***Tech Exercise Guidelines***

* Use your preferred programming language. However, our preferred programming language is Java or Javascript.
* Submit your solution by creating a new repository on Github with a readme file that details how to build, Run tests, etc.
* Integrate API Testing code to CI ( eg. circleCI ) such a way that any new commits to code in Github repository trigger builds and run the API tests.
* Use all coding standards, design patterns, OOPS concepts, etc.

***UI Automation***

1. Navigate to https://www.service.nsw.gov.au/
2. Search for "Apply for a number plate"

Validate the navigation to appropriate page

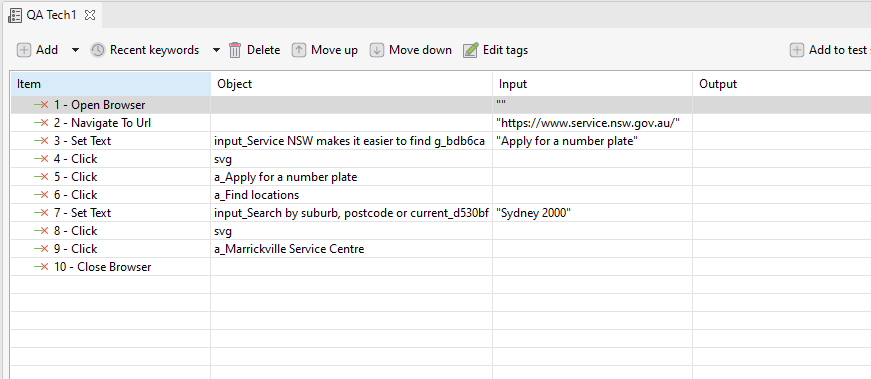
1. Click on Locate us button
2. Enter suburb "Sydney 2000"
3. It should use service centre named as "Marrickville Service Centre"

Make above test as data-driven so that search items can be replaced with anything in step 4 and filtered result in Step 5

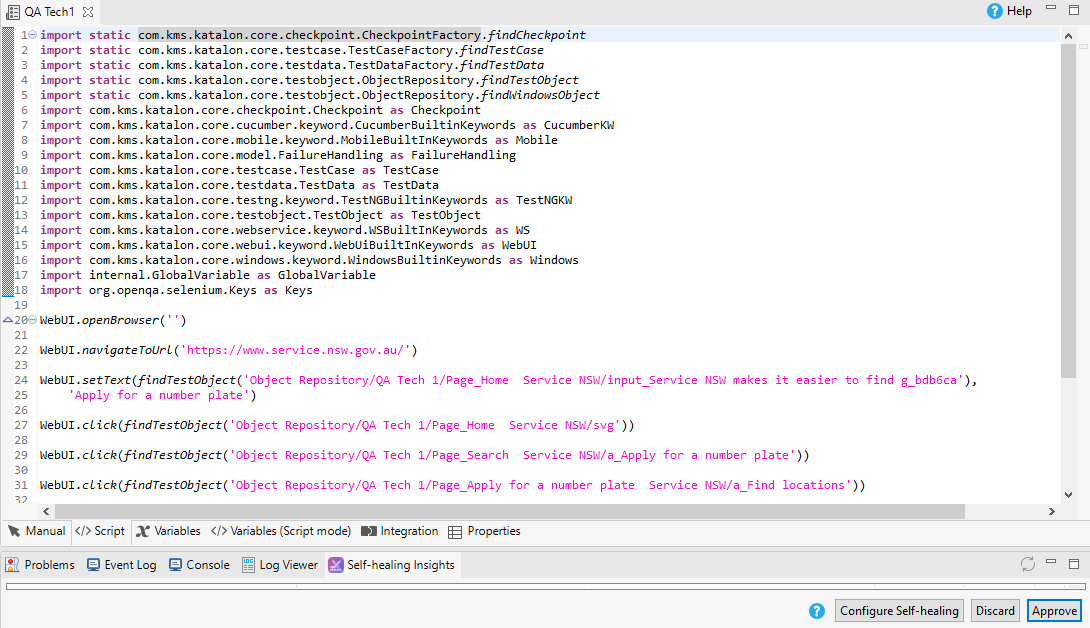
Like “Sydney Domestic Airport 2020” and it should be having “Rockdale Service Centre”

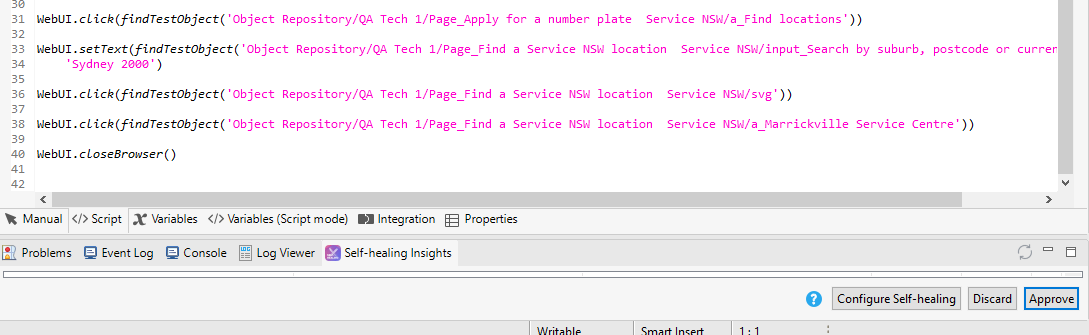
Answer :

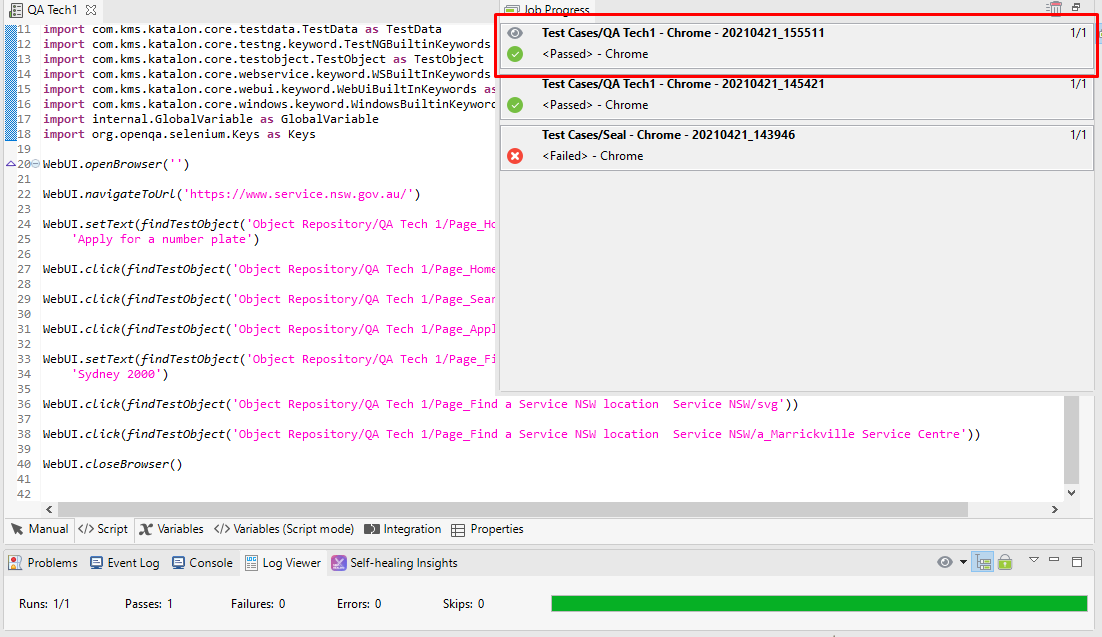
1. Navigate “Apply for number plate” & Enter suburb “Sydney 2000”, “Marricville Service Centre” (Using Katalon V7.7)



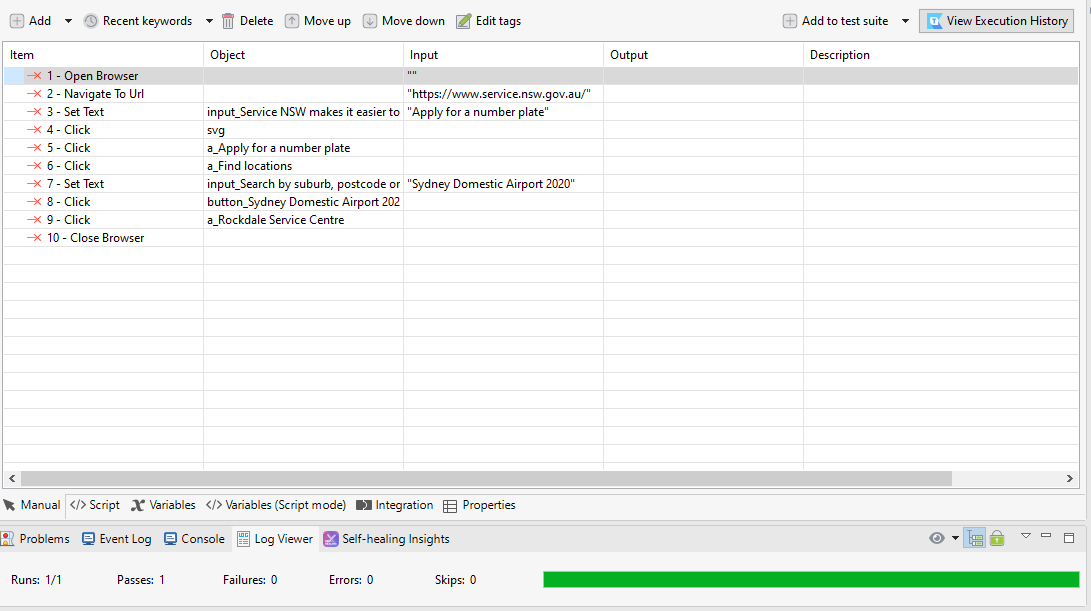
Script

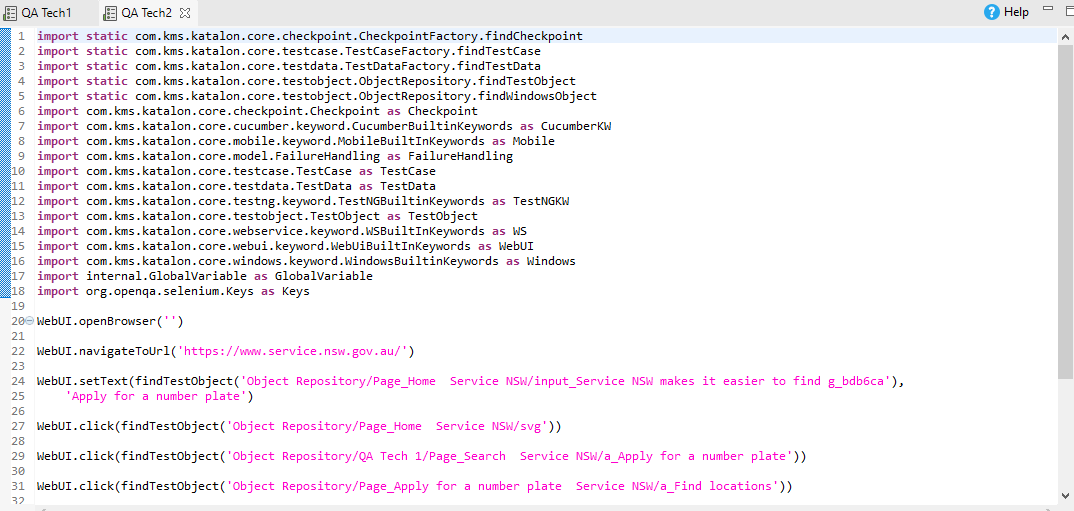


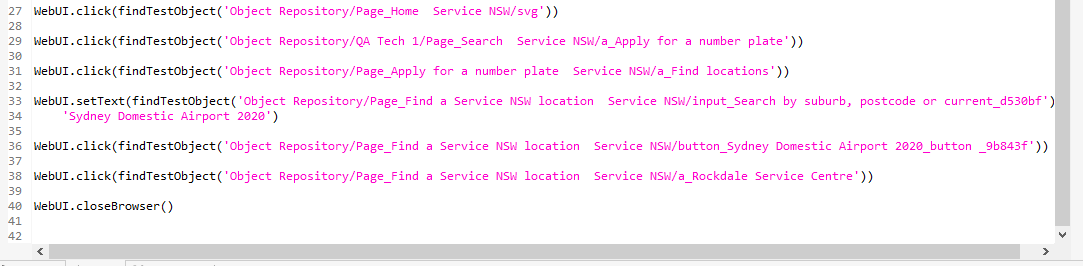


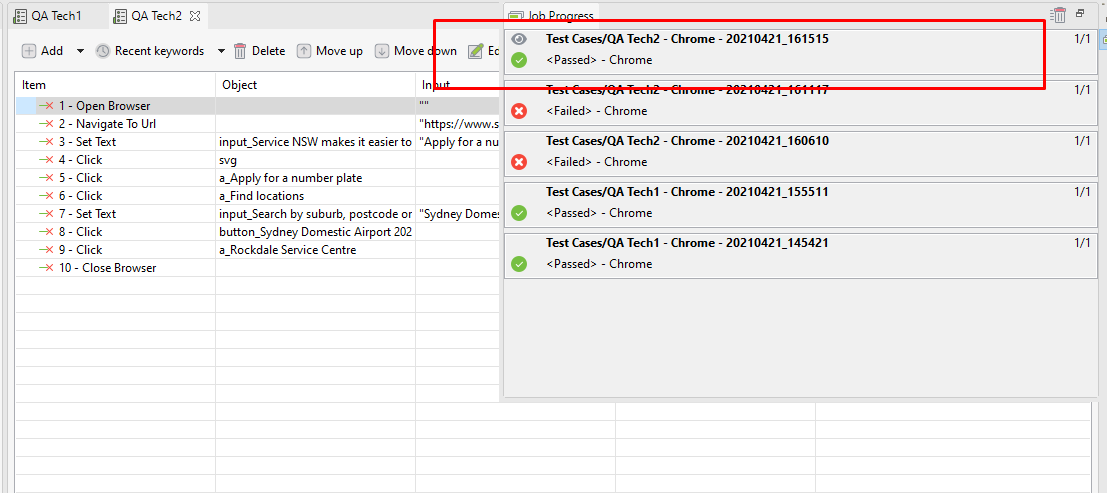


1. Navigate “Apply for number plate” & Enter suburb “Sydney Domestic Airport 2020”, “Rockdale Service Centre” (Using Katalon V7.7)









***API Automation***

Register to get the API token key - https://www.weatherbit.io/account/create

Navigate to **https://www.weatherbit.io/api/swaggerui/weather-api-v2#!/**andautomatebelowAPIs

**a. GET /current?lat={lat}&lon={lon}** for values {lat} as 40.730610 and {lon} as -73.935242

It should parse the response and get the value of the /data/state\_code

**b. GET /forecast/3hourly?postal\_code={postal\_code}**

It should parse the response and get the value of the { timestamp\_utc, weather} for all the data entries

***Bonus Points***

1. Write Performance tests for the API Testing where you hit both APIs  
• 10 times in a 1 minute duration  
• Fetch the test data from csv file multiple lat, long and another csv file for the post codes

Submit Performance Tests scripts in Github with proper readme file and all necessary steps.  
2. Provisioning UI Automation tests to run against local, headless browser [ set as configuration]