**Components Of Log4j**

**Logger:**

* Logger is a class, in org.apache.log4j.\*
* We need to create Logger object one per java class
* Logger methods are used to generate log statements in a java class instead of SOPs
* So in order to get an object of Logger class, we need to call a static factory method [ factory method will gives an object as return type ]
* We must create Logger object right after our class name.

final static Logger log = Logger.getLogger(YourClassName.Class)

Note:  while creating a Logger object we need to pass either fully qualified class name or class object as a parameter, class means current class for which we are going to use Log4j.

* Logger object has some methods, actually we used to print the status of our application by using these methods only

We have totally 5 methods in Logger class

* + debug()
  + info()
  + warn()
  + error()
  + fatal()

As a programmer it’s our responsibility to know where we need to use what method

* **Priority of these methods:**

**debug** < **info** < **warn** < **error** < **fatal**

**Appender:**

* Appender job is to write the messages into the external file or database or smtp.
* Logger classes generates some statements under different levels and this Appender takes these log statements and stores in some files or database
* In log4j we have different Appender  implementation classes
* FileAppender [ writing into a file ]
  + RollingFileAppender
  + DailyRollingFileAppender
* ConsoleAppender [ Writing into console ]
* JDBCAppender [ For Databases ]
* SMTPAppender [ Mails ]
* SocketAppender [ For remote storage ]
* SocketHubAppender
* SyslogAppendersends
* TelnetAppender

**Layout:**

* This component specifies the format in which the log statements are written into the destination by the appender.
* We have different type of layout classes in log4j
* SimpleLayout
* PatternLayout
* HTMLLayout
* XMLLayout