

Referred By

#3 SQL Skills - Part 1

[A] Design and populate an associative table called **people_vehicle** to support the following:

- Mary has a Truck
- John has two Cars and two Trucks
- Chen has no vehicles

Share the contents of the **people_vehicle** table in your results in an email.

[B] Design and populate an associative table called **people_pet** to support the following:

- Mary has no pets
- John has a Dog
- Chen has a Parrot and a Cat

Share the contents of the **people_pet** table in your results in an email.

people	
id	name
1	John
2	Mary
3	Chen

vehicle	
id	name
1	Truck
2	Car
3	Bike

pet	
id	name
1	Dog
2	Cat
3	Parrot

#3 SQL - Part 2

NOTE: Feel free to use PostgreSQL, MySQL or Oracle specific dialects for [C]

[C] Write a query that generates exactly this result set:

name	vehicles	pets
John	Car=2, Truck=2	Dog=1
Mary	Truck=1	None
Chen	None	Parrot=1, Cat=1

[D] Write a query that finds all people who have at least 2 vehicles or 2 pets.
Share the query in your response.

people	
id	name
1	John
2	Mary
3	Chen

vehicle	
id	name
1	Truck
2	Car
3	Bike

pet	
id	name
1	Dog
2	Cat
3	Parrot