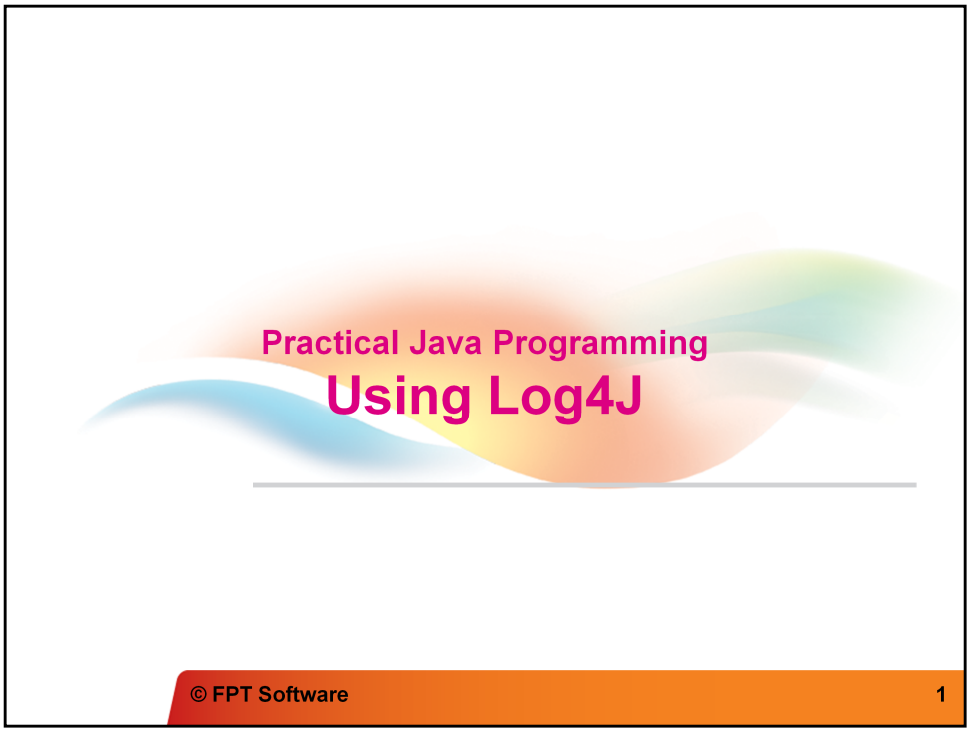


<*Lưu ý: không xóa nội dung cũ mà viết thêm lên trên như nhật ký*>

Tài liệu đào tạo: <*môn học, buổi học*>

Phiên bản tài liệu:

Người cập nhật:

Ngày cập nhật:

Tóm tắt nội dung cập nhật chính:

Ngày ban hành sử dụng:

Agenda

- Overview
- Add Log4J library
- Create configuration file
- Add logging initialization statements
- Add logging code
- Appender overview
- Log level overview
- Layout overview
- Filter overview

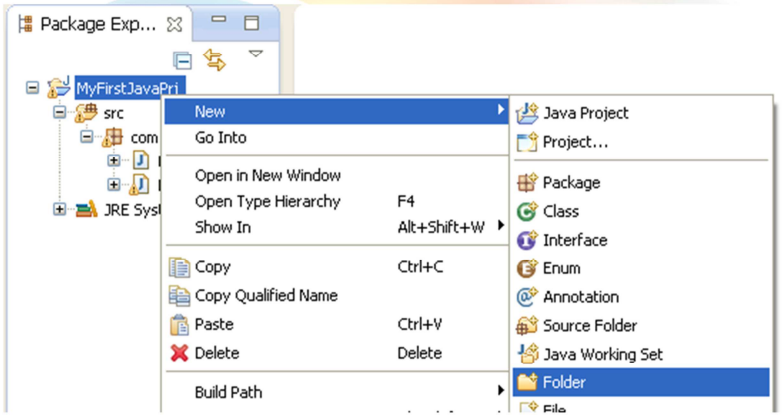
OVERVIEW

Log4J Overview

- Log message in any place of code
- Control many kinds of output by appenders
- Customize the logging behavior for a specific code part
- Filter the log message by log level, keyword
- Format the output message by log layout

LOG4J LIBRARY

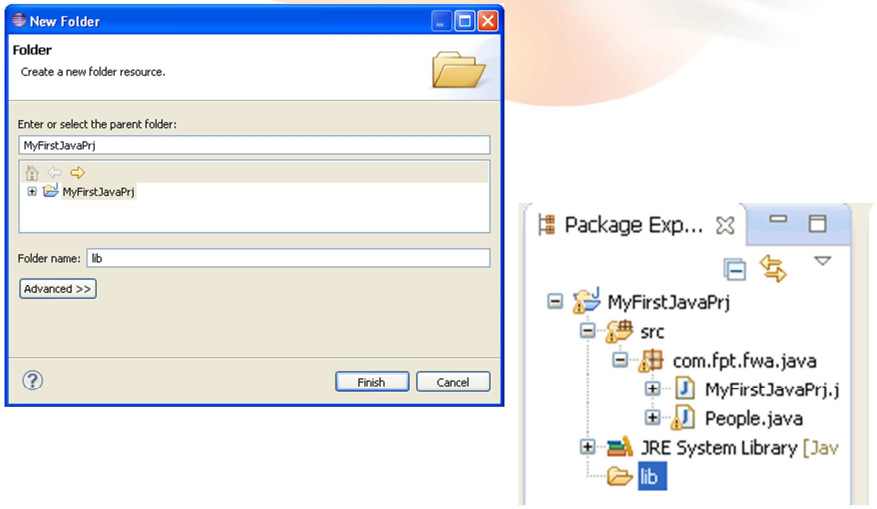
Add library folder



Determine where the library stays

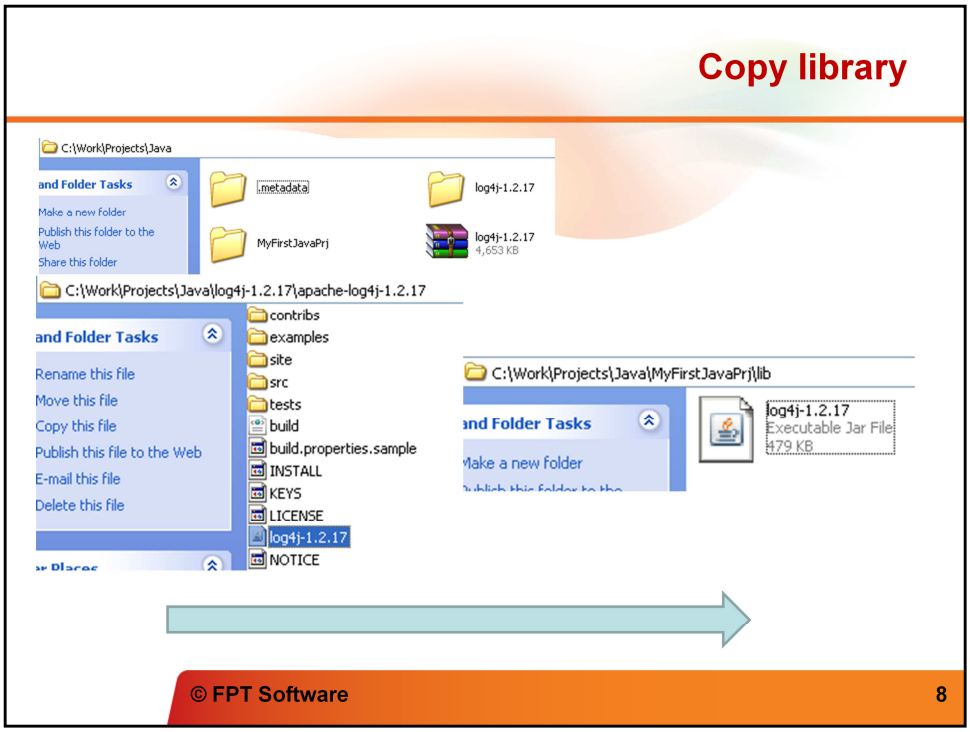
Then add library to project

Add library folder dialog



Determine where the library stays

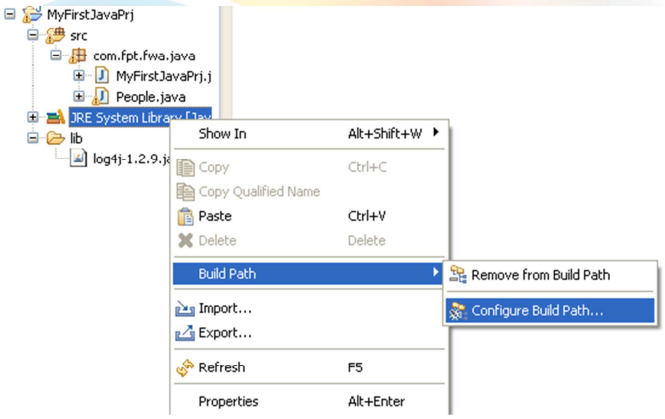
Then add library to project



Determine where the library stays

Then copy library to the lib folder

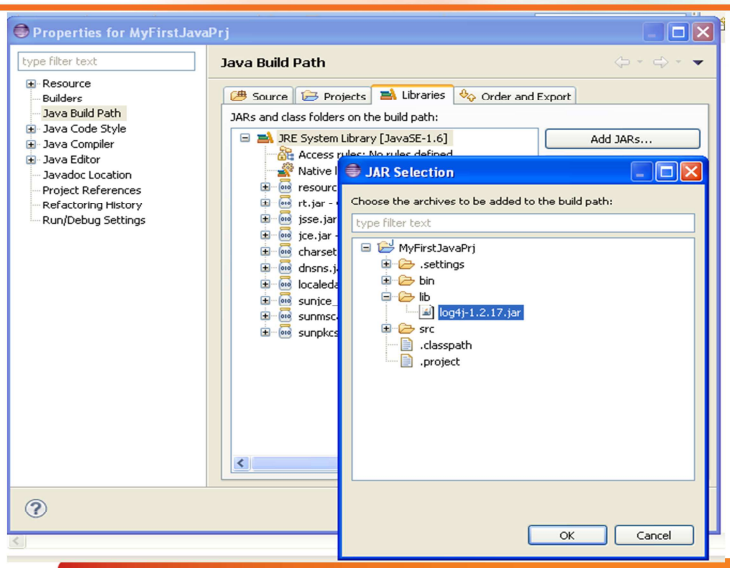
Add library



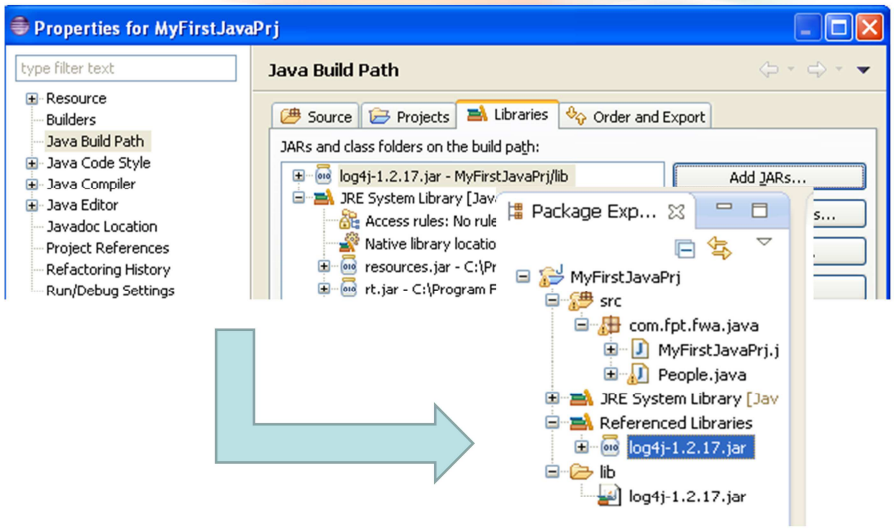
Determine where the library stays

Then add library to project

Add library dialog



Add library result

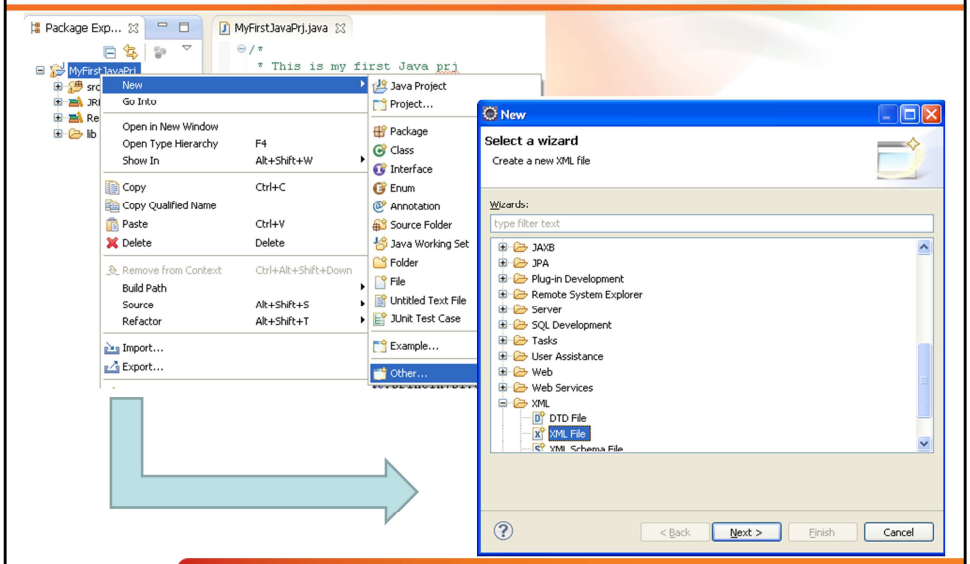


LOG4J CONFIGURATION FILE

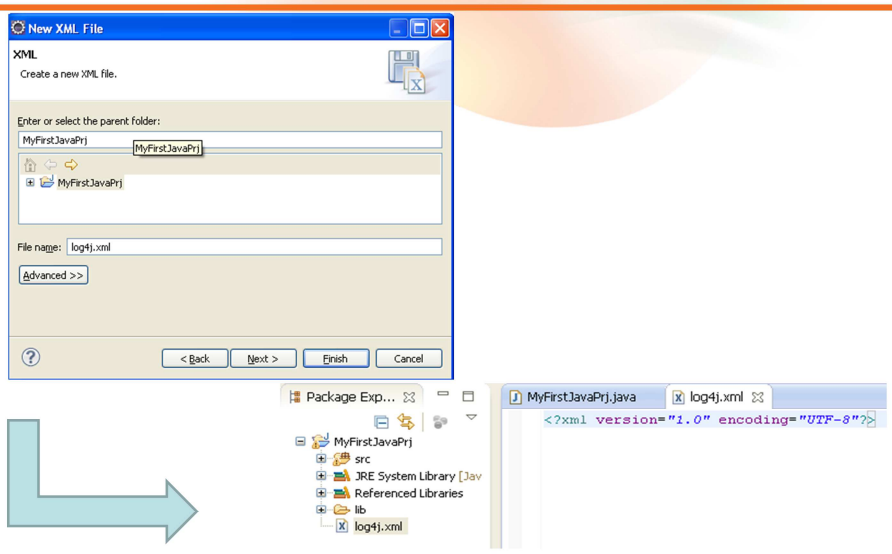
Content

- Add a new configuration file
- Register to Assembly
- Main structure
- Appender Section
- Logger Section

Creation


Package Exp...
MyFirstJavaPrj.java
/*
* This is my first Java prj
MyFirstJavaPrj
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Source Alt+Shift+S
Refactor Alt+Shift+T
Import...
Export...
Java Project
Project...
Package
Class
Interface
Enum
Annotation
Source Folder
Java Working Set
Folder
File
Untitled Text File
JUnit Test Case
Example...
Other...
New
Select a wizard
Create a new XML file
Wizards:
type filter text
JAXB
JPA
Plug-in Development
Remote System Explorer
Server
SQL Development
Tasks
User Assistance
Web
Web Services
XML
DTD File
XML File
< Back
Next >
Finish
Cancel

New file created



Main structure

MyFirstJavaPrj.java | log4j.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd" >
<log4j:configuration debug="true"
    xmlns:log4j='http://jakarta.apache.org/log4j/'>

    <!-- Appender -->

    <!-- Logger -->

</log4j:configuration>
```

Appenders

MyFirstJavaPrj.java | log4j.xml | People.java

```
<!-- FileAppender -->
<appender name="FileAppender" class="org.apache.log4j.RollingFileAppender">
    <param name="file" value="mylogfile.txt" />
    <param name="append" value="true" />
    <param name="MaxBackupIndex" value="5" />
    <param name="MaxFileSize" value="1KB" />
    <layout class="org.apache.log4j.PatternLayout">
        <param name="ConversionPattern"
            value="%d{dd MMM yyyy HH:mm:ss,SSS} [%t] %-5p %c %x - %m%n" />
    </layout>
</appender>

<!-- ConsoleAppender -->
<appender name="ConsoleAppender" class="org.apache.log4j.ConsoleAppender">
    <layout class="org.apache.log4j.PatternLayout">
        <param name="ConversionPattern"
            value="%d{dd MMM yyyy HH:mm:ss,SSS} [%t] %-5p %c %x - %m%n" />
    </layout>
    <filter class="org.apache.log4j.varia.StringMatchFilter">
        <param name="stringToMatch" value="begin" />
        <param name="acceptOnMatch" value="true" />
    </filter>
    <filter class="org.apache.log4j.varia.DenyAllFilter" />
</appender>
```


Present detail about any element of appender, attention to filter

Loggers

MyFirstJavaPrj.java | log4j.xml | People.java

```
    </appender>

    <!-- specific logger for class Person                -->
    <!-- in package com.fpt.fwa.java                     -->
    <!-- to set the logger for whole package, put only   -->
    <!-- package name                                    -->
    <logger name="com.fpt.fwa.java.People" additivity="false">
        <level value ="debug" />
        <appender-ref ref="ConsoleAppender"/>
        <appender-ref ref="FileAppender"/>
    </logger>

    <!-- Required Root logger -->
    <root>
        <level value ="info" />
        <appender-ref ref="FileAppender" />
    </root>

</log4j:configuration>
```

Summary

- Add a new configuration file: Add xml file to the root folder
- Main structure: Appenders and loggers
- Appender Section: File rolling and console
- Logger Section: Root required and specific logger is optional

LOG4J INITIALIZE

Initialization code

MyFirstJavaPrj.java | log4j.xml | People.java

```
package com.fpt.fwa.java;

//Import log4j classes.
import org.apache.log4j.Logger;
import org.apache.log4j.xml.DOMConfigurator;

/*
```

```
// any program starts with Main method, with or without
public static void main(String[] args) {

    DOMConfigurator.configure("log4j.xml");
    logger.info("Entering application.");
```

LOGGING

Add logging

MyFirstJavaPrj.java | log4j.xml | People.java

```
package com.fpt.fwa.java;

//Import log4j classes.
import org.apache.log4j.Logger;
import org.apache.log4j.xml.DOMConfigurator;

public class MyFirstJavaPrj {

   // Define a static logger variable so that it references the
   // Logger instance named "MyFirstJavaPrj".
   static Logger logger = Logger.getLogger(MyFirstJavaPrj.class);

          // any program starts with Main method, with or withc
          public static void main(String[] args) {

              DOMConfigurator.configure("log4j.xml");
              logger.info("Entering application.");

              People p1 = new People("FPT 03", "03.03.2013");
              System.out.println(p1.toString());

              logger.info("Exiting application.");
          }
```

Multi level logging

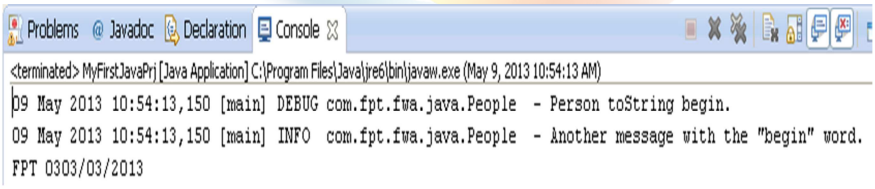
MyFirstJavaPrj.java | log4j.xml | People.java

```
    }

    public String toString(){
        logger.debug("Person toString begin.");

        SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
        if (!isValide) return "Is not valide object";

        logger.info("Another message with the \"begin\" word.");
        logger.debug("Person toString end.");

        return name + df.format(dayOfBirth);
    }
}
```

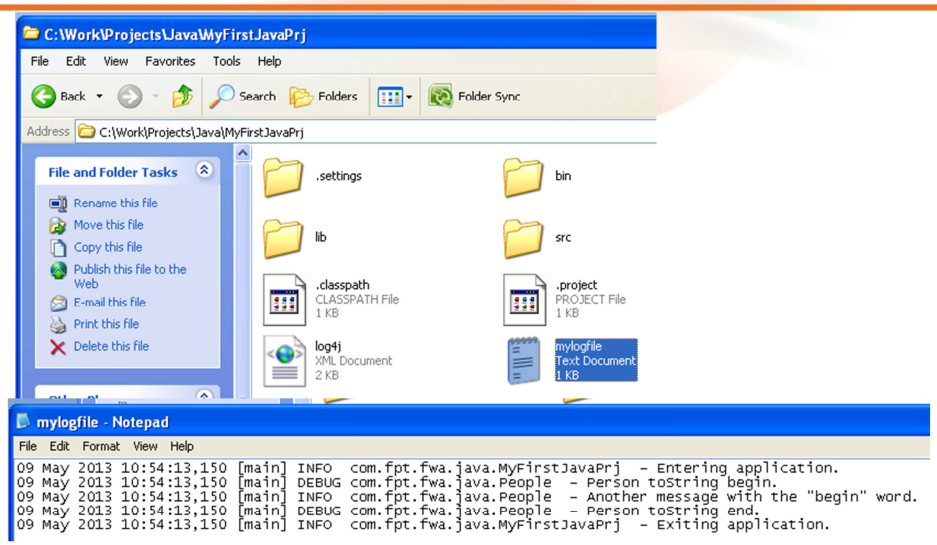

Last log event is not catch cause of Filter

RUNNING RESULT

Console Output

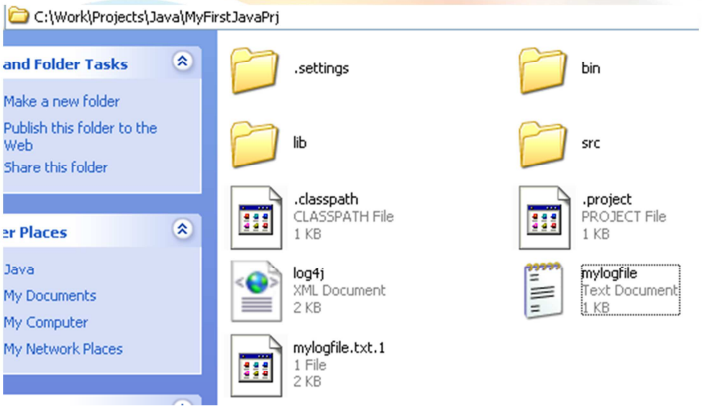


File Output



Point out the folder and file name

File Output: rolling



The last line is not sent out cause of filter

File Output: rolling file

The last line is not sent out cause of filter

APPENDERS

Appender using

MyFirstJavaPrj.java | log4j.xml | People.java

```
<!-- FileAppender -->
<appender name="FileAppender" class="org.apache.log4j.RollingFileAppender">
    <param name="file" value="mylogfile.txt" />
    <param name="append" value="true" />
    <param name="MaxBackupIndex" value="5" />
    <param name="MaxFileSize" value="1KB" />
    <layout class="org.apache.log4j.PatternLayout">
        <param name="ConversionPattern"
            value="%d{dd MMM yyyy HH:mm:ss,SSS} [%t] %-5p %c %x - %m%n" />
    </layout>
</appender>

<!-- ConsoleAppender -->
<appender name="ConsoleAppender" class="org.apache.log4j.ConsoleAppender">
    <layout class="org.apache.log4j.PatternLayout">
        <param name="ConversionPattern"
            value="%d{dd MMM yyyy HH:mm:ss,SSS} [%t] %-5p %c %x - %m%n" />
    </layout>
    <filter class="org.apache.log4j.varia.StringMatchFilter">
        <param name="stringToMatch" value="begin" />
        <param name="acceptOnMatch" value="true" />
    </filter>
    <filter class="org.apache.log4j.varia.DenyAllFilter" />
</appender>
```


Present detail about any element of appender, attention to filter

Appenders

- File rolling
- Console
- File
- Daily File rolling
- JDBC – to Database
- SMTP – by email
- Other…

All == Debug

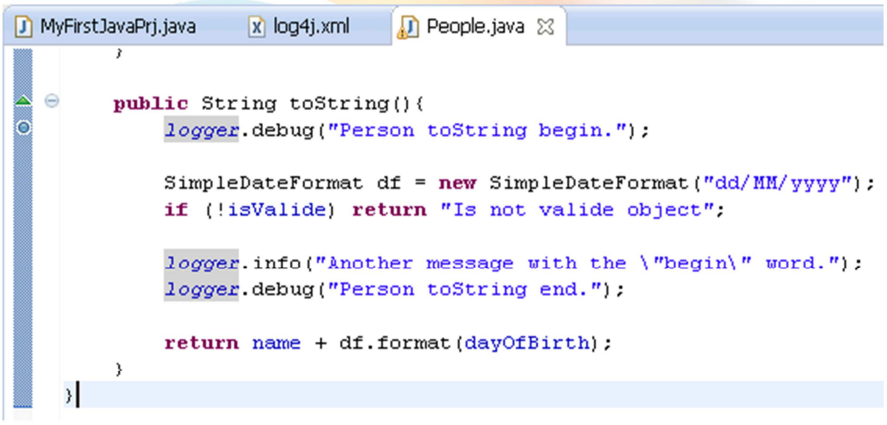
LOG LEVELS

Loggers

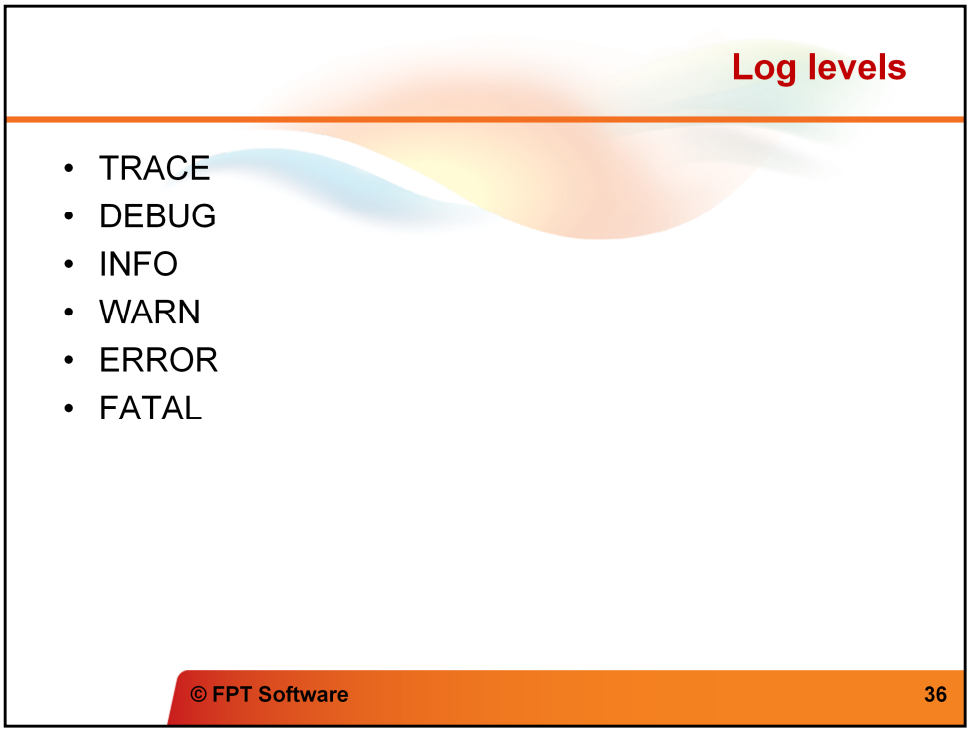
MyFirstJavaPrj.java | log4j.xml | People.java

```
    </appender>

    <!-- specific logger for class Person                -->
    <!-- in package com.fpt.fwa.java                     -->
    <!-- to set the logger for whole package, put only   -->
    <!-- package name                                    -->
    <logger name="com.fpt.fwa.java.People" additivity="false">
        <level value ="debug" />
        <appender-ref ref="ConsoleAppender"/>
        <appender-ref ref="FileAppender"/>
    </logger>

    <!-- Required Root logger -->
    <root>
        <level value ="info" />
        <appender-ref ref="FileAppender" />
    </root>

</log4j:configuration>
```

Log level using



All == Debug



Order from Trace to Fatal

Logger with level Trace will display msg for all log level events

Logger with level Debug will display msg for all log level event from Debug to Fatal and so on

LOG PROPAGATE/ADDITIVITY

Log propagate/Additivity

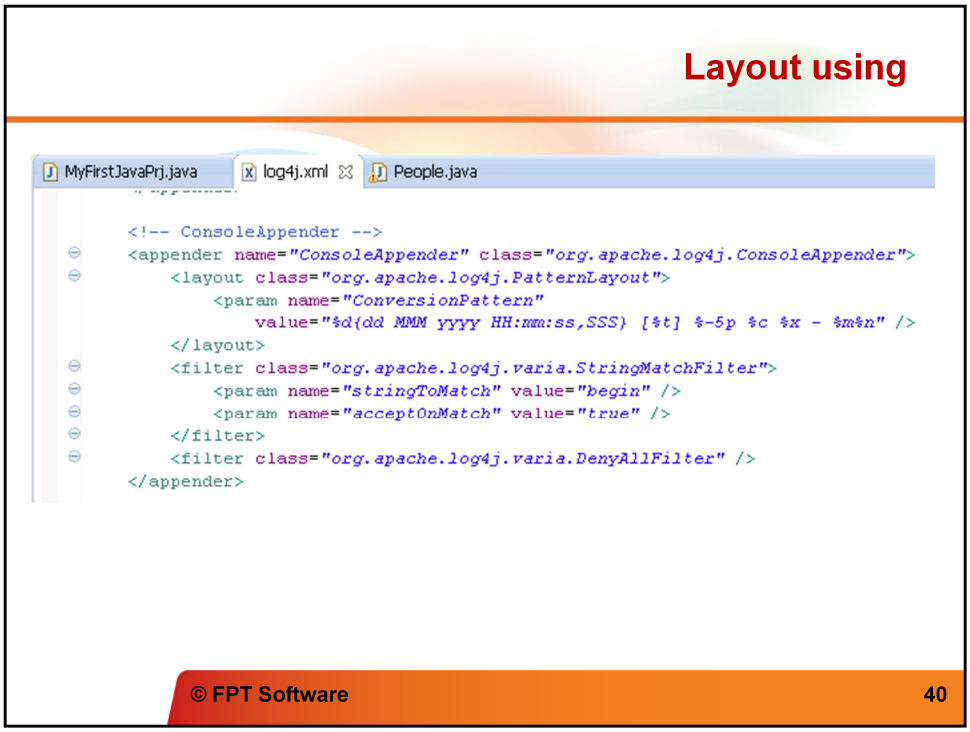
MyFirstJavaPrj.java | log4j.xml | People.java

```
    </appender>

    <!-- specific logger for class Person                -->
    <!-- in package com.fpt.fwa.java                     -->
    <!-- to set the logger for whole package, put only   -->
    <!-- package name                                    -->
    <logger name="com.fpt.fwa.java.People" additivity="false">
        <level value ="debug" />
        <appender-ref ref="ConsoleAppender"/>
        <appender-ref ref="FileAppender"/>
    </logger>

    <!-- Required Root logger -->
    <root>
        <level value ="info" />
        <appender-ref ref="FileAppender" />
    </root>

</log4j:configuration>
```

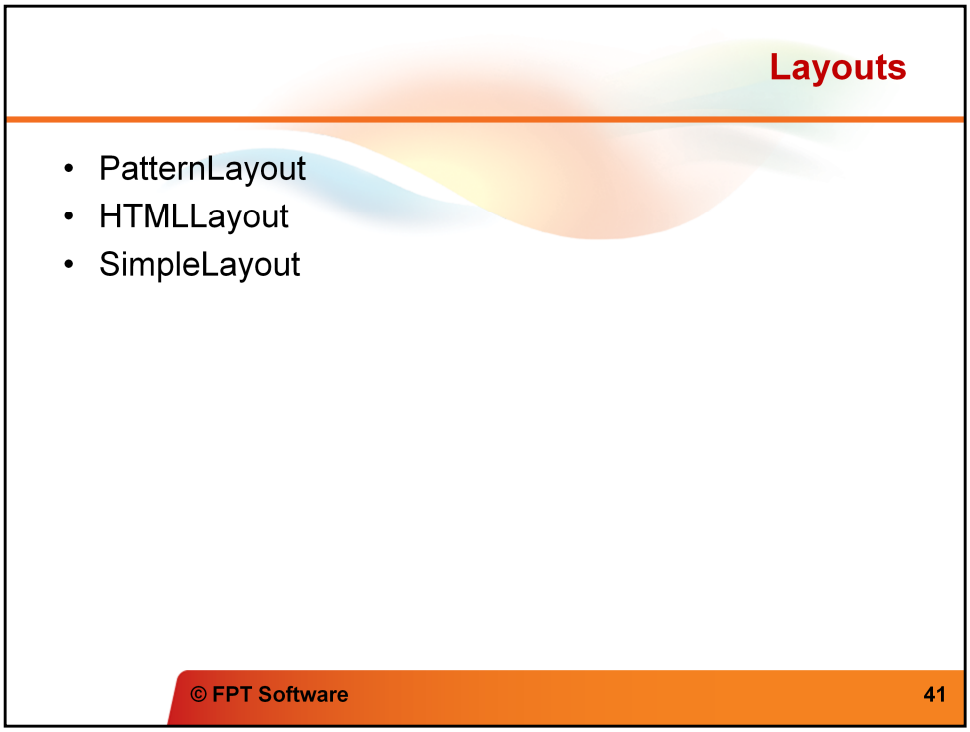

All == Debug

LAYOUT

Layout using



All == Debug



PatternLayout Formats the logging event according to a flexible set of formatting flags.

HTMLLayout output in HTML table

SimpleLayout Formats the logging event very simply: [level] - [message]

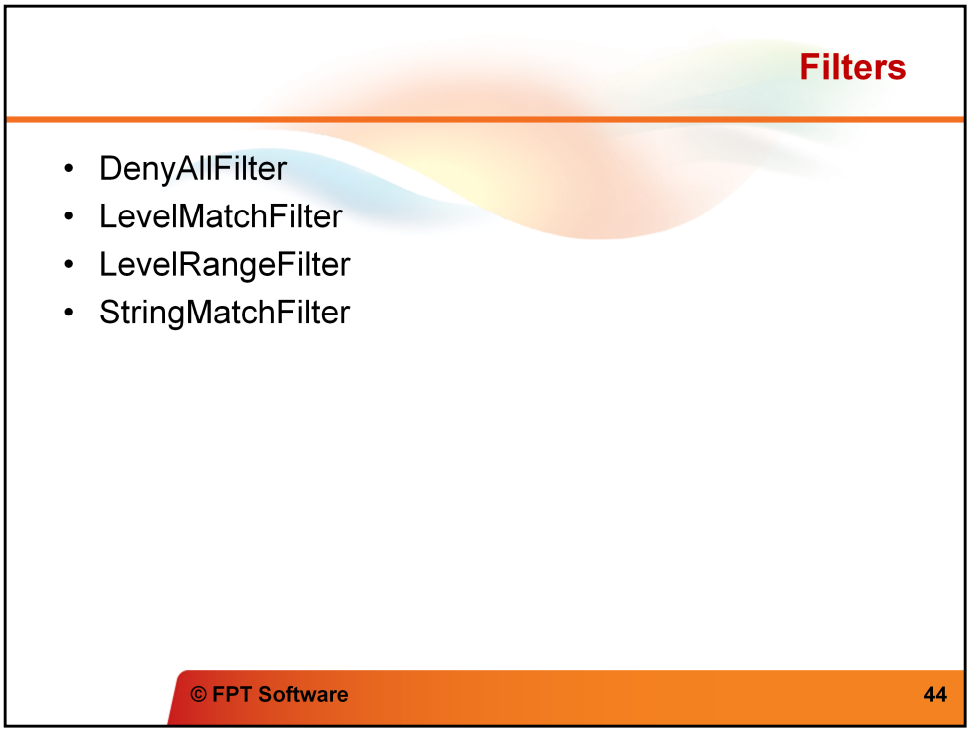
FILTER

Filter using

MyFirstJavaPrj.java | log4j.xml | People.java

```
<!-- ConsoleAppender -->
<appender name="ConsoleAppender" class="org.apache.log4j.ConsoleAppender">
    <layout class="org.apache.log4j.PatternLayout">
        <param name="ConversionPattern"
            value="%d{dd MMM yyyy HH:mm:ss,SSS} [%t] %-5p %c %x - %m%n" />
    </layout>
    <filter class="org.apache.log4j.varia.StringMatchFilter">
        <param name="stringToMatch" value="begin" />
        <param name="acceptOnMatch" value="true" />
    </filter>
    <filter class="org.apache.log4j.varia.DenyAllFilter" />
</appender>
```


All == Debug

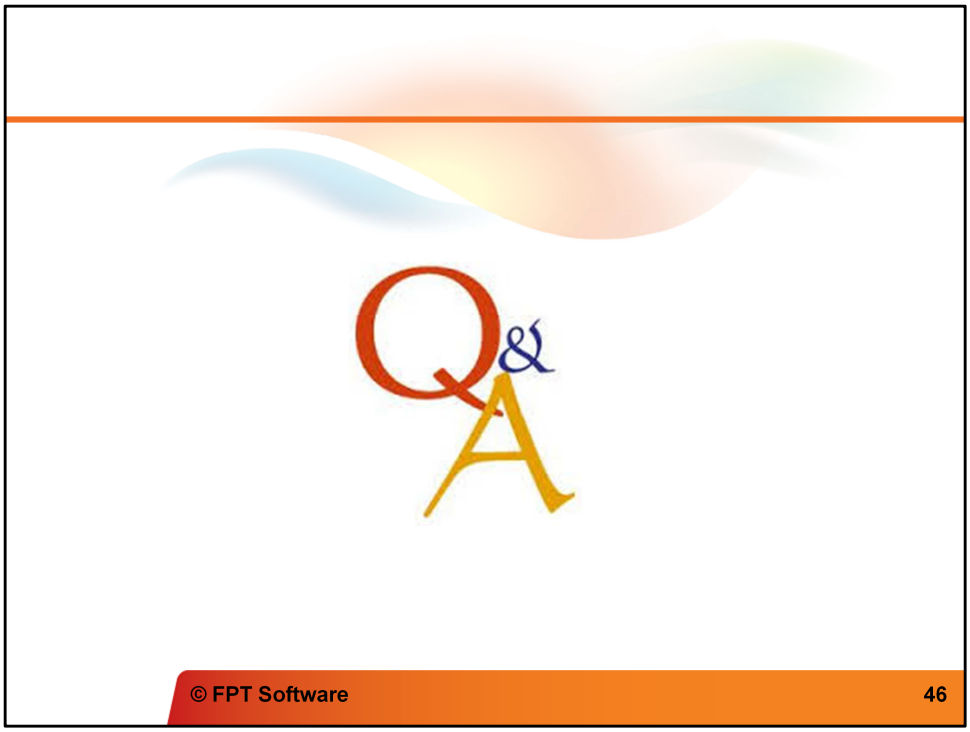


Attention to filter order

DenyAllFilter	Drops all “other” logging events.
LevelMatchFilter	An exact match to the event's level.
LevelRangeFilter	Matches against a range of levels.
StringMatchFilter	Matches a substring from the event's message.

Lesson summary

- Add Log4J library: add reference in build path
- Create configuration file: xml file
- Add logging initialization statements: in main java file
- Add logging code: in each module
- Appender overview: File rolling, console, SQL
- Log level overview: 6 options
- Layout overview: output format
- Filter overview: output restriction


Q&A
© FPT Software
46