Context menu (left panel):

Menu item	Shortcut
New	>
Go Into	
Open in New Window	
Open Type Hierarchy	F4
Show In	Alt+Shift+W >
Show in Local Terminal	>
Copy	Ctrl+C
Copy Qualified Name	
Paste	Ctrl+V
Delete	Delete
Remove from Context	Ctrl+Alt+Shift+Down
Build Path	>
Source	Alt+Shift+S >
Refactor	Alt+Shift+T >
Import...	
Export...	
Source	>
Refresh	F5
Close Project	
Assign Working Sets...	

MANIFEST.MF editor:

```
1 Manifest-Version: 1.0
2 Main-Class: main.Teste
3 Class-Path: libs/log4j-api-2.22.1.jar libs/log4j-core-2.22.1.jar
```

Export dialog:

Export

Select

Export resources into a JAR file on the local file system.

Select an export wizard:

type filter text

- > EJB
- > Install
- ∨ Java
  - JAR file
  - Javadoc
  - Runnable JAR file
- > Java EE
- > Plug-in Development
- > Run/Debug
- > Tasks
- > Team
- > Web
- > Web Services
- > XML

< Back | Next > | Finish | Cancel

JAR Export

JAR File Specification

Define which resources should be exported into the JAR.

Select the resources to export:

```
> ■ 🗁 TesteLog4j2
```

☑ X .classpath
☑ X .project

☑ Export generated class files and resources

☐ Export all output folders for checked projects

☐ Export Java source files and resources

☐ Export refactorings for checked projects. Select refactorings...

Select the export destination:

JAR file: `C:\tmp\Teste.jar` ⌄   Browse...

Options:

☑ Compress the contents of the JAR file

☐ Add directory entries

☐ Overwrite existing files without warning

? | < Back | Next > | Finish | Cancel

JAR Export

JAR Packaging Options
Define the options for the JAR export.

Select options for handling problems:
☑ Export class files with compile errors
☑ Export class files with compile warnings

☐ Create source folder structure

☑ Build projects if not built automatically

☐ Save the description of this JAR in the workspace

Description file: [   ] Browse...

< Back   Next >   Finish   Cancel

JAR Export

JAR Manifest Specification
Customize the manifest file for the JAR file.

Specify the manifest:

○ Generate the manifest file

☐ Save the manifest in the workspace

☐ Use the saved manifest in the generated JAR description file

Manifest file: /TesteLog4j2/META-INF/MANIFEST.MF [Browse...]

◉ Use existing manifest from workspace

Manifest file: /TesteLog4j2/META-INF/MANIFEST.MF [Browse...]

Seal contents:

○ Seal the JAR [Details...]

◉ Seal some packages   Nothing sealed [Details...]

Select the class of the application entry point:

Main class: [   ] [Browse...]

? [< Back] [Next >] [Finish] [Cancel]

```
Matheus.Aguiar@INFOOKI-048 MINGW64 /c/tmp
$ ls -l
total 1932
-rw-r--r-- 1 Matheus.Aguiar 197121 1976532 Feb  9 19:27 Teste.jar
drwxr-xr-x 1 Matheus.Aguiar 197121       0 Feb  9 18:39 libs/

Matheus.Aguiar@INFOOKI-048 MINGW64 /c/tmp
$ cd libs

Matheus.Aguiar@INFOOKI-048 MINGW64 /c/tmp/libs
$ ls -l
total 2184
-rw-r--r-- 1 Matheus.Aguiar 197121  335001 Feb  7 16:49 log4j-api-2.22.1.jar
-rw-r--r-- 1 Matheus.Aguiar 197121 1900022 Feb  7 16:49 log4j-core-2.22.1.jar

Matheus.Aguiar@INFOOKI-048 MINGW64 /c/tmp/libs
$ |
```

```
Matheus.Aguiar@INFOOKI-048 MINGW64 /c/tmp
$ LIBS=c:/tmp/libs/*

Matheus.Aguiar@INFOOKI-048 MINGW64 /c/tmp
$ export LIBS

Matheus.Aguiar@INFOOKI-048 MINGW64 /c/tmp
$ echo $LIBS
c:/tmp/libs/log4j-api-2.22.1.jar c:/tmp/libs/log4j-core-2.22.1.jar

Matheus.Aguiar@INFOOKI-048 MINGW64 /c/tmp
$ java -cp $LIBS -jar Teste.jar
Erro: Não foi possível localizar nem carregar a classe principal c:.tmp.libs.log4j-core-2.22.1.jar
Causada por: java.lang.ClassNotFoundException: c:.tmp.libs.log4j-core-2.22.1.jar

Matheus.Aguiar@INFOOKI-048 MINGW64 /c/tmp
$ |
```