Frequently Asked Questions

Friday, March 31, 2017   2:34 PM

1) **How to run Protractor Batch Run?**

   Commands to do Batch Run for Protractor Projects:

   FactSet Installer Needed Yes Test cases:

   C:/Tellus/Utils> ruby utilsdriver.rb Runtests args[0] args[1] args[2] args[3]   {Eg: ruby Utilsdriver.rb Runtests Live Production LargeSetCAccess default}

   args = [Staging (Devel/QA/Live/Prebuild etc.), HTML Reports Menu (Production/Staging/Devel), CAccess, default]

   we need to pass any of the set of options from the array to the above command.

   FactSet Installer needed No Test Cases:

   C:/Tellus> node RunProtractor.js

   To do Batch runs on VMs we need to follow below steps:
   a. **T Drive mapping:**
      net use T: \\tellusdevb01.pc.factset.com\vms\appqa_vms\FDSQAR_C-<serial number>   [Ideal way is to use Tellus Serials only]

   b. **FactSet Installation:**
      msiexec.exe /i C:\temp\FactSet_Setup_2016_2A_x64.msi ALLUSERS=1 /qb FACTSET_FACTSETDIR="T:\FactSet"

      Here Assuming FactSet msi file is located at C:/temp hence gave the path as "C:/temp/__.msi". You can give the file location where it is stored.

   c. copy Gems and gem_setup.rb file from **\\fpeadminb01\Data\QAAuto** [Exclusively for FactSet Installer Needed **Yes** Test Cases]

   d. Now use the above mentioned commands to run the scripts.

2) **Why Protractor and not Selenium for Angular Applications:**
   Let us take an example of automating a web browser using selenium web driver.

   Verify Page Title: The code will look like:

   public static void main(String[] args)

   {

       // Create a new instance of the Firefox driver

       WebDriver driver = new FirefoxDriver();

       // (1) Go to a page

       driver.get("http://www.google.com");

       // (2) Locate an element

       WebElement element = driver.findElement(By.name("q"));

       // (3-1) Enter something to search for

       element.sendKeys("Purdue Univeristy");

       // (3-2) Now submit the form. WebDriver will find the form for us from the element

       element.submit();

       // (3-3) Wait up to 10 seconds for a condition

       WebDriverWait waiting = new WebDriverWait(driver, 10);

       waiting.until(ExpectedConditions.presenceOfElementLocated(By.id("pnnext")));

       // (4) Check the title of the page

       if(driver.getTitle().equals("purdue univeristy - Google Search"))

         System.out.println("PASS");

       else

         System.err.println("FAIL");

       //Close the browser

       driver.quit();

   }

   If you see the above code, you can observe that there are lot of the waits in the middle of the code. This is because selenium web driver does not wait until it receives a response. So the script will contain lots of waits and sleeps.

   Advantage Of using Protractor is: Protractor by default wait until it receive a response from the http requests and will avoid sleeps in the test cases.

   The above code with protractor looks like:

       it('Testing the Title',function(){

         browser.get('www.google.com');

         var ref = element(by.name('q'));

         ref.sendKeys('Purdue Univeristy');

ref.submit();

expect(browser.getTitile()).toEqual('purdue univeristy - Google Search');

});

In the above code, Protractor will wait untill it receives response from each request.

One more Advantage is: If your application wanted to automate is pure Angular application, Protractor provides some new locator strategies and functions which are very helpful to automate the AngularJS application waitForAngular, By.binding, By.repeater, By.textarea, By.model, WebElement.all, WebElement.evaluate, etc. and all the above are not provided by Selnium.

3) The very basic question is difference between **isPresent**() and **isDisplayed**() -
   ○ **isPresent**()   - returns **true** if element reference is present in the Dom.
                            - returns **false** if element reference is not present in the Dom.
   ○ **isDisplayed**() - returns **true** if element that is present in the Dom is Displayed on the browser.
                            - returns **false** if element reference is present in the Dom and not displayed on the browser.
                            - throws an **Error** if element reference is not present in the Dom.

4) Difference between **getText**() and **getAttribute**('value') -
   ○ getText() - To get the visible **innerText** of an element including sub-elements without any leading or trailing whitespaces. Example includes getting the text of the drop downs or buttons.
   ○ getAttribute('value')- If you observe any of the input element you do not see any text that you entered in the input. The text of the input is stored in value attribute of the input tag. So, you need to use getAttribute('value').

5) **Getting Inner and Outer Html code:** We have been using **getInnerHtml**() and **getOuterHtml**() functions in some cases if we cannot able to see the elements in the DOM. But, with the latest version of protractor they are not working as expected as they are depreciated but you can access them as:
   eleRef.getAttribute('innerHTML'), getAttribute('outerHTML').

6) **by.js:**
   ○ If any of your applications have checkbox / radio buttons in the slick grid and if you are using slick grid api to get the cell nodes and as tf-checkbox will be the child element to the cell node you can use element(by.js(expression)) which is same like using **webdriver.WebDriver.executeScript.**
   ○ One advantage of by.js over execute script is we can get the reference of the cell node and can search the child elements of it. The code snippet I used is:
```
var ele = element( by.js( function( rowIndex, colIndex ) {
    var slickObject = $( '.tf-slick-grid' ).data( '$tfSlickGridController' );
    var parentID = '', newId = '';
    return slickObject.grid.getCellNode( rowIndex, colIndex );
}, rowIndex, colIndex ));
Utilities.scrollElementToVisibility( ele );
return ele.element( by.xpath( './/tf-checkbox-control' ) );
```
   ○ You can also go through the Protractor API documentation for this: http://www.protractortest.org/#/api?view=webdriver.By.js

7) **browser.ignoreSynchronization** Uses:
   ○ If you wanted to run the scripts without waiting for any ongoing $http requests to finish, then you can set this as browser.ignoreSynchronization = true.
   For example: If you have a step to verify if Loading icon is appeared and if you cannot handle through automation if it finishes loading in very less time, you can make browser.ignoreSynchronization = true in the preceding **it block** of the verification step and make browser.ignoreSynchronization = false after the verification **it block**. Example Code here is:
```
it('Should click on "Done" button',function(){
    EditTileOptions.getButton('Done').click();

    //Ignoring synchronization
    browser.ignoreSynchronization=true;
});

it('Verifying if "Factors" tile starts calculating',function(){

    //Should wait until loading icon appears
    Utilities.waitUntilElementAppears(AT4MainPage.getLoadingIcon('Factors','Factors'),12000);

    //Verifying if Loading icon appears
    expect(AT4MainPage.getLoadingIcon('Factors','Factors').isPresent()).toBeTruthy();
});

it('Verifying if "Factors" report is blank',function(){
    browser.ignoreSynchronization=false;

    AT4MainPage.isTileDataDisplayed().then(function(){
        expect(false).customError('"Factors" report is calculated and is not blank');
        CommonFunctions.takeScreenShot();
    },function(){});
});
```
   If you observe the above code you can see that `Utilities.waitUntilElementAppears(AT4MainPage.getLoadingIcon('Factors','Factors'),12000);` is used this is because as you are making ignoreSynchronization to true it will not even for Loading icon to appear. So used waitUntillElementAppears for the loading icon to appear.
   ○ An alternative way of using browser.ignoreSynchronization is to access the standard webdriver API directly like:

   expect(browser.driver.findElement(by.css('.message')).getText().toBe('Loading...');

   And if you have any case of accessing the elements in the non-angular page you can use this. One example here is you can observe in **loginToApplication()** in **factset-login-page.po** while entering the credentials to login page.

8) Dealing with element References:

   Let us discuss through an example:

   CustomFractilingDialog.prototype.getCell = function(rowName, colName) {
       this.getColumnIndex(colName).then(function(colIndex) {
        return element(by.xpath('//*[@id="custom-fractiles"]//*[contains(@class, "slick-row") and descendant::*' +
        '[normalize-space(.)="'+rowName+'"]]//*[contains(@class, "slick-cell l'+colIndex+' r'+colIndex +'")]'));
       });
     };

   this code does not return anything because the return is just to the **getColumnIndex** function but not the whole function which results as no return to the actual getCell().

   So in this case we need to do as:

   CustomFractilingDialog.prototype.getCell = function(rowName, colName) {
       return this.getColumnIndex(colName).then(function(colIndex) {
        return element(by.xpath('//*[@id="custom-fractiles"]//*[contains(@class, "slick-row") and descendant::*' +
        '[normalize-space(.)="'+rowName+'"]]//*[contains(@class, "slick-cell l'+colIndex+' r'+colIndex +'")]'));
       });
     };

   then the above code will return the value to actual getCell() function.

9) The scenario where to return the values and where to use promises? Fulfilling and rejecting?
   For Example:

   CustomFractilingDialog.prototype.getAllColumns = function() {

```
var arrColumns = [];
element.all( by.xpath( '//*[@id="custom-fractiles"]//*[contains(@class,"slick-header-column") and @title]' ) ).
    map( function( eleRefs, index ) {
        return {
            eleText: eleRefs.getText(),
            index: index
        };
    } ).then( function( arrElements ) {
        var arrColumns = [];
        for ( var i = 0; i < arrElements.length; i++ ) {
            arrColumns.push( arrElements[ i ].eleText );
        }
    } );

return arrColumns ;
};
```

The above code will return an empty array as Protractor execution is synchronous, it do not return arrColumns[] after the elements are pushed. So for this we can write as

```
CustomFractilingDialog.prototype.getAllColumns = function() {
    var defer = protractor.promise.defer();
    var promise = defer.promise;

    element.all( by.xpath( '//*[@id="custom-fractiles"]//*[contains(@class,"slick-header-column") and @title]' ) ).
        map( function( eleRefs, index ) {
            return {
                eleText: eleRefs.getText(),
                index: index
            };
        } ).then( function( arrElements ) {
            var arrColumns = [];
            for ( var i = 0; i < arrElements.length; i++ ) {
                arrColumns.push( arrElements[ i ].eleText );
            }
            defer.fulfill( arrColumns );
        } );

    return promise;
};
```

Note: While dealing with Deferred objects, if you once fulfill the promise we cannot overwrite/fulfill the promise again in the same function.

10) Ways to **Clear the Input Value**:
   a. inputRef.clear();
   b. Input.sendKeys(**webdriver**.Key.chord(**webdriver**.Key.CONTROL,**'a'**,**webdriver**.Key.DELETE),**webdriver**.Key.ENTER);
   c. Low Priority: Function clear(elem, length){
```
        length = length || 100;
        var backspaces = '';
            For ( var I = 0; I < length; i++ ) {
                backspaces += protractor.Key.BACK_SPACE;
            }
            Elem.sendKeys(backspaces);
        }
```
11) Select multiple elements using Control Click:
   a. Holding the Control Key: browser.actions().keyDown(protractor.Key.CONTROL).perform();
   b. Select a single element or loop through elements to click. Ex: element.click().
   c. Releasing the Control Key: browser.actions().keyUp(protractor.Key.CONTROL).perform();

   Example: Please don't worry about the function provided here. Following is the example provided just for reference:

```
        it('Should select "Sector Number" from the available section',function(){
            EditDocumentRiskModel.getElement('Available','Sector Number').click();
        });

        it('Should verify Sector Number is highlighted from the Available section',function(){
            EditDocumentRiskModel.getElement('Available','Sector Number').getAttribute('class').then(function(value){
                if(value.indexOf('selected')===-1){
                    expect(false).customError('"Sector Number" is not highlighted from the Available section');
                    CommonFunctions.takeScreenShot();
                }
            });
        });

        it('Should hold ctrl key select "Beta Factor1" from the available section',function(){
            // Holding the Control Key
            browser.actions().keyDown(protractor.Key.CONTROL).perform();

            EditDocumentRiskModel.getElement('Available','Beta Factor1').click();

            //Releasingthecontrolkey
            browser.actions().keyUp(protractor.Key.CONTROL).perform();
        });
```
   Additional Information: keyDown() can only work for modifier keys such as Alt, SHIFT, Control. For the rest use sendKeys().

12) Right Click on Element Reference:
   a. browser.actions().click(eleRef,protractor.Button.RIGHT).perform();

```
   b. browser.actions().mouseMove(eleRef).perform();//Move the mouse over the element
      browser.actions().click(protractor.Button.RIGHT).perform();//Perform right click operation
```

13) To get the CSS Attributes: element.getCssValue(**'background-color'**) you can use color, font-weight, text-decoration

14) In few application there are few scenarios where if we enter value to input as 'aa'. It will show in UI as 'AA' but if we use input.getAttribute('value') it returns 'aa'. So to get the validated value we should do as:
```
   1. Return browser.executeScript(function( sectionName ) {
          Return angular.element(document.evaluate('//*[@id="factors-tile-content"]//tf-accordion//tf-accordion-section[@section-title="'+sectionName+'"]//*[@data-qa-class="input-formula"]',document,null,XPathResult.FIRST_ORDERED_NODE_TYPE,null).singleNodeValue).scope().$ctrl.formula;
      },sectionName);

      Here $ctrl.formula is the value of ng-model attribute which can be observed in the Dom of the input reference
```
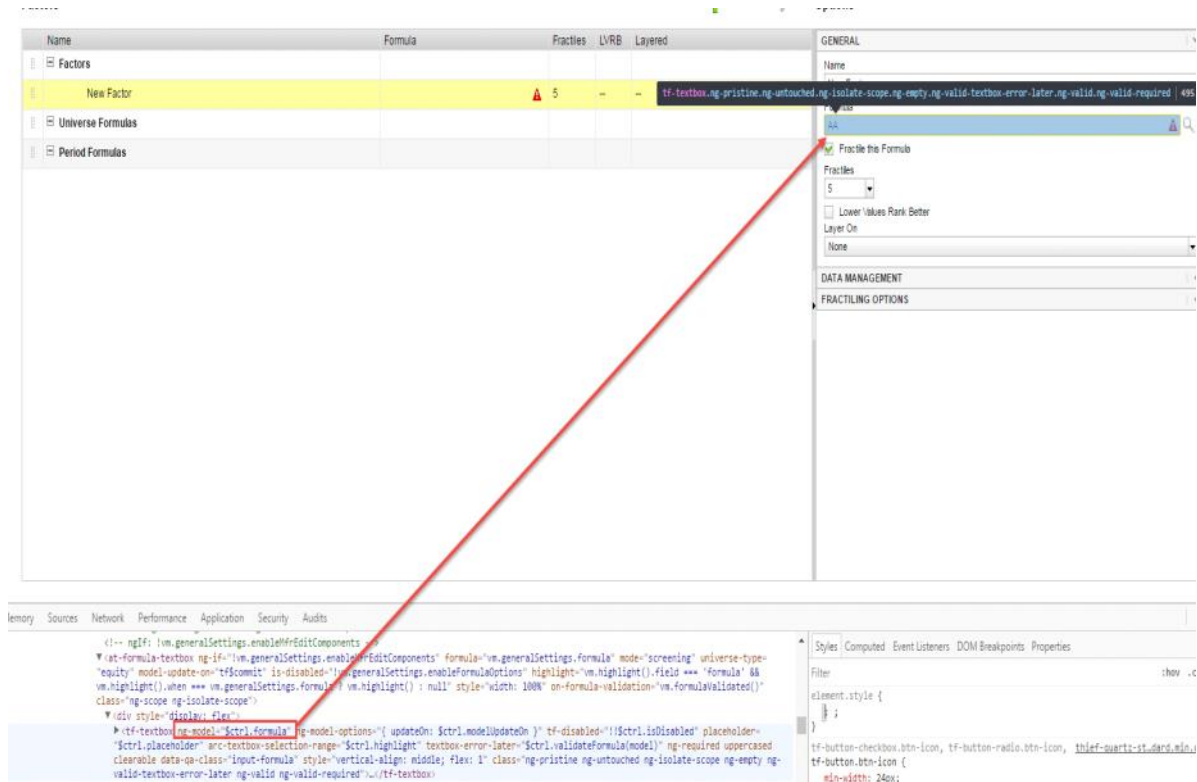
2. it('Verifying if Period Formula text area contains "AA"', function() {
        function getModelValue(options) {
            return browser.executeScript(function(options) {
                return _.get($(options.selector).scope(), options.modelPath);
            }, options);
        }

        // to check the period formula:
        getModelValue({
            selector: '*[data-qa-id="input-period-formula"]',
            modelPath: '$ctrl.columnDefinition.periodFormula' }).then(function(value) {
            if (value !== 'AA') {
                expect(false).customError('Error: Period Formula text area contains "'+value+'"');
                CommonFunctions.takeScreenShot();
            }

        });
    });

3. **Fix for "Failed: webdriver is not defined"**
   We are now unable to access Keys using **webdriver** object instead we should use **protractor** object to access KEY's
   We need to update **webdriver** with **protractor**.

   A quick fix for this can be adding
   webdriver = protractor; in index.js or all-page-objects.js

15) A List of Protractor Keys List Link: <u>Keys</u>

Changes to be done after upgrading to Protractor v5.x

1. Update path to **chromedriver.exe** as this binary file is not present at webdriver-**manager/selenium** location.
2. As the version of **jasmine-spec-reporter** is updated you have to change the way you require the same in cong.js file.
   Now you have to update with this: **SpecReporter = require('jasmine-spec-reporter').SpecReporter**.
3. Change the way we inject cookies to this: **browser.manage().addCookie({name:'LimaOverrideCookie',
value:overrideValue, ":", domain:domain});**

From <http://is.factset.com/rpd/Summary.aspx?messageId=30598134>

Boiler Plate Code for Protractor projects

Thursday, February 08, 2018   2:14 PM

Subject	Boiler Plate Code for Protractor projects
From	Pramod Devi
To	QAAutomationTeam; Team - Analytics QA Automation; Vijay Kumar Rojra
Cc	QAAutomation TLs
Sent	Wednesday, February 07, 2018 8:15 PM

Hello Team,

I've created boiler code template for Protractor scripts. You can find the code at below location.

Location: H:\Department\QualityAssurance\QAAutomation\Automation\Boiler Plate\Protractor Boiler.

For this we need to install third part local server like Xampp to run the project and place the code in the server directory.

Step to download Boilerplate code for test case:
- Navigate to url: http://localhost/protractorboiler/test.php (example url).
- Click on 'Choose File' and select the excel file (i.e. excel file of required test case downloaded from the justifier).
- Select the optional fields.
- Click on '**Get Spec'** button.
- Click on '**Click here to download**' link.

Please let me know if you have any issue.

Thanks,
Pramod Devi
QA Automation Associate
FactSet Systems India Pvt. Ltd.
M: +918686861760
pdevi@factset.com
www.factset.com

FACTSET 〉 SEE THE ADVANTAGE

FINANCIAL DATA | ANALYTICS | TECHNOLOGY | SERVICES

R&D, Findings

Using isChecked() when checkbox is not available will never fail.

Subject	**Using isChecked() when checkbox is not available will never fail.**
From	Vasavi Tallada
To	Analytics QA Automation
Cc	QAAutomation TLs
Sent	Friday, March 31, 2017 1:08 PM

Hi All,

When we are using thief helpers, if there is a scenario to verify if the checkbox is checked, usually we will verify it like this

```
ThiefHelpers.getCheckBoxClassReference('Multi-Lot Actual Weights').isChecked().then(function(flag) {
  if (!flag) {
   expect(false).customError('"Multi-Lot Actual Weights" check box is  unchecked');
    CommonFunctions.takeScreenShot();
  }
});
```

```
Even If there is no checkbox with the specified name the test step will never fail because the thief helpers are using isPresent() to check the class attribute.
```

```
isChecked() {
      return this._checkbox.element(by.css('tf-checkbox-control.tf-checkbox-checked')).isPresent();
    }
```

If you want to log error in scenarios like this verify the element presence first and then the status of it. Please see the below code for reference.

```
ThiefHelpers.getCheckBoxClassReference('Multi-Lot Actual Weights').getElement().isDisplayed().then(function(bool) {
  if (bool) {
    ThiefHelpers.getCheckBoxClassReference('Multi-Lot Actual Weights').isChecked().then(function(flag) {
      if (!flag) {
        expect(false).customError('"Multi-Lot Actual Weights" check box is unchecked');
        CommonFunctions.takeScreenShot();
      }
    });
  } else {
    expect(false).customError('"Multi-Lot Actual Weights" checkbox is not present in Risk accordion');
    CommonFunctions.takeScreenShot();
  }
});
```

Thanks,
Vasavi.

Browser.call()-Murali

1. If you have a scenario where you need to run a part of code after forEach you can do the following.

```
Browser.call(function(){
        forEach();
}).then(function(){
        //Dependent code
});
```

2. We know that exception handling in protractor is challenging because of its asynchronous nature. We can use Browser.call() handle exceptions as it captures the response from a method call and wraps it in an promise. So in case any exception is thrown then it will be captured in the promise and handled in rejection statement

```
Browser.call(function(){
    //webDriverJS/Javascript call
}).then(function(){
    // Code to be performed if call was success
},function(){
    //Exception thrown from the call will be handled here.
});
```

Capturing CPU Usage in the script - Santhoshi

Tuesday, March 07, 2017   12:47 AM

Subject	**Capturing CPU Usage in the script**
From	Santhoshi Chepuru
To	Analytics Automation
Sent	Monday, March 06, 2017 9:42 PM

Hi All,

We can capture CPU Usage in the script. We have an npm library with name '**windows-cpu**' you can now install the library and can use it. We can also capture Chrome Load on CPU, node load on CPU etc.

We need to discuss regarding where to implement this. I feel, this will be very helpful specially while testing the Charts test cases because all the components are responding with some delays and we can see many errors during CRON runs.

My idea here is, can we capture this in takeScreenShot() method so that we can print as a log message. Needs to discuss on this.

Sample Code:

```javascript
var cpu = require('windows-cpu');

describe('Performance Test', function() {

  it('CPU Test', function(done) {
    cpu.totalLoad(function(err,res) {
      if (err) {
        console.log('Error' + err);
      }

      console.log('CPU usage is ' + res + '%');
      done();
    });
  });

  it('Node Test', function(done) {
    cpu.nodeLoad(function(err,res) {
      console.log('Node is using ' + res.load + '% of total CPU Usage');
      console.log('Results Found PID' + res.found[0].pid);
      console.log('Results Found Process' + res.found[0].node);
      console.log('Results Found Load' + res.found[0].load);
      if (err) {
        console.log('Error' + err);
      }

      done();
    });
  });

  it('Chrome Test', function(done) {
    cpu.findLoad('chrome', function(err,res) {
      console.log('Chrome is using ' + res.load + '% of total CPU Usage');
      var arr = res.found;
      for (var i = 0; i < arr.length; i++) {
        console.log('pid: ' + arr[i].pid + '-- Process: ' + arr[i].process + '-- Load:
```

```
' + arr[i].load);
        }

      if (err) {
        console.log('Error' + err);
      }

      done();
    });
  });

});
```

Thanks,
Santhoshi.

```
' + arr[i].load);
        }

      if (err) {
        console.log('Error' + err);
```

Utility to enable input box in a grid - Charuvaka

Wednesday, March 08, 2017   8:36 PM

Hi All,

We have been using references of the cell nodes to double click/click and enable the input box in a grid. Slick grid API facilitates us with a function **gotoCell** to enable the input box by taking Row Index and Col Index as parameters. Please have a look at this Snippet.

```
/**
* @function goToCell
* @description This function is Enable Cell Node
* @param {number} rowIndex Index of the Row
* @param {number} colIndex Index of the Column
* @param {bool} forceEdit to double click and enable Input Box.
*/
UtilitiesSlickgrid.prototype.goToCell = function ( rowIndex, colIndex, forceEdit ) {

    browser.driver.executeScript( function ( colIndex, rowIndex, forceEdit ) {
        var row = rowIndex;
        var col = colIndex;
        var forceEdit = forceEdit;
        var slickGridObject = $('.tf-slick-grid').data( '$tfSlickGridController' );
        slickGridObject.grid.gotoCell( row, col, forceEdit );
    }, colIndex, rowIndex, forceEdit );

};
```

Regards,
Charvaka.

Disable Remember Password dialog

Thursday, March 30, 2017   5:20 PM

Subject	**Disable Remember Password dialog**
From	Santhoshi Chepuru
To	Analytics Automation
Sent	Tuesday, March 28, 2017 6:23 PM

Hi All,

With the new version of chrome- 57.x We are seeing Remember Password dialog appearing through automation runs. In order to disable it please add below in the capabilities.

```
capabilities: {
    'browserName': 'chrome',
    'chromeOptions': {
        'args': ['start-maximized', 'disable-extensions'],
        'prefs': {
            'credentials_enable_service': false,
            'profile': {
                'password_manager_enabled': false
            }
        }
    }
},
```

Thanks,
Santhoshi Chepuru
Automation Analyst II, QA Automation
FactSet Systems India Pvt. Ltd.
csanthoshi@factset.com
www.factset.com

FACTSET) SEE THE ADVANTAGE   FINANCIAL DATA | ANALYTICS | TECHNOLOGY | SERVICES

Solving Mouse Hover issues using jquery - Santhoshi.

- In a few of our applications sometime the hovering function -
  browser.actions().mouseMove(eleRef).perform() do not hover the element when the physical pointer is on the automation window. So, to solve this use

```
browser.driver.executeScript('$("tf-infobox tf-spotlight a").mouseover();'); -jquery function for hovering.
The equivalent xpath for this is '//tf-infobox//tf-spotlight//a'.
```

Accessing text inside execute script

Tuesday, April 04, 2017   10:34 PM

Subject	**Accessing text inside execute script**
From	Santhoshi Chepuru
To	Analytics Automation
Sent	Tuesday, April 04, 2017 9:05 PM

Hi All,

While working with slick grid API if we want to get the text using
getCellNode() function inside **browser.driver.executeScript** function we cannot use
getText()inside execute script block since it is protractor function. So, if you want
to get the value from **getCellNode**() inside execute script function use
slickObject.**grid**.**getCellNode(i, j)**.**innerText**
Using this we can able to access the text for the element.

```
browser.driver.executeScript(function() {
  var slickObject, value, col, columnData = [];
  slickObject = $('.tf-slick-grid').data('$tfSlickGridController');
  for (var i = 1; i < 4; i++) {
    for (var j = 1; j < 4; j++) {
      columnData.push({rowIndex: i, cellValue: slickObject.grid.getCellNode(i, j).inne
rText});
    }
  }
  return columnData;
}).then(function(colData) {
  console.log(colData);
});
```

Result is: [Object { cellValue: '1.01', rowIndex: 1 },
  Object { cellValue: '1.53', rowIndex: 1 },
  Object { cellValue: '-0.51', rowIndex: 1 },
  Object { cellValue: '2.94', rowIndex: 2 },
  Object { cellValue: '3.00', rowIndex: 2 },
  Object { cellValue: '-0.06', rowIndex: 2 },
  Object { cellValue: '1.92', rowIndex: 3 },
  Object { cellValue: '2.10', rowIndex: 3 },
  Object { cellValue: '-0.18', rowIndex: 3 }]

Thanks,
Santhoshi Chepuru
Automation Analyst II, QA Automation
FactSet Systems India Pvt. Ltd.
csanthoshi@factset.com
www.factset.com

Function to skip it/describe blocks on the run

Monday, April 17, 2017   10:10 PM

Subject	**Function to skip it/describe blocks on the run**
From	Murali Krishna Thototta Govindan
To	Avanthi Rentala; Bhupender Agarwal; Deekshith kumar Lingoju; Eyme Ann Mathew; Mahendra Tribhuvan Sura; Mamatha Jyothi Akula; Priyanka Muddalapuram; Vasu Nakka
Sent	Saturday, November 12, 2016 12:32 AM
Attachments	jasmineOve ride

Hello,

        Attached to this mail is a js file which can allow protractor runs to skip any given it/describe blocks on a given condition. This method overrides the existing it/describe block of the jasmine with new parameter named 'disabled'. When last variable is set to true, corresponding it/describe block will be skipped. So we can create a flag on a step and use this flag as 3rd parameter to the dependent step. So in this way if the flag value is 'true' then that step will be skipped with added message on the log.
Usage:

1. Add 'require('<path>/jasmineOveride.js')' in onPrepare() function.
2. Call to the it/describe block is as follows:

```
        Describe/it(<string>,<function()>,<disabled flag>);

        Eg: describe("This is a sample test",function(){
                Var bFlag=false;
                    It("This step is executed",function(){
                            //Execution steps
                            bFlag=true;
                    },false);
                    It("This step will be skipped",function(){
                            //This will be skipped
                    },bFlag);

            },false)
```

Note: This method is still bare, we can make tweaks to make it efficient .
        Please let me know your thoughts.

-Murali

Creating multiple XML file for every run.

Wednesday, April 19, 2017   6:50 PM

Subject	**Creating multiple XML file for every run.**
From	Komal Agrawal
To	AppQA Hyderabad Automation Team
Cc	Mahendra Tribhuvan Sura; Bhupender Agarwal
Sent	Wednesday, April 19, 2017 6:29 PM

Hi Team,

If you have a requirement to create multiple log XML files for every test run then please add the highlighted code to "onPrepare function" of conf.js file

```
onPrepare: function() {
  var date = new Date();

  //Reading the current timestamp
  var time = date.getDate() + '' + date.getHours() + '' + date.getMinutes() + '' + date.getSeconds() + '' +
date.getMilliseconds();
  var jasmineReporters = require('jasmine-reporters');
  var junitReporter = new jasmineReporters.JUnitXmlReporter({
    savePath: __dirname + '\\Logs',
    consolidateAll: true,

    //concatenating the timestamp with the file name
    filePrefix: 'junitresults-' + time,
  });
  jasmine.getEnv().addReporter(junitReporter);

  CommonFunctions = require(__dirname + '/PageObjects/common-files/common-functions.po');
  CommonFunctions.setCookieAndNavigateTo('factset.com');

  // Add module to give more verbose output
  var SpecReporter = require('jasmine-spec-reporter');
  jasmine.getEnv().addReporter(new SpecReporter({displayStacktrace: 'all'}));

},
```

And also if you want to run the scripts multiple times then create a node file with the following code and run it in the command prompt of your project location.
Example:- C:/Tellus> node run.js

run.js:-

```
var fs = require('fs');
var exec = require('child_process').exec;
try {
RunProtractor();
}
```

```
catch(e){
console.log(e);
}

function RunProtractor(){
exec(__dirname+'/../node_modules/.bin/protractor '+__dirname+'/conf.js',{maxBuffer : 2000 * 1024},
function (error, stdout, stderr) {
  if (error !== null) {
   fs.appendFile(__dirname+'/NodeModuleInstallationLog.txt','exec error while running protractor
conf.js:'+error+'\n', function(err){
    if( err ){
     console.log( 'While copying error occured protractor conf.js command execution'+ err);
    }
   } );
  }
  if (stderr !== null && stderr !== "") {
   fs.appendFile(__dirname+'/NodeModuleInstallationLog.txt','stderr while running protractor conf.js
command'+stderr+'\n', function(err){
    if( err ){
     console.log( 'while copying stderr occured when executing protractor conf.js'+ err);
    }
   } );
  }
  RunProtractor();
} );
}

Thanks,
Komal
```

Module to identify if given date is holiday

Wednesday, April 19, 2017   6:50 PM

Subject	**Module to identify if given date is holiday**
From	V N Kiranmai Malladi
To	AppQA Hyderabad Automation Team
Cc	QAAutomation TLs
Sent	Wednesday, April 19, 2017 6:47 PM

Hi Team,

I found a module which provides dates of holidays for various **countries, states and region's** by type.

Module Name: **date-holidays**

How to install? : npm install date-holidays

Website link: https://www.npmjs.com/package/date-holidays

Usage:
```
var Holidays = require('date-holidays');
var hd = new Holidays('US','nj','va');

US: United States of America
nj: New Jersey
va: Virginia
```

1) Checking if 14th April 2017 is a holiday or not.

```
var date = new Date(' 2017-04-14 10:00:00 UTC+05:30');
var isHoliday = hd.isHoliday(date);
console.log(isHoliday);
```
isHoliday -> Will return an **object** consisting of date, start, end, name and type.

2) Checking if 16th April 2017 is a holiday

```
var date = new Date(' 2017-04-16 10:00:00 GMT-0500');
var isHoliday = hd.isHoliday(date);
console.log(isHoliday);
```
isHoliday -> Will return **false,** as Weekends are not listed in the Holidays

3) Checking if 16th April 2017 is a holiday

```
var date = new Date(' 2017-04-16 10:00:00 GMT-0500');
var isHoliday = hd.isHoliday(date);
console.log(isHoliday);
```
isHoliday -> Will return **false,** as Weekdays are not listed in the Holidays

NOTE: This module return **object** only if the given date is an holiday in the given country/state/regions. For weekends/weekdays it returns false.

I am sure this module will help us automate scenarios related to date more precisely now.

Thank you,
V N Kiranmai Malladi
QA Automation Analyst
FactSet Systems India Pvt. Ltd.
Vnkiranmai.malladi@factset.com
www.factset.com

Regular expression is to used as parameter in thief grid helpers

Thursday, April 27, 2017   6:05 PM

Subject	**Regular expression is to used as parameter in thief grid helpers**
From	Komal Agrawal
To	AppQA Hyderabad Automation Team
Cc	Mahendra Tribhuvan Sura; Deekshith kumar Lingoju; Bhupender Agarwal; Eyme Ann Mathew
Sent	Thursday, April 27, 2017 6:01 PM

Hi Team,

If you get a chance to work on thief grid and you are not able to access a particular column whose label is combination of special character and alphabet. It's because the thief grid helper functions accepts a pattern for matching.

https://gitlab.factset.com/thief/angular-grid/blob/master/src/grid/grid-helper.js#L63

and if we pass a string which is combination of special character and alphabets it is used to construct a **RegExp.** That is why it gets failed to access.
So construct a RegExp and pass it, so the function receives appropriate pattern match.

Example: column Name :-   Fee (BPS)
          RegExp to be passed:- Fee \\(BPS\\)

For more details on regular expression visit this link https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/RegExp

Thanks,
Komal

Creating a reproducer and Information needed to file gilab issue for thief team

Wednesday, May 10, 2017   11:15 PM

Subject	Creating a reproducer
From	Santhoshi Chepuru
To	AppQA Hyderabad Automation Team
Sent	Wednesday, May 10, 2017 5:44 PM

Hi Team,
Following are the instructions to be followed to create a reproducer while creating issues in thief-helpers library.

1) The FactSet link for creating a reproducer is : http://jsbin.factset.com
2) The API to explore the thief components is http://unixdeva16.factset.com/~fwi/thief-angular/versions/1.31.0/docs/#/api/thief/css
3)

The Sample structure for creating a reproducer is:

```
<!DOCTYPE html>
<html>
<head>
<script src="//cdn.factset.com/jquery/2.1.1/jquery.js"></script>
<script src="//cdn.factset.com/lodash/3.10.1/lodash.js"></script>
<script src="//cdn.factset.com/moment/2.13.0/moment.js"></script>
<script src="//cdn.factset.com/angular/1.5.11/angular.js"></script>
<script src="//cdn.factset.com/angular-animate/1.5.11/angular-animate.js"></script>
<script src="http://unixdeva16.factset.com/~fwi/thief-angular/versions/curr/dist/js/thief-angular.js">
</script>  //This link contains all the controller code
<link href="http://unixdeva16.factset.com/~fwi/thief-angular/versions/curr/dist/css/thief-angular-quartz.css"
rel="stylesheet" type="text/css" />
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width">
 <title>JS Bin</title>
</head>
<body ng-app="thief" class="ng-scope"> // Here we are writing ng-app=thief as the thief-angular.js controller code is
with the name thief and you can verify in thief-angular.js

Sample code here

</body>
</html>
```

All you need to do for creating a new reproducer is, check the sample thief component code that is provided in the API and attach in the body tag, run it and verify in the console. We don't need to write any controller code for the basic functionalities like clicking the element, collapsing/expanding the element because everything is embedded in thief-angular.js file. After you create a reproducer you need to confirm with your app Engineers if that is correct. But, there may be cases where we need to write the controller code for some of the thief components and one of them is thief-grid. I am trying to write one controller code for thief grid now.

From the API I picked the sample code of Listbox and Options pane code and created a reproducer for those. The links are:
1) Listbox: http://jsbin.factset.com/suvag
2) Options Pane: http://jsbin.factset.com/nezih

If everything looks good then you should automate the reproducer and attach the reproducer link and spec file along with the description while creating an issue in thief-helper library.

The above instructions are based on my findings and sharing this as it may be helpful to the team if we wanted to create a reproducer for the basic functionalities.

I will take a short session on creating a reproducer some time in future if required.

Thanks,
Santhoshi Chepuru
Automation Analyst II, QA Automation
FactSet Systems India Pvt. Ltd.
csanthoshi@factset.com
www.factset.com

Information needed to file gitlab issue
1. **Thief-angular-testing** version used.
2. **Thief-angular** version used by the application.
3. Number of test scripts/test cases affected by the issue.
4. Number of applications affected due to this issue.
5. Is this issue affecting the product release.
6. Reproducer to replicate the same issue which will help the thief engineers to narrow down the issue.

Possible solution for "Do not make functions within a loop" error

Tuesday, May 16, 2017   4:13 PM

Subject	Possible solution for "Do not make functions within a loop" error
From	Shireesha Gaddam
To	Analytics Automation
Sent	Monday, May 15, 2017 11:13 PM

Hi Team,

I think the below way of coding will solve the lint error "Do not make functions within a loop".

Instead of defining the whole function in the loop we can define in separate function and call that function with in the loop.

Example code:

Initial code.-> Gives lint error.

```
UtilitiesSlickgrid.getDataLength('main').then(function(len) {
for (var i = 0; i < len; i++) {
UtilitiesSlickgrid.getCellNode('main', i, 0).then(function(cellRef) {
  cellRef.getCssValue('font-weight').then(function(font) {
    if (font === 'bold') {
      browser.actions().mouseMove(cellRef).perform();
      Utilities.isToolTipAppeared('ACTM').then(function(bool) {
        if (bool) {
          cellRef.getText().then(function(text) {
            PDBMainPage.isTreeExpanded('Portfolio Dashboard',
text).then(function(bool) {
              if (bool) {
                PDBMainPage.getSectionExpandOrCollapseIcon('Portfolio Dashboard',
text).click().then(function() {
                  PDBMainPage.getSectionExpandOrCollapseIcon('Portfolio Dashboard',
text).click().then(function() {
                  }, function() {

                    expect(false).customError('Unable to expand');
                    CommonFunctions.takeScreenShot();
                  });
                }, function() {

                  expect(false).customError('Unable to collapse');
                  CommonFunctions.takeScreenShot();
                });
              }
            });
          });
        }
      });
    }
  });
});
}
});
```

It can be called in the following way:

```
var expandAndCollapse = function(i) {
  UtilitiesSlickgrid.getCellNode('main', i, 0).then(function(cellRef) {
    cellRef.getCssValue('font-weight').then(function(font) {
      if (font === 'bold') {
        browser.actions().mouseMove(cellRef).perform();
        Utilities.isToolTipAppeared('ACTM').then(function(bool) {
          if (bool) {
            cellRef.getText().then(function(text) {
              PDBMainPage.isTreeExpanded('Portfolio Dashboard',
text).then(function(bool) {
                if (bool) {
                  PDBMainPage.getSectionExpandOrCollapseIcon('Portfolio Dashboard',
text).click().then(function() {
                    PDBMainPage.getSectionExpandOrCollapseIcon('Portfolio Dashboard',
text).click().then(function() {
                    }, function() {

                      expect(false).customError('Unable to expand');
                      CommonFunctions.takeScreenShot();
                    });
                  }, function() {

                    expect(false).customError('Unable to collapse');
                    CommonFunctions.takeScreenShot();
                  });
                }
              });
            });
          }
        });
      }
    });
  });
};


it('Verify the "ACTMs" which are in "Bold" should be able to expand and collapse',
function() {
  UtilitiesSlickgrid.getDataLength('main').then(function(len) {
    for (var i = 0; i < len; i++) {
      expandAndCollapse(i);
    }
  });
});
```

No need to write any new logic in such cases.

Thanks,
Shireesha.

Switching back to main application from iframe

In protractor we have a utility (**CommonFunctions.switchToiFrame()**) to switch to Iframe inside application. However we can switch back again to main application from Iframe by using the following line in your scripts.

```
browser.switchTo().defaultContent();
```

Utility to wait until chart changes are reflected

Wednesday, July 26, 2017   3:29 PM

Function:

```
getWaitFunction = function( options ) {
    return function() {
        return browser.driver.executeScript( function( inputObj ) {
            var currentValues = {};
            currentValues[ 'chartSelector' ] = inputObj.chartSelector;
            var chart = $( inputObj.chartSelector ).data( 'chart' );
            _.each( inputObj.chartObjAttributeMap, function( attributeMap,
chartObjId ) {
                var chartObj = chart.getObjectById( chartObjId );
                var attributeKeys = _.keys( attributeMap );
                currentValues[ 'chartObjAttributeMap' ] = {};
                currentValues[ 'chartObjAttributeMap' ][ chartObjId ] = {};
                _.each( attributeKeys, function( attributeKey ) {
                    currentValues[ 'chartObjAttributeMap' ][ chartObjId ]
[ attributeKey ] = chartObj.getAttribute( attributeKey );
                } );
            } );
            _.each( inputObj.resolvePropertyStringMap, function( propertyStringMap,
chartObjId ) {
                var chartObj = chart.getObjectById( chartObjId );
                var propertyKeys = _.keys( propertyStringMap );
                currentValues[ 'resolvePropertyStringMap' ] = {};
                currentValues[ 'resolvePropertyStringMap' ][ chartObjId ] = {};
                _.each( propertyKeys, function( propertyKey ) {
                    currentValues[ 'resolvePropertyStringMap' ][ chartObjId ]
[ propertyKey ] = chartObj.getResolvedPropertyString( propertyKey );
                } );
            } );
            return currentValues;
        }, options ).then( function( results ) {
            return _.isEqual( results, options );
        } );
    };
};
```

How to call it in Spec?

```
browser.wait( UtilitiesFDSChartJS.getWaitFunction( {
    chartSelector: '#chart',
    chartObjAttributeMap: {
        FC_CHART: {
            CanvasBGColor: 26316,
            CanvasBGAlpha: 145.35000000000002
        }
    }
} ), 10000, 'Options not set properly on main chart' );
```

FC_CHART: is the object id on which we need to verify the attributes mentioned. (If it is some series mention series id etc.)

You can have this utility in charting utilities and use whenever needed. One thing you have to do is changing chart controller statement which is highlighted above.

Drag and Drop functionality

Monday, August 07, 2017   11:26 PM

Subject	Drag and Drop functionality
From	Santhoshi Chepuru
To	Analytics QA Automation
Sent	Monday, August 07, 2017 11:03 PM

Hi Team,

Drag and Drop thief helpers is implemented for At4 thief-grid. This was implemented very long time before. There were few issues from Grid side and thief-helpers side so the test cases were moved to Road Blocked. But with the latest version of updated thief-angular-testing to 2.2.1 and thief-angular-grid to 1.2.0 no longer issues were seen. The only thing we should do is, should select an element that needs to be dragged and implement the code to perform and drag and drop operation.

The Code for Drag and Drop is:

```
it('Drag and drop operation', function() {
  var grid = new ThiefAngularGrid.Grid(by.css('tf-local-grid'));
  var column = grid.columnByLabel('Name');
  var dragRow = grid.rowByLabel(column, 'Industry').element();
  var dropRow = grid.rowByLabel(column, 'Profit').element();

  dragRow.click();

  var draggable = new TestHelpers.Draggable(dragRow);
  var dropZone = new TestHelpers.Dropzone('after', dropRow);
  draggable.drop(dropZone);
});
```

I think few of the applications were using thief-grid. So, they can speak with their Engineers if they support drag and drop functionality for the thief grid and can automate the same.

Regards,
Santhoshi Chepuru
Automation Analyst II, QA Automation
FactSet Systems India Pvt. Ltd.
csanthoshi@factset.com
www.factset.com

FACTSET ⟩ SEE THE ADVANTAGE

FINANCIAL DATA | ANALYTICS | TECHNOLOGY | SERVICES

Getting started with VS code editor

Thursday, October 12, 2017   4:52 PM

- Install Visual studio code.
- Open your project in VSC editor.
- **For the configuration settings to run our conf.js file**: In Debug section click on the cogwheel button and (select node.js) edit the **launch.json**. It is similar like web storm configurations.
  ```
  "name": "Launch",
  "type": "node",
  "request": "launch",
  "program": "C:/node_modules/protractor/built/cli.js",
  "stopOnEntry": false,-
  "args": ["${workspaceRoot}/conf.js"]
  ```

The following attributes are mandatory for every launch configuration:

- type - the type of debugger to use for this launch configuration. Every installed debug extension introduces a type, for example, node for the built-in node debugger, or php and go for the PHP and Go extensions.
- request - the request type of this launch configuration. Currently supported are launch and attach.
- name - friendly name which appears in the Debug launch configuration dropdown.

Here are some optional attributes available to all launch configurations:

- preLaunchTask - to launch a task before the start of a debug session, set this attribute to the name of a task specified in tasks.json (located under the workspace's .vscode folder).
- internalConsoleOptions - control visibility of the Debug Console panel during a debugging session
- debugServer - for debug extension authors only: connect to the specified port instead of launching the debug adapter

Many debuggers support some of the following attributes:

- program - executable or file to run when launching the debugger
- args - arguments passed to the program to debug
- env - environment variables
- cwd - current working directory for finding dependencies and other files
- port - port when attaching to a running process
- stopOnEntry - break immediately when the program launches
- console - what kind of console to use, for example, internalConsole, integratedTerminal, externalTerminal.
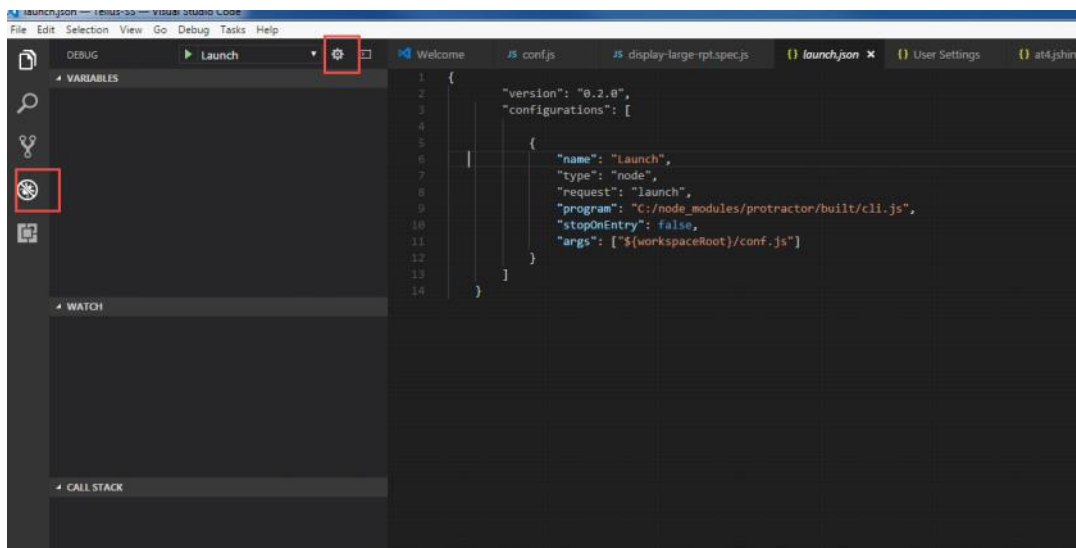
- **Installing the required modules:** From the extensions tab you can install all the required modules. (Adding the screenshot that includes the modules that I've installed).
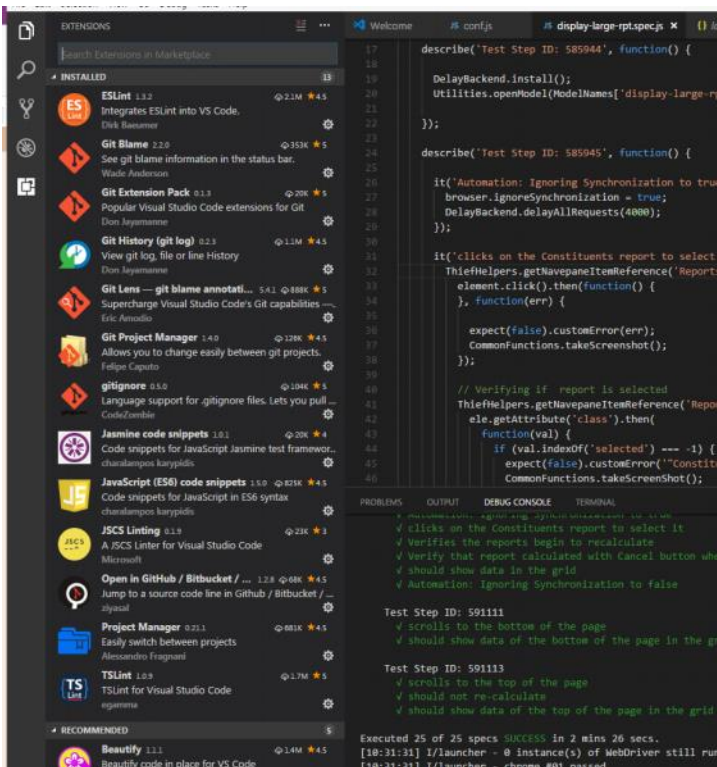
- **Setting User settings:** Click on the cog Wheel from down left and select settings or select settings from File menu to set the User settings. Here you can edit the settings as per our project requirements.
- Go to the required setting and click on the edit button and select the option suitable for the project.

-

- Following are the changes that I've added

```
"gitProjectManager.baseProjectsFolders": [
"C:\\"
],
"gitProjectManager.ignoredFolders": ["node_modules"],
"jscs.configuration": {
"plugins": [ "@fds/lint-standards/jscs" ],
"preset": "factset"
},
"javascript.format.insertSpaceAfterFunctionKeywordForAnonymousFunctions": false,
"javascript.format.insertSpaceAfterSemicolonInForStatements": false,
"editor.tabSize": 2,
"editor.insertSpaces": false
```
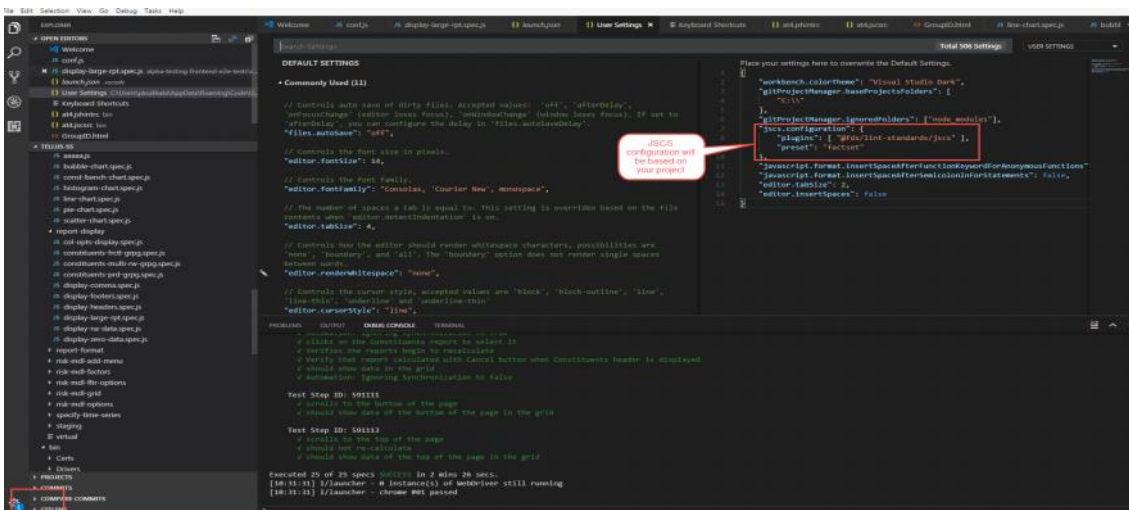

- To run the script normally select Run without debugging option from Debug menu or **CTRL+F5.**
- For Debug: Click on **Run** button in DEBUG section or **F5**.
- To fix all jscs errors in your file go to the file and **Alt+Shift+F** or right click on the file and select **Format Document** option.

Connecting to specific node on Selenium-Grid

Friday, August 11, 2017   2:41 PM

Added the binary path in ChromeOptions and I am able to connect to the node:
Reference:

```
{
  browserName: 'chrome',
  version: browserVersion,
  chromeOptions: {
    args: ['start-maximized', '--disable-extensions'],
    binary: 'C:/selenium/chromes/Chrome54/chrome.exe',
  },
}
```

I tried to connect to the node using ruby chrome options you provided. But, I am unable to connect to the Node. Could you please let us know the Key to specify the binary path.

Note: Binary path varies from node to node. Make sure the Chrome.exe is available at the binary path set.

Regards,
Charvaka.

From <http://is.factset.com/rpd/summary.aspx?messageId=33312120&commentId=33341247>

'[controller]' is better written in dot notation

Friday, November 03, 2017   3:05 PM

Subject	**Lint error**
From	Komal Agrawal
To	Analytics QA Automation
Sent	Friday, November 03, 2017

When we use controller helpers(up, down, clear) of transfer box or any other component as explained in transferbox helpers index.spec.js, it is throwing a lint error " **'[controller]' is better written in dot notation**".

I found a workaround which is working for me.

Current usage as in Index.spec.js: ref.target[controller]();
Workaround snippet : ref.target. controller();

Control Flow while injecting overrides - Charuvaka

Tuesday, November 07, 2017   4:29 PM

Subject	**Control Flow while injecting overrides.**
From	Charuvaka Thallapelli
To	Bhupender Agarwal; Charuvaka Thallapelli; Deekshith kumar Lingoju; Durga Prasad Vemula; Komal Agrawal; Mahendra Tribhuvan Sura; Priyanka Kudikala; Ranvijay Kumar singh; Santhoshi Chepuru; Shireesha Gaddam; Sruthi Kura; Sudha Chakinala; Usha Sai Masabattula; V N Kiranmai Malladi; Vasavi Tallada
Sent	Tuesday, November 07, 2017 4:23 PM

Hi,

Few of us are aware of the execution flow to inject overrides, just to iterate here is the control flow to inject overrides via fiddler:

- **CommonFunctions.setOverrideCookie**() function is to set cookie to the HTTP-Request.
- `browser.manage().addCookie({ name: `**`'LimaOverrideCookie'`**`, value: override, domain: domain })` will add a cookie with name: LimaOverrideCookie and value: override Ex:(devel,sat_to_devel) to the HTTP-Request.

Have a look at the below screenshot to understand where the LimaOverrideCookie is added:

Below screenshot's are Cookie information captured from HTTP Request Headers using Fiddler.
With **chromeDriver 2.28**
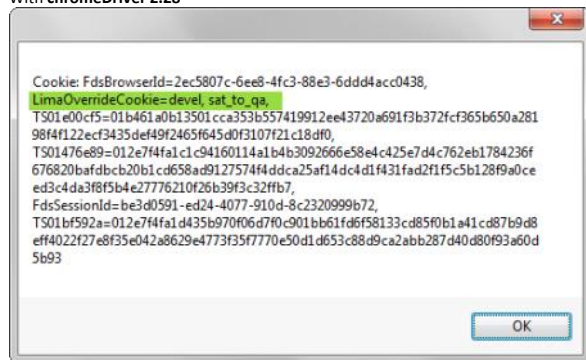
With **chromeDriver 2.33**:
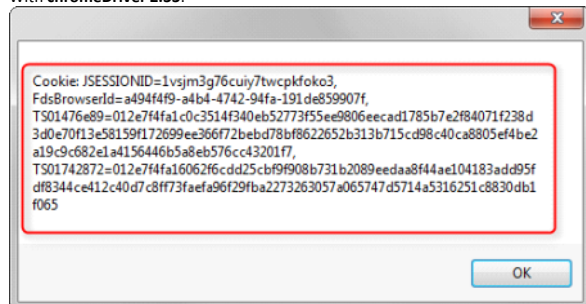
You can see that LimaOverrideCookie is not added in the second case.

- If the Cookie is successfully added
  - Fiddler will catch the request Headers as shown below:

- Now onBeforeRequest() function in CustomRules.js file will get the headers and check if **LimaOverrideCookie** is available in Cookie[], Look at the below snippet:

```
if(oSession.host.Contains("factset.com") && oSession.oRequest["Cookie"])
    {
        var cookies = oSession.oRequest["Cookie"].Split(';'); -→ This
Cookie is has no LimaOverrideCookie in the second-case(with 2.33)
            MessageBox.Show("Cookie: " + cookies);

        for(var i = 0; i < cookies.length; i++)
        {
            var c = cookies[i];

            if(c.IndexOf("LimaOverrideCookie") > -1 &&
c.IndexOf("=") > -1) → This is false as LimaOverrideCookie is not
available.
            {
                oSession.oRequest["X-FDS-Override-Name"] =
cookies[i].Substring(cookies[i].IndexOf('=') + 1);
            }

                    if(c.IndexOf("FDSAV2SessionCookie") > -1 &&
c.IndexOf("=") > -1) →  → This is false as FDSAV2SessionCookie is not
available.
            {
                oSession.oRequest["X-Fdsav2-Session-Id"] =
cookies[i].Substring(cookies[i].IndexOf('=') + 1);
            }
        }
    }
```

- As the above tow conditions are not satisfied using 2.33Driver, **LimaOverrideCookie/FDSAV2SessionCookie** are not injected.

Please correct me if I am wrong.

Regards,
Charvaka.

Automation Issues

Nice Findings

Tuesday, March 20, 2018   6:26 PM

RunKit to utilize Node Modules without installing

Tuesday, March 20, 2018   6:27 PM

Subject	**FW: RunKit to utilize Node Modules without installing**
From	Thallapalli Charvaka
To	Bhupender Agarwal
Sent	Tuesday, March 20, 2018 4:56 PM

Hi Team,

We have been working with different node modules on a daily basis. We may come across few modules which we install just to do a POC.

https://npm.runkit.com/
Here is **RunKit** powered by **npm** to utilize the node modules without installing. All you need is require the module in Chrome and use.

This is a `playground` to test code. It runs a full `Node.js` environment and already has all of `npm`'s 400,000 packages pre-installed, including `openurl`. Try it out:

```
1  var _ = require("lodash"  4.17.5  )
2  _.capitalize('charvaka');
```

Save on **RunKit**   Node 8 ⇕   help   ▶ run

    "Charvaka"

Utilize RunKit to make your job easy ☺

Regards,
Charvaka.