Prolifics
Global Technology Solutions Provider

JPL Plugin 3.*

April 2021

JPL Plugin enables Panther developers to create and update Panthers' JPL source code within the Eclipse IDE.
In addition to creating Panther Projects, Panther Libraries and JPL files, the plugin also validates JPL syntax. The Content Assist feature enables the selection and insertion of existing JPL code elements to complete partially entered JPL code.

This document outlines features of the JPL Plugin

Features

- Create Panther Projects
- Create Panther Libraries
- Import Panther Libraries
- Export Panther Libraries
- Create JPL files
- JPL syntax validated for correctness
- Converts binary screens, reports and menus to ASCII
- Search and Compare files
- Content assist after "->" , i.e. @widget("screen1")->title="Login Screen"
- Content assist after "_", i.e. " sm_" or " dm_"
- Hyperlinking – for smart switching between call and procs
- Source Code Control Support

Software Requirements:

1. Eclipse IDE (Neon/Oxygen/Release 2020-06 or any latest version):
   To install this plugin you must install Eclipse onto your system. You may download it from http://www.eclipse.org/downloads/eclipse-packages/
2. A suggested Eclipse Package is Eclipse IDE for Java Developers. The package should contain tools for JAVA EE and Web applications.

Note: - if you don't know which Eclipse version to select, just download Neon 3 from below link. https://www.eclipse.org/downloads/packages/release/neon/3

Installation Instructions:

Follow the step-by-step instructions to install the plugin.

Install JPL Editor Plugin

1. Download JPL Editor Plugin JPL Plugins.zip from GitHub https://github.com/ProlificsPanther/JPLPlugin/releases
2. Run eclipse.exe and create a workspace.
3. Once your workspace opens, go to the **Help** menu and select "**Install New Software**".

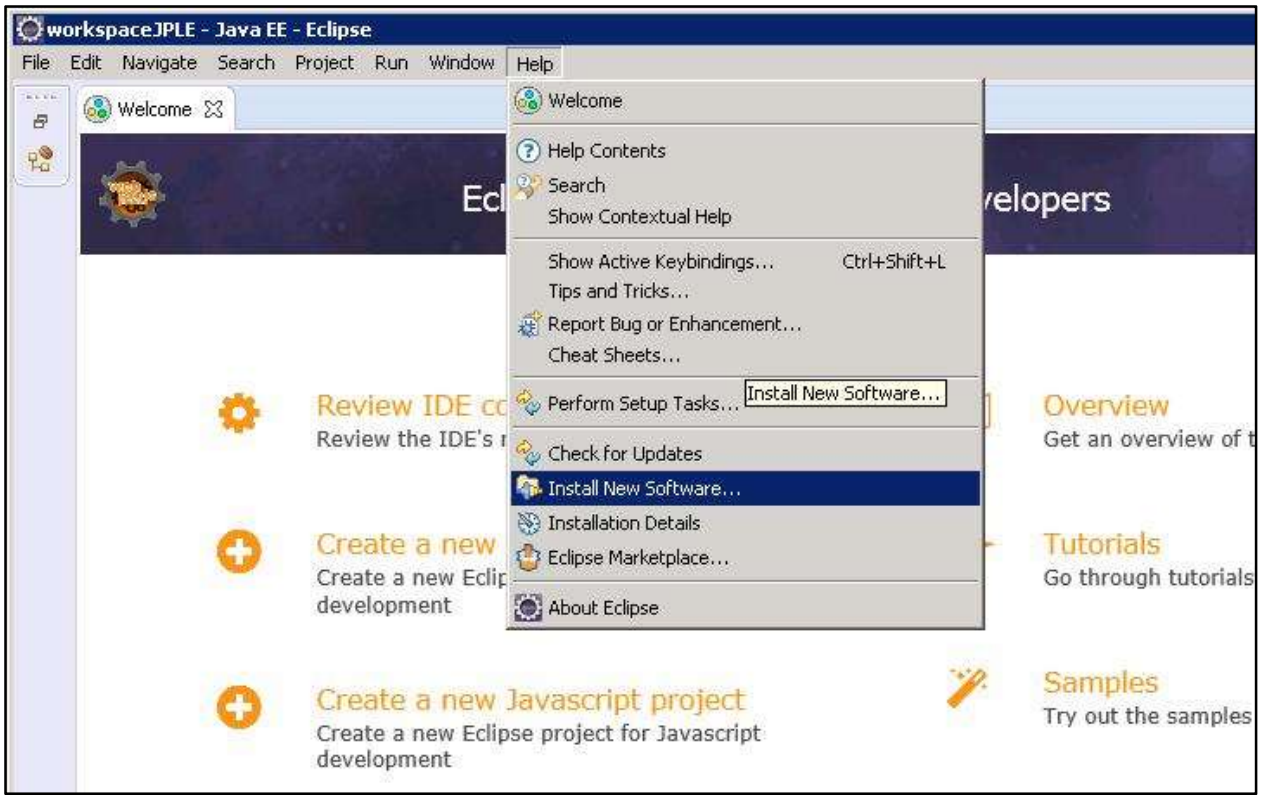

Fig 1

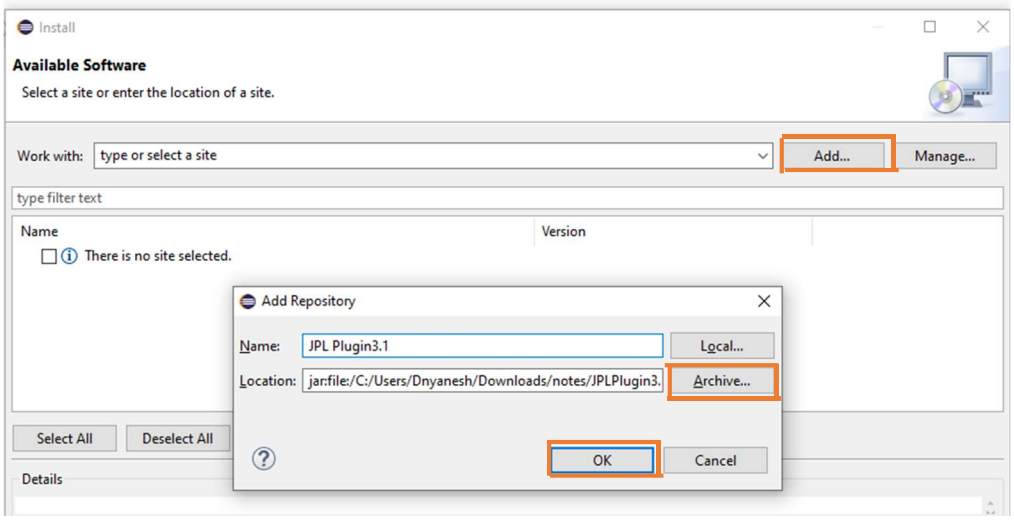

Fig 2

4. Click the Add button and choose Archive then Browse to the zip folder i.e. JPL Editor Plugin. Click Ok. [Fig 2]
5. Unselect 'Group items by category' in the Install window and select the check box 'JPL Plugin 3.*. Click Next. [Fig 3]

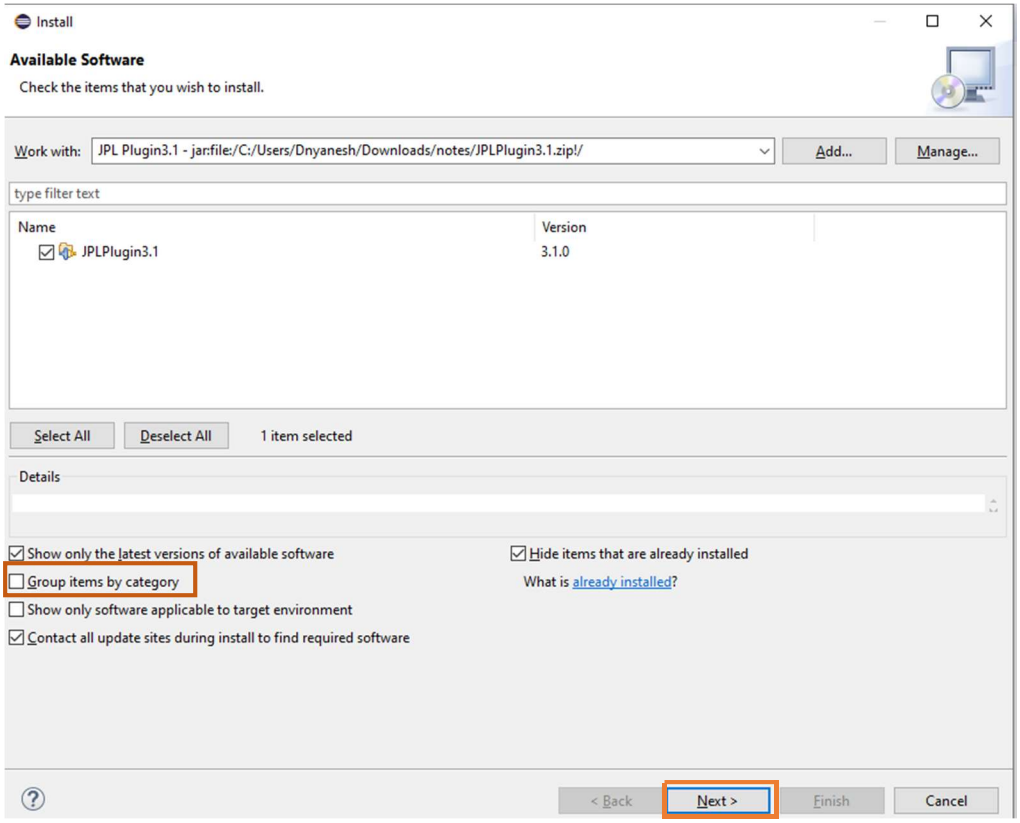

Fig 3

6. Choose the JPL Plugin 3.* then Press Next.
7. Next choose “I accept the terms of the license …” radio button. Click Finish. You may ignore the Security Warning.
8. If you encounter any errors, redo Step 2 but this time choose Help->Install New Software and uncheck “Contact all update sites during install to find required software”.
9. Restart Eclipse. The installation of JPL Editor Plugin is now complete. You may verify this in Eclipse’s installation detail. You will see JPL Plugin 3.* appear in the list. Now, the very first thing to do is to create a new Panther Project.

Create a Panther Project

1. Please make sure SMBASE is set in your environment. It must point to the location of your Panther installation
   SMBASE=C:\Prolifics\Panther553 (Windows)
   SMBASE=/usr/develop/prolifics/panther553 (Linux)
2. From the Eclipse menu, click File-> New->Other->Panther -> Create Panther Project. [Fig 4]
3. Enter a Project name. For example “CurrDev2016” then press Finish.
4. If Eclipse requests that you add “Xtext nature” to your project, please click Yes.

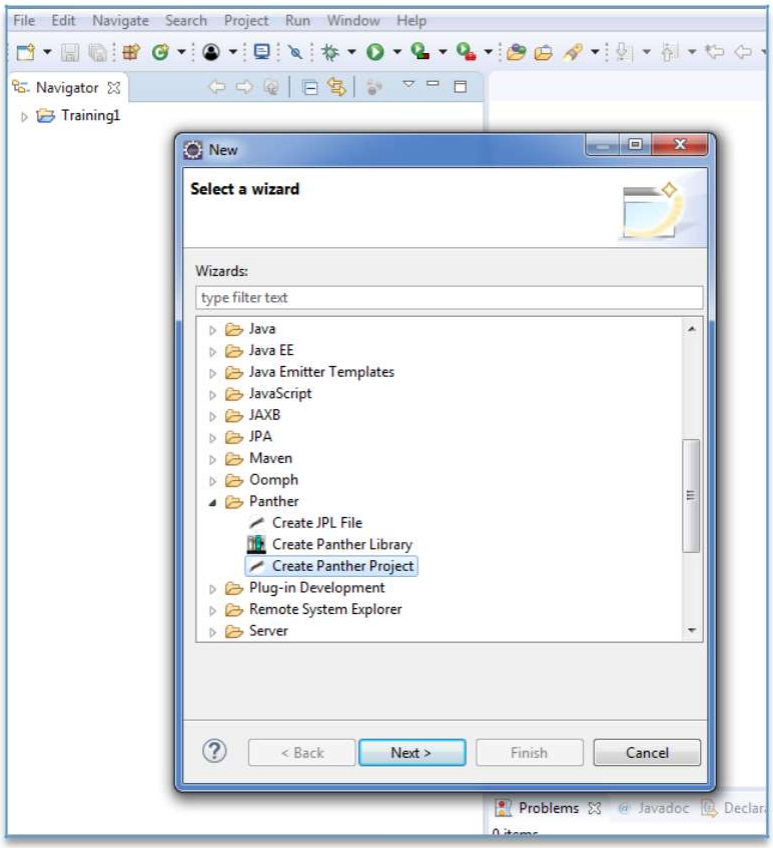

Fig 4

Import a Panther Library

1. Your Panther library can contain screens, jpl, menus, reports, html, JavaScript, Java files etc. Right click on your newly created Panther project and select import from the context menu.[Fig 5]
2. Select Import Panther Library from the import wizard.[Fig 6] .
3. Click on Panther Library and then click on “Next”.
4. Browse to your Panther library e.g. test.lib.
5. Click on “Finish”.

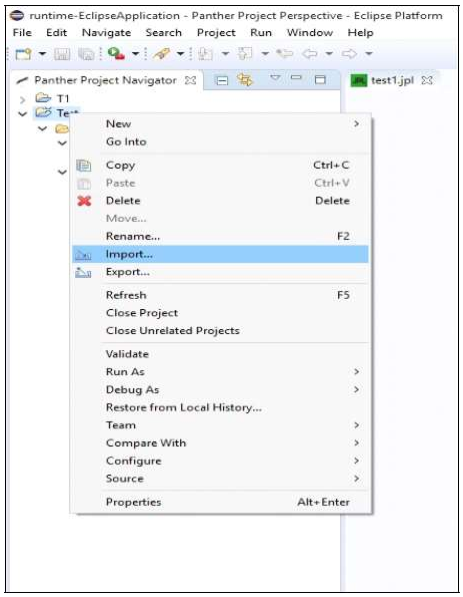

Fig 5

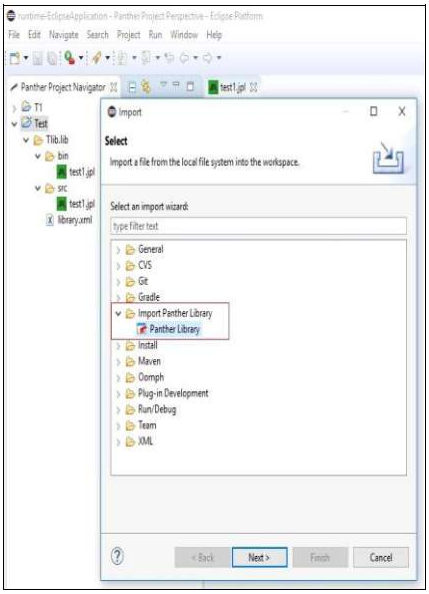

Fig 6

6. The entire content of the Panther library is placed into the <bin> sub-folder of the library folder that is shown in Panther Project Explorer View in Eclipse. Note that a Panther library file contains no folder structure itself. A <src> folder is also created within the library folder. Its content is dynamically generated from binary files in the <bin> folder during the import process. Panther utilities, such as formlib, f2asc, and jpl2bin from your Panther installation, are executed automatically in order to accomplish these tasks. Only some file types within the <bin> folder, such as Panther screens and binary JPL files, are processed in this manner in order to create source files from them. These source files may be edited with the Eclipse editors that are provided by this plugin. Files in the <bin> folder are then regenerated from the <src> files when the project is built. By default, this occurs automatically when <src> files are edited. [Fig 7]

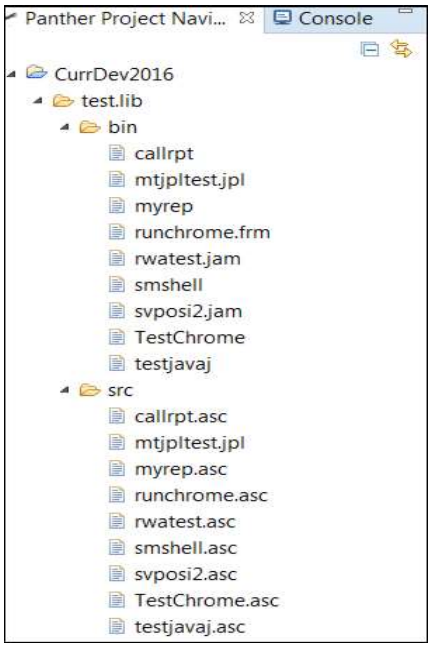

Fig 7

Create a Panther Library

1. Right click on Project -> New -> Other -> Panther -> Create Panther Library. [Fig 8]
2. It contains <src> and <bin> directories and one library.xml. You can change the extension of generated binaries by adding jpl/bin in the <jpl-binary-filename-extension> tag [Fig 9]

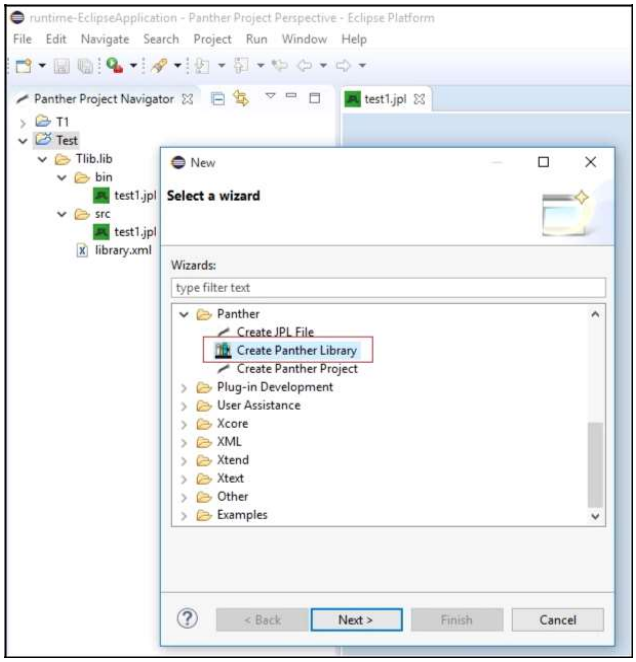

Fig 8

Fig 9

Create a New JPL File

1. Right-mouse click on the <src> folder in which you would like to create a jpl file in. Choose New -> Other -> Panther -> Create JPL File.[Fig 10]
2. The filename MUST have a ".jpl extension, then press Finish.
3. When making an update in any ASCII file, once you save your changes, the file is compiled to binary and stored in the <bin> folder.

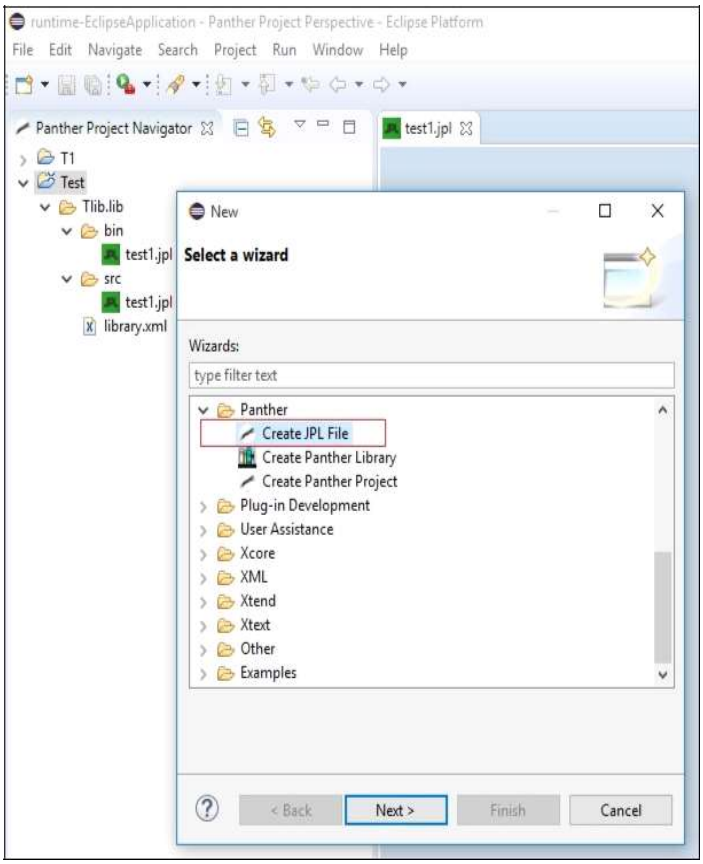

Fig 10

Sample Screen after installing plugin:

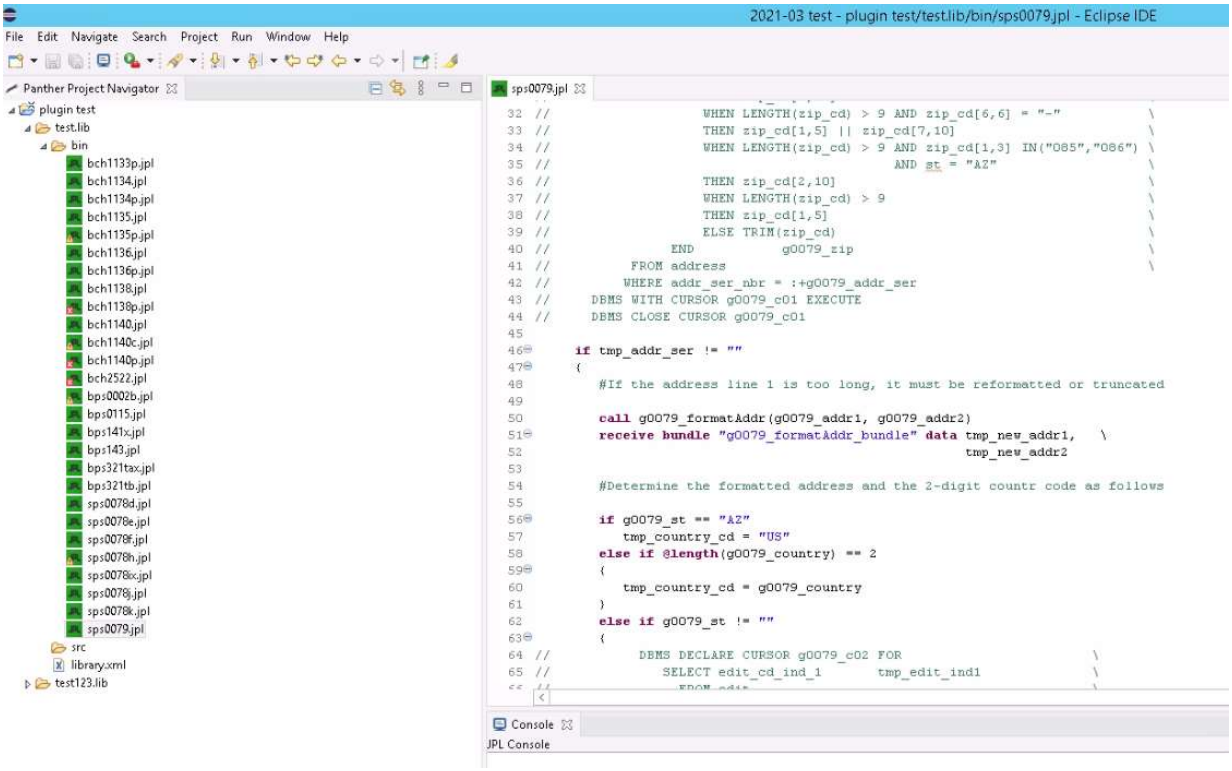

Fig 11

Export a Panther Library

1. Right click on the Panther project library, for example Shazam.lib. [Fig 11]. Select export from the context menu.
2. Select Export Panther Library.
3. Then click on “Panther Library” and click on “Next”.
4. Browse the destination directory. This is where you will export your Panther library to.
5. Click on “Finish” button.

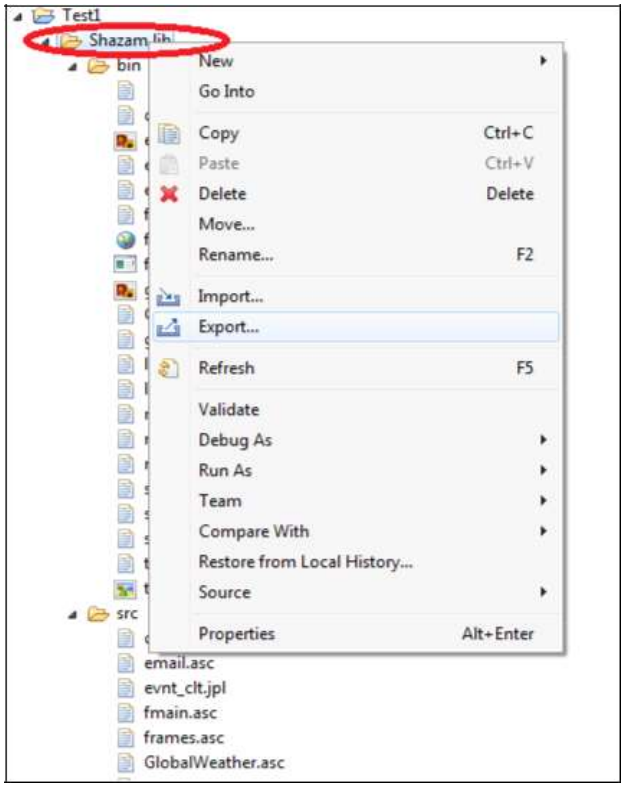

Fig 12

NOTE: Due to a known defect, when exporting the library you must select both <src> and <bin> folders to be exported, or else the library file is not created. However, only files in the <bin> folder are ever included within the exported library, which is the intended behavior.

You may also add new files directly to the <bin> folder that should be added to the exported library.

YouTube Reference:

1. **Installation:**
   https://www.youtube.com/watch?v=ymqthkjxJlc&t=41s&ab_channel=ProlificsPanther
2. **JPL Plugin 3.2:**
   https://www.youtube.com/watch?v=ALmpeBqZTeQ&ab_channel=ProlificsPanther

Uninstalling the Plugin

1. If you need to remove an old JPL Plugin, uninstall it using Help->About Eclipse.
2. Click on Installation Details and select the "JPL Plugin Editor", press uninstall. Confirm that you would like to uninstall it. Then Restart Eclipse.