**Steps to add log4j**

1. **Add dependencies “**slf4j-api**”,”** log4j**”**
2. **Use standard “log4j.properties” file, keep in project**
3. **Create an object of logger and give logger name**

**public** **static** Logger *logger*;

@Test

**public** **void** f()

{

*logger*= *Logger*.*getLogger*("My\_New\_log");

PropertyConfigurator.*configure*("log4j.properties");

*logger*.info("Hello");

}



<dependency>

<groupId>org.slf4j</groupId>

<artifactId>slf4j-api</artifactId>

<version>2.0.6</version>

</dependency>

<dependency>

<groupId>log4j</groupId>

<artifactId>log4j</artifactId>

<version>1.2.17</version>

</dependency>

</dependencies>

|  |  |
| --- | --- |
| **Level** | **Description** |
| ALL | All levels including custom levels. |
| DEBUG | Designates fine-grained informational events that are most useful to debug an application. |
| INFO | Designates informational messages that highlight the progress of the application at coarse-grained level. |
| WARN | Designates potentially harmful situations. |
| ERROR | Designates error events that might still allow the application to continue running. |
| FATAL | Designates very severe error events that will presumably lead the application to abort. |
| OFF | The highest possible rank and is intended to turn off logging. |
| TRACE | Designates finer-grained informational events than the DEBUG. |