



# Logging & Log4J

**DURGA ADIMULAM**

FEBRUARY 18, 2020

# Logging

---

Logging is the process of writing log messages during the execution of a program to a central place. This logging allows you to report and persist error and warning messages as well as info messages (e.g., runtime statistics) so that the messages can later be retrieved and analyzed.

The object which performs the logging in applications is typically just called **Logger**

# Why Logging

If we use SOP (`System.out.print()`) statements to print log messages, then we can run into some disadvantages like:

- We can print log messages on the console only. So, when the console is closed, we will lose all of those logs.
- We can't store log messages in any permanent place. These messages will print one by one on the console because it is a single-threaded environment.

To overcome these problems, the Log4j framework came into the picture. Log4j is an open source framework provided by Apache for Java projects.

# Advantages of Logging

---

- Quick Debugging
- Problem Diagnosis
- Easy Maintenance
- Cost and Time Savings

# Log4J

---

Log4j is a reliable, fast and flexible logging framework (APIs) written in Java, which is distributed under the Apache Software License.

Log4j is highly configurable through external configuration files at runtime.

Log4J has three main components :

- Logger
- Appender
- Layout

# Log4J Features

---

- It is thread-safe.
- It is optimized for speed.
- It is based on a named logger hierarchy.
- It supports multiple output appenders per logger.
- Logging behavior can be set at runtime using a configuration file.

Logging is an important component of the software development. A well-written logging code offers quick debugging, easy maintenance, and structured storage of an application's runtime information.

# Log4J Components

## Logger :

Logger is a class in the *org.apache.log4j.\** package. We have to initialize one Logger object for each Java class. We use Logger's methods to generate log statements. Log4j provides the factory method to get Logger objects.

```
private static Logger logger = LogManager.getLogger(AppClass.class)
```

## Methods in Logger class

```
logger.info()
```

```
logger.debug()
```

```
logger.warn()
```

```
logger.fatal()
```

```
logger.error()
```

# Log4J Components

## Levels :

Level is a class in the org.apache.log4j.\* package. Each level has a different priority order as below

ALL < DEBUG < INFO < WARN < ERROR < FATAL < OFF

Level	Comments
DEBUG	Log messages that are useful to debug an application
INFO	Log messages that highlights the progress of the application at coarse-grained level
WARN	Log messages that are potentially harmful situations
ERROR	Logs error events that might still allow the application to continue running
FATAL	Logs very severe error events that will presumably lead the application to abort



# Log4J Components

---

## Appender :

Appender is used to write messages into a file or DB or SMTP.

Log4j has different types of appenders.

- ConsoleAppender
- DailyRollingFileAppender
- FileAppender
- JDBCAppender
- SMTPAppender
- RollingFileAppender

# Log4J Components

---

## Layout :

This is used to define the formatting in which logs will print in a repository.

We have different types of layouts

- PatternLayout
- SimpleLayout
- XMLLayout
- HTMLLayout

- Write Program to calculate simple interest, and compound interest
- Write Program to estimate house construction cost, based on various materials  
Construction cost per square feet is
  - 1200INR if we use standard materials
  - 1500INR if we use above standard materials
  - 1800INR if customers needs high standard material
  - 2500INR if customer needs high standard material and fully automated home
  - Input will be material standard, total area of house and if they want fully automated home

#### Notes:

1. Write the above programs by following clean code.
2. Don't use main method
3. Don't use System.out.println
4. Write client application to use above services
5. Use Java for completing these tasks
- 6. Use Logger to print messages**

<https://epa.ms/EPAMPEP2020SESSION4TASK>

# References

---

<https://logging.apache.org/log4j/2.x/>

<https://logging.apache.org/log4j/2.x/manual/configuration.html>

<https://howtodoinjava.com/log4j2/>

<https://logging.apache.org/log4j/1.2/apidocs/org/apache/log4j/PatternLayout.html>