Protractor Guidelines

StepStone

INCONTACT

*Created Date: July 09, 2018*

*Updated Date: Nov 01, 2018*

*Updated Version: 1.3*

*Created By: LogiGear Team*

Contents

[Automation Environment Setup 4](#_Toc532393046)

[I. Setup Java 4](#_Toc532393047)

[II. Setup Protractor 4](#_Toc532393048)

[1. Install Node.js 4](#_Toc532393049)

[2. Install Protractor 4](#_Toc532393050)

[3. Setup and update WebDriver for IE and Edge 4](#_Toc532393051)

[III. Setup Git 9](#_Toc532393052)

[1. Download Git 9](#_Toc532393053)

[2. Use command to setup Git 9](#_Toc532393054)

[3. Use SourceTree as Git client to setup Git 10](#_Toc532393055)

[Jenkins Server & Hourly Builds 15](#_Toc532393056)

[I. Jenkins Server Information 15](#_Toc532393057)

[II. How to use Jenkins 16](#_Toc532393058)

[1. Login to Jenkins 16](#_Toc532393059)

[2. Check result and console log 17](#_Toc532393060)

[3. Executing test cases on Jenkins 21](#_Toc532393061)

[III. How to execute test cases on Visual Studio Code 22](#_Toc532393062)

[1. Install Visual Studio Code 22](#_Toc532393063)

[2. Executing 24](#_Toc532393064)

[Working With Protractor and SourceTree 27](#_Toc532393065)

[I. Protractor Structure 27](#_Toc532393066)

[1. build 27](#_Toc532393067)

[2. node\_modules 27](#_Toc532393068)

[3. src 28](#_Toc532393069)

[4. Others 28](#_Toc532393070)

[II. Protractor naming convention 29](#_Toc532393071)

[1. Files 29](#_Toc532393072)

[2. Classes 29](#_Toc532393073)

[3. Elements 29](#_Toc532393074)

[4. Methods 31](#_Toc532393075)

[5. Test cases 32](#_Toc532393076)

[a. Test cases Name 32](#_Toc532393077)

[b. Comment in test case 32](#_Toc532393078)

[c. Verify point in test case 33](#_Toc532393079)

[III. Coding conventions vs Programming style 33](#_Toc532393080)

[IV. Working with SourceTree 38](#_Toc532393081)

[1. Pull code 38](#_Toc532393082)

[2. Push code (we need to pull code before pushing new code) 39](#_Toc532393083)

[3. Resolve Conflict 43](#_Toc532393084)

[Troubleshooting 45](#_Toc532393085)

[I. Unable to use command “*tsc*” 45](#_Toc532393086)

[1. Error Behavior 45](#_Toc532393087)

[2. Solution 45](#_Toc532393088)

# Automation Environment Setup

## Setup Java

* Window must be installed Java 8
* Link download: 
  + JDK (version: 8u171): <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
  + JRE (version: 8u171): <http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>
* After installing Java, path in system variables (System Properties > Environment Variables) should be “ C:\Program Files\Java\jdk1.8.0\_171\bin” (window 64 bit)

## Setup Protractor

### Install Node.js

* Link download: https://nodejs.org/en/
* Current version is: 8.11.1
* After installing Node.js, path in system variables (System Properties > Environment Variables) should be “C:\Program Files\nodejs\ & %AppData%\Roaming\npm” (window 64 bit)

### Install Protractor

* Use npm to install globally with below command on Command Prompt:
  + npm install -g protractor
  + npm install -g typescript
  + npm install -g jasmine2-protractor-utils

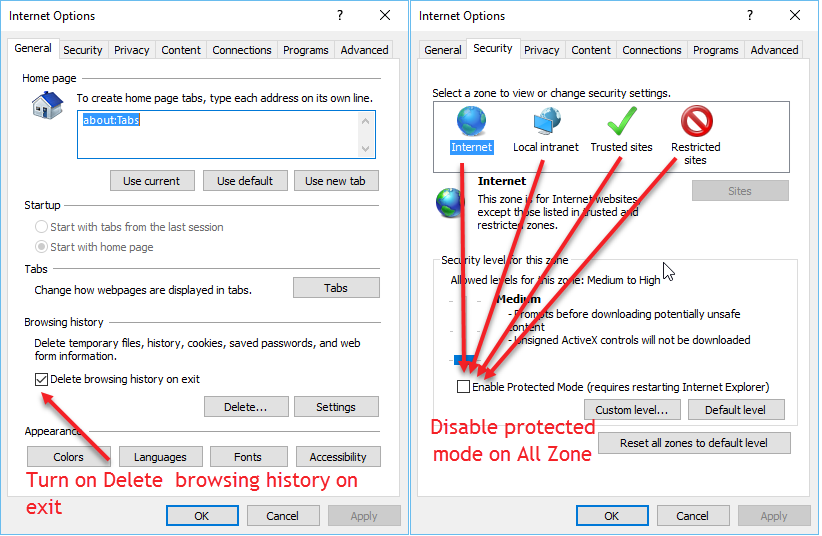
### Setup and update WebDriver for IE and Edge

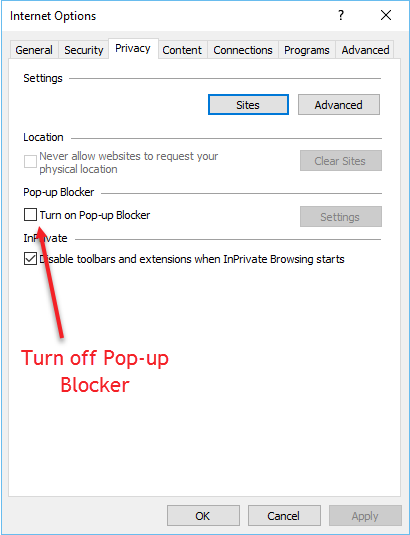
#### Internet Explorer

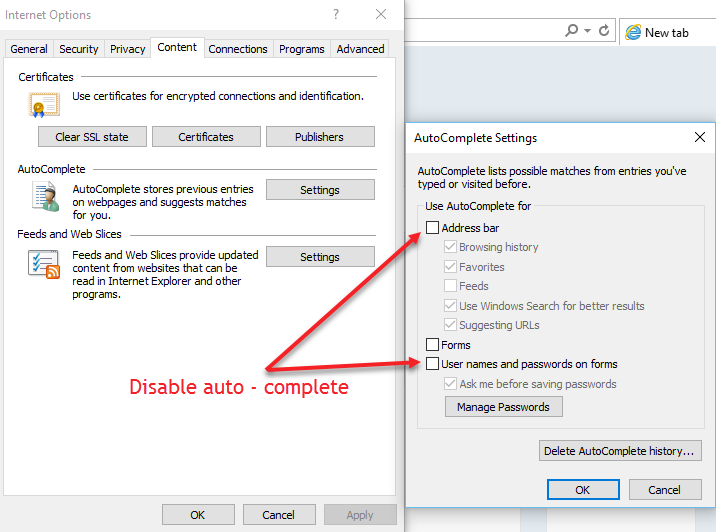
* **Install IE WebDriver to run Protractor:**

Open cmd and enter: webdriver-manager update --ie

* **Setting Internet Explorer as below:**

****





After that click OK button

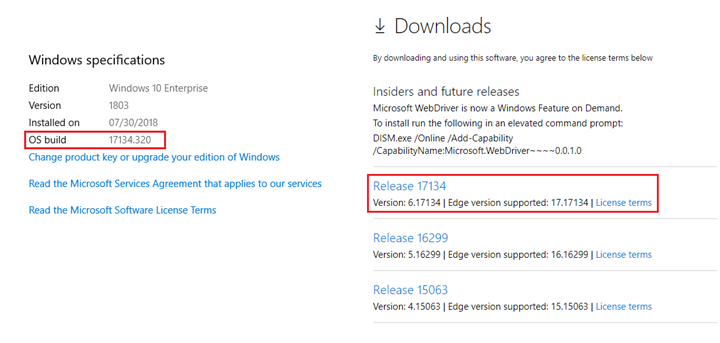
#### Edge

* **Install Edge to run protractor:** 
  + Follow link to download Edge driver: <https://developer.microsoft.com/en-us/microsoft-edge/tools/webdriver/>

**Note:**

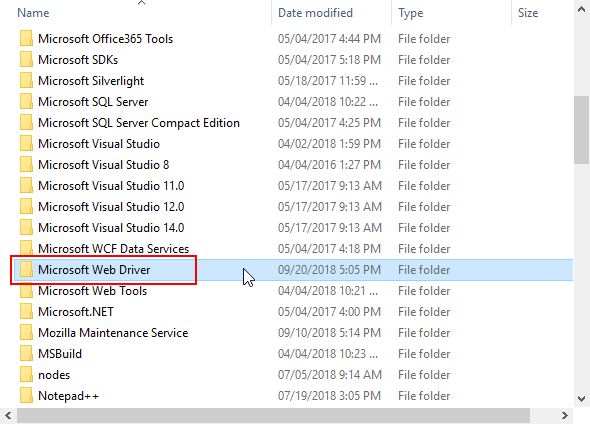
Download the correct Microsoft WebDriver server version for your OS build.

To find your correct build number: Go to **Start** > **Settings** > **System** > **About** and locate the number next to OS Build on the screen.

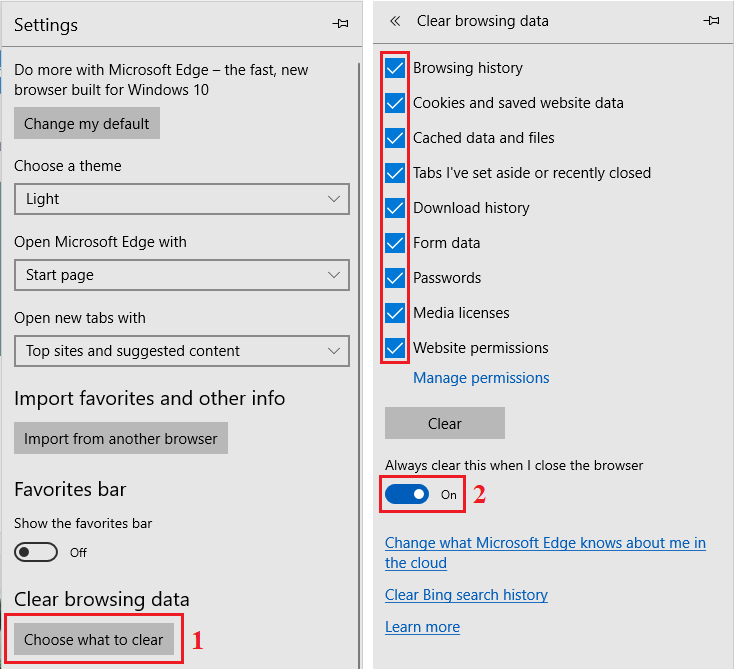


**Noted:** We need to update to lasted window version to use the lasted Edge driver. The older Edge version has a lot of issue while running Protractor.

After that, we copy downloaded Edge driver to folder “**C:\Program Files (x86)\Microsoft Web Driver**” (create if not exist)



**Setting Edge as below:**



* **Before running protractor:**

Open cmd and enter: webdriver-manager update

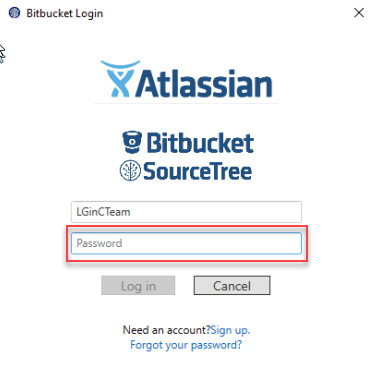
## Setup Git

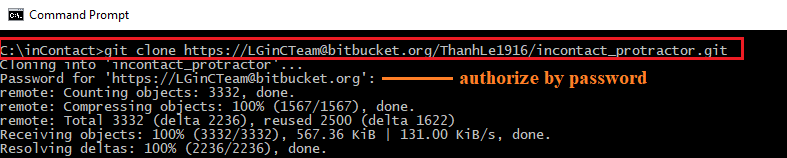
### Download Git

* Link download: <https://git-scm.com/download/win>
* Current version is: 2.18.0
* After installing Git, path in System variables (System Properties > Environment Variables) should be “C:\Program Files\Git\cmd” (window 64 bit)

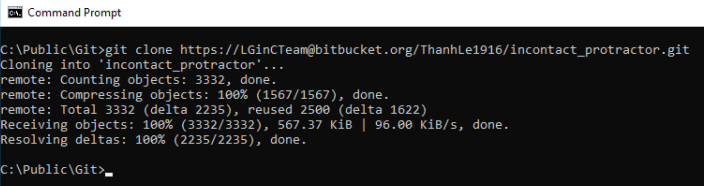
### Use command to setup Git

* Information:
  + Repository URL: https://LGinCTeam@bitbucket.org/ThanhLe1916/incontact\_protractor.git
  + Password: logigear123!
* Step 1: Navigate to clone git repo folder.
  + cd + “repository folder path”
* Step 2: Clone git repository.
  + git clone https://LGinCTeam@bitbucket.org/ThanhLe1916/incontact\_protractor.git
* Step 3: Enter password on authentication pop up or on the command line (logigear123!)



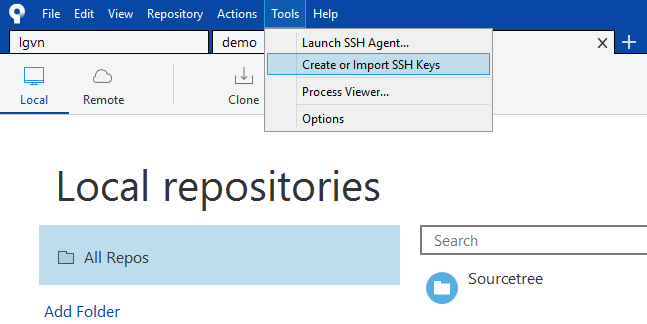


* Step 4: Wait until cloning completely.

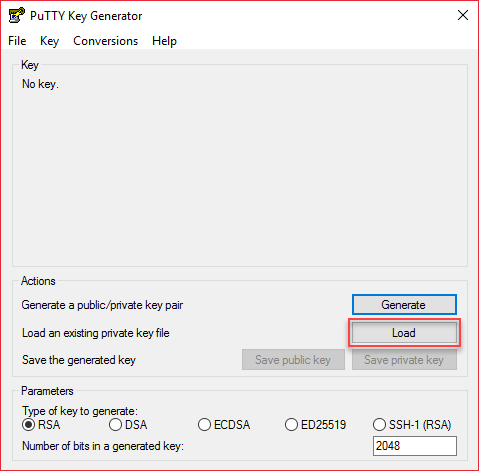


### Use SourceTree as Git client to setup Git

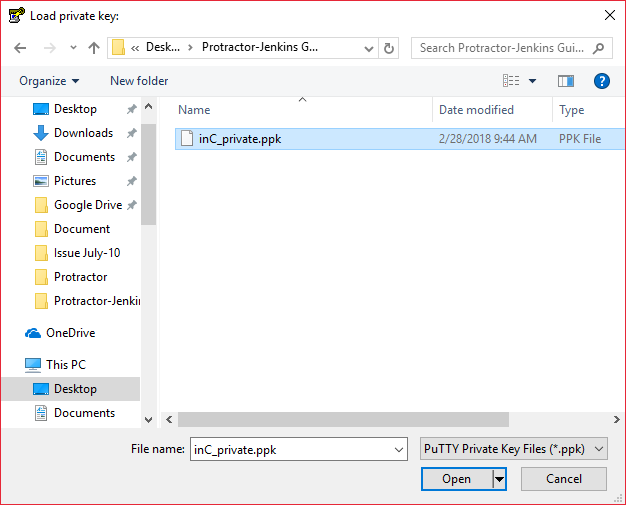
* Information:
  + Repository URL: [git@bitbucket.org:ThanhLe1916/incontact\_protractor.git](mailto:git@bitbucket.org:ThanhLe1916/incontact_protractor.git)
  + Security: We’ve using SSH key (Please use the attached private key “inC\_private.ppk”).
  + Passphrase for Private Key: logigear123!
  + Atlassian account:
    - US team: incuser001@gmail.com / logigear123!
    - VN team: [lgincontactteam@gmail.com /](mailto:lgincontactteam@gmail.com%20/) logigear123!
* Step 1: Download and install [SourceTree](https://blog.sourcetreeapp.com/)
* Step 2: Import SSH Keys
  + Select Tools Menu > Create or Import SSH Keys



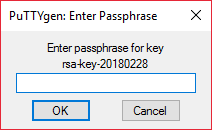
* + Click on “Load” button to import SSH keys



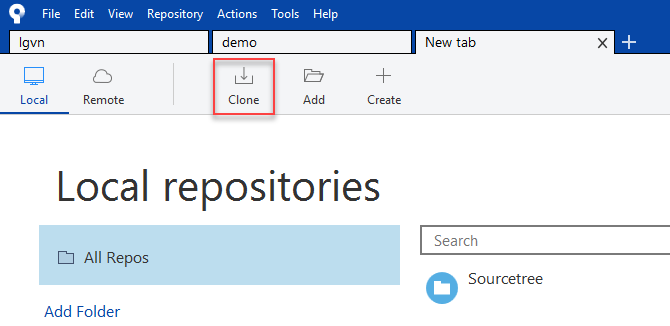
* + Select SSH key “inc\_private.ppk”



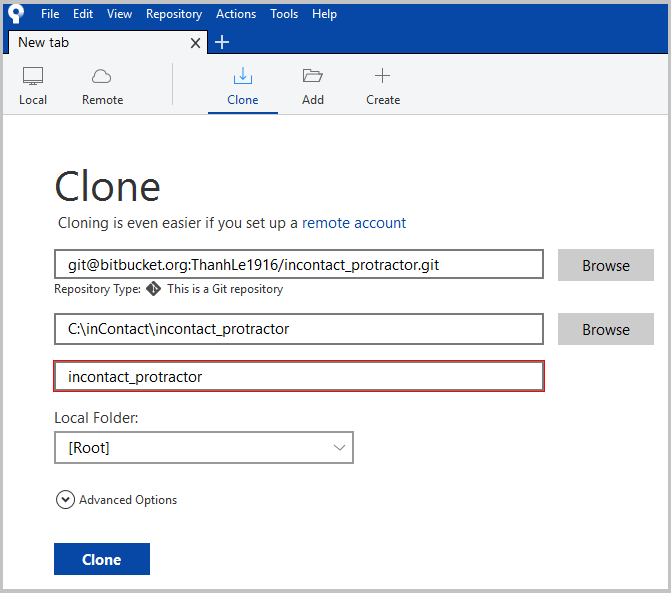
* + Enter SSH key Passphase (logigear123)



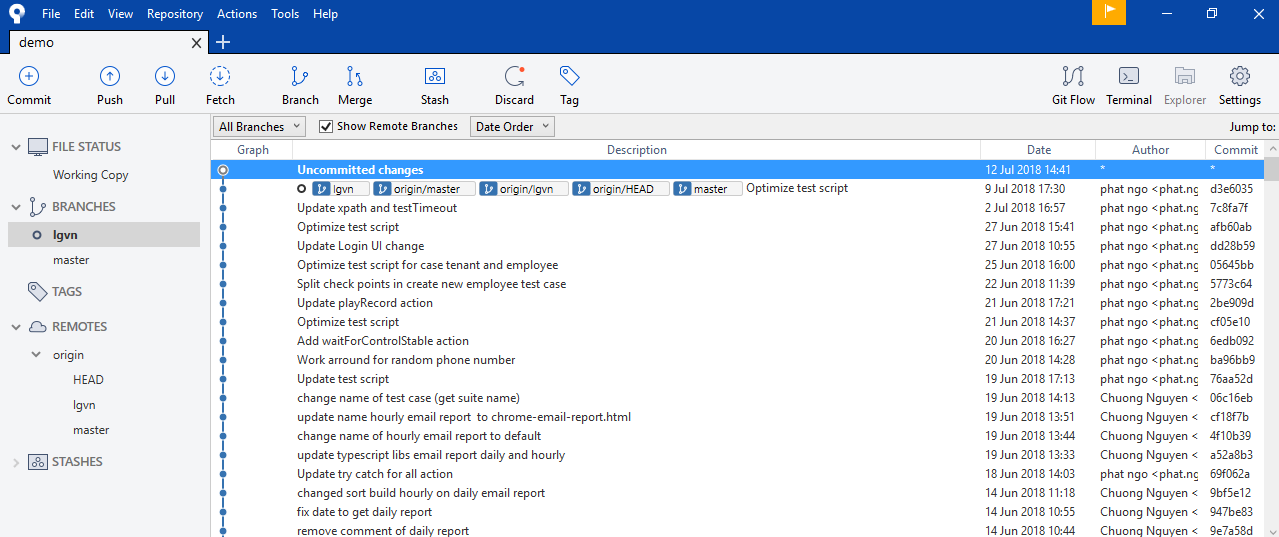
* Step 3: Clone git repository
  + Select “Clone” on menu to show Clone page.



* + Input Source Path: git@bitbucket.org:ThanhLe1916/incontact\_protractor.git
  + Input Destination Path
  + Click “Clone” button to download git repository.



* Step 4: Wait until cloning completely.



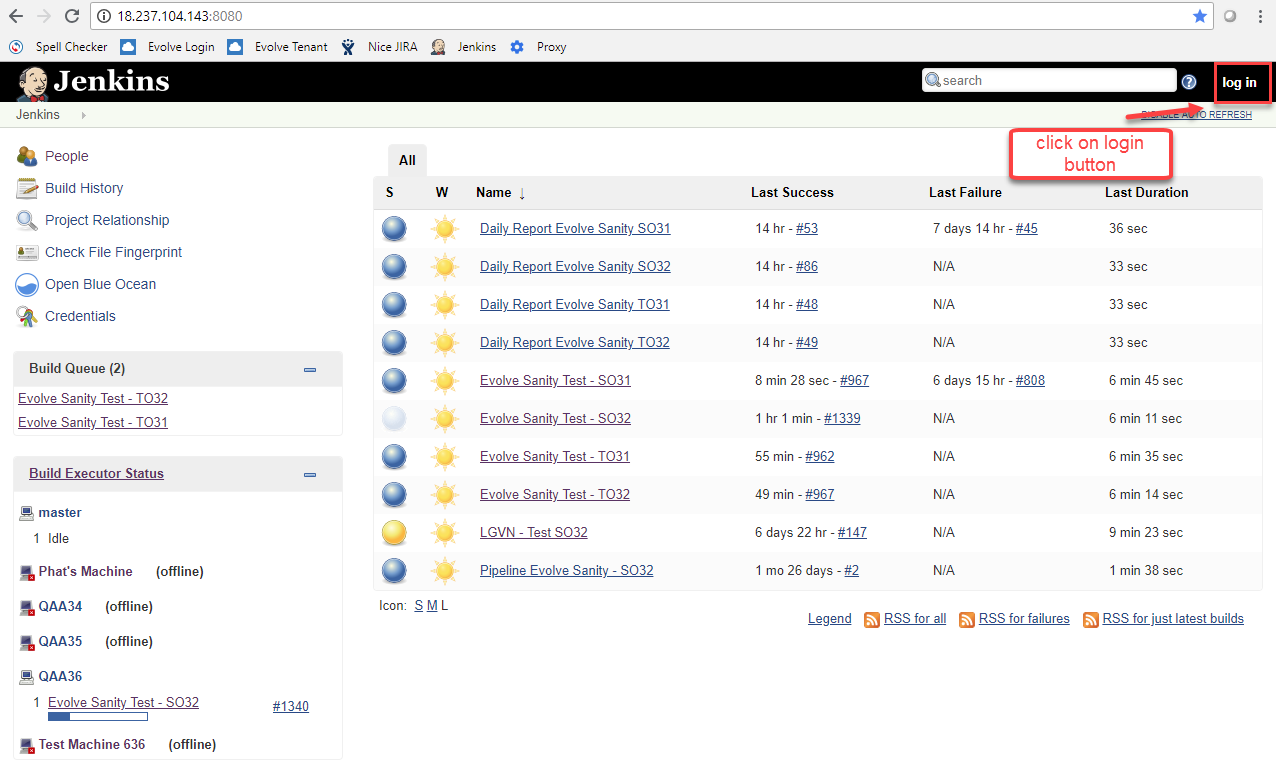
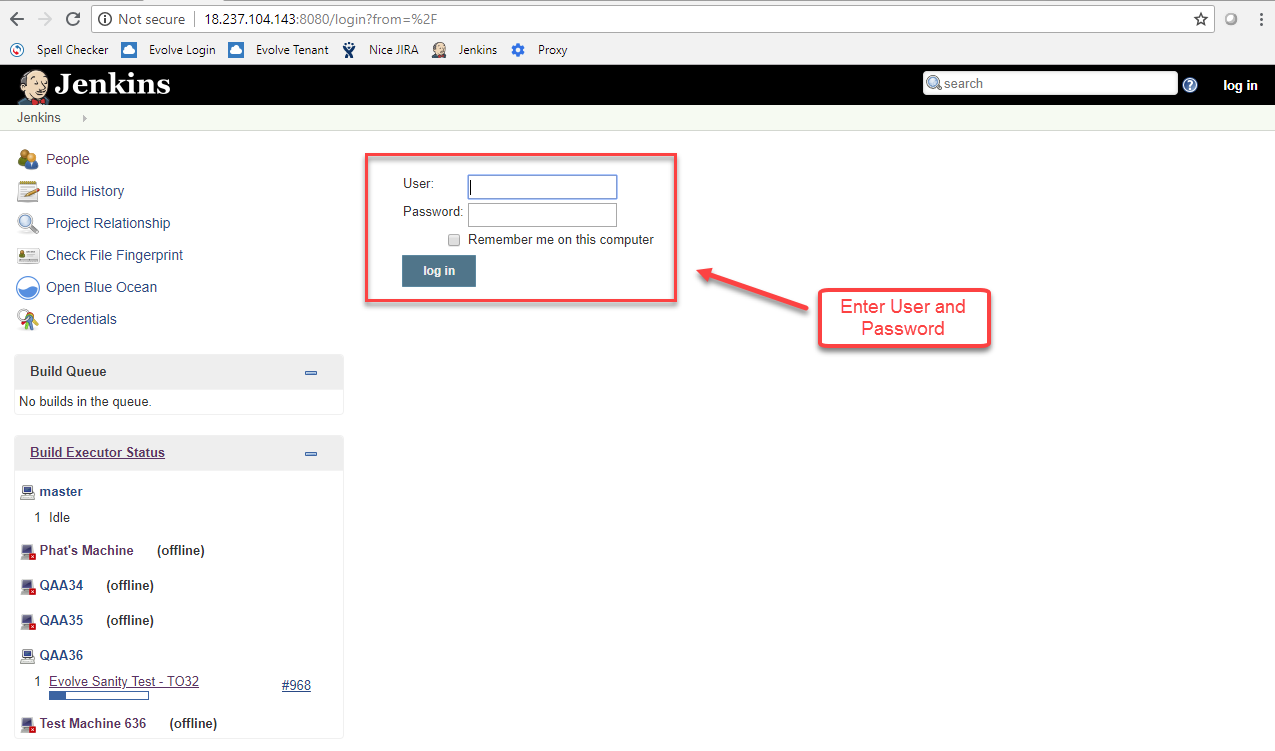
# Jenkins Server & Hourly Builds

## Jenkins Server Information

* Link server: http://18.237.104.143:8080/
* Account: 
  + Username: inContactAdmin
  + Password: logigear123
* Evolve Sanity Hourly builds:
  + Evolve Sanity Test - SO31
  + Evolve Sanity Test - SO32
  + Evolve Sanity Test - TO31
  + Evolve Sanity Test - TO32
* Running machine: Eng-QAA36.in.lab
* Time to run: It will start at the beginning of each hour (e.g. 12:00, 1:00..)
* **Note:** We should restart an execute machine to prevent memory leak.

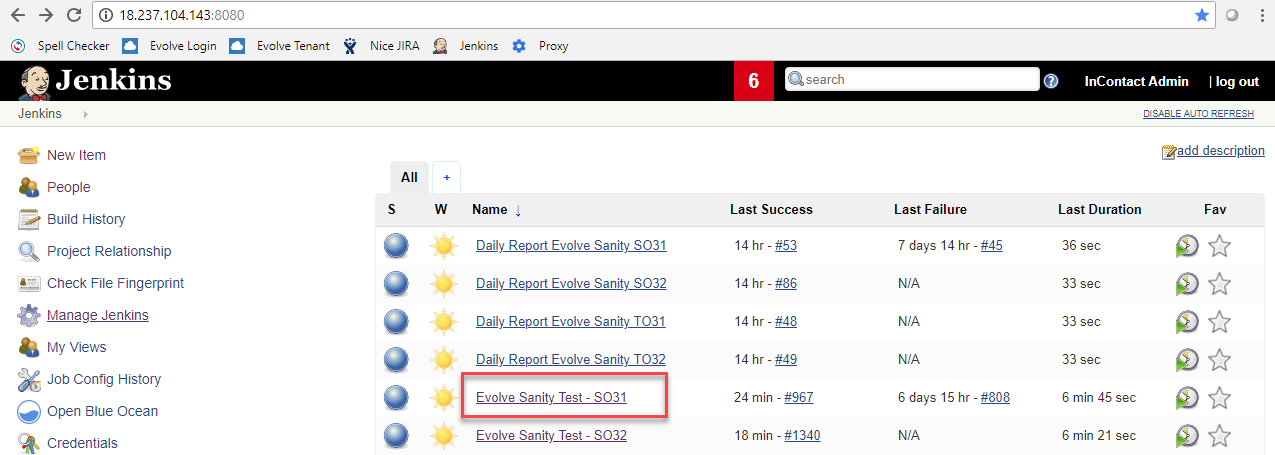
## How to use Jenkins

### Login to Jenkins

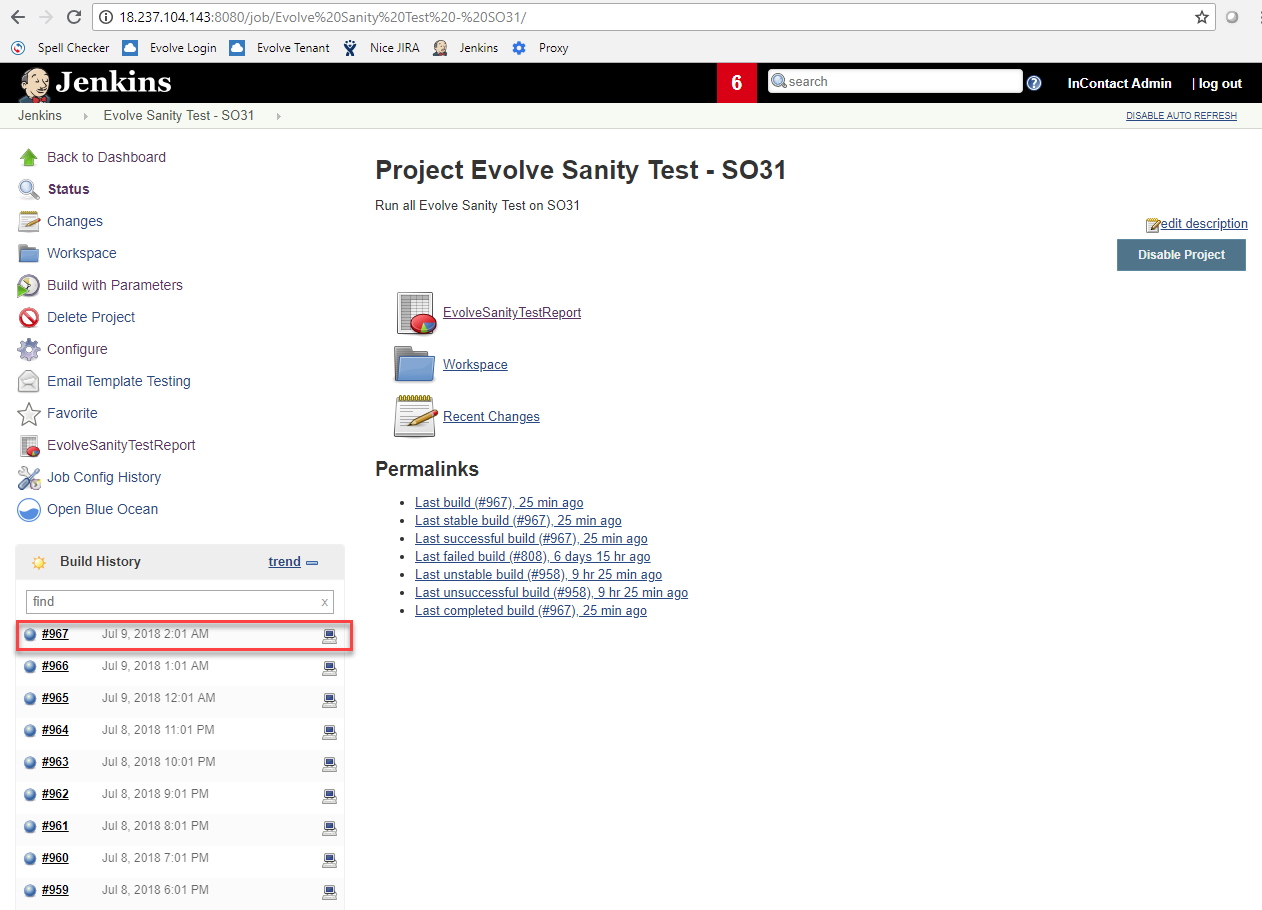
* Click on login button
* Enter User and Password

### Check result and console log

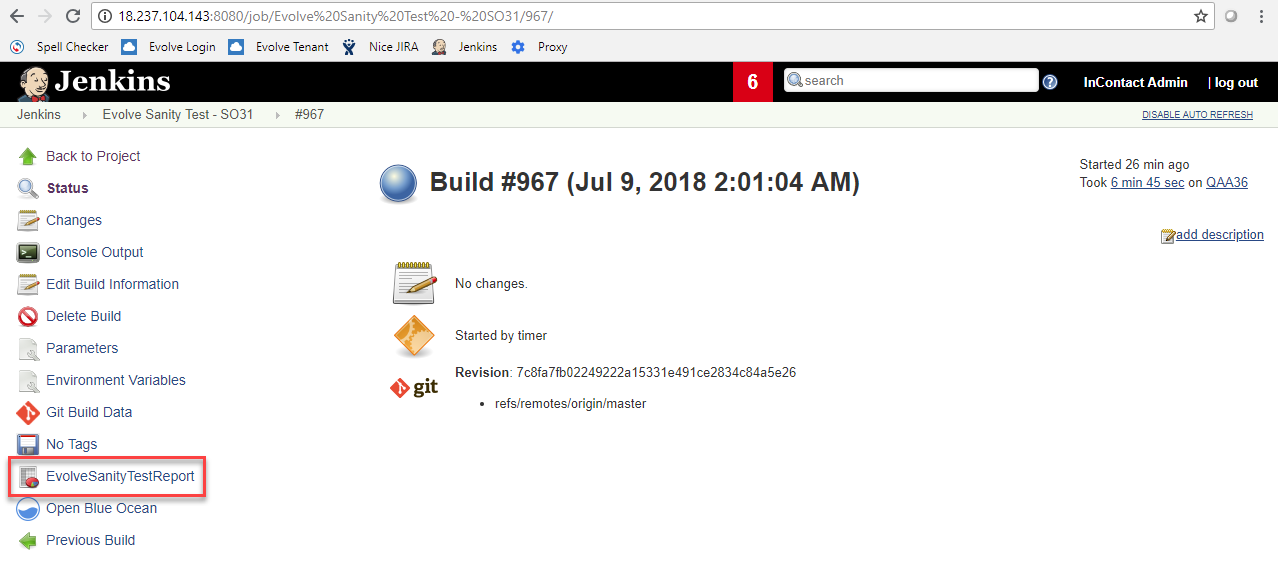
* Select hourly build such as “Evolve Sanity Test - SO31”



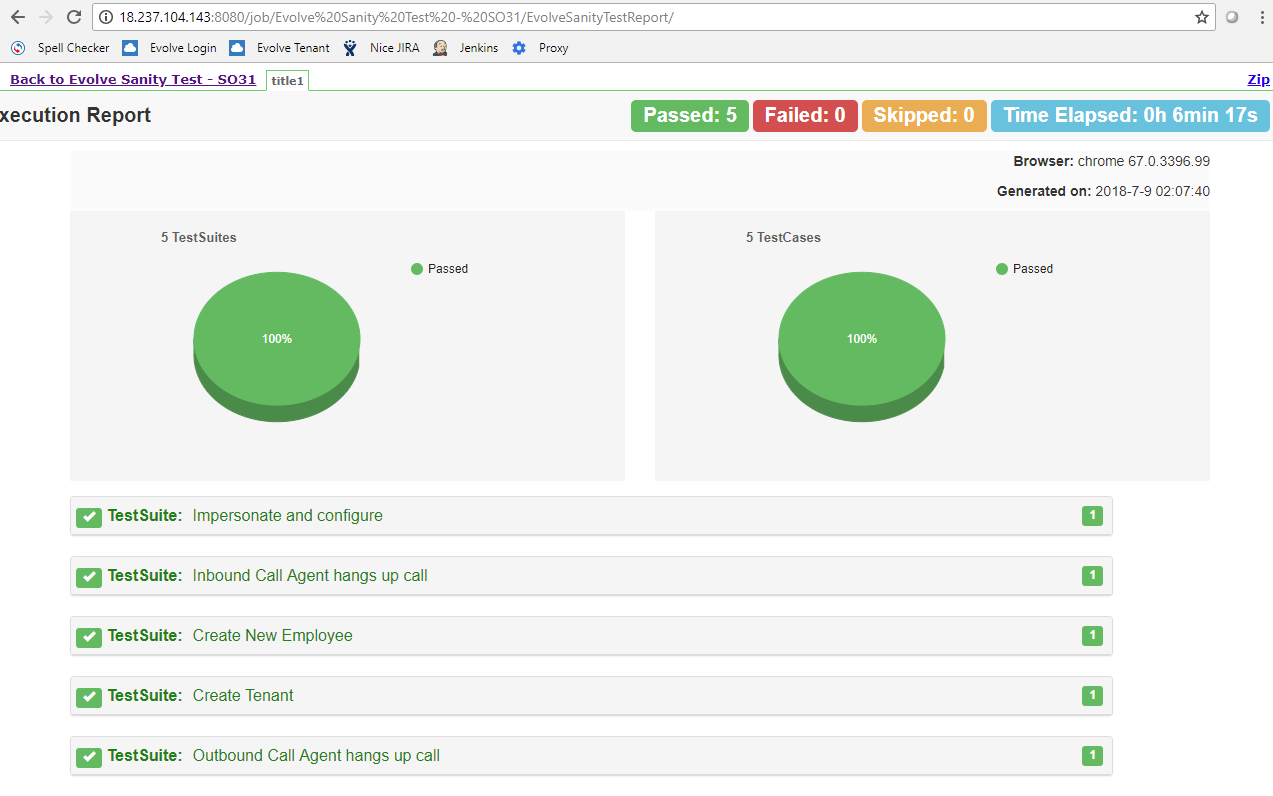
* Select build number you want to check

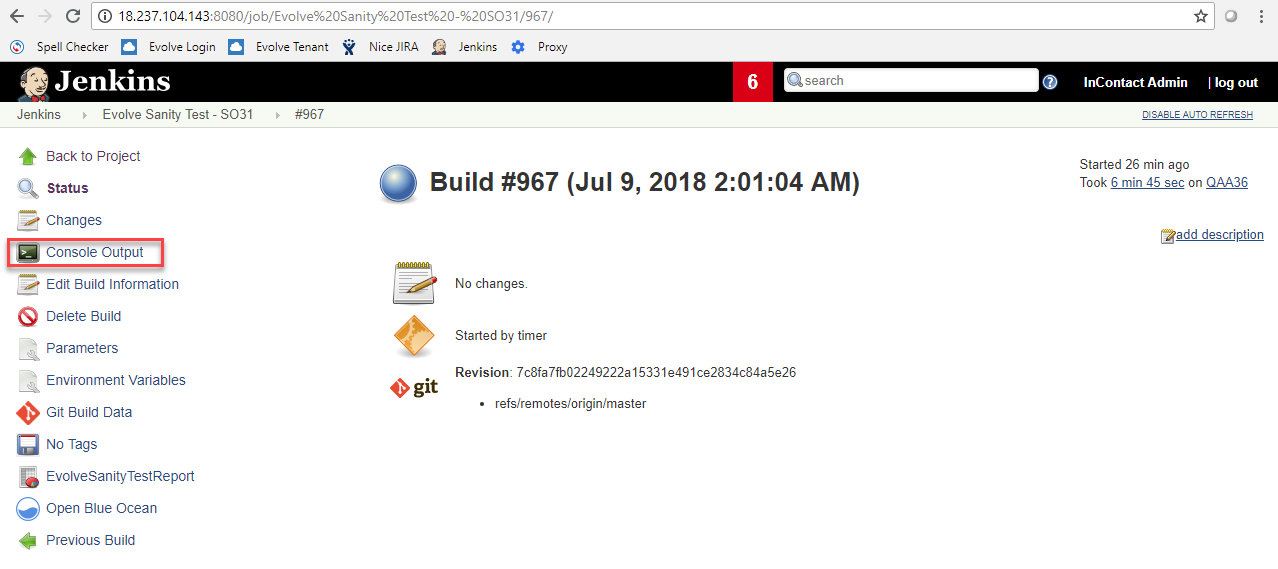


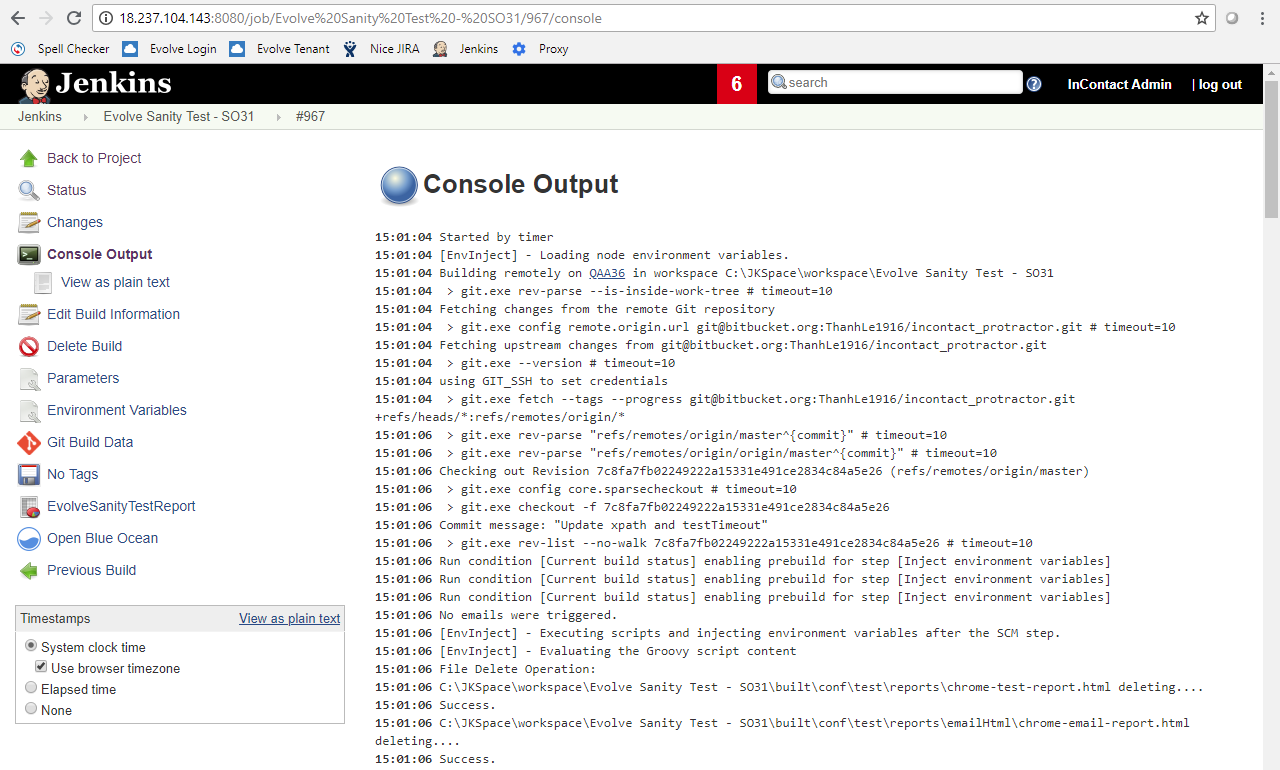
* Select “EvolveSanityTestReport” --> “chrome-test-report.html” to check result





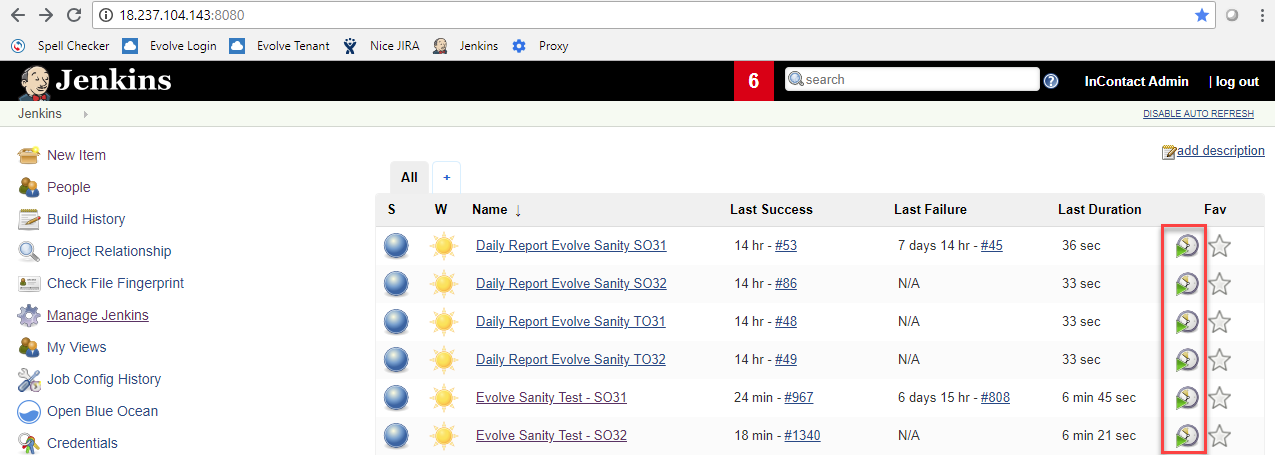


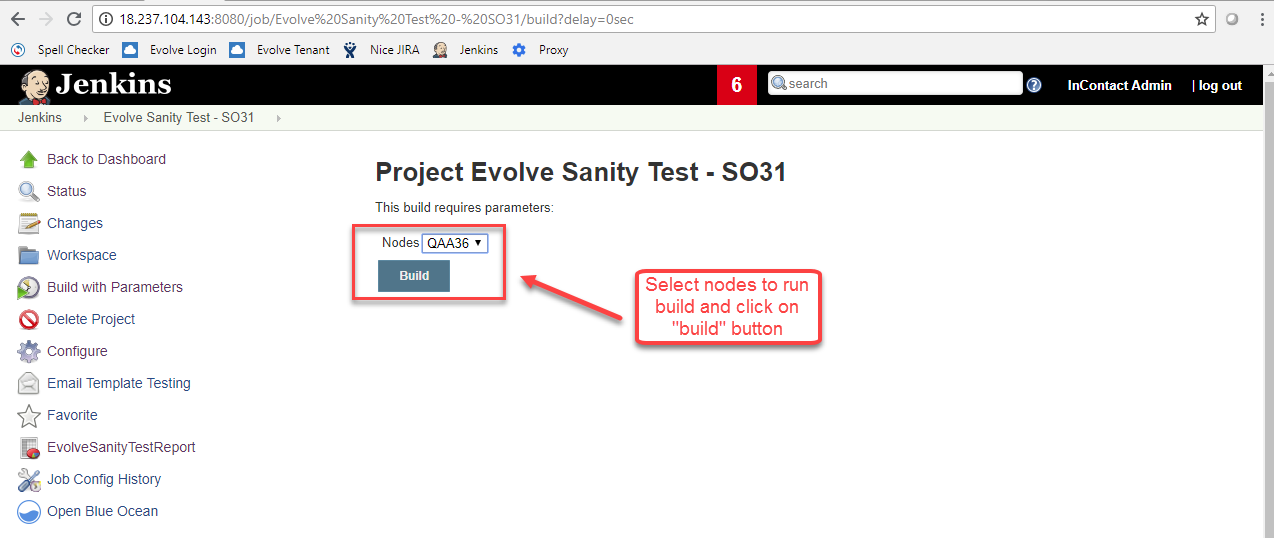
* Select “Console Output” to check console log



### Executing test cases on Jenkins

* Login to Jenkins server: <http://18.237.104.143:8080/>
* Select “schedule build” icon

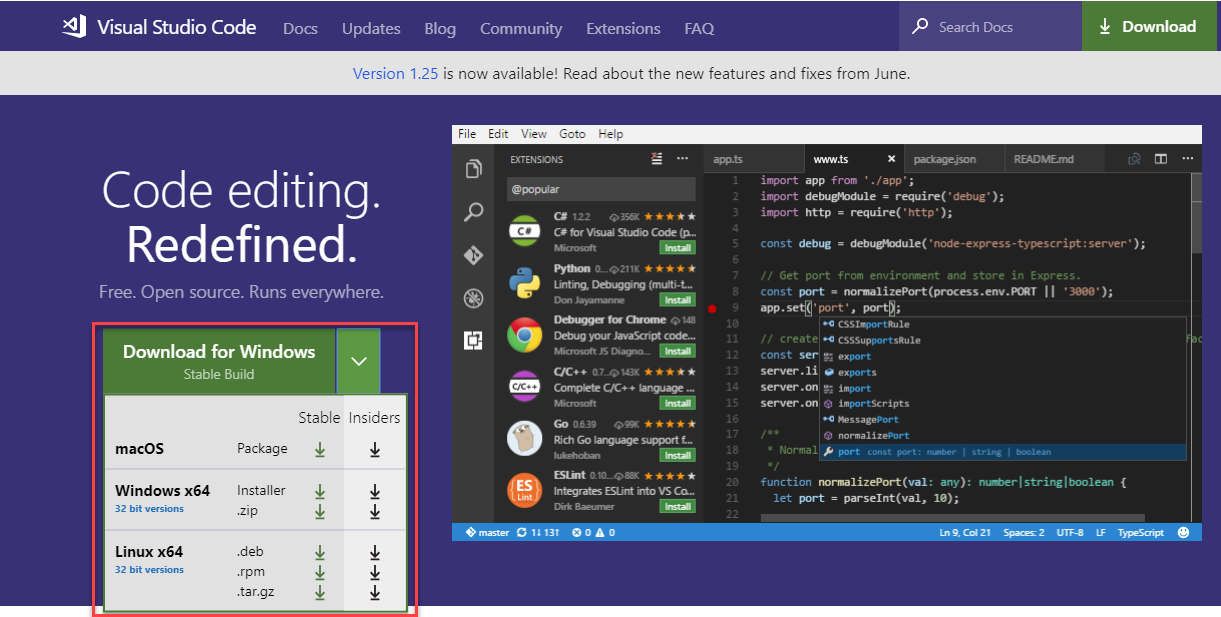




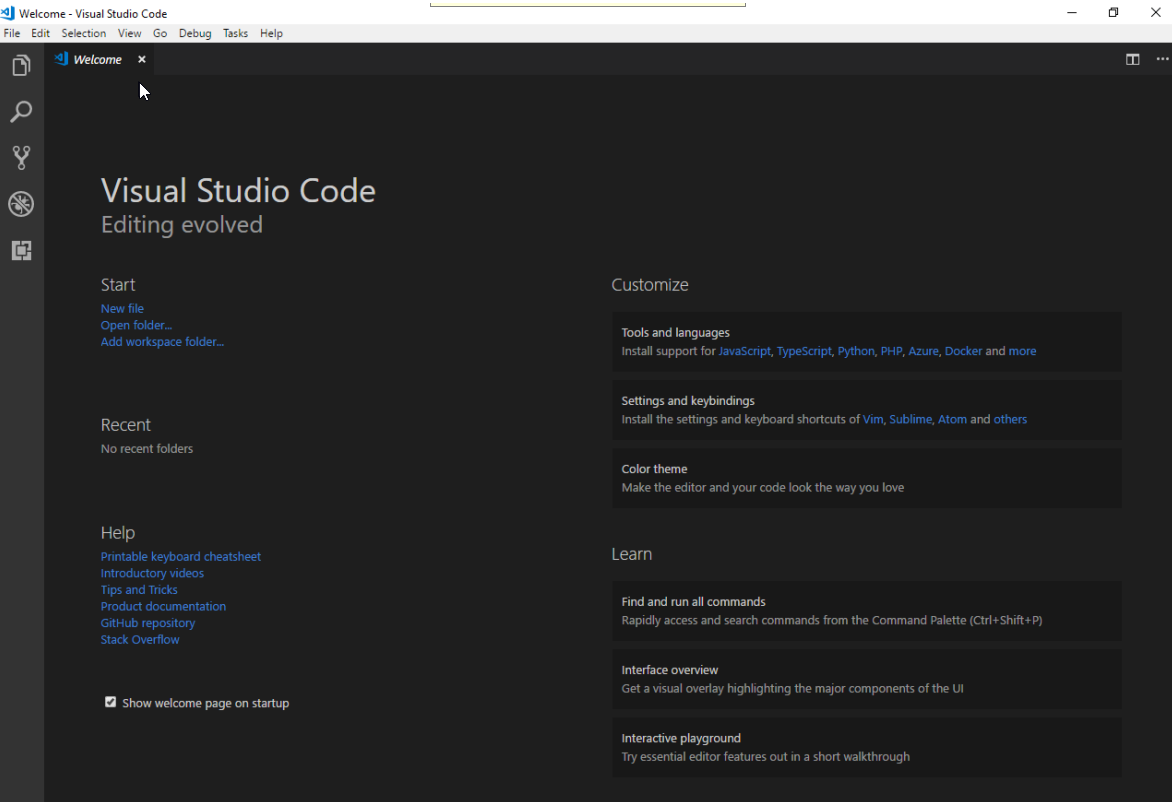
## How to execute test cases on Visual Studio Code

### Install Visual Studio Code

* Go to <https://code.visualstudio.com>.

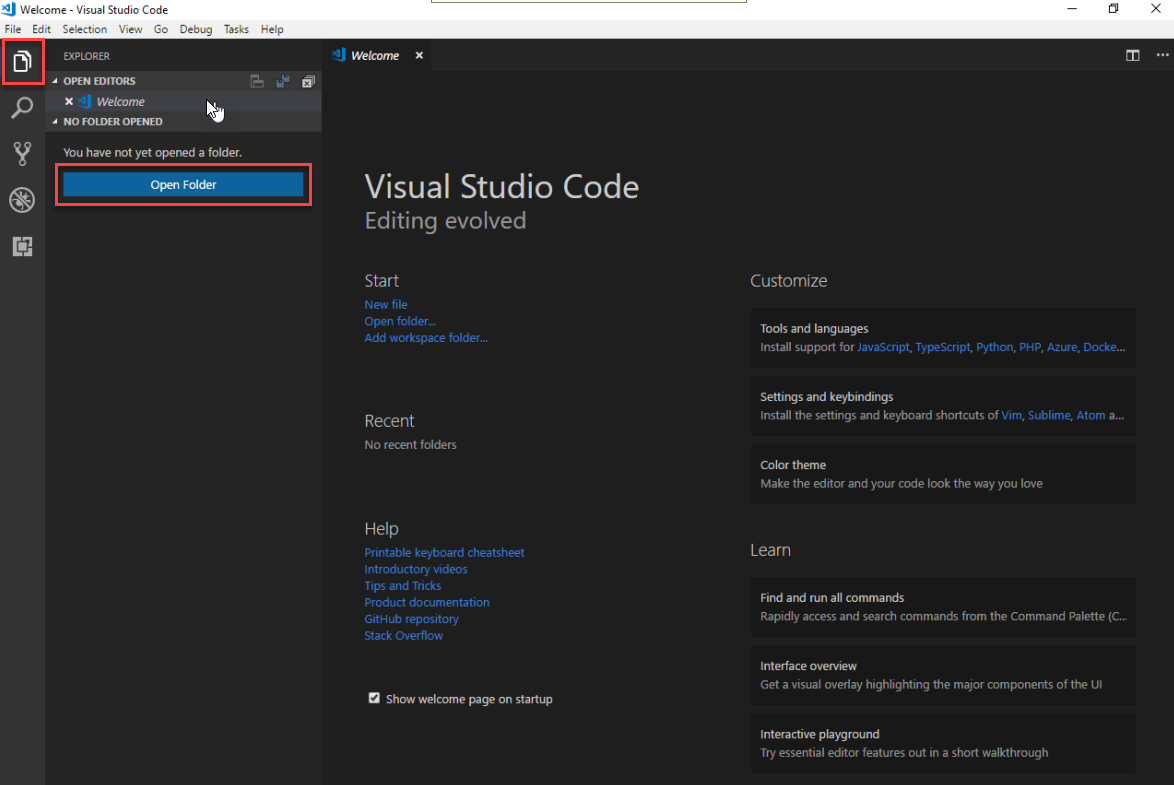


* Download and install Visual Studio Code. Open it after finishing.

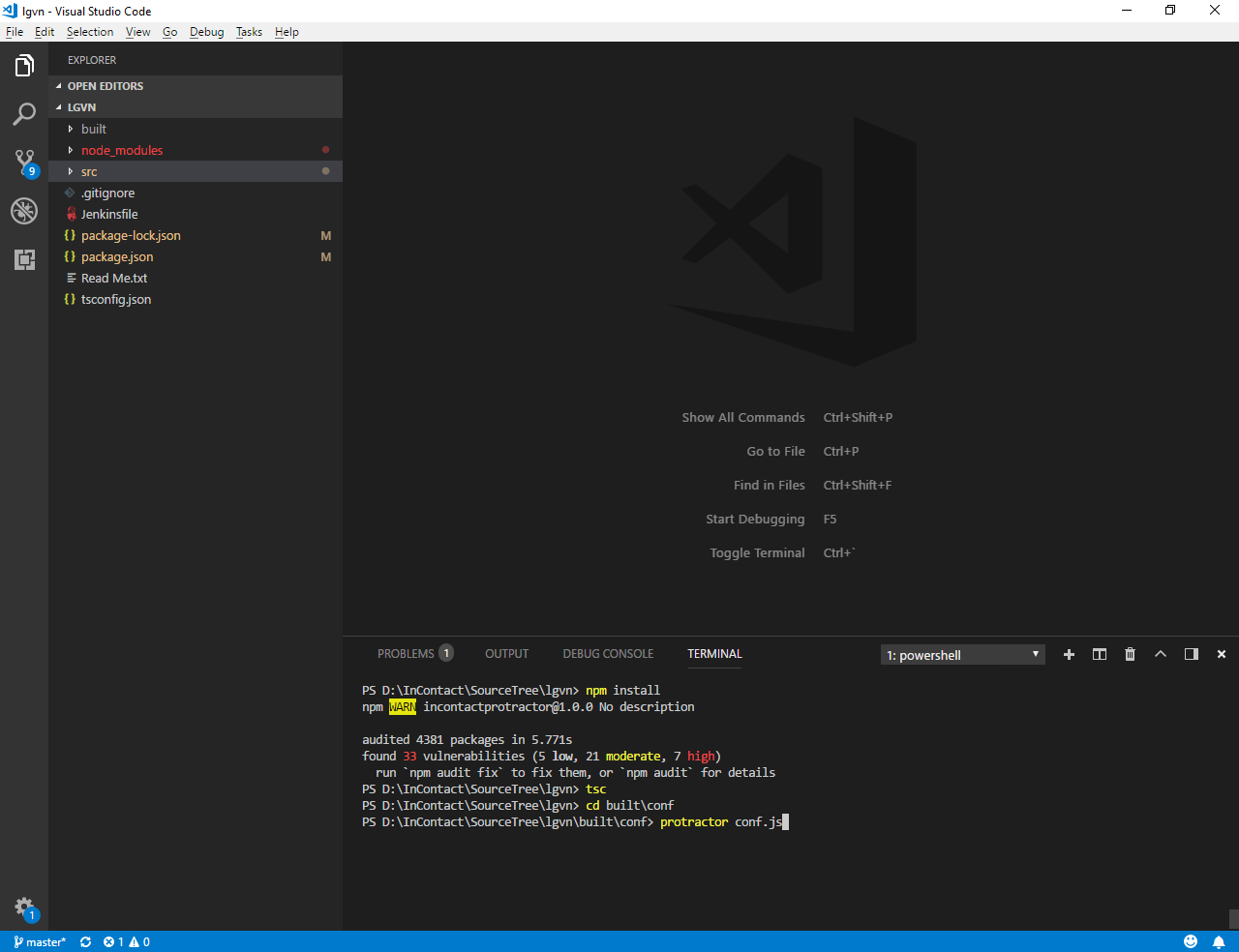


### Executing

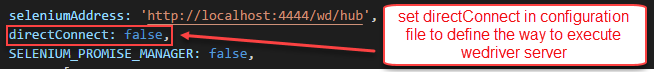
* Click on Explorer icon and opening folder that contains your source code



* Using command below to execute protractor:
  + npm install: install all required packages for running protractor.
  + tsc: transforming typescript into JavaScript.
  + cd build\conf: go to “conf” folder after transforming to execute the configuration file.
  + protractor “configuration file name” (protractor conf.js): executing configuration file.
    - protractor set-to31-conf.js: set cluster TO31 to run test cases. (change configuration file name to set other clusters)
    - protractor conf.js: run all evolve test cases on “conf” file.

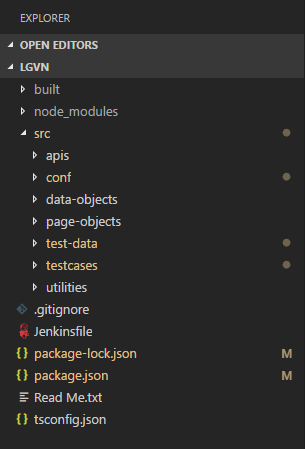


* **Note**: we need to start the “*Selenium Server*” before executing test cases.
  + Go to configuration file (/src/conf/conf.js)
  + Modify “*directConnect*” value:
    - true: start server directly. We don’t need any setup step.
    - false: we need to use command “*webdriver-manager start*” to start Selenium Server.



# Working With Protractor and SourceTree

## Protractor Structure



### build

* Contain all codes after transforming from typescript to JavaScript.
* This is a place where we use to execute test script.

### node\_modules

* Contain all required packages needed for project.
* It’s automatically installed and updated after using command “npm install”.

### src

* This is main place where we are working on.
* **apis**: contain all classes related to API.
* **conf**: configuration file use to run specs.
* **data-object**: object use to get or set data in project (agent, employee, tenant, etc.)
* **page-object**: contain locators and methods in one page (login page, employee page, etc.)
* **test-data**: contain all data use in project
* **testcases**: contain test cases
* **utilities**: contain all general actions (methods) which is used in project

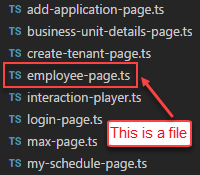
### Others

* **package.json** and **package-lock.json**: manage packages in project
* **tsconfig.json**: typescript configuration file.

## Protractor naming convention

### Files

* All characters should be lowercases.
* Should have dash “-“ between two words.



### Classes

* Use PascalCase (uppercase first letter of each word).
* No space or any special character between two words.

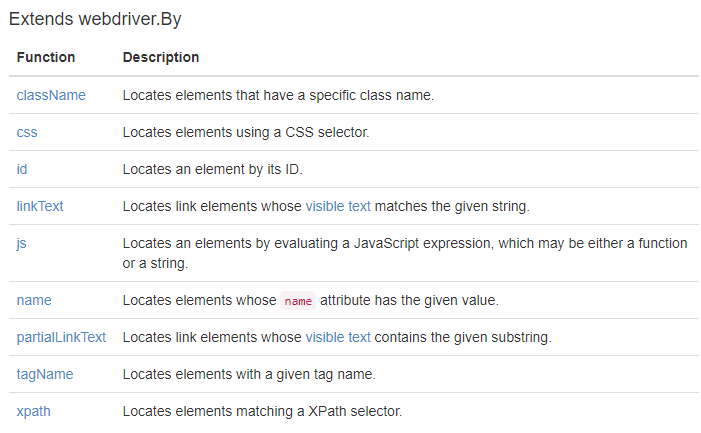


### Elements

* Use camelCase (first letter of first word is lower case, and first letter of each word after that should be uppercase).
* No space or any special character between two words.
* Use “protected” for access modifier.
* Should have element type before each word (button: btn, label: lbl, combobox: cbb, etc.)

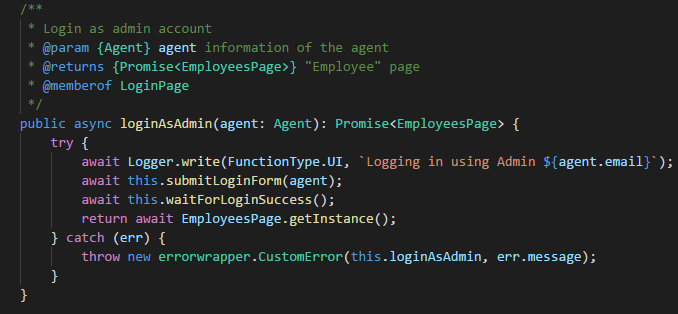
Element should be captured using **Webdriver.By**  with **Xpath** as first priority:





### Methods

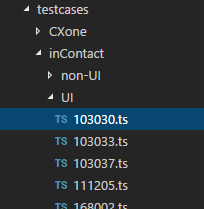
* Use camelCase (first letter of first word is lower case, and first letter of each word after that should be uppercase).
* No space or any special character between two words.
* Use “public” for access modifier.
* Having “Logger” for methods performing action steps (Going to…, Selecting…, Clicking on…., etc.)
* Having try…catch for all methods.
* Should have clear and detail document for each action.
* Method’s name should start with the verb.



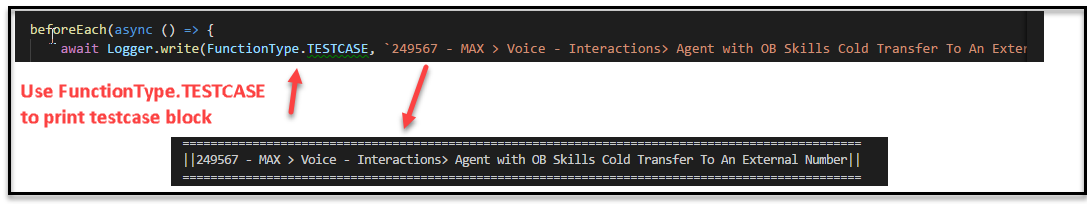
### Test cases

### Test cases Name

* One file should contain only one test case.

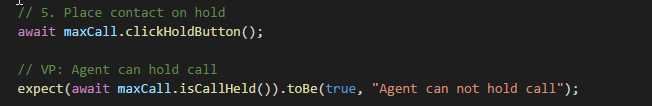


* Should have “Logger” for test case’s name.
* Should use ‘FunctionType.TESTCASE’ for test case name.
* Should put test case name at the beginning of before Each.



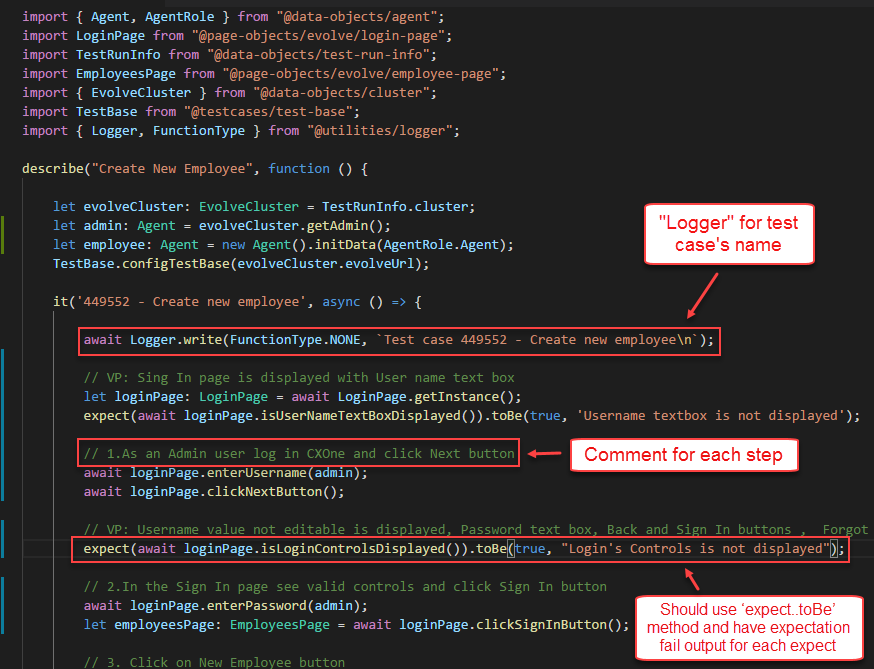
### Comment in test case

* Should have comment for each step and verify point.
* Should have comment for additional wait.

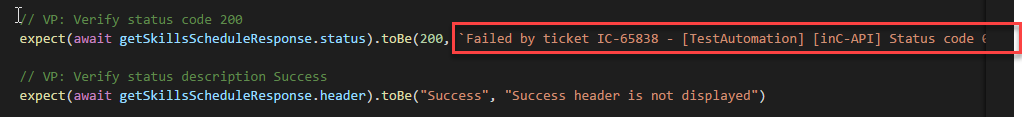


### Verify point in test case

* Should follow Page Object Model design pattern.
* Should use ‘expect...toBe’, ‘epect…toContain’, etc. methods and have expectation fail output for each expect.

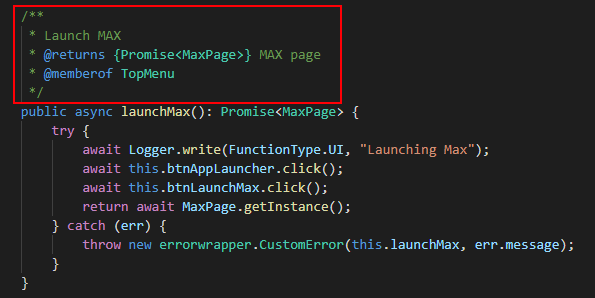


* Should add ticket for known issue



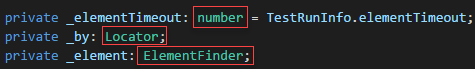
## Coding conventions vs Programming style

* All methods should have document.

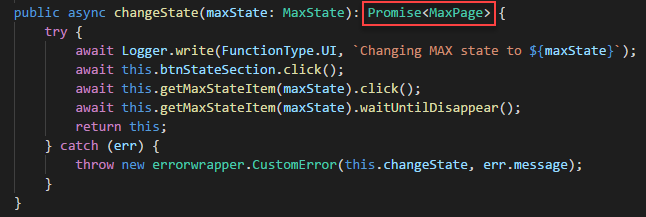


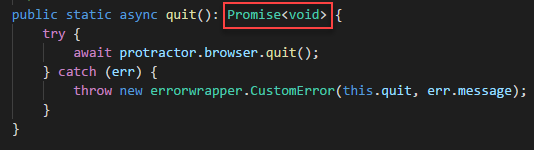
* All variables should have primitives type.



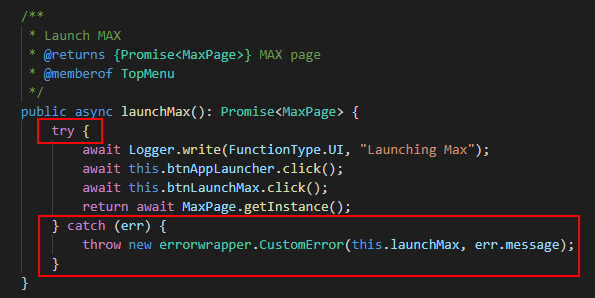


* All methods should have return type.

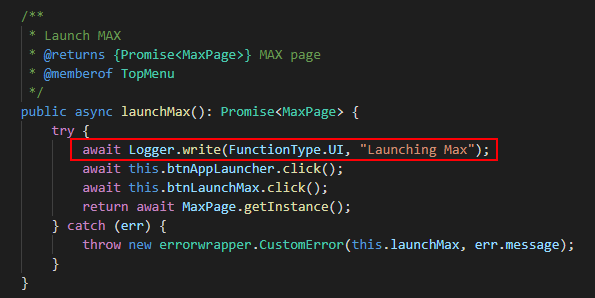


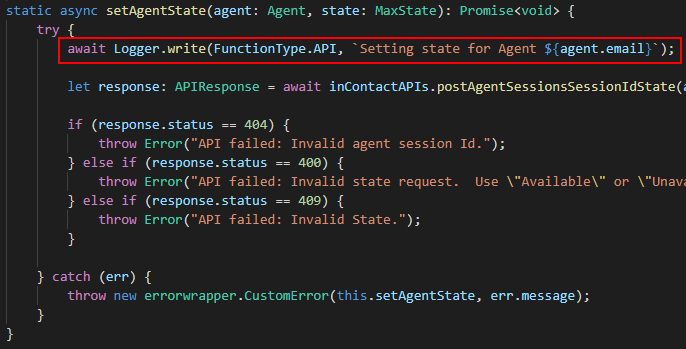


* All methods should have try and catch.



* All UI and API methods should have logger.





* Only use ***let*** instead of ***var*** for assigning references.
* Should use back quote (`) for string which contain parameter. Ex: `The username should be ${username}`
* Use // for single line comments. Place single line comments on a newline above the subject of the comment. Put an empty line before the comment unless it’s on the first line of a block.
* Use /\*\* ... \*/ for multi-line comments.
* Following prefix type of element name when capturing locators:

**\*Note**: Converting elements which have different type than usual but same function to normal "Element Type". Ex: element type: span, div, header, and paragraph --> label.

* + **Common Type:**

|  |  |  |
| --- | --- | --- |
| **Element type** | **Abbreviation** | **Description** |
| button | btn | Element seems like a button |
| label | lbl | All element type is **text** type |
| textbox | txt |  |
| link | lnk |  |
| checkbox | chk |  |
| radio button | rdo |  |
| combobox | cbb | A combination of an input text field and a list of options. You can type in stuff in the input field and the list should update to highlight a possible match |
| list box | lb | Allows the user to select one or more items from a list contained within a static, multiple line text box |
| dropdown list | ddl | A list of options that drops down when you click on the arrow button. You can only choose from a list of predefined options |
| table | tbl |  |
| image | img |  |
| dialogbox | dlg |  |
| frame | frm |  |
| date picker | dtp |  |
| text area | txa |  |
| popup | pop |  |
| Icon | Ico |  |
| div | div |  |
| list | lst |  |

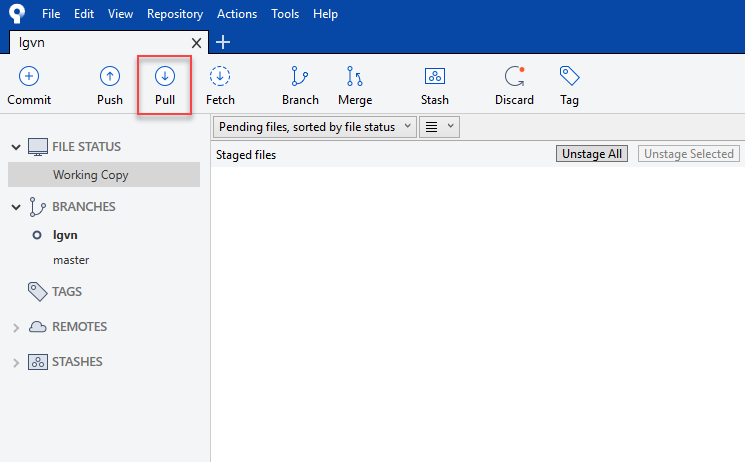
* + **Complex Type:**

|  |  |  |
| --- | --- | --- |
| **Element type** | **Abbreviation** | **Description** |
| menu item | mi |  |
| menu | mn |  |
| message box | msg |  |
| main window | mw |  |
| push button | pb |  |
| progress bar | pgb |  |
| page list | pl |  |
| panel | pnl |  |
| popup list | pul |  |
| radiobutton group | rg |  |
| radio list | rl |  |
| scroll bar | slb |  |
| tab | tab |  |
| text field | tf |  |
| toggle button | tg |  |
| toolbar | tlb |  |
| tree view | trw |  |
| tool tip | tt |  |

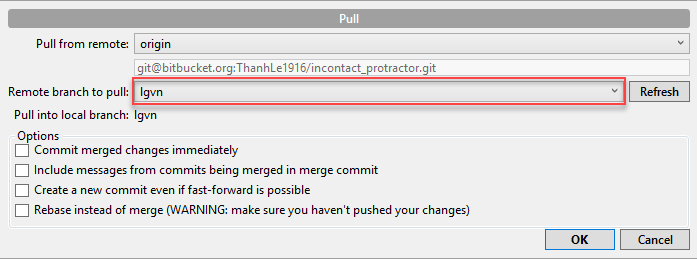
## Working with SourceTree

### Pull code

* Click on “Pull” button

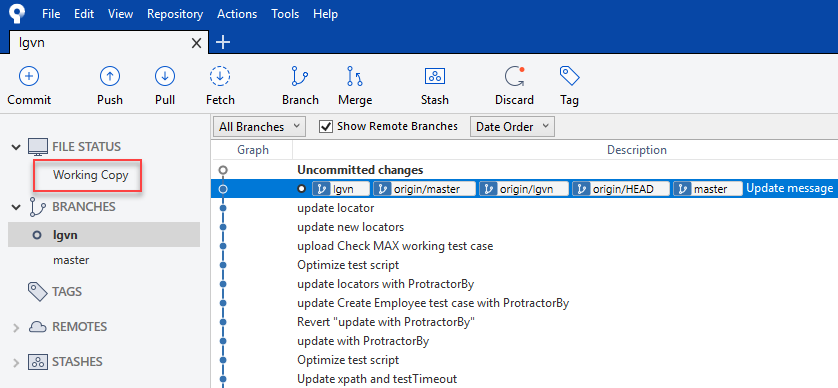


* Select branch for pulling and click “OK” button

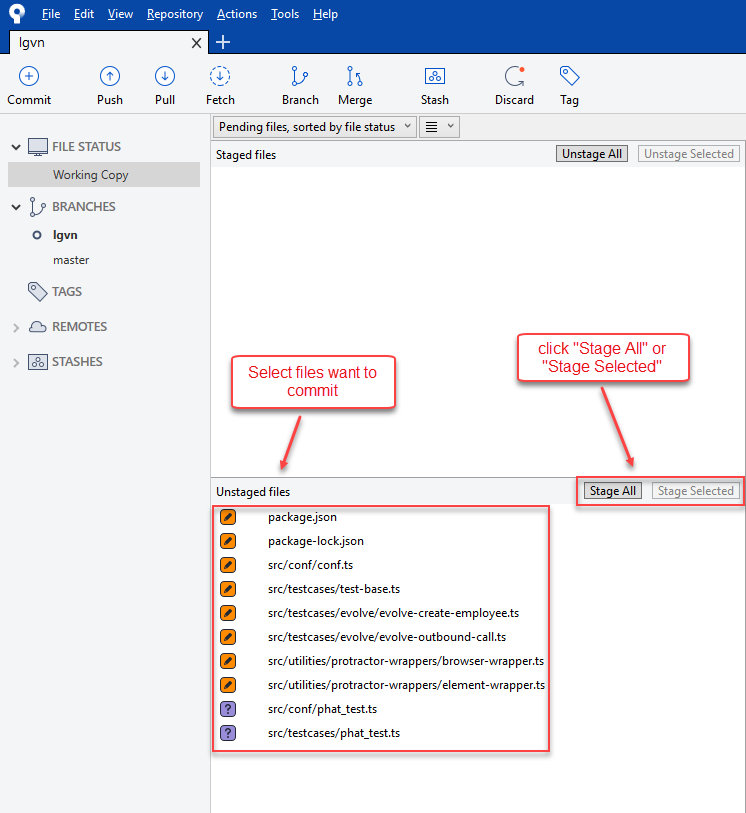


### Push code (we need to pull code before pushing new code)

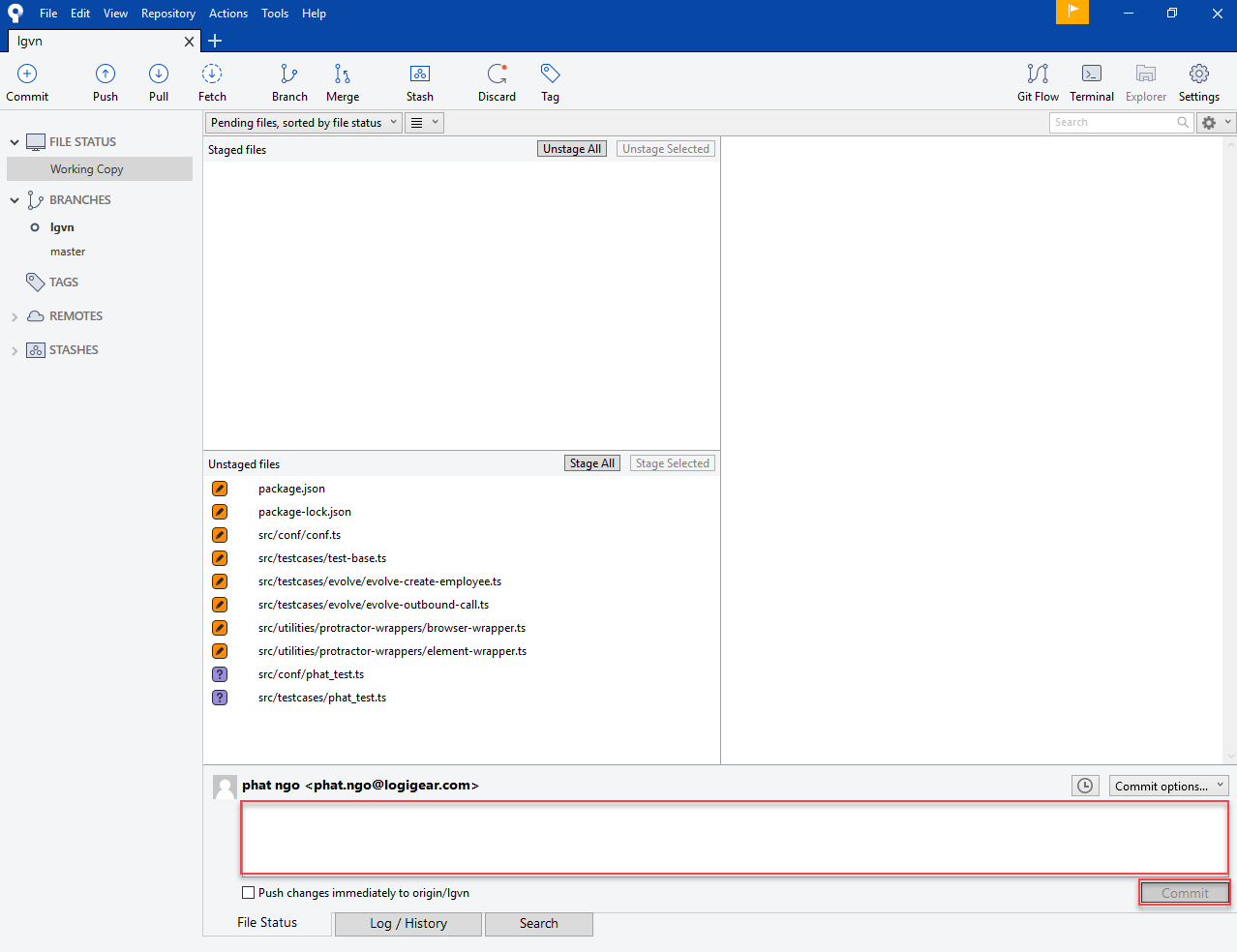
* Click on “Working Copy”



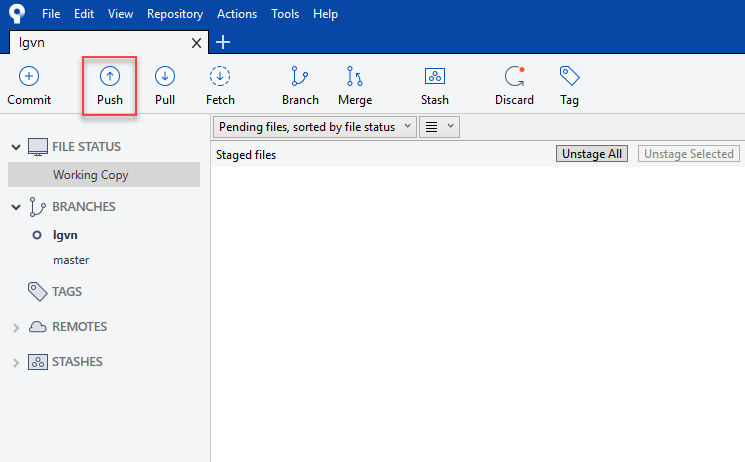
* In “Unstaged” files window, click on “Stage Selected” to stage one file or “Stage All” to stage all files



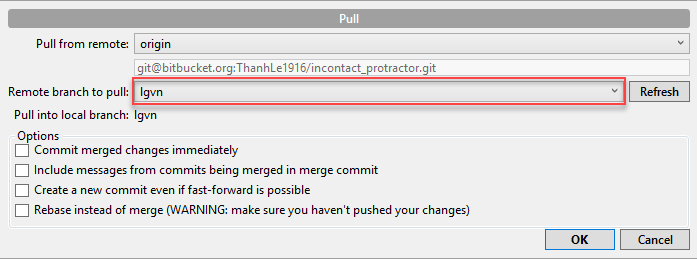
* Input comment and click on “Commit” button



* After commit, click “Push” button

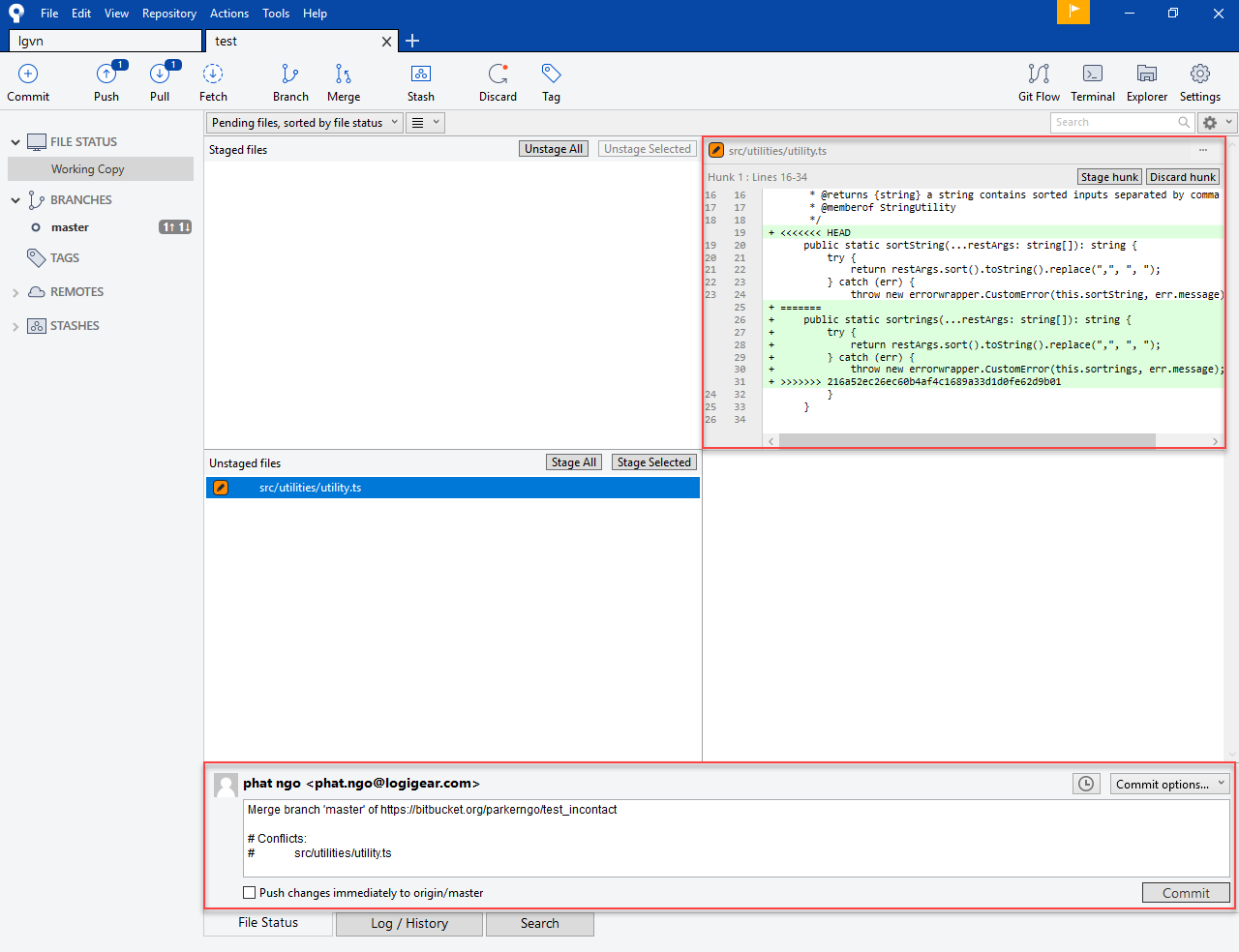


* Select branch for pushing code and click on “Push” button

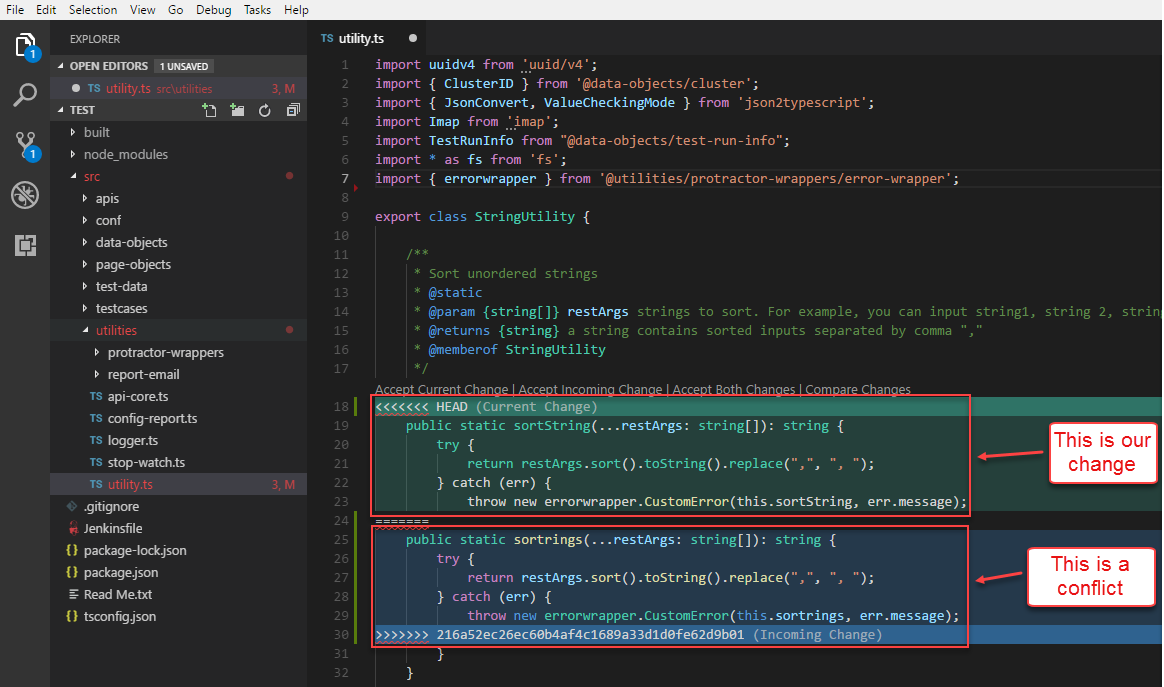


### Resolve Conflict

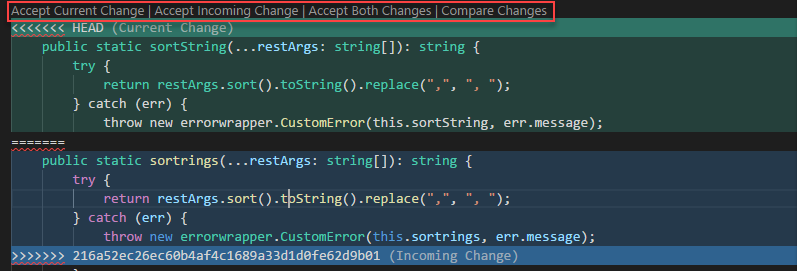
* If we see the Working Copy window as below after pulling code. You need to resolve the conflict.



* We can use Visual Studio to do that.
  + Check the difference between our code and a conflict.



* + Choose “Option” that best fix in your case.



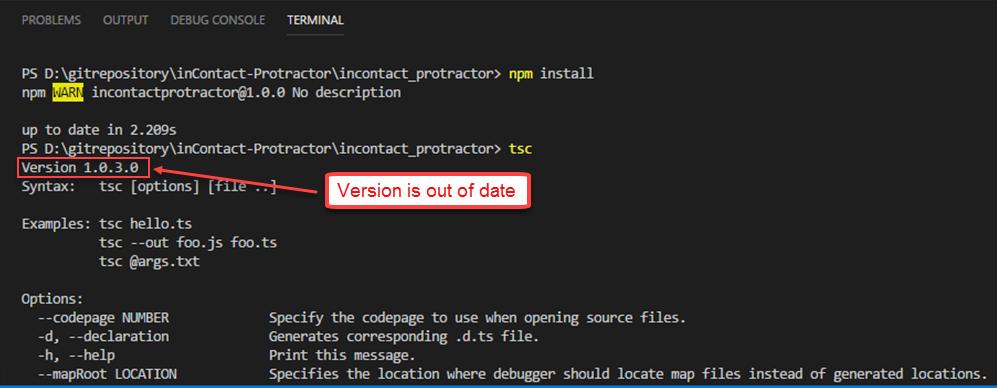
* + Save the change and follow steps to Push code.

# Troubleshooting

## Unable to use command “*tsc*”

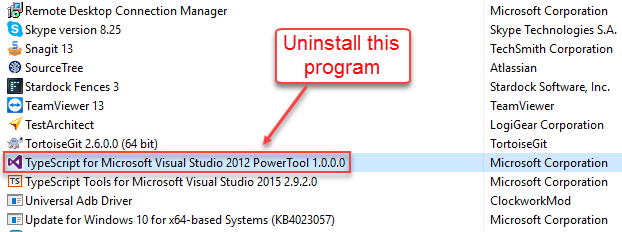
### Error Behavior

* The protractor will tell us we are using wrong syntax of “*tsc*” command and display old version.

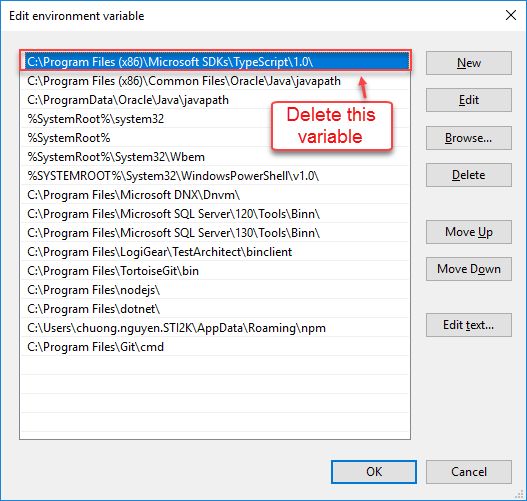


### Solution

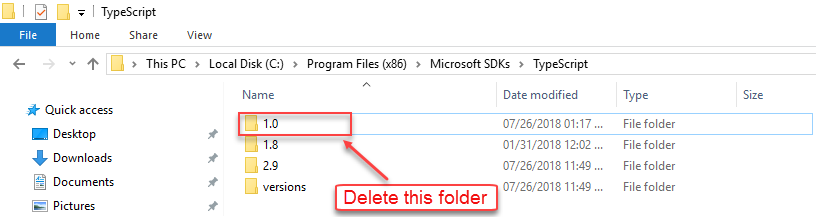
* There is a version of typescript was installed before(install manually by using plugin of Visual Studio or automatically install when installing Visual Studio) and system variable is set to this typescript’s PATH. All we need to do is remove this typescript and re-install latest version using ‘*npm*’.
  + Open ‘Control Pane -> Programs and Features’ and find ‘TypeScript for Microsoft Visual Studio…’. If you found that program, please uninstall it.



* + Open ‘System Variables’ and delete Microsoft typescript variables if it exists.



* + Go to ‘C:\Program Files (x86)\Microsoft SDKs\TypeScript’ folder and delete ‘1.0’ folder if it exists.



* + After remove all things of old typescript, please restart your machine to make it effective.
  + Hit command “npm install -g typescript@latest” to make sure we installed latest version.