Automation Framework using Appium

Version 1.0

Author: **Prabodh Kelkar** ([prabodh.kelkar@gmail.com](mailto:prabodh.kelkar@gmail.com) / +91-9850089719)

**Dependencies and Config details**:

App under test:

* eBay (downloaded from the link…

https://drive.google.com/open?id=1sWgqPr\_NF6N8vOYoBBjop7B9Cvrkg-KZ

Framework dependency:

* Java - 1.8.0\_161
* Appium – 1.7.2 or 1.8.X Beta

IDE used:

* IntelliJ IDEA

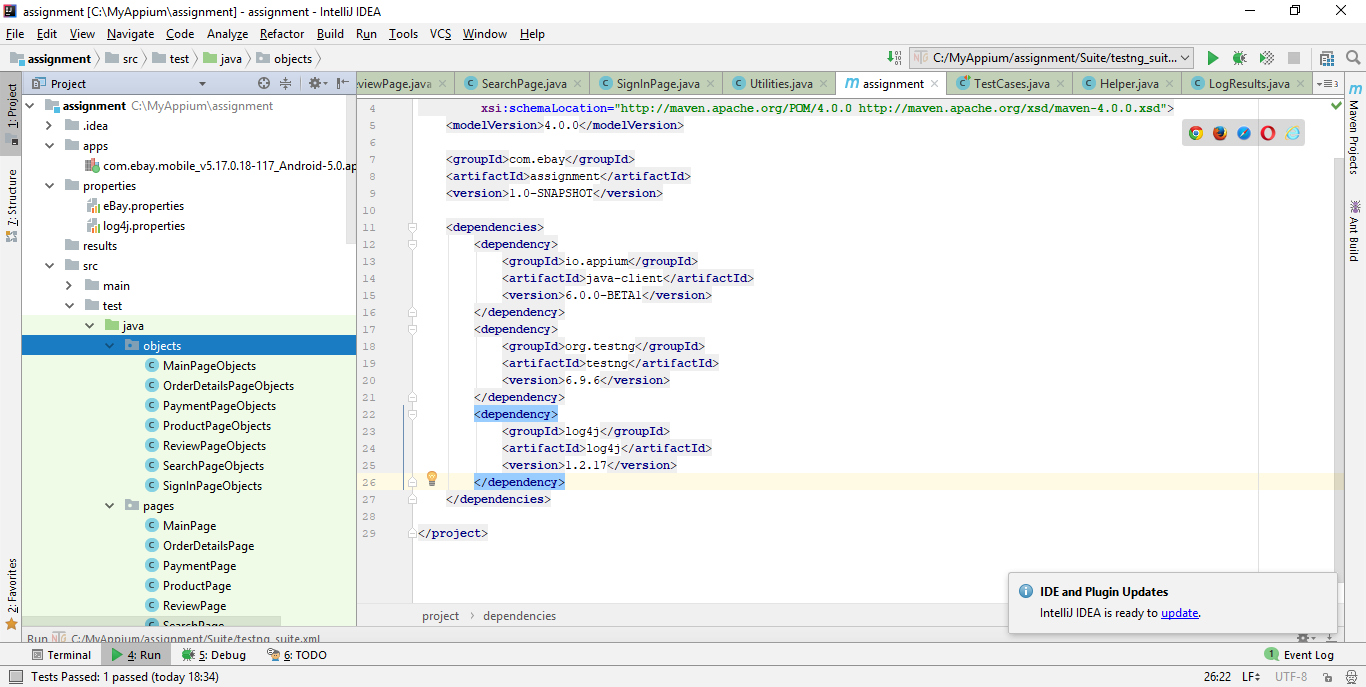
Devices used for testing:

* Google Pixel 1 with Android 8.1.0

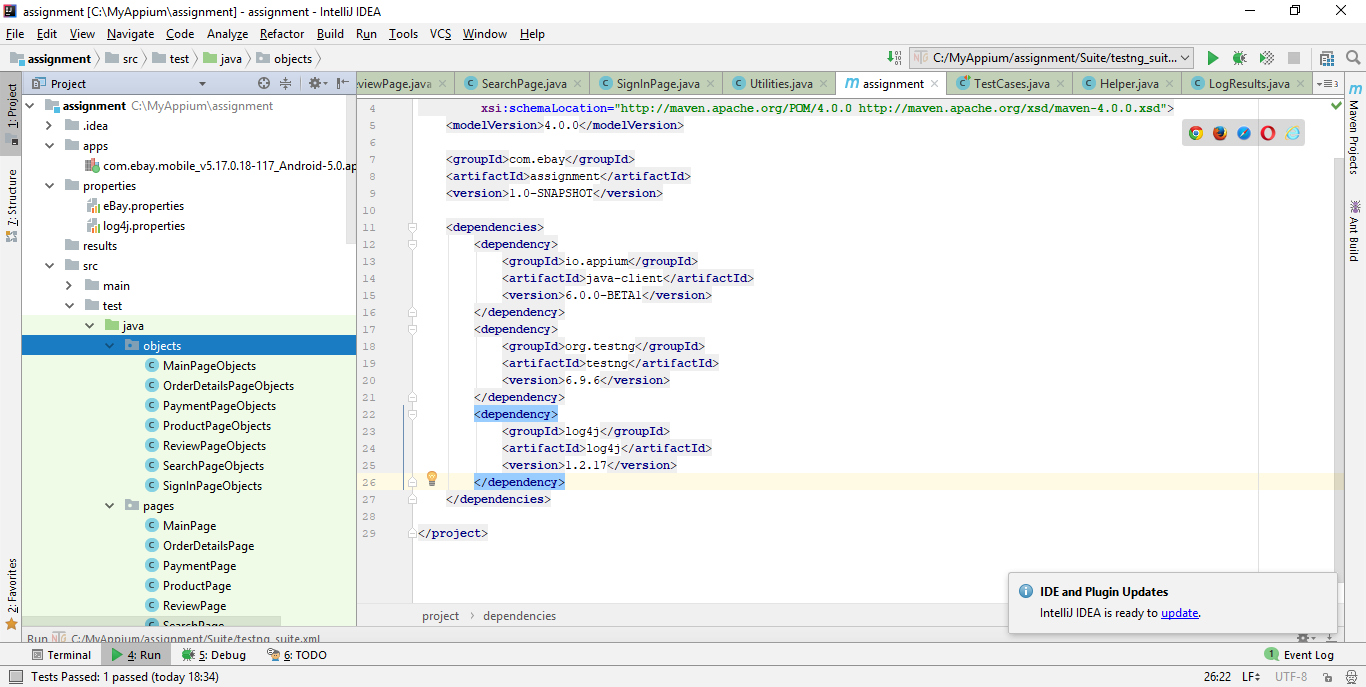
**Automation Framework developed:**

Components of Automation Framework:

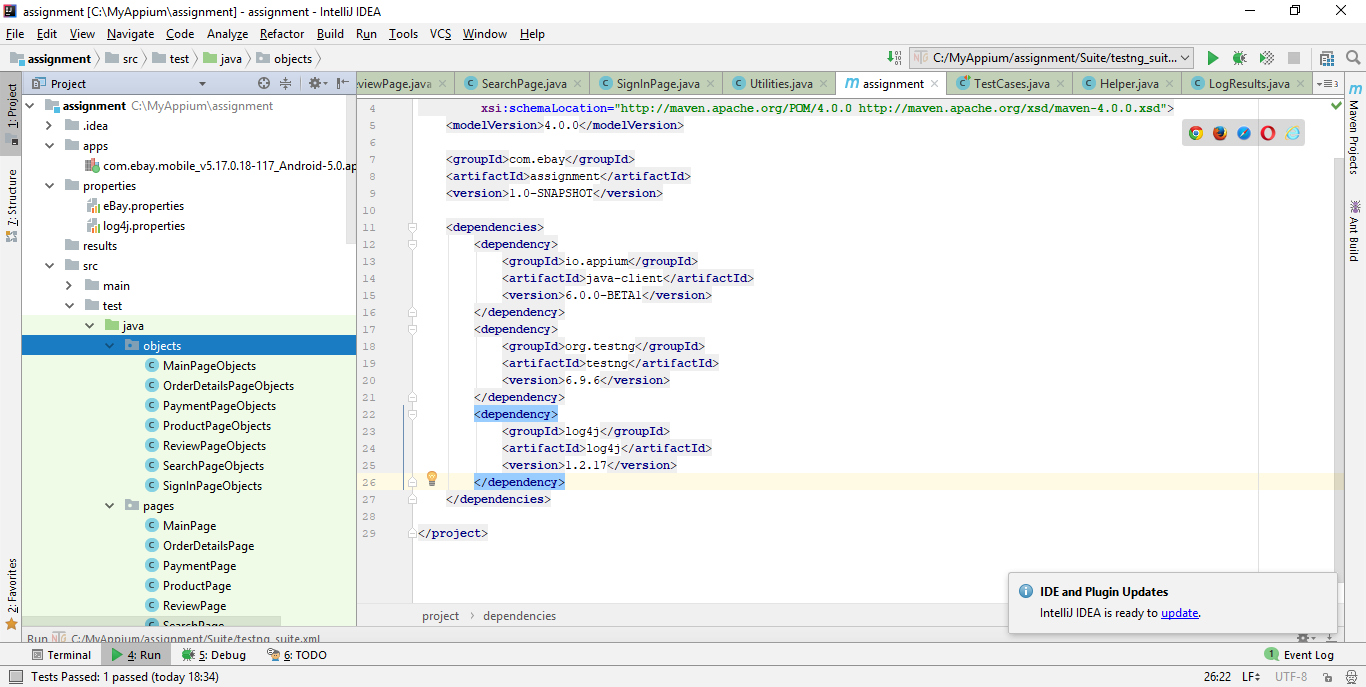
1. App:
   1. **app** directory of project is where “.apk” file for application under test is placed (com.ebay.mobile\_v5.17.0.18-117\_Android-5.0.apk in this case)



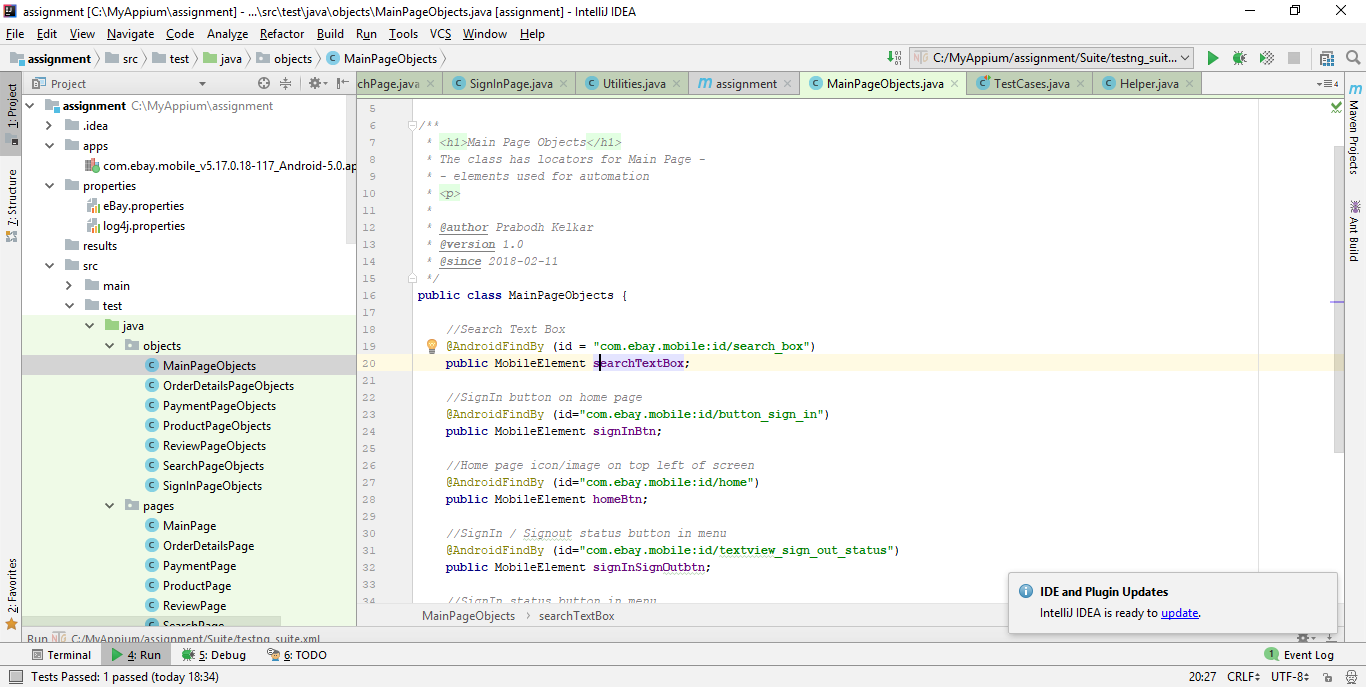
1. Properties files: Reside in **properties** directory
   1. eBay.properties: This file contains properties to be used for Appium driver initialization, user credentials to login into eBay app, Search criteria to search the product
   2. log4j.properties: This file contains properties to be used for log generation via log4j



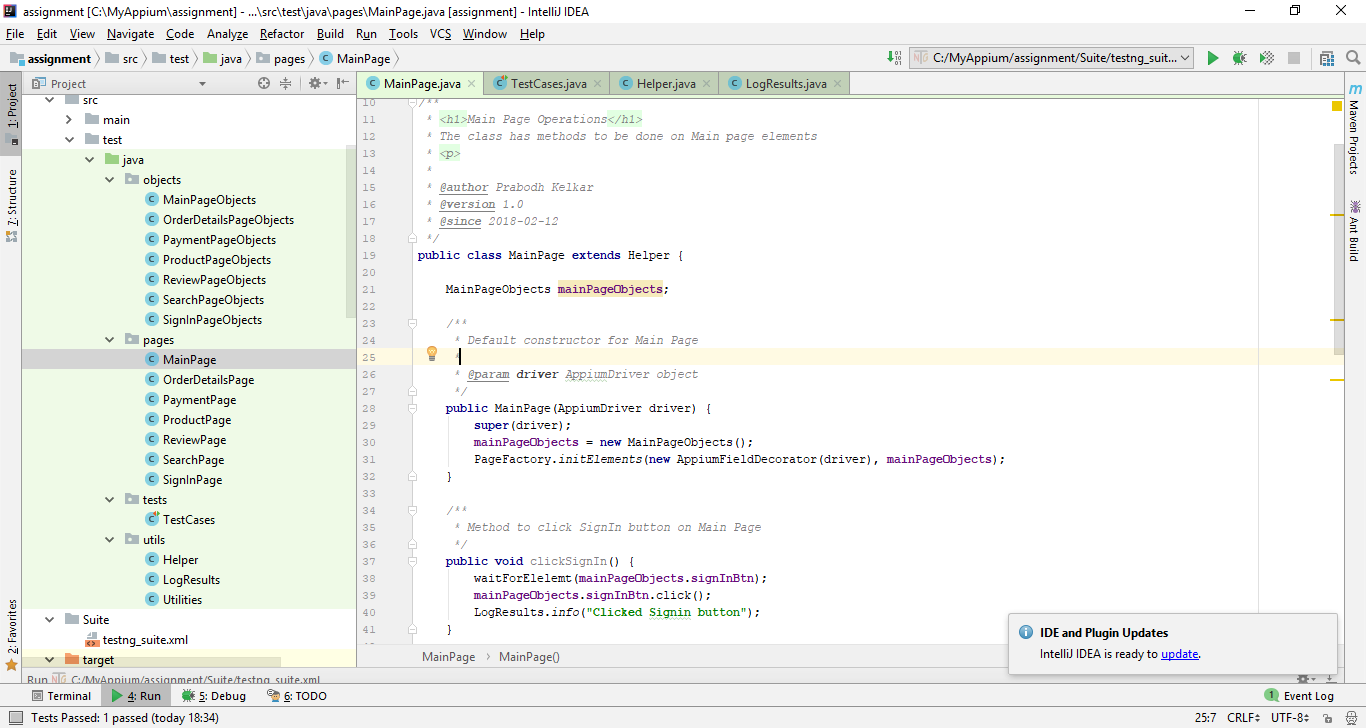
1. Results: This is the directory where logs will be generated after running the test case



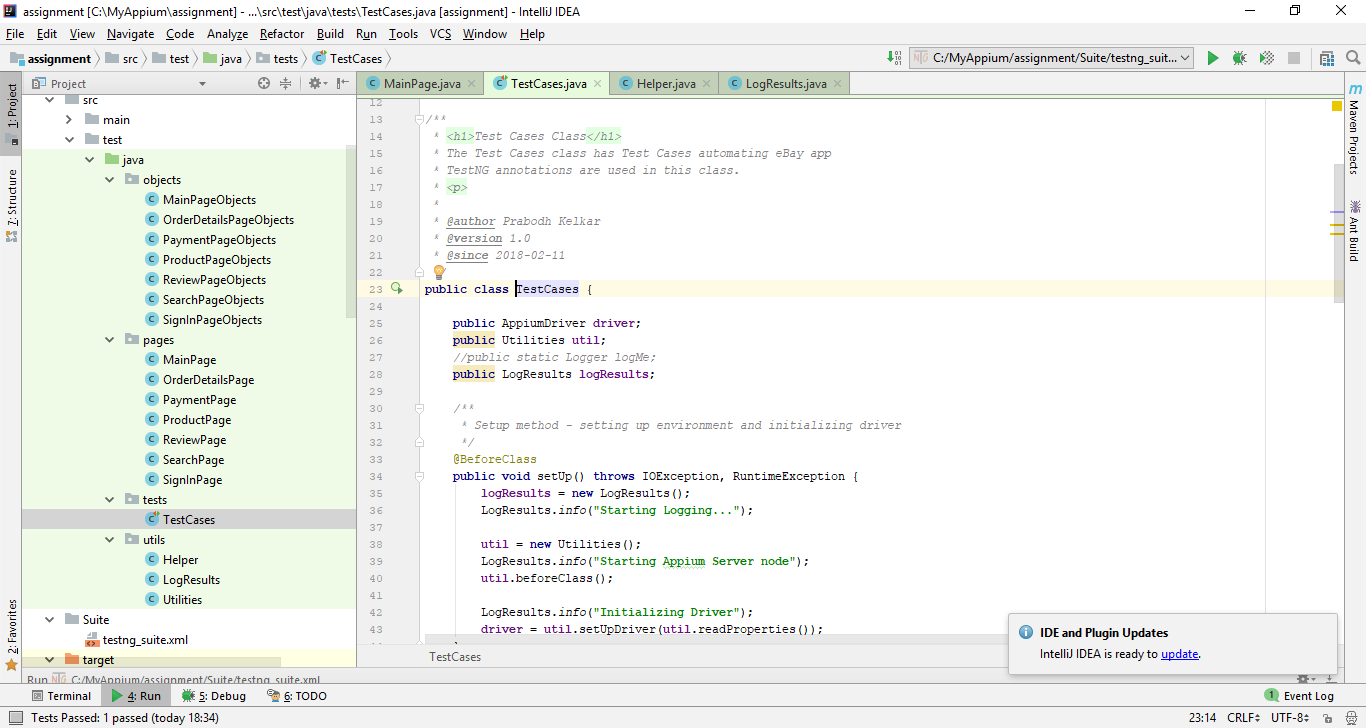
1. Automation framework structure / code structure: Under \src\test\java we have 4 packages as below.
   1. **Objects**: This package contains Page Objects (java classes) for each page in app. Each page has locator information in Page Factory format.



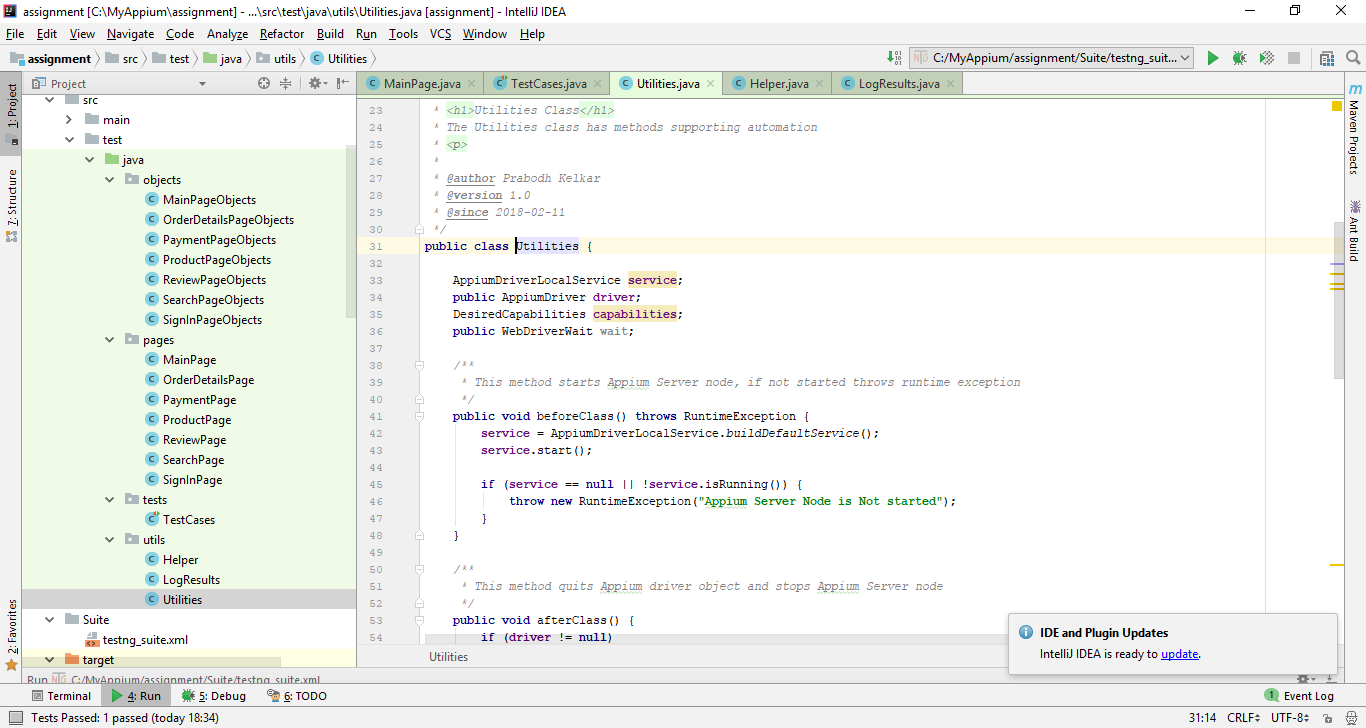
* 1. **Pages**: This package contains Pages (java classes) for each page in app. Each page has small methods performing actions on objects/elements from respective Page.



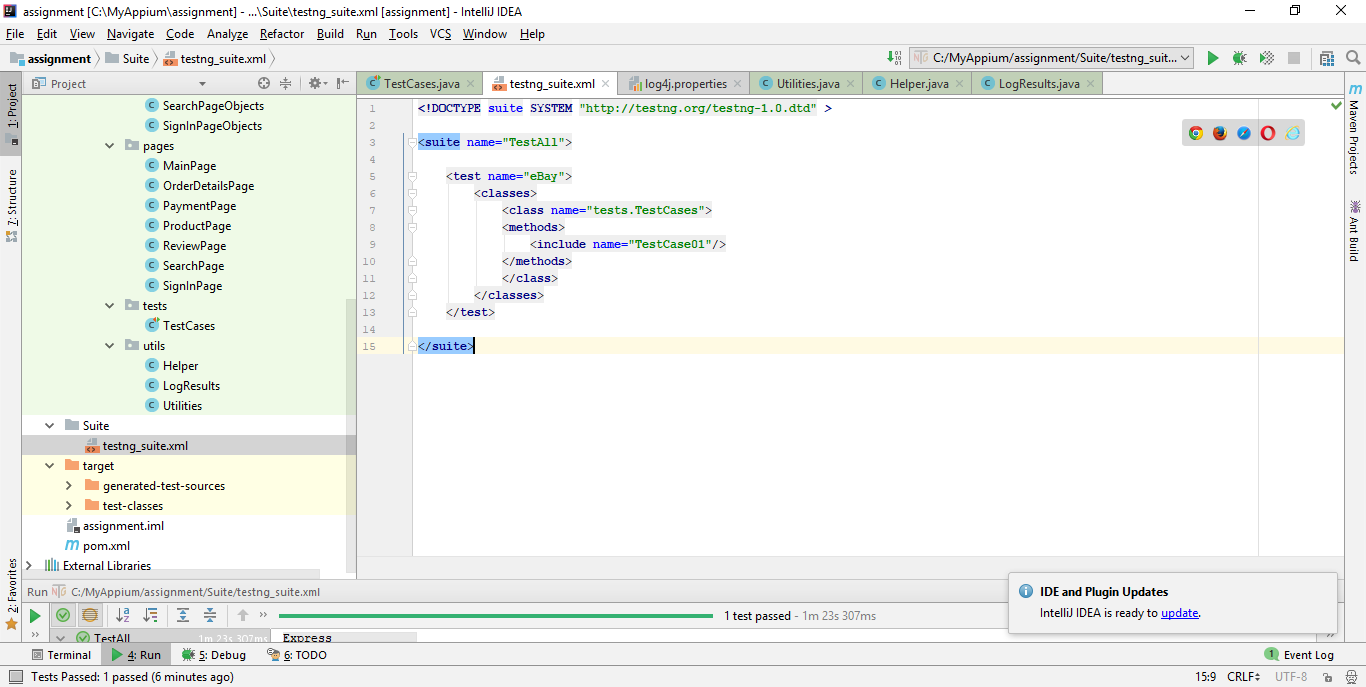
* 1. **Tests**: This is TestNG class with Test Case methods written into it using TestNG annotations. @BeforeClass, @Test, @AfterClass notations are used and we can extend this to use further TestNG annotations as well as attributes like priority / group



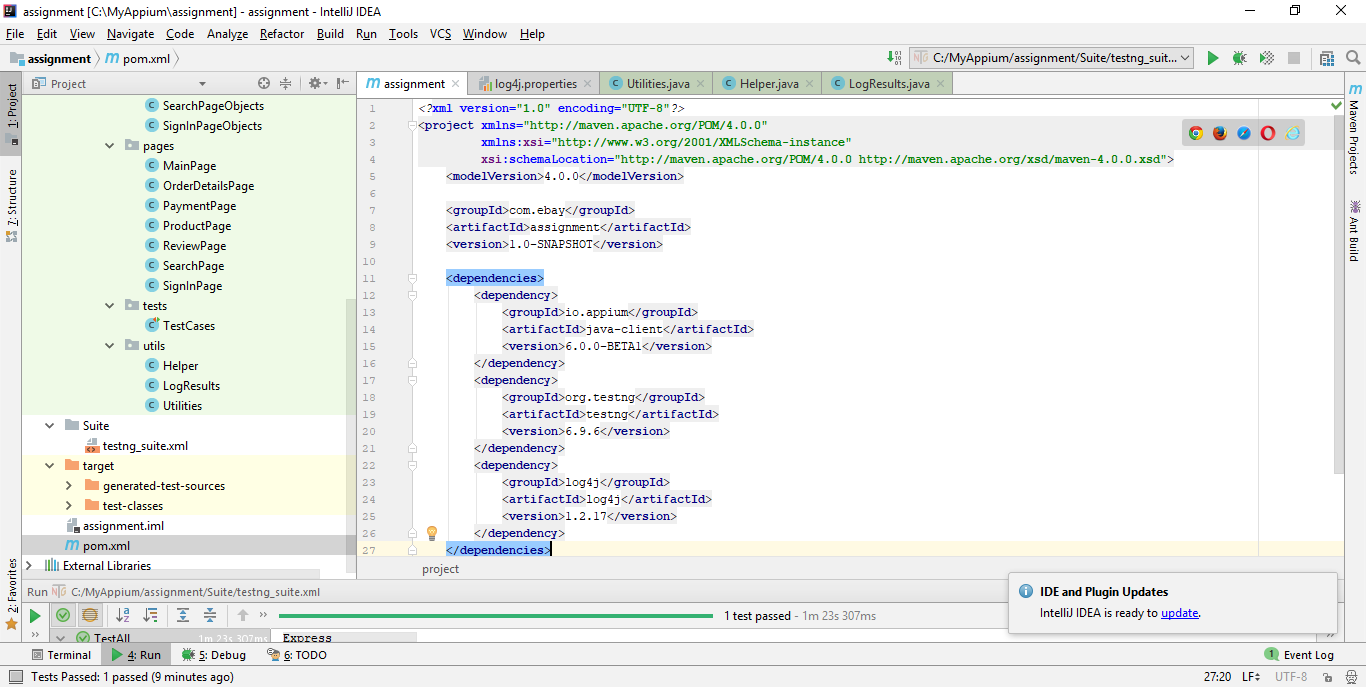
* 1. **Utils**: This package contains utility/helper classes for automation.
     1. Utilities class - has methods related to Appium service start, Driver initialization and teardown functionality
     2. Helper class - has common method about “wait” for element
     3. LogResults class - has methods to write log entries in log file



1. Suite: This directory contains testing\_suite.xml. This is a suite file, where one can add entries for new classes, methods to run all tests as test suite. As this is TestNG, we can configure to generate HTML Test Reports.



1. pom.xml: Project Object Model file used by Maven, where one can add dependency entries



In current framework, I have not implemented “**method chaining**” as in the eBay app under test, control flows are not constant. But, if we have control flow in defined fashion, method chaining can be implemented avoiding the need to declare objects for all required pages in Test Case script.

Advantages of framework:

1. Configurable items are kept outside – in properties file. So, code change should not be required if relevant parameters change – improving **maintainability**
2. Locators are part of single file – for each page. So, if there is any change in locators at any given point of time, just single Page Object file containing respective element should be changed – without touching any other code – improving **maintainability**
3. For any new page additions, respective Page Object file and Page file should be created, no update to be done on other code – improving **scalability**
4. TestNG notations used – so, possibility to include selective test cases from test suite – providing **flexibility**
5. With Maven, adding a new dependency is easy – improving **scalability**
6. Appium service start / stop considered as part of automation framework – providing **convenience**
7. Dynamic “path” used in project. So, no need to have specific folders created on machine executing test scripts – providing **convenience**
8. Log file config parameters are kept outside – in properties file – improving **maintainability**