LOGGING

* JAVA (SINCE 1.4 version) logging in inbuilt java.util.logging
* Java.util.handler --> 1)How to handle message 2) console file(file handler)
* Logging that came from JDK kinda sucked. (Performance issue, not flexible, need more features)
* So ppl need for a logging library , out of box functionality in jdk not good enough so ppl go for open source library ----> some of the oldest popular open source library Log4j
* Log 4j declaration

private static final Logger *logger* = LoggerFactory.*getLogger*(HelloWorldApplication.class);

🡪 instead handlers Log4j has Appeneders

1. File Appeners
2. Console Appeneders
3. Rolling file appenders (that can create a new log file for eachday)
4. JDBC appenders (directly log to the database)
5. SMTP (send log message via email)

With Formatting Option

If we are switch to one library to another library. We have to go back change and where you were intracting with library.

TO rectify ppl started using FAÇADE library

1)APACHE COMMONS

2) SLF4J (Became super duper popular)

Concept of SLF4J is simple. It has no functionality only interfaces

Acutal Logging library

Log4j Jar(Actually Do logging)

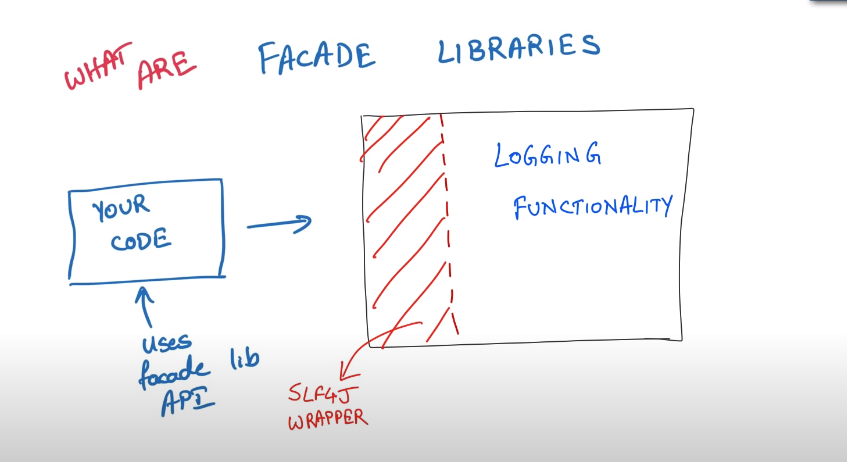
Façade Lib

SLF4J API + calls to actual lib

Your Code

Implement SLF4J

When we write code it’ll call Façade lib and that’ll call another actual library that might be Log4j or whatever we want.



And Two current popular Library

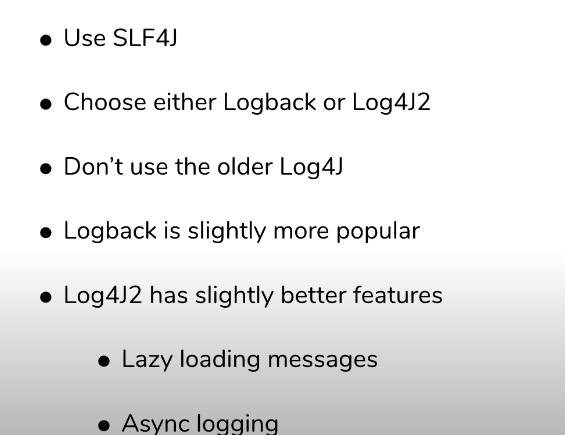
1. LogBack
2. Log4J2

And those to support SLF4J

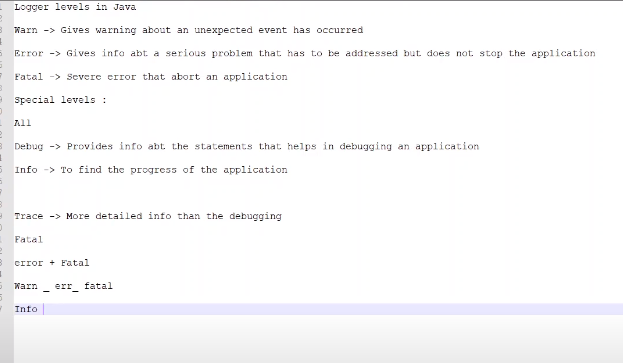
If two application are there.. one is using LogBack and another app using Log4J2

Instead of that We can go for SLF4J.. it supports both.

Recommended Logging



LOGGING LEVELS



Info – info, warn , error, fatal

Debug – debug, info , warn , error, fatal

All – all log