Contents

System Requirements

Operating System

This codebase can be setup on Windows or MacOS.

Library

This was developed using Java 1.8.0_231. To re-build this codebase you should have same version of JDK installed. If you do not have it, you may get it from Oracle's website and install it.

Tools

You would need following tools to work with codebase.

1. Eclipse
2. GIT
3. Eclipse-GIT Extension (Optional)
4. Source Tree (Optional)

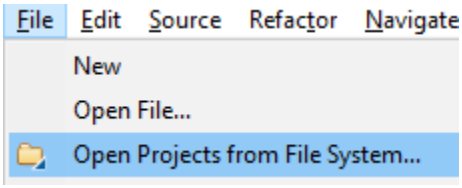
Cloning Repository

You may close the repository from following URL to your system

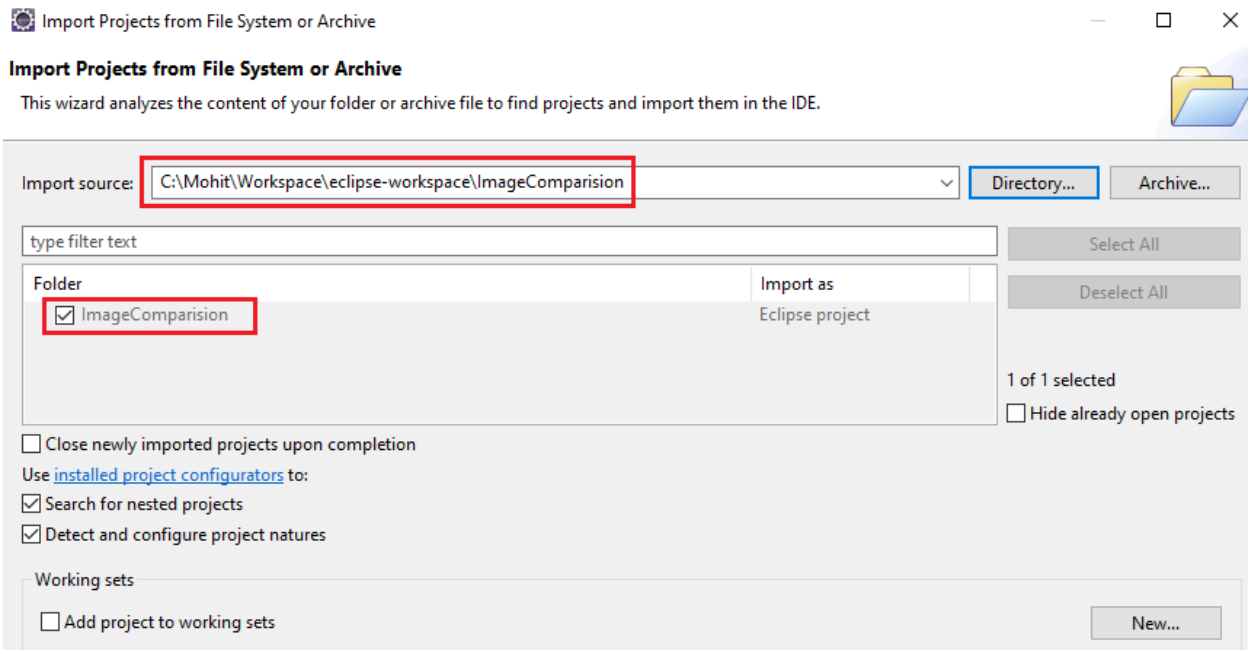
https://github.com/mkthakral/ImageComparisionTool.git

Import to Eclipse

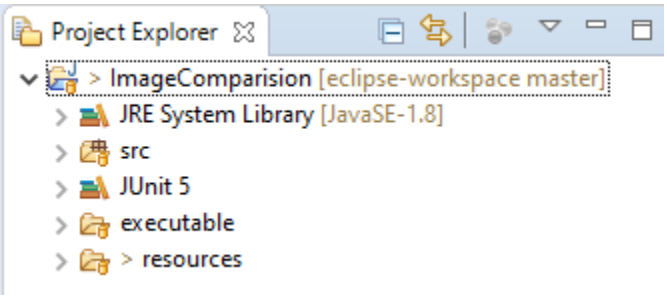
Once you have the codebase on your system, you can import it into eclipse.



Select the codebase path and click "Finish"



Once import is complete, you should see the project structure like this.

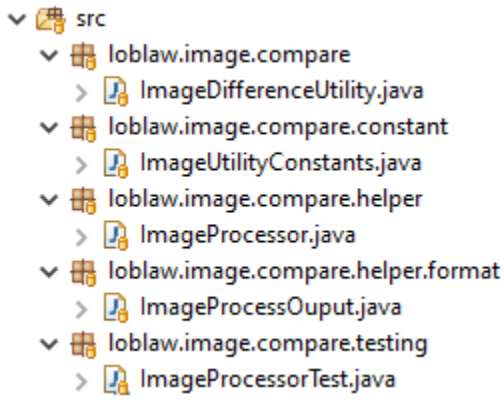


Directory Structure

Directory	Purpose
<Project Home>/src	This has all the code needed to build or run the application. This also has the code for unit testing.
<Project Home>/executable	Here you will find the application's built ready to use versions.
<Project Home>/executable/Version-N	Here you will find executable JAR i.e. the built application for specific version. It also has "changelog.txt" which lists the changes done in this version.
<Project Home>/resources	This directory contains documentation of the project and resources needed to run unit tests.

Understanding Code

Following is the code file structure.



Please find high level function of each file.

File	Purpose
ImageDifferenceUtility.java	Entry point of the application. Creates a user form i.e. GUI. Reaches out to “ImageDifferenceUtility.java” for processing input.
ImageUtilityConstants.java	Contains constant variables used in project, also contains a bit of configurable items like error messages.
ImageProcessor.java	This is the primary working horse of the application which contains algorithm to compare images, reading input, creating output file.
ImageProcessOuput.java	This is a custom class, a format to return response from “ImageProcessor.java” to “ImageDifferenceUtility.java”
ImageProcessorTest.java	This has multiple unit test cases built using Junit.

Modifying Utility

Code Merge

To modify code for any new requirements, you may cut a branch from master branch in GIT repository and make changes. Once you are done with the changes you may get this code merged with master branch.

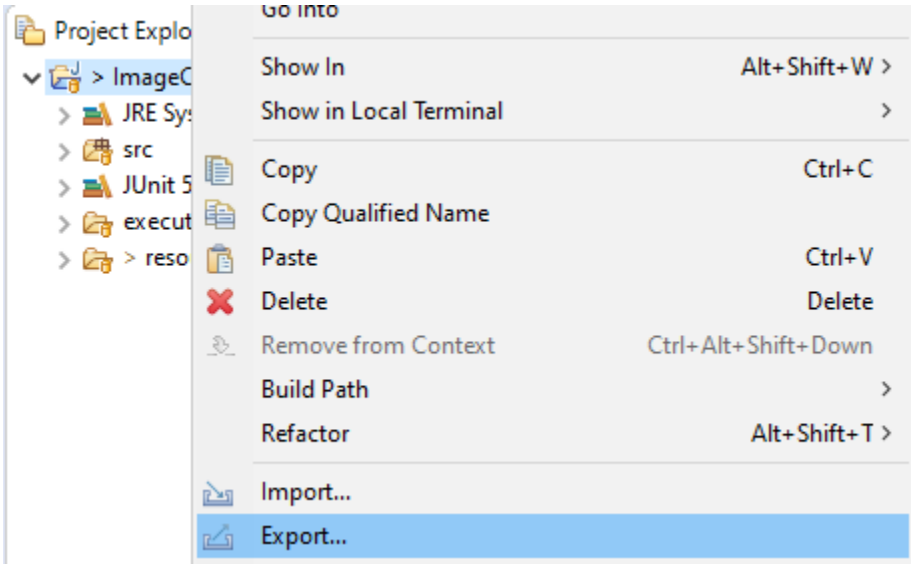
Directory Structure

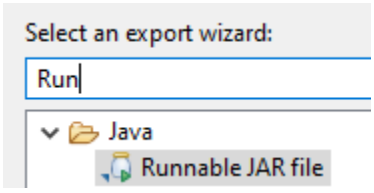
You should create also create following manually.

Create	Purpose
<Project Home>/executable/Version-N	Create this directory structure to host a new version of executable file.
<Project Home>/executable/Version-N/changelog.txt	Create this file and mention what changes are done in this version of application.

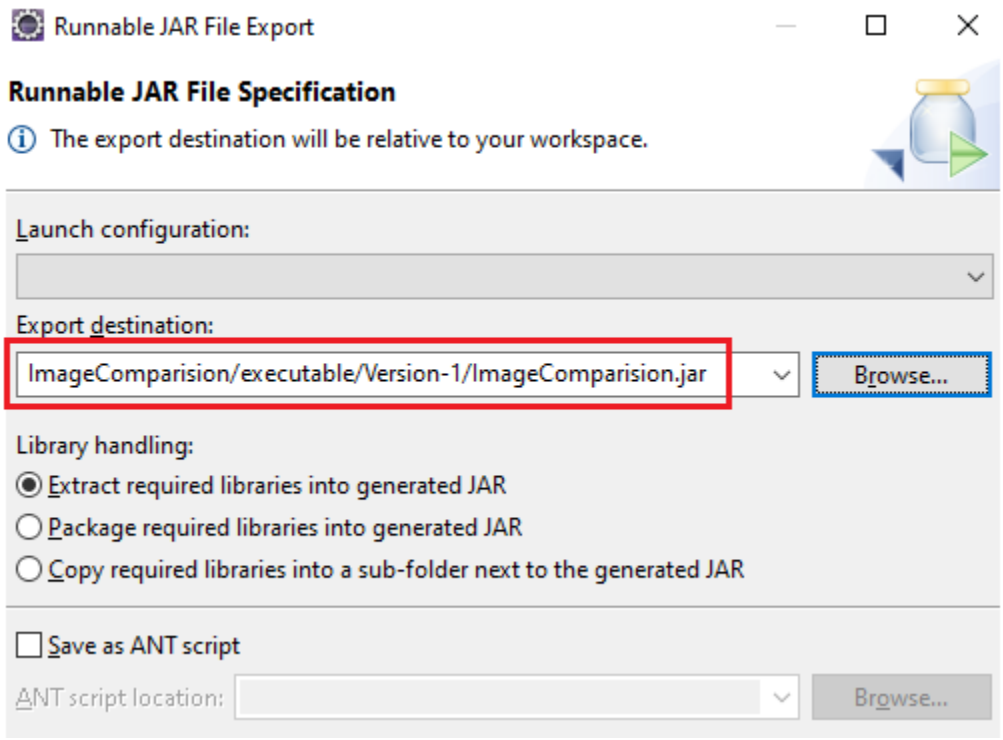
Building Executable

When you are modify the code, you should create a new version of executable JAR. For this





Select the directory created above and mention file name.



This shall create the JAR file at the selected directory.

Check in this directory into GIT.

Test Cases

We have written several unit test cases when you modify the code. Do not forget to review/add/modify test cases according to changes done.

Version

Update version information at `ImageUtilityConstants.java -> VERSION`