**Coding task for QA Automation Engineer @tradeling**

**The website to test**

1. The platform: [binance trading platform](https://www.binance.com/en)

**Tool Used: -**

* **Puppeteer** - Automation tool for scripting
* **Mocha** – Test Runner
* **Web Socket** - For the third scenario related to Web Socket, used Nodejs Web Socket library
* **Prettier** - Used for formatting the code
* **IDE** - Visual Studio Code

**Steps to Run Script: -**

**1.Node.js** should be installed in your system

2. From the project directory path, run below command in terminal

npm i puppeteer --save

**Example**: D:\puppeteer-tradling-project> npm i puppeteer --save

3. From the project directory path, again run one more command in terminal

npm run test

**Example**: D:\puppeteer-tradling-project> npm run test

**Acceptance Criteria: -**

**Scenario 1- Write a first end-to-end test that**

* 1. Starts on the [landing page](https://www.binance.com/en)
  2. Click the link [view more market](https://www.binance.com/en/markets)
  3. Clicks the ETH/BTC button leading to [Pair trading view](https://www.binance.com/en/trade/ETH_BTC)

**Expected Result**: - Verifies that all important information is loaded, e.g. trading chart, limit, market, stop limit input boxes, etc. (leaving this purposefully open to see what you think makes sense to test)

**Actual Result -** Verified that all important information is loaded successfully, CHART\_CONTAINER, EXCHANGE\_TAB, .MARGIN\_TAB, LIMIT\_TAB, MARKET\_TAB, STOP\_LIMIT\_TAB

1. Write a second end-to-end test that
   1. Starts on the [Pair trading view](https://www.binance.com/en/trade/ETH_BTC)
   2. Enters some valid values into the Buy ETH form
   3. Make sure the Total field has the correct value (Total = price \* Amount)
   4. Verifies that you can't submit the form since you are not logged in

**Expected Result: -**

* Verify that valid values is entered into the Buy ETH form.
* Verify that Amount is entered correctly (used amount = 50)
* Verify that user is not able to submit when user is not logged in.

**Actual Result: -**

* Verified that valid values is entered into the Buy ETH form.
* Verified that Amount is entered correctly. (used amount = 50)
* Verified that user is not able to submit when user is not logged in.

1. Write a data last end-to-end (WebSocket’s) test that
   1. Starts on the [Pair trading view](https://www.binance.com/en/trade/ETH_BTC)
   2. Under the hood, the UI uses WebSocket’s to fetch tickers
   3. Verifies that the data is loaded in the tickers,
   4. Verifies there is a consistent stream of data.
   5. Verifies the time it takes the socket connection to connect is less than 1 second

**Expected Result: -**

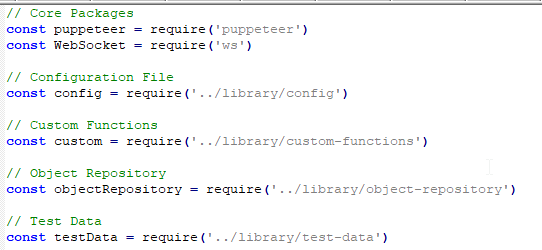
* 1. Verifies that the data is loaded in the tickers,
  2. Verifies there is a consistent stream of data.
  3. Verifies the time it takes the socket connection to connect is less than 1 second

**Actual Result: -**

* 1. Verified that the data is loaded in the tickers.
  2. Verified there is a consistent stream of data.
  3. Verified the time it takes the socket connection to connect is less than 1 second

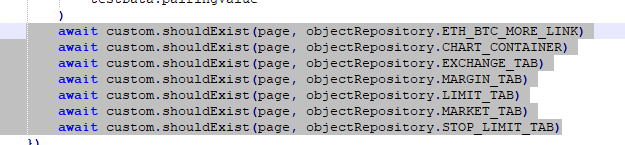
**Code explanation: -**

**Package, File, Custom Function, Test Data, Object Repository**

****

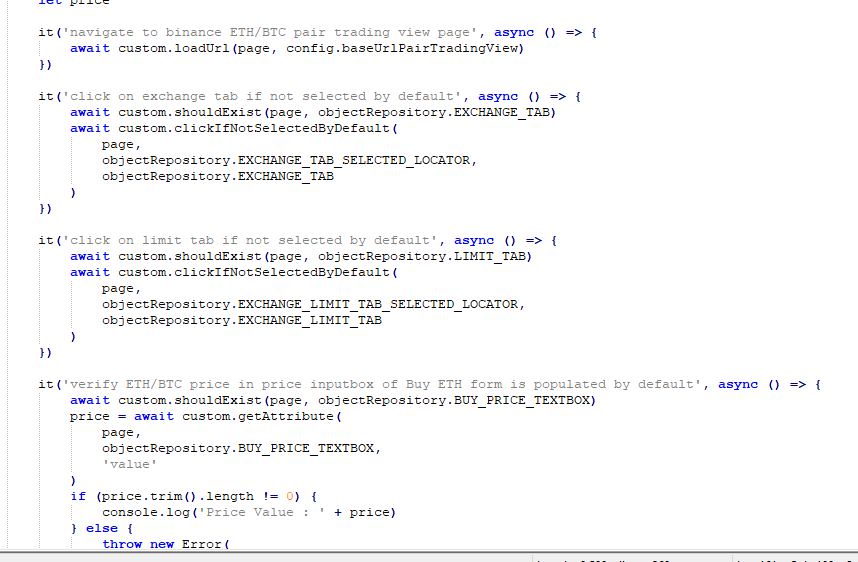
**Verified below information as per “First End-End Test”:**

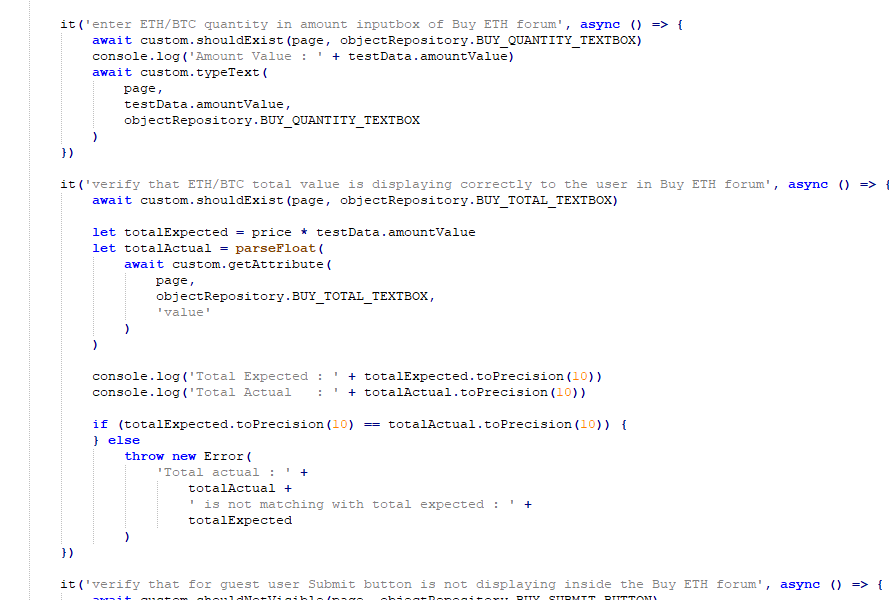
**CHART\_CONTAINER, EXCHANGE\_TAB, MARGIN TAB, LIMIT\_TAB, MARKET\_TAB, STOP\_LIMIT\_TAB**

****

**Verified below information as per “Second End-End Test”:**

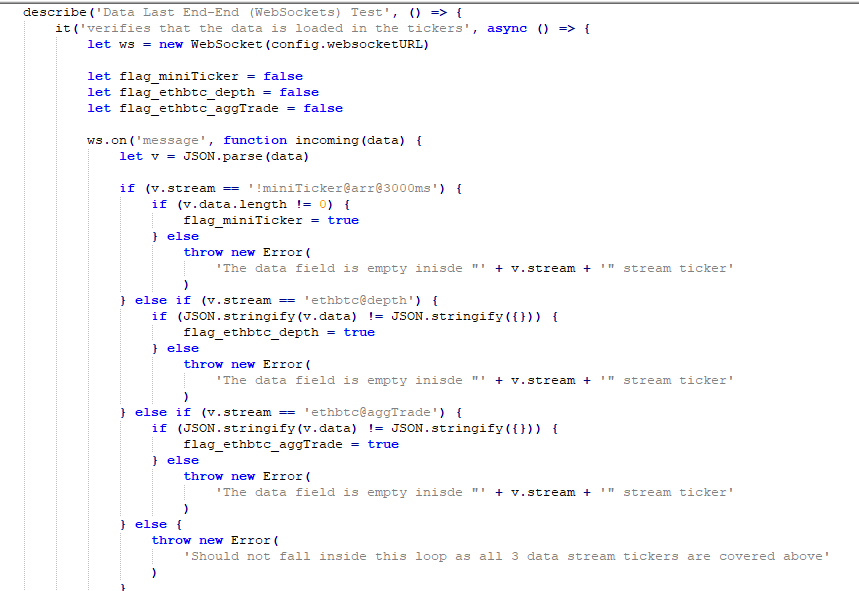
* Verify that ETH/BTC total value is displaying correctly to the user in Buy ETH forum
* Verify ETH/BTC price in price inputbox of Buy ETH form is populated by default
* Verify that for guest user Submit button is not displaying inside the Buy ETH forum
* Verify that for guest user Login and Registry Now links are displaying inside the Buy ETH forum

****

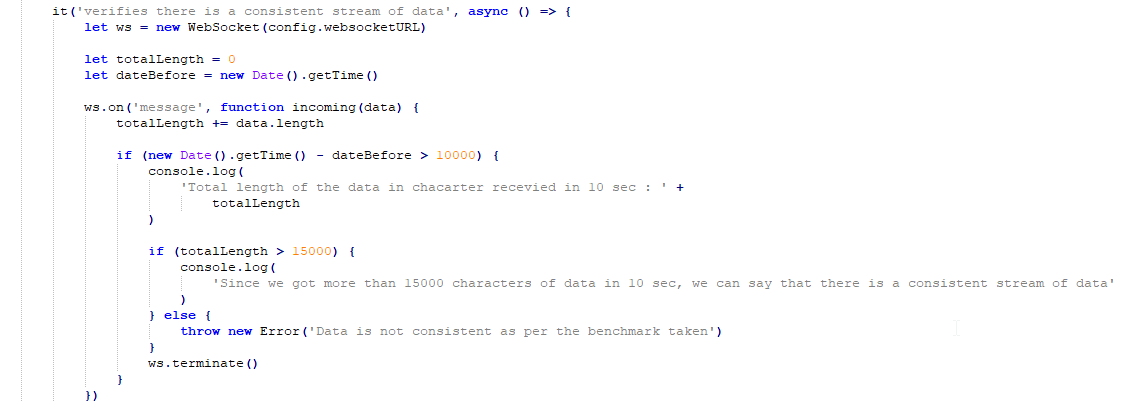
****

**Verified below information as per “Data Last End-End”:**

* Verifies that the data is loaded in the tickers



* Verifies there is a consistent stream of data
* Total length of the data in character received in 10 sec
* Since we got more than 15000 characters of data in 10 sec, we can say that there is a consistent stream of data

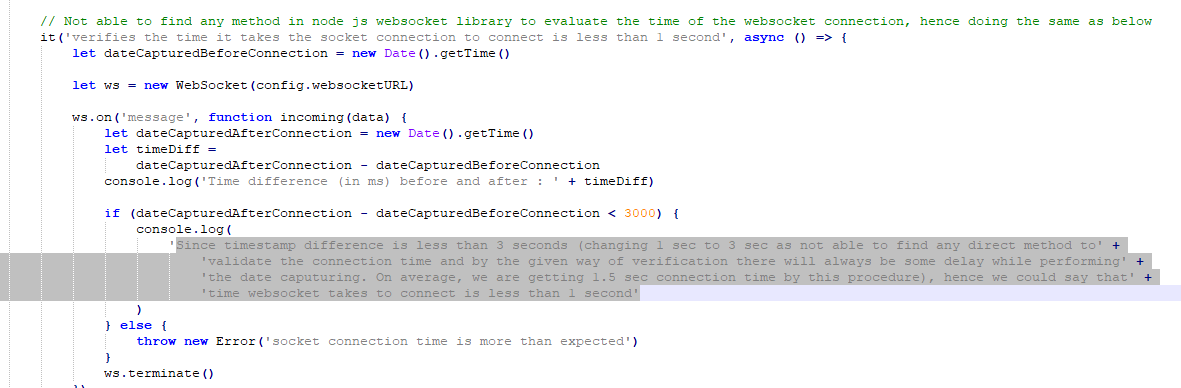
****

* Not able to find any method in NodeJS WebSocket library to evaluate the time of the WebSocket connection, hence doing the same as below
* Verifies the time it takes the socket connection to connect is less than 1 second
* Since timestamp difference is less than 3 seconds (changing 1 sec to 3 sec as not able to find any direct method to' +

'validate the connection time and by the given way of verification there will always be some delay while performing' +

'the date capturing. On average, we are getting 1.5 sec connection time by this procedure), hence we could say that' +

'time WebSocket takes to connect is less than 1 second'

****