|  |  |  |
| --- | --- | --- |
| N0 | Steps | Description |
| 1 | Setup.txt | Follow setup.txt to copy the onSpring/onSpring.properties on user.home |
| 2 | Dependencie | You need to set in your PATH  C:\tools\gradle-7.5.1  More: <https://docs.gradle.org/7.5.1/release-notes.html> |
| 3 | Java | C:\Program Files\Java\jdk-18.0.2 |
| 4 | Eclipse project | Run gradle.bat eclipse. The dependencies should get sorted |
| 5 | Launch the junit test via eclipse | Using 1 run configuration:  io.OnLogBackLauncher |
| 6 | Explanation | /\*\*  \* Configuration files:  \* the logger is defined in a file in classpath: either in logback.xml / logback-test.xml  \* Once found the loggers are defined by 3 parts:  \* - 1/ the name see a/.  \* - 2/ the level  \* - 3/ the appender to use: format of the line + whether log file or console to use.  \* a/ nameof logger and package of java: If the java class starts with the package name that coresponds to the appender  \* b/ A ladder of statuses are defined with most important level:  \* error most important  \* info  \* debug  \* trace  \* So if you chose the the less important level (trace) all the more important levels for a specific appender will be written in logs of this appender.  \*  \*/ |
| 7 | Work without configuration file with the console | ***logger***.info("We are using info logs with Olivia");  ***logger***.error("We are using error logs with Olivia");  ***logger***.debug("We are using error logs with Olivia");  ***logger***.trace("We are using error logs with Olivia");  /\*\*  \* by default it would write in the console without need for logback.xml  \* configured 11:26:14.843 [main] INFO io.OnLogBackLauncher - We are using info  \* logs with Olivia 11:26:14.852 [main] ERROR io.OnLogBackLauncher - We are  \* using error logs with Olivia 11:26:14.852 [main] DEBUG io.OnLogBackLauncher -  \* We are using error logs with Olivia  \*\*/ |