**Installing Java Code Generator Plug-in under Eclipse**

1. Make sure, you have the Eclipse Plug-in Development Environment and Titan Designer Feature installed under Eclipse.
2. On the **Package Explorer** pane in Eclipse,right-click an empty spot and select **Import**.
3. In the pop-up window, under **General**, select **Existing Projects into Workspace**. Click Next.
4. Select the radio button named **Select root directory** then click **Browse** next to it and select the org.eclipse.titan.codegenerator directory, wherever you’ve downloaded it. Click **Finish**.
5. On the **Package Explorer** pane, click the arrow next to the newly imported org.eclipse.titan.codegenerator project to expand it.
6. Make sure that no other instances of the Code Generator plugin are installed in Eclipse (this is true if the coffee cup icon is not present on the toolbar). To uninstall existing instances of the plugin, go to Help/Installed Software. If it does not help, empty the eclipse/dropins folder. A search for files containing the text „codegenerator” in the workspace/.metadata folder can also help, as this way, you can get the paths on which, other instances of the plugin’s JAR are stored and then remove them by hand.
7. Go to org.eclipse.titan.codegenerator/src/org.eclipse.titan.codegenerator/walker.properties and set the variable javafile.path to the directory where you would like the Java files to be generated.
8. Double-click plugin.xml. Scroll to the bottom of the Overview tab, and click **Export Wizard** in the **Exporting** section.
9. In the pop-up window, make sure org.eclipse.titan.codegenerator is selected in the Available Plug-ins and Fragments section, then on the Destination select radio button **Install into host. Repository:**
10. Click **Finish**, and restart Eclipse when prompted to.

**Running the generated code**

1. Similarly to the above, import the contents of ttcnjava.zip as a project into the workspace.
2. Open your source TITAN project and open one of the TTCN-3 files in it.
3. Click on the coffee cup icon („Generate JAVA code from the TTCN sources”).
4. Go to the directory you specified as javafile.path, and drag-and-drop the newly generated java files to ttcnjava/src/org.eclipse.titan.javagen on the Package Explorer pane in Eclipse.
5. Set the number of hosts used for test execution in ttcnjava/org.eclipse.titan.ttcn3java. TTCN3JavaAPI/MC.java, as the value of variable HCNUM.
6. Set the IP address of the host where the main test component should be started, in the same file, variable MTCIP.
7. Set whether you’d like a detailed log or a less detailed one in variable DEBUGMODE, in the same class. You can set it for each host controller in the corresponding ttcnjava/org.eclipse.titan.javagen/HC.java files as well.
8. Set the output file of the logger in ttcnjava/org.eclipse.titan.ttcn3java.TTCN3JavaAPI/ TTCN3Logger.java, line 52.
9. Build the newly created Java project, if necessary.
10. Right-click ttcnjava/org.eclipse.titan.ttcn3java.TTCN3JavaAPI/MC.java, select Run As, then select Java Application.
11. Right-click ttcnjava/org.eclipse.titan.javagen/HC.java, select Run As, then select Java Application. Repeat this for each host that is used for test execution. After the last host controller is connected to the main controller, test execution starts automatically.
12. To see the results of the test execution, either open the log file you specified, or see the results on the Eclipse console (the log file contains more entries even when not in debug mode).

**Note:** Please be aware that in the current state of the generator, when creating a test component, the create function has to be parameterized with the name of the test component and the IP address of the host on which it has to be created (so two charstring parameters in total). Please, also be aware that in the current state of the generator, the control parts of TTCN-3 modules are not processed. Instead, the name of the test case to be executed should be explicitely specified in ttcnjava/org.eclipse.titan.ttcn3java.TTCN3JavaAPI/MC.java.