################################################################################

以下内容摘自http://blog.sina.com.cn/s/blog\_5ed94d710101go3u.html  
#①配置根Logger，其语法为：   
#log4j.rootLogger = [level],appenderName,appenderName2,...   
#level是日志记录的优先级，分为OFF,TRACE,DEBUG,INFO,WARN,ERROR,FATAL,ALL   
##Log4j建议只使用四个级别，优先级从低到高分别是DEBUG,INFO,WARN,ERROR   
#通过在这里定义的级别，可以控制到应用程序中相应级别的日志信息的开关   
#比如在这里定义了INFO级别，则应用程序中所有DEBUG级别的日志信息将不被打印出来   
#appenderName就是指定日志信息输出到哪个地方。可同时指定多个输出目的   
################################################################################   
################################################################################   
#②配置日志信息输出目的地Appender，其语法为：   
#   
#log4j.appender.appenderName = fully.qualified.name.of.appender.class   
#log4j.appender.appenderName.optionN = valueN   
#   
#Log4j提供的appender有以下几种：   
#1)org.apache.log4j.ConsoleAppender(输出到控制台)   
#2)org.apache.log4j.FileAppender(输出到文件)   
#3)org.apache.log4j.DailyRollingFileAppender(每天产生一个日志文件)   
#4)org.apache.log4j.RollingFileAppender(文件大小到达指定尺寸的时候产生一个新的文件)   
#5)org.apache.log4j.WriterAppender(将日志信息以流格式发送到任意指定的地方)   
#   
#1)ConsoleAppender选项属性   
# -Threshold = DEBUG:指定日志消息的输出最低层次   
# -ImmediateFlush = TRUE:默认值是true,所有的消息都会被立即输出   
# -Target = System.err:默认值System.out,输出到控制台(err为红色,out为黑色)   
#   
#2)FileAppender选项属性   
# -Threshold = INFO:指定日志消息的输出最低层次   
# -ImmediateFlush = TRUE:默认值是true,所有的消息都会被立即输出   
# -File = C:\log4j.log:指定消息输出到C:\log4j.log文件   
# -Append = FALSE:默认值true,将消息追加到指定文件中，false指将消息覆盖指定的文件内容   
# -Encoding = UTF-8:可以指定文件编码格式   
#   
#3)DailyRollingFileAppender选项属性   
# -Threshold = WARN:指定日志消息的输出最低层次   
# -ImmediateFlush = TRUE:默认值是true,所有的消息都会被立即输出   
# -File = C:\log4j.log:指定消息输出到C:\log4j.log文件   
# -Append = FALSE:默认值true,将消息追加到指定文件中，false指将消息覆盖指定的文件内容   
# -DatePattern='.'yyyy-ww:每周滚动一次文件,即每周产生一个新的文件。还可以按用以下参数:   
#              '.'yyyy-MM:每月   
#              '.'yyyy-ww:每周   
#              '.'yyyy-MM-dd:每天   
#              '.'yyyy-MM-dd-a:每天两次   
#              '.'yyyy-MM-dd-HH:每小时   
#              '.'yyyy-MM-dd-HH-mm:每分钟   
# -Encoding = UTF-8:可以指定文件编码格式   
#   
#4)RollingFileAppender选项属性   
# -Threshold = ERROR:指定日志消息的输出最低层次   
# -ImmediateFlush = TRUE:默认值是true,所有的消息都会被立即输出   
# -File = C:/log4j.log:指定消息输出到C:/log4j.log文件   
# -Append = FALSE:默认值true,将消息追加到指定文件中，false指将消息覆盖指定的文件内容   
# -MaxFileSize = 100KB:后缀可以是KB,MB,GB.在日志文件到达该大小时,将会自动滚动.如:log4j.log.1   
# -MaxBackupIndex = 2:指定可以产生的滚动文件的最大数   
# -Encoding = UTF-8:可以指定文件编码格式   
################################################################################   
################################################################################   
#③配置日志信息的格式(布局)，其语法为：   
#   
#log4j.appender.appenderName.layout = fully.qualified.name.of.layout.class   
#log4j.appender.appenderName.layout.optionN = valueN   
#   
#Log4j提供的layout有以下几种：   
#5)org.apache.log4j.HTMLLayout(以HTML表格形式布局)   
#6)org.apache.log4j.PatternLayout(可以灵活地指定布局模式)   
#7)org.apache.log4j.SimpleLayout(包含日志信息的级别和信息字符串)   
#8)org.apache.log4j.TTCCLayout(包含日志产生的时间、线程、类别等等信息)   
#9)org.apache.log4j.xml.XMLLayout(以XML形式布局)   
#   
#5)HTMLLayout选项属性   
# -LocationInfo = TRUE:默认值false,输出java文件名称和行号   
# -Title=Struts Log Message:默认值 Log4J Log Messages   
#   
#6)PatternLayout选项属性   
# -ConversionPattern = %m%n:格式化指定的消息(参数意思下面有)   
#   
#9)XMLLayout选项属性   
# -LocationInfo = TRUE:默认值false,输出java文件名称和行号   
#   
#Log4J采用类似C语言中的printf函数的打印格式格式化日志信息，打印参数如下：   
# %m 输出代码中指定的消息   
# %p 输出优先级，即DEBUG,INFO,WARN,ERROR,FATAL   
# %r 输出自应用启动到输出该log信息耗费的毫秒数   
# %c 输出所属的类目,通常就是所在类的全名   
# %t 输出产生该日志事件的线程名   
# %n 输出一个回车换行符，Windows平台为“\r\n”，Unix平台为“\n”   
# %d 输出日志时间点的日期或时间，默认格式为ISO8601，也可以在其后指定格式   
#    如：%d{yyyy年MM月dd日 HH:mm:ss,SSS}，输出类似：2012年01月05日 22:10:28,921   
# %l 输出日志事件的发生位置，包括类目名、发生的线程，以及在代码中的行数   
#    如：Testlog.main(TestLog.java:10)   
# %F 输出日志消息产生时所在的文件名称   
# %L 输出代码中的行号   
# %x 输出和当前线程相关联的NDC(嵌套诊断环境),像java servlets多客户多线程的应用中   
# %% 输出一个"%"字符   
#   
# 可以在%与模式字符之间加上修饰符来控制其最小宽度、最大宽度、和文本的对齐方式。如：   
#  %5c: 输出category名称，最小宽度是5，category<5，默认的情况下右对齐   
#  %-5c:输出category名称，最小宽度是5，category<5，"-"号指定左对齐,会有空格   
#  %.5c:输出category名称，最大宽度是5，category>5，就会将左边多出的字符截掉，<5不会有空格   
#  %20.30c:category名称<20补空格，并且右对齐，>30字符，就从左边交远销出的字符截掉   
################################################################################   
################################################################################   
#④指定特定包的输出特定的级别   
#log4j.logger.org.springframework=DEBUG   
################################################################################   
  
#OFF,systemOut,logFile,logDailyFile,logRollingFile,logMail,logDB,ALL   
log4j.rootLogger =ALL,systemOut,logFile,logDailyFile,logRollingFile,logMail,logDB   
  
#输出到控制台   
log4j.appender.systemOut = org.apache.log4j.ConsoleAppender   
log4j.appender.systemOut.layout = org.apache.log4j.PatternLayout   
log4j.appender.systemOut.layout.ConversionPattern = [%-5p][%-22d{yyyy/MM/dd HH:mm:ssS}][%l]%n%m%n   
log4j.appender.systemOut.Threshold = DEBUG   
log4j.appender.systemOut.ImmediateFlush = TRUE   
log4j.appender.systemOut.Target = System.out   
  
#输出到文件   
log4j.appender.logFile = org.apache.log4j.FileAppender   
log4j.appender.logFile.layout = org.apache.log4j.PatternLayout   
log4j.appender.logFile.layout.ConversionPattern = [%-5p][%-22d{yyyy/MM/dd HH:mm:ssS}][%l]%n%m%n   
log4j.appender.logFile.Threshold = DEBUG   
log4j.appender.logFile.ImmediateFlush = TRUE   
log4j.appender.logFile.Append = TRUE   
log4j.appender.logFile.File = d:/log.log    
log4j.appender.logFile.Encoding = UTF-8   
  
#按DatePattern输出到文件   
log4j.appender.logDailyFile = org.apache.log4j.DailyRollingFileAppender   
log4j.appender.logDailyFile.layout = org.apache.log4j.PatternLayout   
log4j.appender.logDailyFile.layout.ConversionPattern = [%-5p][%-22d{yyyy/MM/dd HH:mm:ssS}][%l]%n%m%n   
log4j.appender.logDailyFile.Threshold = DEBUG   
log4j.appender.logDailyFile.ImmediateFlush = TRUE   
log4j.appender.logDailyFile.Append = TRUE   
log4j.appender.logDailyFile.File = ../Struts2/WebRoot/log/DailyFile/log4j\_Struts   
log4j.appender.logDailyFile.DatePattern = '.'yyyy-MM-dd-HH-mm'.log'   
log4j.appender.logDailyFile.Encoding = UTF-8   
  
#设定文件大小输出到文件   
log4j.appender.logRollingFile = org.apache.log4j.RollingFileAppender   
log4j.appender.logRollingFile.layout = org.apache.log4j.PatternLayout   
log4j.appender.logRollingFile.layout.ConversionPattern = [%-5p][%-22d{yyyy/MM/dd HH:mm:ssS}][%l]%n%m%n   
log4j.appender.logRollingFile.Threshold = DEBUG   
log4j.appender.logRollingFile.ImmediateFlush = TRUE   
log4j.appender.logRollingFile.Append = TRUE   
log4j.appender.logRollingFile.File = ../Struts2/WebRoot/log/RollingFile/log4j\_Struts.log   
log4j.appender.logRollingFile.MaxFileSize = 1MB   
log4j.appender.logRollingFile.MaxBackupIndex = 10   
log4j.appender.logRollingFile.Encoding = UTF-8   
  
#用Email发送日志   
log4j.appender.logMail = org.apache.log4j.net.SMTPAppender   
log4j.appender.logMail.layout = org.apache.log4j.HTMLLayout   
log4j.appender.logMail.layout.LocationInfo = TRUE   
log4j.appender.logMail.layout.Title = Struts2 Mail LogFile   
log4j.appender.logMail.Threshold = DEBUG   
log4j.appender.logMail.SMTPDebug = FALSE   
log4j.appender.logMail.SMTPHost = SMTP.163.com   
log4j.appender.logMail.From = xly3000@163.com   
log4j.appender.logMail.To = xly3000@gmail.com   
#log4j.appender.logMail.Cc = xly3000@gmail.com   
#log4j.appender.logMail.Bcc = xly3000@gmail.com   
log4j.appender.logMail.SMTPUsername = xly3000   
log4j.appender.logMail.SMTPPassword = 1234567   
log4j.appender.logMail.Subject = Log4j Log Messages   
#log4j.appender.logMail.BufferSize = 1024   
#log4j.appender.logMail.SMTPAuth = TRUE   
  
#将日志登录到MySQL数据库   
log4j.appender.logDB = org.apache.log4j.jdbc.JDBCAppender   
log4j.appender.logDB.layout = org.apache.log4j.PatternLayout   
log4j.appender.logDB.Driver = com.mysql.jdbc.Driver   
log4j.appender.logDB.URL = jdbc:mysql://127.0.0.1:3306/xly   
log4j.appender.logDB.User = root   
log4j.appender.logDB.Password = 123456   
log4j.appender.logDB.Sql = INSERT INTOT\_log4j(project\_name,create\_date,level,category,file\_name,thread\_name,line,all\_category,message)values('Struts2','%d{yyyy-MM-ddHH:mm:ss}','%p','%c','%F','%t','%L','%l','%m')