1. Explain what is Log4j?

Answer: Log4j is a Java-based logging utility used for generating log statements from applications.

It allows developers to control log output and format, set log levels, and direct log messages to

various outputs like the console, files, or external systems. Log4j helps in tracking and debugging

application behavior by providing a flexible and efficient logging framework.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

2. Explain why to use Apache Log4j?

Answer:

\* Being open-source its completely free to use.

\* You can easily save log information into either files or even databases.

\* Can be used for projects of any sizes small or large.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

3. Mention what are the three principal components of Log4j?

Answer: The three principal components of Log4j are

\* Loggers

\* Appenders

\* Layout

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

4. What are the features of log4j?

Answer: Following are features of log4j:

\* It is thread-safe.

\* It is optimized for speed.

\* It is based on a named logger hierarchy.

\* It supports multiple output appenders per logger.

\* It supports internationalization.

\* It is not restricted to a predefined set of facilities.

\* Logging behavior can be set at runtime using a configuration file.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

5. Inside logger component what are the different log levels?

Answer: Different log levels inside logger components are

\* All -

\* Debug -

\* 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.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

6. What is purpose of TRACE log level?

Answer: TRACE - Designates finer-grained informational events than the DEBUG.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

7. What is the purpose of Appender object?

Answer: Appender Object: This is a lower-level layer of log4j architecture which provides Appender

objects. The Appender object is responsible for publishing logging information to various preferred

destinations such as a database, file, console, UNIX Syslog, etc.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

8. Mention what are the different types of Appenders?

Answer: Some of the Appenders type include

\* ConsoleAppender logs to standard output

\* FileAppender prints logs to some file

\* Rolling file appender to a file with maximum size

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

9. Explain what is layouts in log4j?

Answer: Layout in log4j is responsible for formatting logging information in different styles.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

10. What is the purpose of ObjectRenderer object?

Answer: ObjectRenderer: The ObjectRenderer object is specialized in providing a String representation

of different objects passed to the logging framework. This object is used by Layout objects to prepare

the final logging information.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

11. How will you create a logger in any class?

Answer: Any other named Logger object instance is obtained through the second method by

passing the name of the logger. The name of the logger can be any string you can pass,

usually a class or a package name as we have used in the last chapter and it is mentioned below −

static Logger log = Logger.getLogger(log4jExample.class.getName());

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

12. How log4j file is defined?

Answer: Log4j file is defined by the name log4j.properties, it keeps properties in key-value pairs.

By default, the log manager looks for a file name log4j.properties in the CLASSPATH.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

13. Mention what are the other support objects in Log4j?

Answer: There are other support objects in Log4j framework they are

\* Level Object

\* Filter Object

\* Object Renderer

\* Log Manager

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

14. Explain what is the command to write your logging information into a file?

Answer: To write your logging information into a file, you would need to use a command

org.apache.log4j.FileAppender

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

15. Mention what are the logging methods provided by logger class?

Answer: Logger class provides a variety of methods to handle logging activities. To obtain a

logger object it provides two static methods

\* Public static logger getRootLogger();

\* Public static logger getLogger(String name);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

16. In log4j how can you log into the database?

Answer: The log4j API provides the object org.apache.log4j.jdbc. JDBCAppender object can put

logging information in a particular database.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

17. Explain what are the system properties checked by log4j?

Answer: The system properties checked by log4j are

\* Log4j debug, if true, log4j will show internal debugging messages to the console

\* defaultInitOverride, if true, log4j will not execute default initialization

\* configuration, URL for default initialization configuration file

\* configurationClass, Class name for configurator to execute default initialization configuration file

\* ignoreTCL, if true, the thread class loader will be overlooked when loading classes

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

18. Mention what is the role of filter in log4j?

Answer: Filter objects in log4j decide whether logging request should be handled by a particular

Appender or ignored.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

19. Explain how can you get multiple processes to log to the same file?

Answer: You may have each process log to a socket Appender. The receiving socket server can

receive all the events and send them to a single log file.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

20. Mention what is the difference between Threshold and LevelRangeFilter in log4j?

Answer: Both Threshold and LevelRangeFilter does the same thing. However threshold should

be faster. Filters enable you to implement your own logic, and you can also link them together

if required. If you need a basic threshold functionality, then “threshold” function will be enough.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

21. Explain what are the format characters used in log4j?

Answer: The format characters used in log4j are:

\* L: it is used to output the line number from where the logging request was processed or issued

\* M: It is used to output the application supplied message related with the logging event

\* P: It is used to output the priority of the logging event

\* C: It is used to output the class name of the caller issuing the logging request

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

22. Explain what is package level logging in log4j?

Answer: Package level logging is the standard logging of log4j, with this you would determine

the package and the associated level.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

23. Mention what does .class mean in log4j context?

Answer: In log4j context, .class is used to get the full name of your class and that string is

used to configure this logger object.

For example : logger.getlogget (Myclass.class)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

24. Explain how can you get multiple processes to log to the same file?

Answer: You may have each process log to a socket Appender. The receiving socket server can

receive all the events and send them to a single log file.