LOG BACK

Logback is provided out of the box with Spring Boot when you use one of the Spring Boot starter dependencies, as they include spring-boot-starter-logging — providing logging without any configuration that can be altered to work differently if required. There are two ways of providing your own configuration. If you only need simpler alterations, they can be added to a properties file, such as application.properties or for more complex needs, you can use XML or Groovy to specify your settings.

‘

It gives you the following information:

 Date and Time that gives the date and time of the log

 Log level shows INFO, ERROR or WARN

 Process ID

 The --- which is a separator

 Thread name is enclosed within the square brackets []

 Logger Name that shows the Source class name

 The Log message

Console Log Output

The default log messages will print to the console window. By default, “INFO”, “ERROR” and “WARN” log messages will print in the log file.

File Log Output

By default, all logs will print on the console window and not in the files. If you want to

print the logs in a file, you need to set the property logging.file or logging.path in the

application.properties file.

Log Levels

Spring Boot supports all logger levels such as “TRACE”, “DEBUG”, “INFO”, “WARN”,

“ERROR”, “FATAL”, “OFF”. You can define Root logger in the application.properties file as

shown below:

logging.level.root=WARN

Note: Logback does not support “FATAL” level log. It is mapped to the “ERROR” level log.

Configure Logback

Logback supports XML based configuration to handle Spring Boot Log configurations.

Logging configuration details are configured in logback.xml file. The logback.xml file

should be placed under the classpath.