Migration to OPENJDK 12

Download openjdk 12 and unzip it in the java folder in your system. Set the environment variables in your system. Download the latest eclipse, June 2019 release and Tomcat 9.x

Install Server (as per requirement)

• apache-tomcat-9.0.16

Note: Set the environment variables to OpenJDK 12. Add the jar files to Tomcat folder as well.

Make sure that eclipse and Tomcat are running on java 12 version. Change the compling option in the eclipse to jdk12, and execution environment too using jdk12.

Removed from OpenJDK11

EE modules are deprecated in JDK 9 and removed in 11 and following are the replacements:

- java.xml.ws (JAX-WS) incl. jdk.xml.ws: wsgen, wsimport -> com.sun.xml.ws (by Sun/Oracle, *Eclipse Distribution License v. 1.0*)
- java.xml.bind (JAXB) incl. jdk.xml.bind : Schemagen -> com.sun.xml.bind:jaxb-impl + jaxbapi (by Sun/Oracle, CDDL)
- java.activation (JAF) -> com.sun.activation:javax.activation + avax.activation:javax.activationapi: ((by Sun/Oracle, CDDL)
- java.xml.ws.annotation (Common Annotations) -> javax.annotation + javax.annotation-api (by Glasfish, *CDDL*)
- java.transaction (JTA) -> 'org.glassfish:javax.transaction' (*CCDL*) + javax.transaction-api
- java.corba (CORBA) incl tools -> no replacement
- java.se.ee (all APIs from above)
- Geronimo-jpa_spec jar

Add external jar files **Project<Properties<Build Path** then under **Libraries** click **Add External Jars** add following jar files.

- javax.xml.soap
 - o jakarta.activation-api-1.2.1.jar
 - o jakarta.xml.bind-api-2.3.2.jar
 - o jakarta.xml.soap-api-1.4.1.jar
 - o mimepull-1.9.11.jar
 - o saaj-api-1.3.5.jar
 - o saaj-impl-1.5.1.jar
 - o stax-ex-1.8.1.jar
- javax.xml.bind
 - o javax.activation-api-1.2.0.jar
 - o jaxb-api-2.3.0.jar
 - o jaxb-core-2.3.0.jar
 - o jaxb-impl-2.3.0-b170127.1453.jar
- javax.jws
 - o javax.activation-api-1.2.0.jar

- o javax.annotation-api-1.3.2.jar
- o javax.jws-3.1.2.2.jar
- o javax.xml.soap-api-1.4.0.jar
- o jaxb-api-2.3.1.jar
- o jaxws-api-2.3.1.jar

Note: If still exception persist, like noclassdeffounderror or javax.xml.soap.SOAPException etc. then Add required jar files to WEB-INF/lib

For Gradle Project

Step 1:

Upgrade Eclipse photon current version to 4.10 version.

Step 2:

Install gradle 5.2.0 Set the environment variables in your system.

To upgrade gradle wrapper of eclipse to version 5.2.0, go to command prompt and run command: gradle wrapper --gradle-version 5.2.0

Step 3:

Click on Window<Preferences<Java<Complier. Set the values according to below given picture:

Click on Window<Preferences<Java<Installed JREs. Add jdk12 in the table.

Click on **Window<Preferences<Java<Installed JREs<Execution Environment**. Select JAVASE-12 under the heading <u>Execution Environment</u> and select the checkbox JDK 12 under heading <u>Compatible JREs</u>.

Click on Window<Preferences<Servers<Runtime Environment. Add servers as per requirement.

Click on Apply and Close.

type filter text Compiler > General Configure Project Specific Setti > Ant Gradle > Help JDK Compliance > Install/Update 12 > Java Use default compliance settings Code Coverage Code Coverage > Code Style Enable preview features > Debug Installed JREs JUnit Properties Files Editor > Java EE Source compatibility: JSON Language Servers Model Editor Add variable attributes to generated class files (used by the debugger) Add source file name to generated class files (used by the debugger) > Model Editor > Mylyn > Oomph > Domph	Preferences	— 🗆 X
> General ▲ > Ant Configure Project Specific Setti Gradle JDK Compliance > Help 12 > Install/Update Use 'release' option > Java Use default compliance settings > Appearance Build Path Code Coverage Preview features with severity level: Warning Code Style Generated .class files compatibility: 12 > Debug Source compatibility: 12 Source compatibility: 12 Disallow identifiers called 'assert': Error Disallow identifiers called 'enum': Error Disallow identifiers called 'enum': Error > Java Script JSON Add variable attributes to generated class files (used by the debugger) Add osurce file name to generated class files (used by the debugger) > Moven Model Editor Preserve unused (never read) local variables > Mylyn Inline finally blocks (larger class files, but improved performance) Store information about method parameters (usable via reflection)	type filter text	← ▼ ⇒ ▼
 Compiler Debug Editor Installed JREs JUnit Properties Files Editor Java EE Java Script JSON Language Servers Maven Model Editor Mylyn Oomph Disallow identifier and to generated class files (used by the debugger) Source tile name to generated class files (used by the debugger) Add source file name to generated class files (used by the debugger) Add source file name to generated class files (used by the debugger) Source tile name to generated class files (used by the debugger) Source file name to generated class files (used by the debugger) Source file name to generated class files (used by the debugger) Store information about method parameters (usable via reflection) 	 > General > Ant Gradle > Help > Install/Update > Java > Appearance > Build Path Code Coverage > Code Style 	Configure Project Specific Settings 12 ~ ity level: Warning ~
 JavaScript JSON Language Servers Maven Model Editor Mylyn Chassine Generation Add variable attributes to generated class files (used by the debugger) Add source file name to generated class file (used by the debugger) Add source file name to generated class file (used by the debugger) Add source file name to generated class file (used by the debugger) Add source file name to generated class file (used by the debugger) Difference Mylyn Inline finally blocks (larger class files, but improved performance) Store information about method parameters (usable via reflection) 	Compiler Debug Editor Installed JREs JUnit Properties Files Editor	12 ~ 12
> Run/Debug	 > JavaScript > JSON > Language Servers > Maven Model Editor > Mylyn > Oomph > Plug-in Development > Run/Debug 	ated class files (used by the debugger) enerated class files (used by the debugger) ed class file (used by the debugger) cal variables files, but improved performance) d parameters (usable via reflection)
Server Team TextMate Tomcat Validation Restore Defaults Apply	 > Server > Team > TextMate > Tomcat Validation 	Restore Defaults Apply

Step 4:

Click on **Project<Properties<JavaBuildPath** add the required jar files in Classpath and add JRE System Library and link it with jdk12 in Modulepath.

Set the execution environment to jdk12, as shown below.

Preferences				
type filter text	Installed JREs		•	⇔ - ⇔
> General A > Ant Gradle > Help	Add, remove or edit JRE of newly created Java pr Installed JREs:	definitions. By default, the checked J ojects.	RE is added t	to the build path
> Install/Update	Name	Location	Type	Add
✓ Java	⊠ ⇒ idk-12.0.1 (C:\Program Files\Java\idk-12.0.1	Standard	
> Appearance	☐ ➡ idk1.8.0 191	C:\Program Files\Java\idk1.8.0 1	Standard	Edit
Sode Coverage		gg		Duplicate
Code Style				
> Compiler				Remove
> Debug			[[Search
> Editor			L	
> Installed JREs				
JUnit				
Properties Files Editor				
> Java EE				
> JavaScript				
> JSON				
> Language Servers				
Model Editor				
> Mylyn				
> Oomph				
> Plug-in Development				
> Run/Debug				
> Server				
> Team	4			
> TextMate	`			
> Tomcat				
Validation 🗸				Apply
< >				Арріу
? 卢 🖌 🔘		Apply ar	nd Close	Cancel

Click on Apply and Close.

Note: Some code modification has to be carried out in build.gradle to make it compatible with newer version of gradle.

Click on **Project<Properties<Project Facets** set the java version to 12.

Properties for ELP-BE-EDI		_		×
type filter text	Java Compiler		⇔ - d	× •
> Resource Builders	Enable project specific settings	Configure Worl	kspace Sett	ings 🔺
Coverage	JDK Compliance			
Git	Use compliance from execution environment 'JavaSE-12' on the 'Java B	Build Path'		
Java Build Path	Compiler compliance level:		12	\sim
> Java Code Style				
V Java Compiler	✓ Userelease option			
Annotation Processing Building	Se default compliance settings			
Errors/Warnings	Enable preview features			
Javadoc	Preview features with severity level:		Warning	
Task Tags	Generated .class files compatibility:		12	\sim
> Java Editor	concrete and the compact may			_
Javadoc Location	Source compatibility:		12	\sim
Project Facets	Disallow identifiers called 'assert':		Error	\sim
Project References	Disallow identifiers called 'enum':		Error	\sim
Refactoring History	Char Che Connection			
Run/Debug Settings				
Server	Add variable attributes to generated class files (used by the debugger)			
> Task Repository	Add line number attributes to generated class files (used by the debug	ger)		
lask lags	Add source file name to generated class file (used by the debugger)			
Tomcat Validation	Preserve unused (never read) local variables			
WikiText	Inline finally blocks (larger class files, but improved performance)			
	Store information about method parameters (usable via reflection)			
				~
(?)	Ar	oply and Close	Cano	el
U U		in the close	Carro	

Errors and Exceptions:

1. If this " com.sun.net.ssl.HttpsURLConnection" is indicating as an error then-

add one jar file named **com.sun.net.ssl.jar** in the build path of ELP-FE

The above bug of eclipse has been solved in the June 2019 release.