**Import Automation Framework Project**

**Primary steps to complete Set Up Framework** on Local system

1. Download / Clone Java Projects from GIT HUB on local system
2. Open Eclipse and perform Open Eclipse ->File->Import

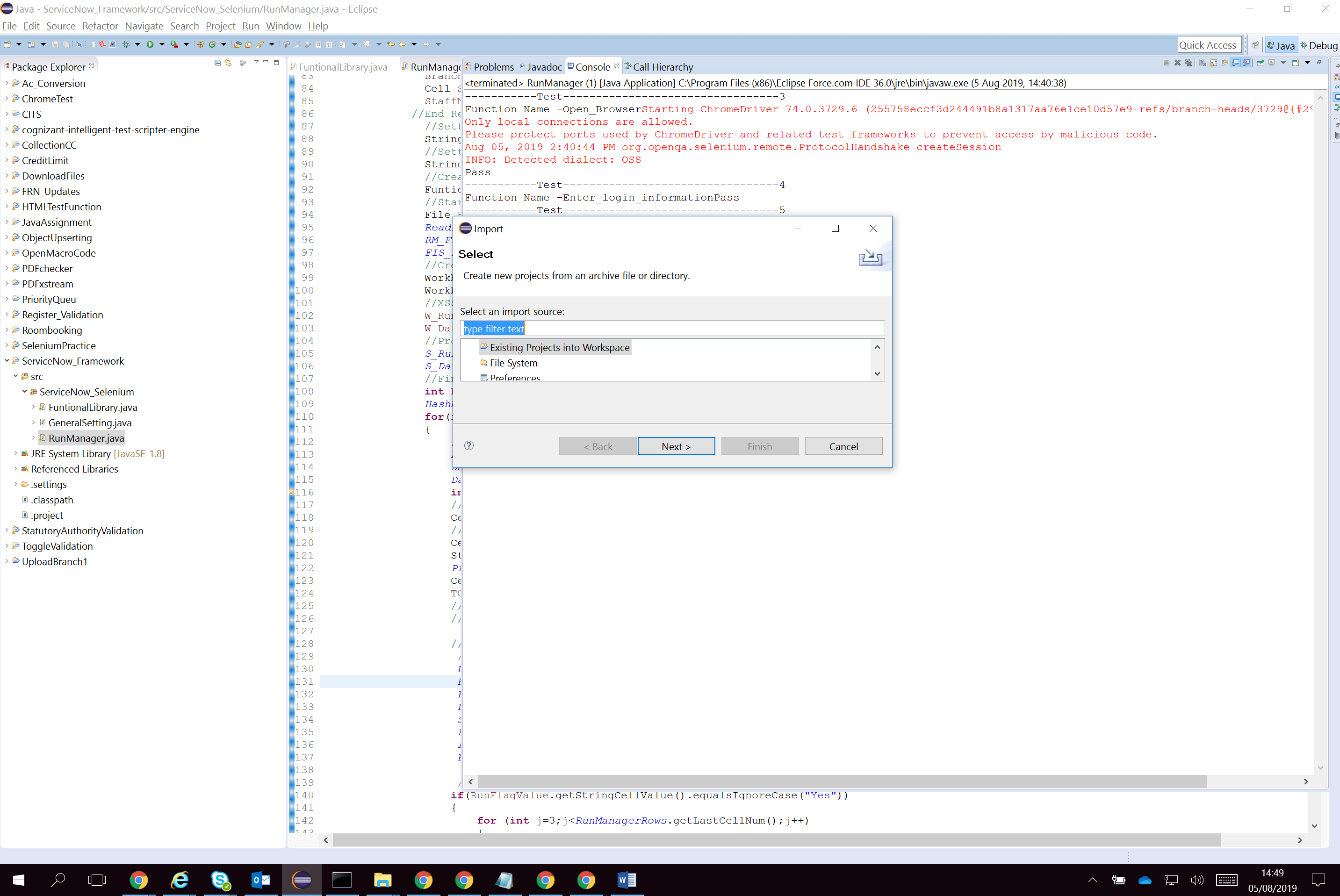
Import Project in Eclipse – To Import existing project under eclipse

Import SELENIUM and APACHE POI Jar Files as well

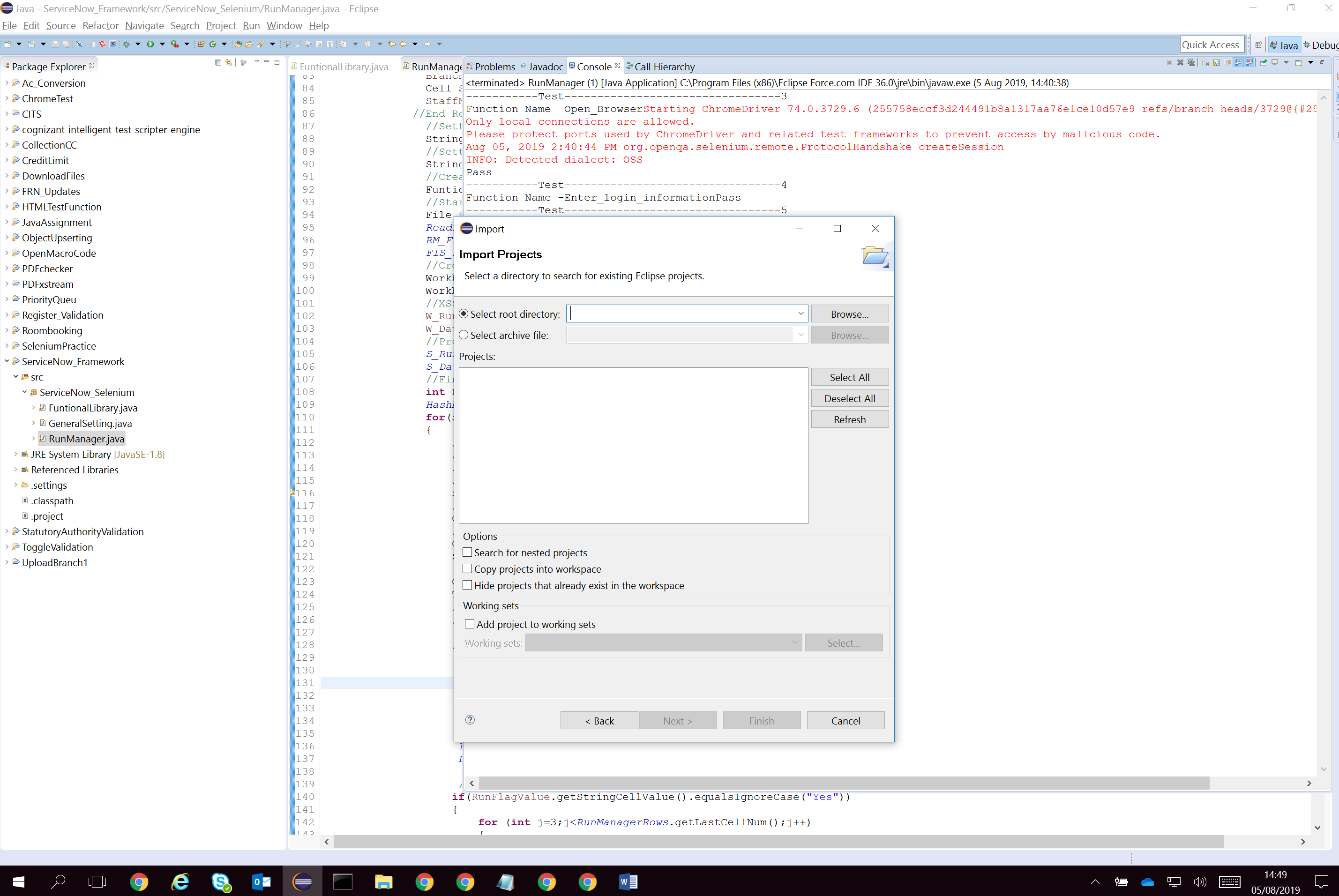
Step – Open Eclipse ->File->Import



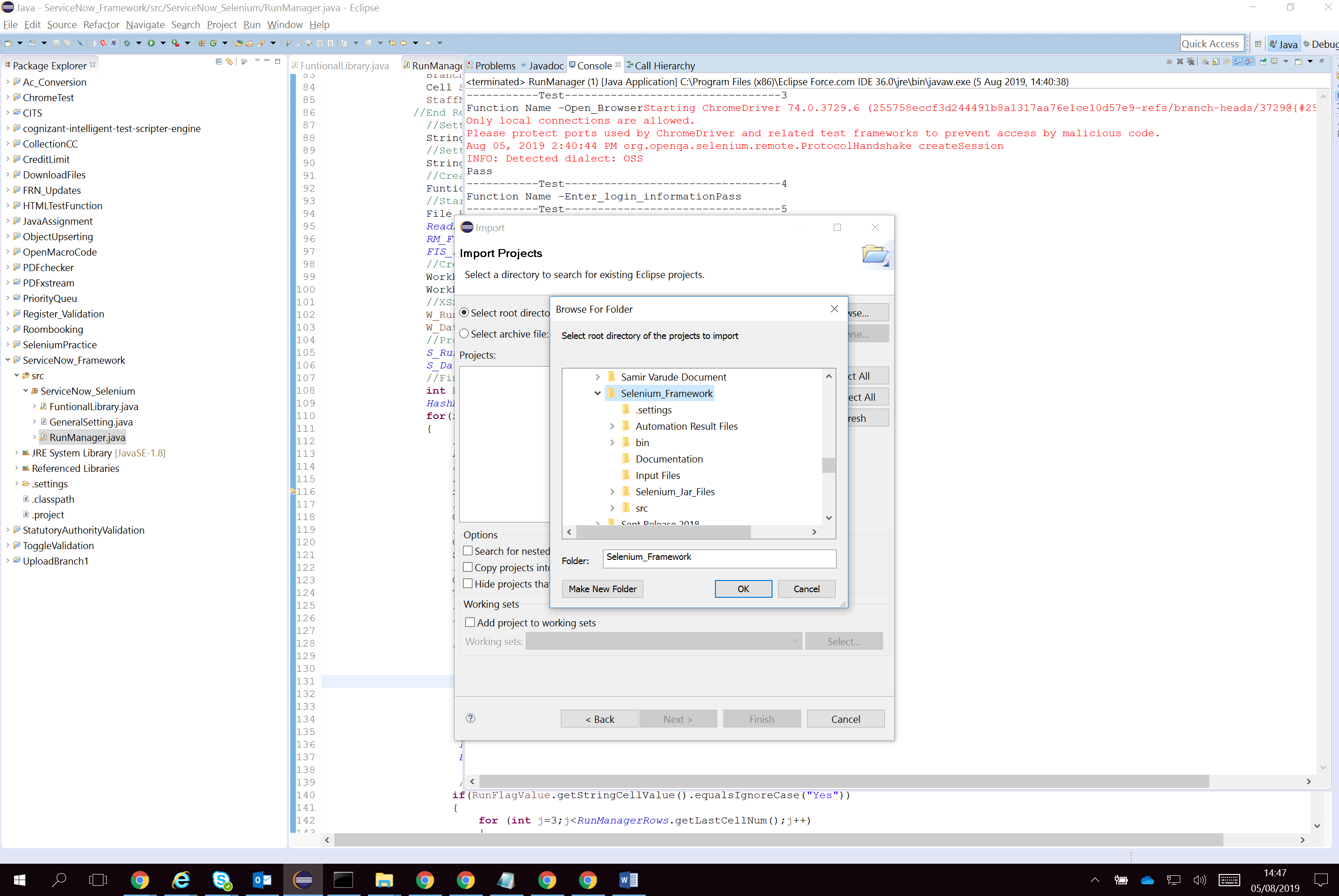
Select General ->Existing Projects into Workspace



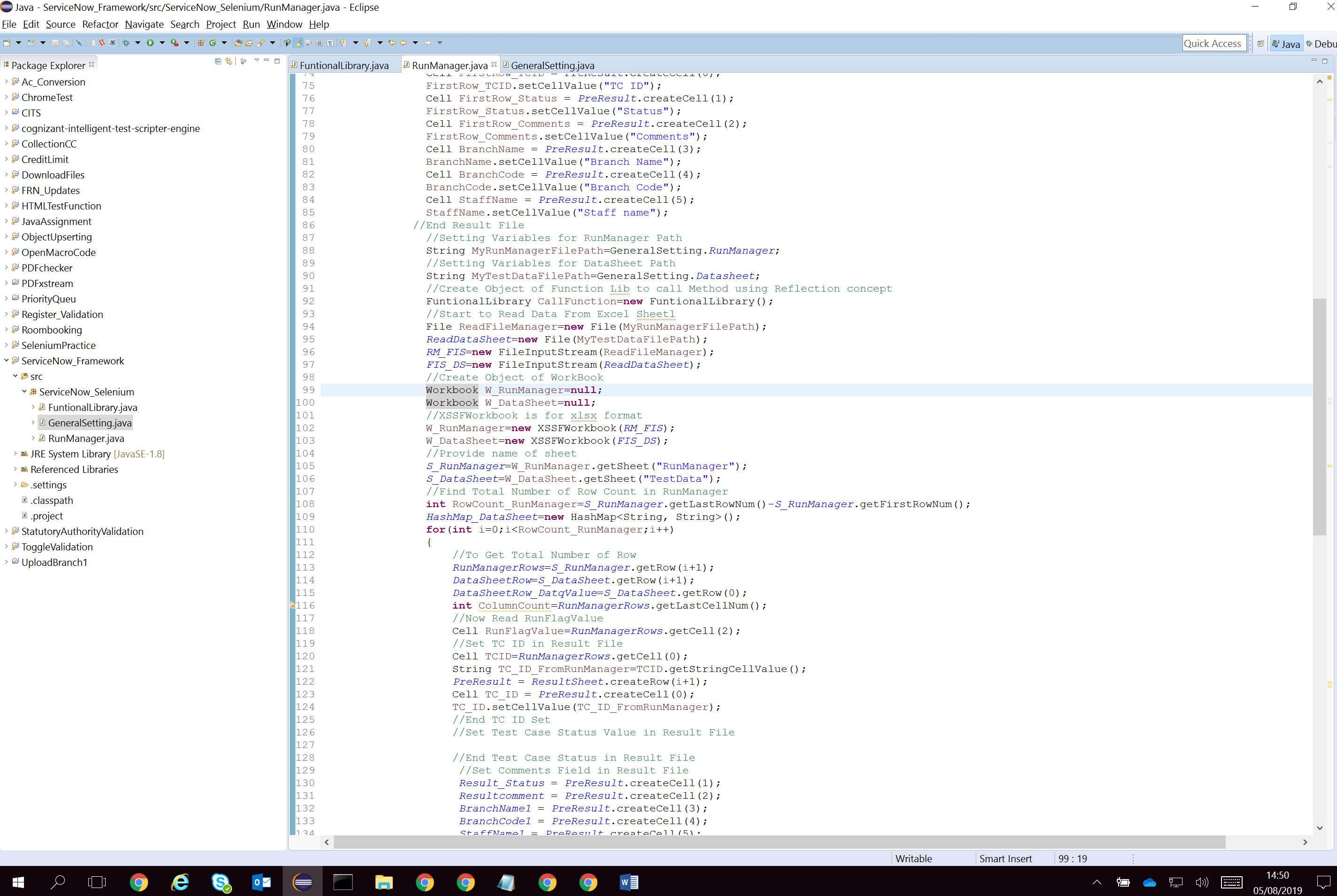
Click on Next button and select root directory



And Click on Finish button



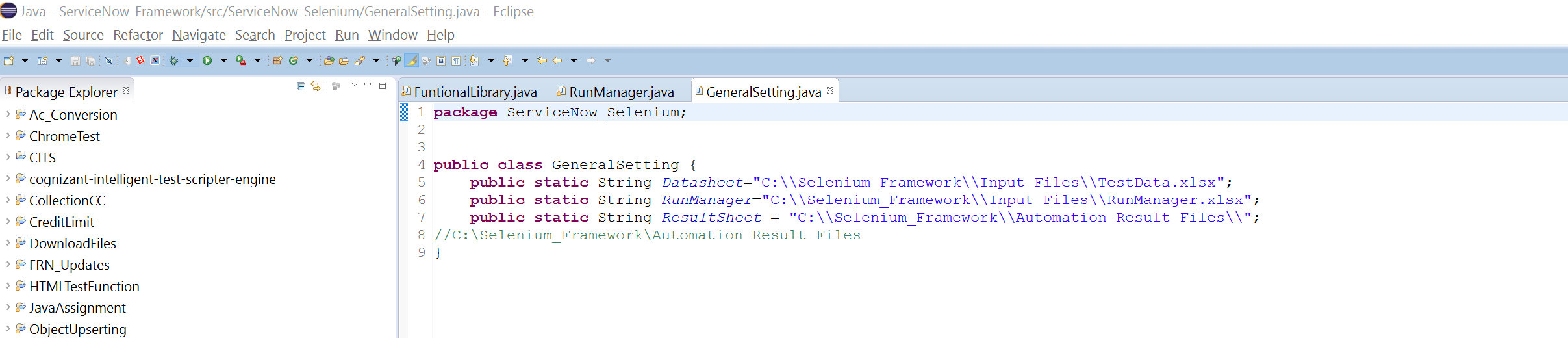
Project has been successfully imported



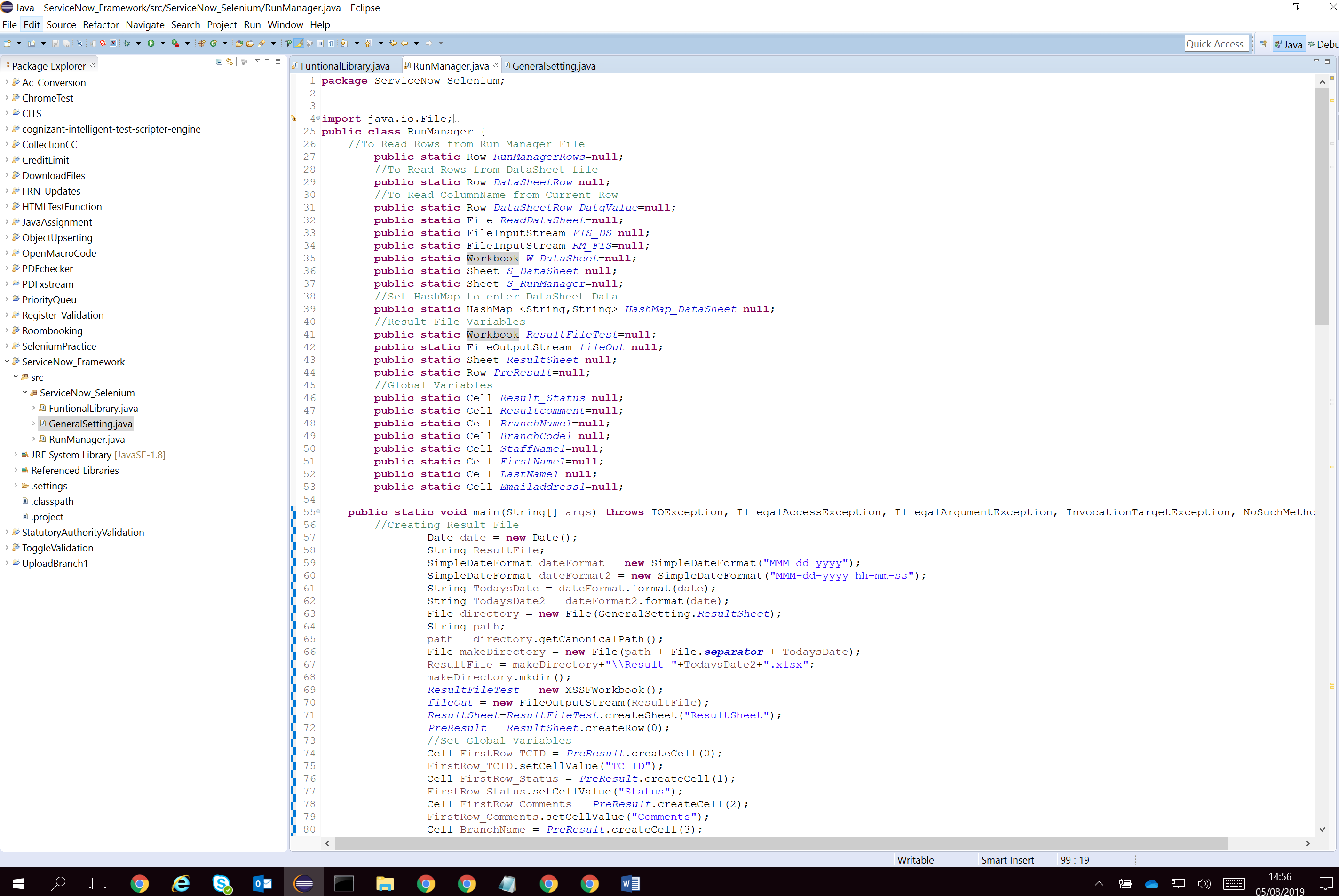
**Validate Framework Core Components**

1. **GeneralSettings Class**: This component is used to capture location of Test Data, Run Manager and Result File location.

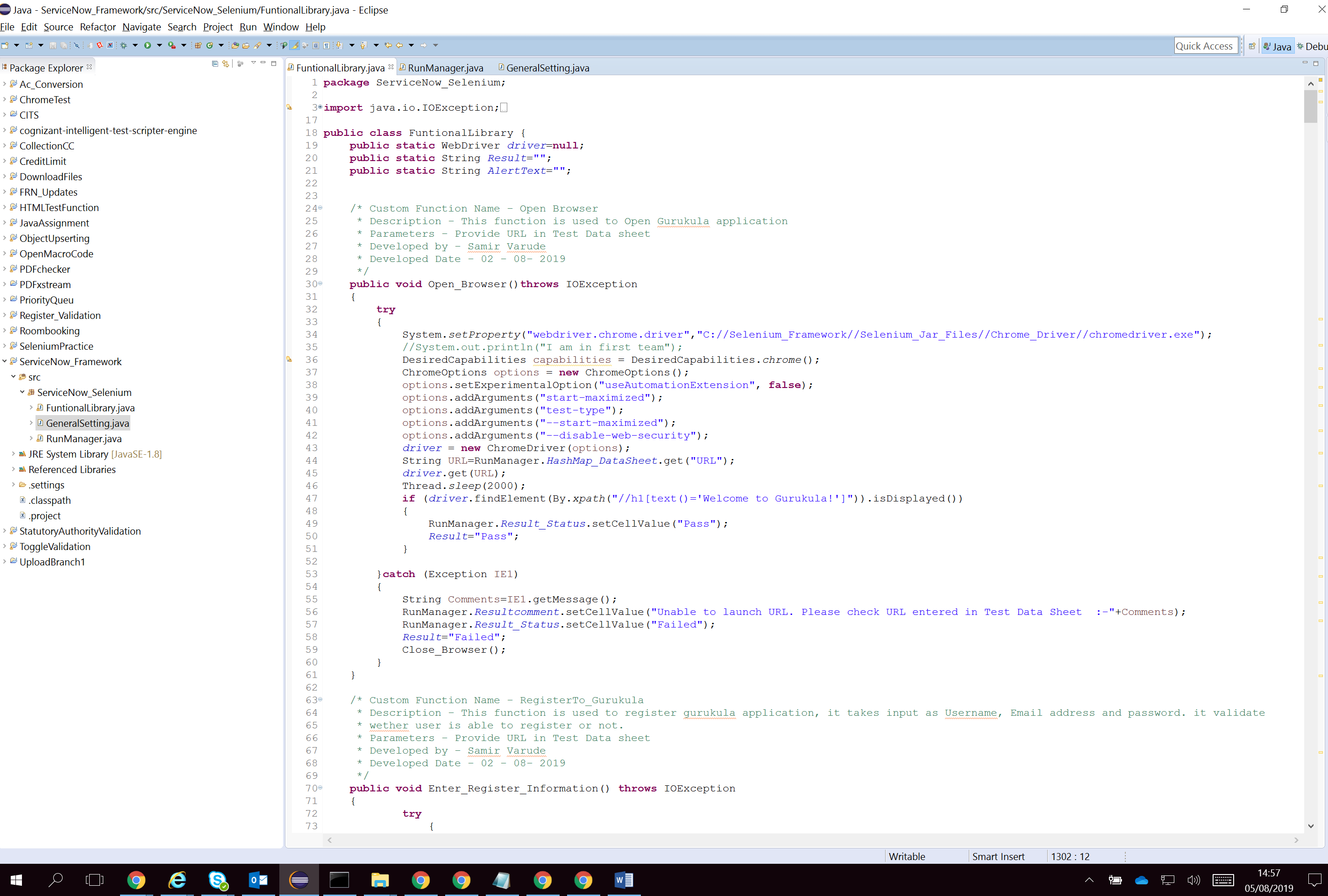
Open General setting class and set file location of Run Manager, Data sheet and Result file



1. **RunManager Class**: This component is core part of this framework. It reads Test data and Run Manager files and start execution of Test scripts. I have used, Apache POI jars to read and write data in Excels



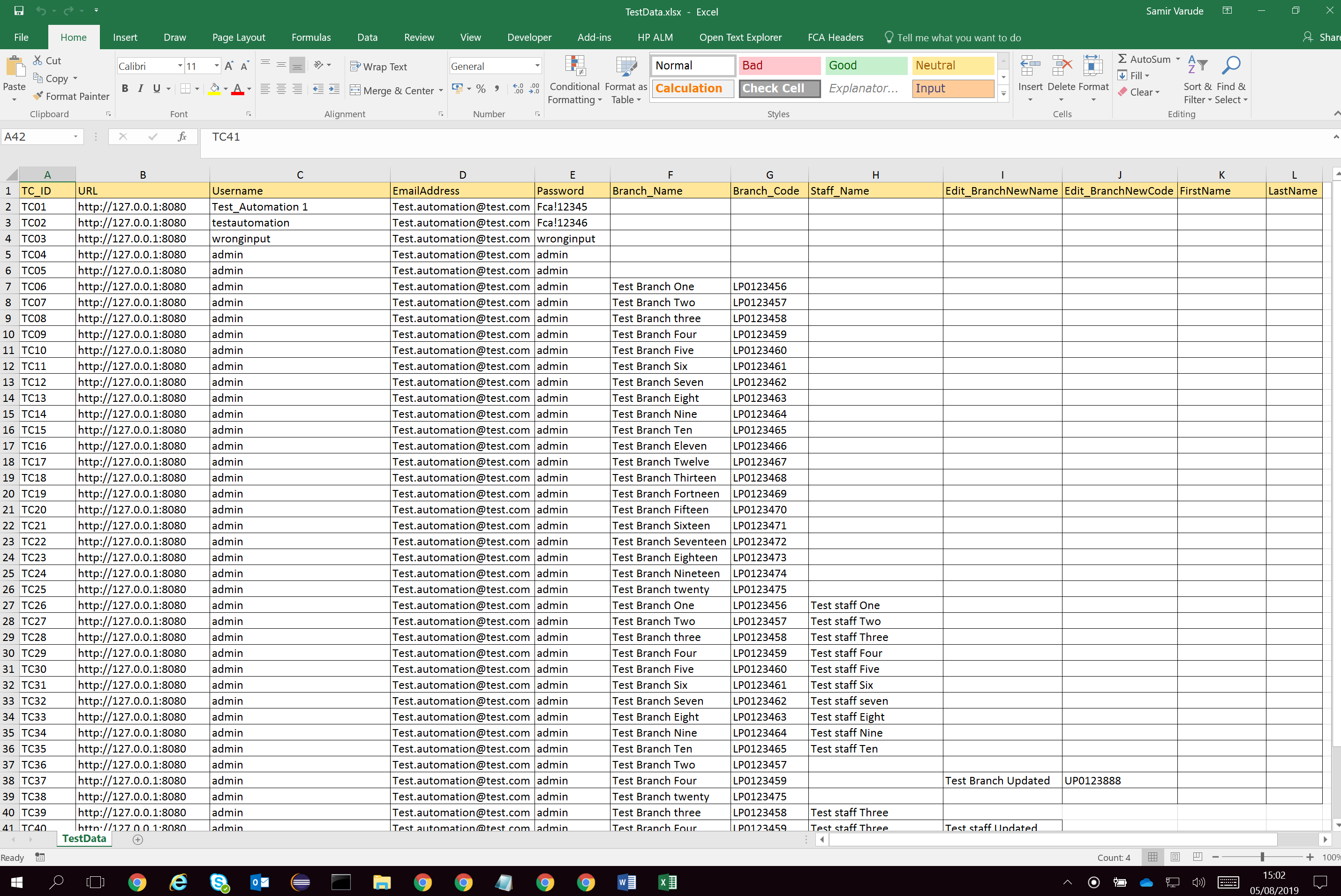
1. FunctionLibrary Class: This component is used to store all functions / Business components which required to complete test flow.



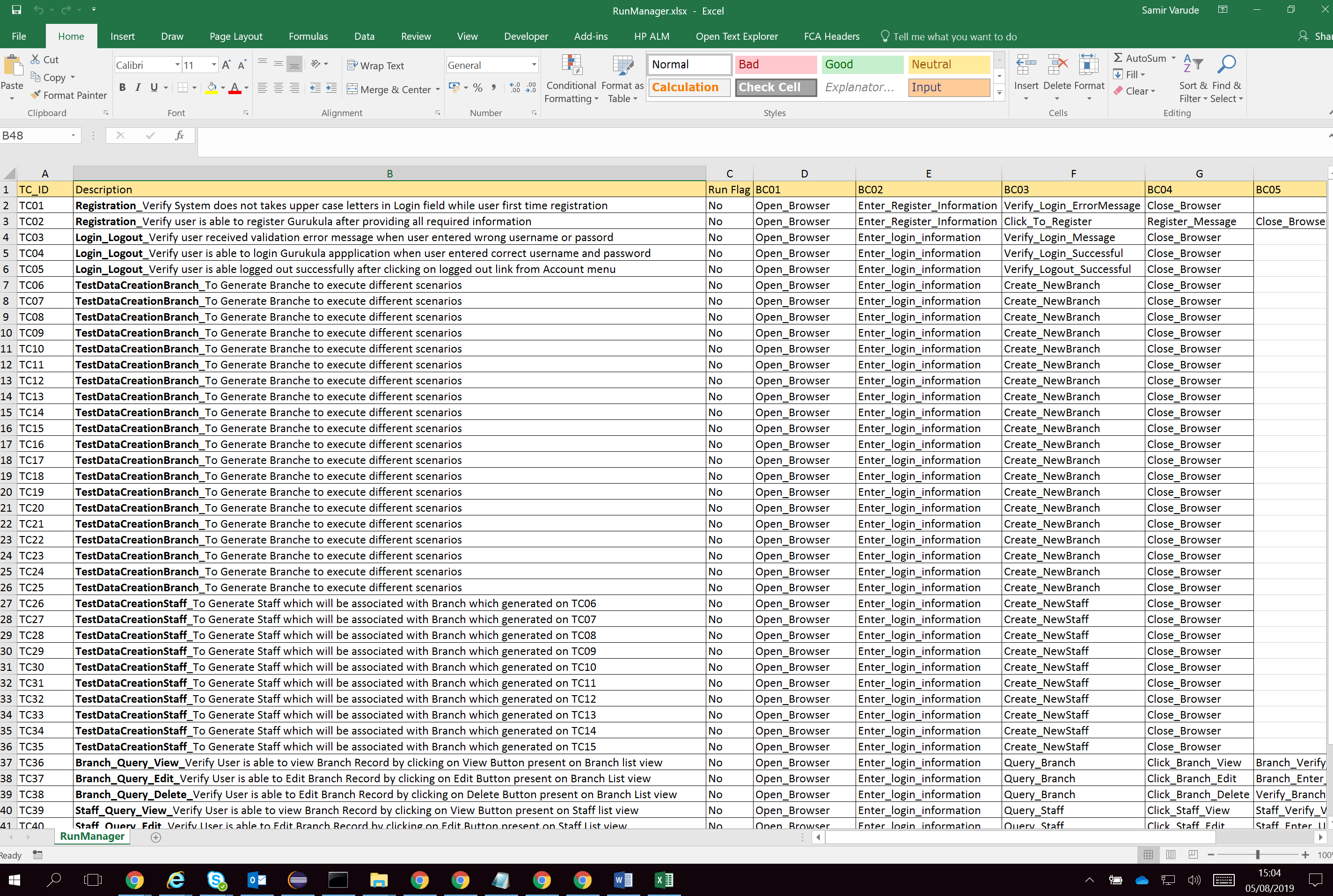
**Validate Input Files**

There are 3 types of Excel require to accomplish Test automation framework as this is keyword driven framework.

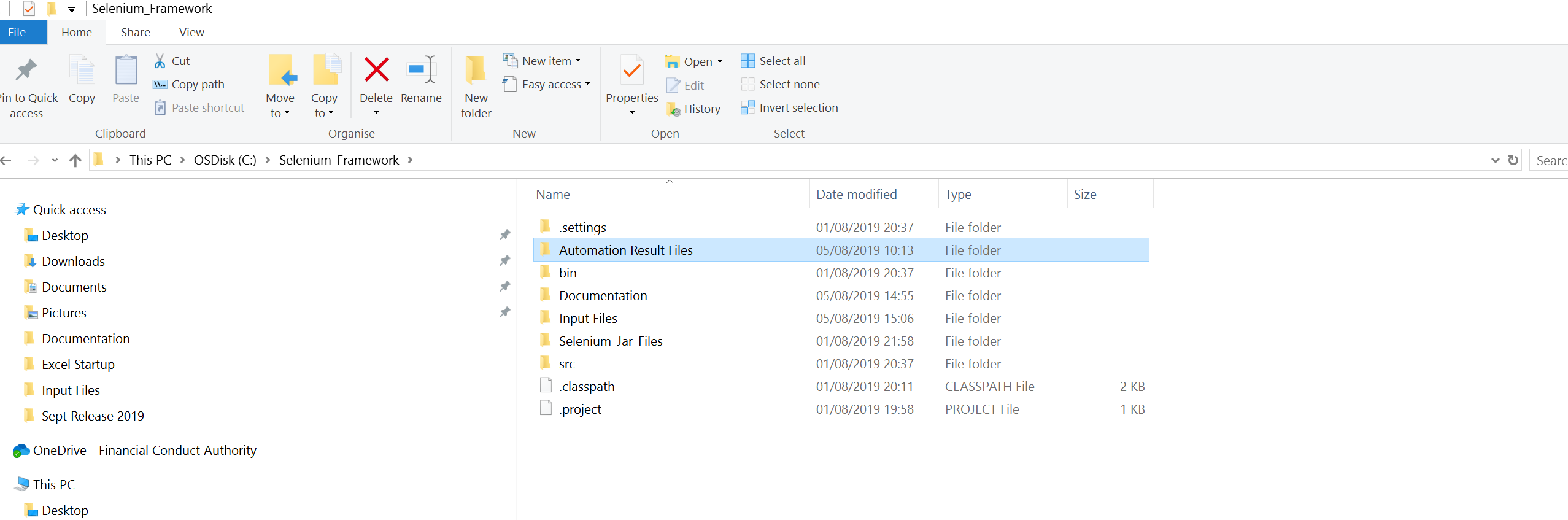
A] TestData Excel: This excel contains require test data to complete test case execution



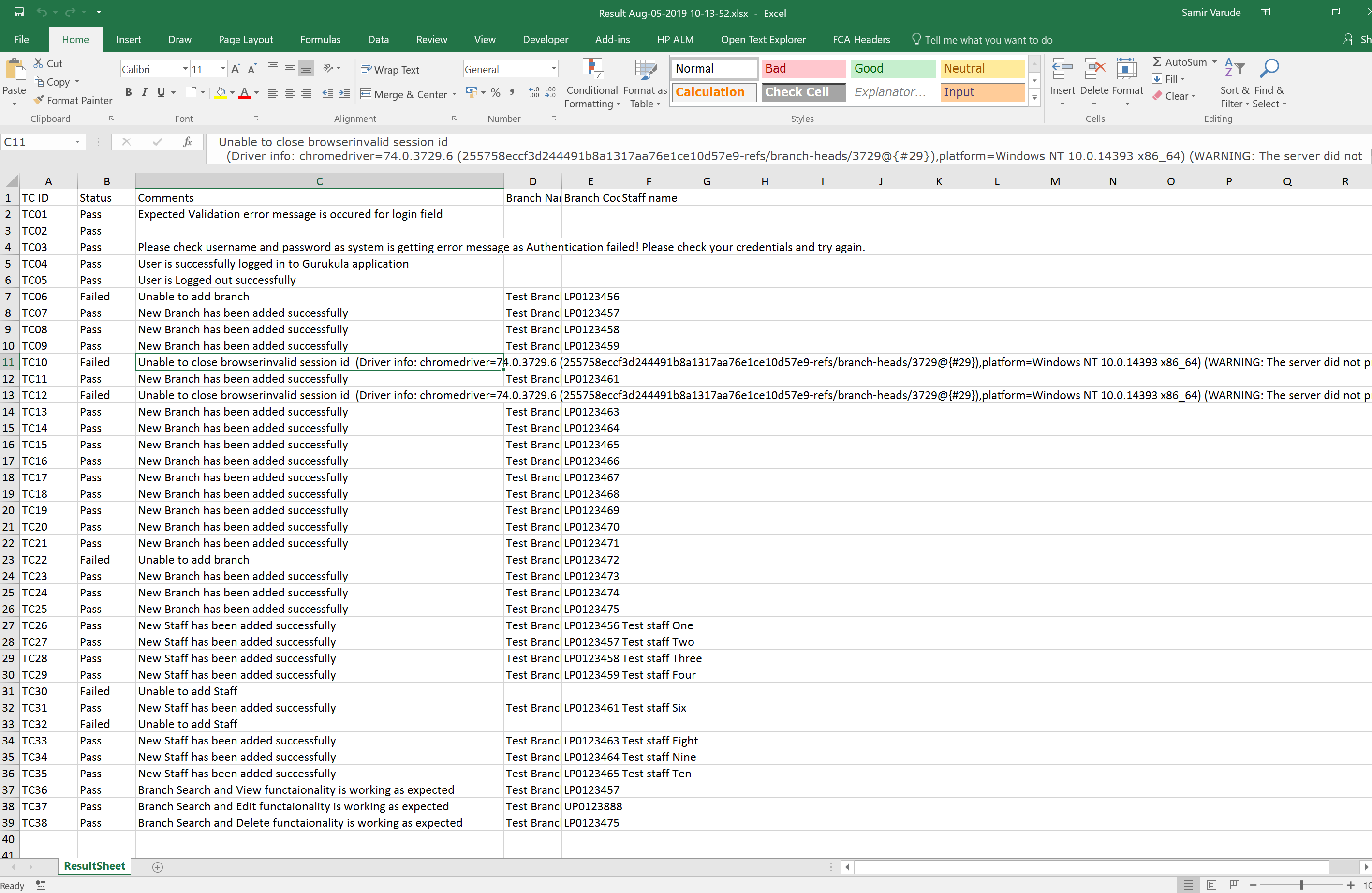
B] RunManager Excel: This excel contains configuration of Business components which will be executed to complete test case execution. User can decide test case execution priority by using Run Flag Column.



C] Result Excel: This file would be generated on each run as its result file. It will have all run time test data including execution statuses of all test cases

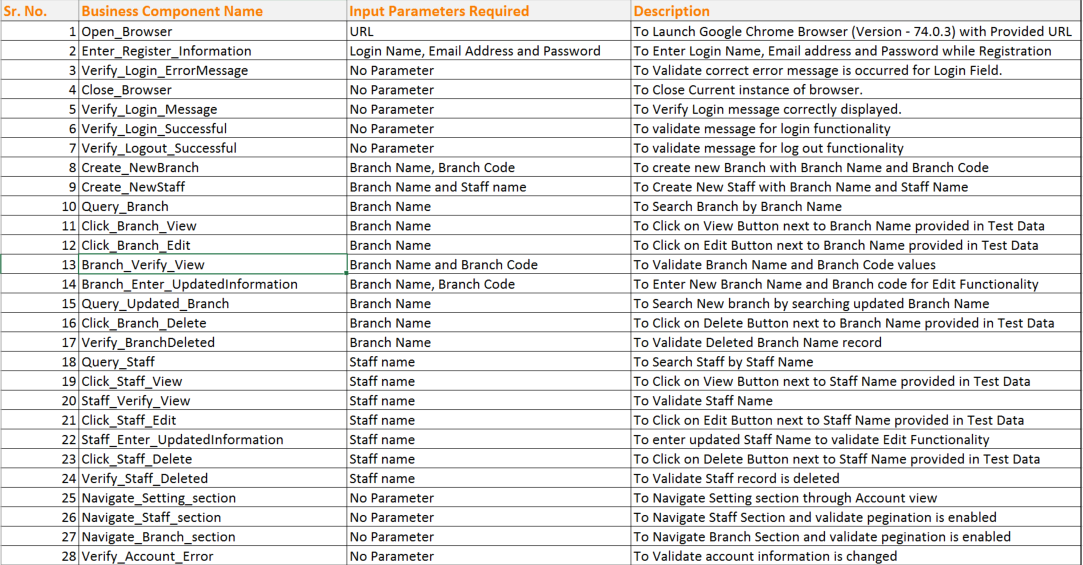


Test Execution report will be generated in Excel format and it will have all details including Test Case ID, Status, Comments (Pass or Failed) and generated run time data for review.



**Once Above mentioned set up is done then we need to run RunManager.JAVA file to kick off Automated script execution**

Custom Functions developed to complete Execution



Limitation of Automation Framework but can be achievable

* No HTML Report generated
* Screenshot feature is not available
* Test Iteration is not possible
* Email / PDF validation is not available
* Require custom functions to connect with JENKIS or Test Management Tool
* Require Java coding knowledge