**Training Setup Guide**

Cucumber/Java Course Prerequisites

1. Mid-level Java core
2. Maven
3. JUnit

# Java Resources

1. Safari: Maven: The Definitive Guide
2. Safari: Apache Maven 3 Cookbook
3. Safari: Getting Started with Apache Maven [Video]
4. Safari: Beginning Java 7 : Chapters 1-6, 8 as a minimum
5. Safari: JUnit in Action, Second Edition

Contents:

1. List of required software
2. Steps for setting up the development environment.

Required Software

1. Oracle JDK 8 Update 65 (This is the latest available in MyServices : Java JDK 7 or above is required)
2. Eclipse IDE Java EE Mars 4.5.1 x64 (Any Eclipse IDE will be fine: This one available in MyServices)
3. Apache Maven 3.0.3+
4. Mozilla Firefox (Firefox ESR 24.3.0 is latest available in MyServices)
5. Oracle Client 11.2​
6. Putty
7. Quest Toad (12.8 is latest available in MyServices)
8. TextPad
9. Tortoise SVN 1.7.x
10. WinSCP
11. Winzip

 Steps to configure development environment

1. Start Eclipse
2. Configure proxy

a.       ​click **Windows>>Preferences**

b.      ​​Search for "Network" on the left side of the panel of Preferences.

c.       ​click on network Connections under (+) general.

d.       select **manual**under drop down of **Active Provider**

e.      Click on First Row – for Schema "HTTP" under Proxy entries and then click**Edit**

* + ​​Enter the value for host as **bcproxy.fanniemae.com** with port as**80**
  + Enable check box for **Required Authentication**
  + Enter your LDAP username and password

f.        click **Save**

1. Download Plugins

a.       Install the following plugins:

* Xtext (2.9.1)
* Natural (0.7.6)
* Subclipse (1.10.13)
* TestNG

​​b.      Follow the below instructions to installing the plugins

* + Click **Help**>>**Eclipse Market Place**
  + Search for plug-in name as shown above
  + ​Install the plug-in, restart Eclipse as required, repeat for other plug-ins

1. Configure JDK in eclipse

a.       ​​Click **Windows>> Preferences**

b.      Click on Compiler under (+) Java on left side panel of Preferences.

c.        Change compiler compliance level to 1.7 or 1.8 depending on which version you have and configure the path of JDK from your System.

1. Download BDD Automation sample code from SVN

a.       Under Package Explorer – right click and select **Import**

b.      Click **SVN**

c.       Under SVN, ​click on **Checkout Projects from SVN** and click **Next**

d.      ​Select option **Create a new repository location**

e.      Type the following URLs one by one to download the code into your work space

<http://plsysadm-cm07:8888/cm-repos/essta/devbranches/snapshots/test-automation-core>

<http://plsysadm-cm07:8888/cm-repos/essta/devbranches/snapshots/training/calculator-example>

<http://plsysadm-cm07:8888/cm-repos/essta/devbranches/snapshots/training/test-automation-stayfit>

<http://plsysadm-cm07:8888/cm-repos/essta/devbranches/snapshots/training/test-automation-stayfit-data>

1. For setting up the maven repositories:

a.       Copy settings file

​​**From:**/test-automation-stayfit-data/src/main/resources/conf/temp/settings.sample.xml

**To:**C:\Documents and Settings\<UserID>\.m2     >>> rename it settings.xml

1. ​Steps to Compile the Code
   1. Execute following Maven for each above project in the following order : core, stayfit, stayfit-data
      1. Right-click the folder in Package Explorer
      2. Go to **Run as** and  select **Maven Build**
      3. Within the new window check "Skip Tests" and enter the following in "Goals" textbox
         * "clean install"
      4. click "Run"
      5. repeat steps 7.(1-5) again, but with "eclipse:eclipse" instead of "clean install"
   2. After the successful execution of maven commands refresh all projects by
      1. Right-clicking their folders
      2. Selecting "Run As"
      3. Selecting "Maven Clean"