TABLA DE CONTENIDO

[TABLA DE CONTENIDO 1](#_Toc503819551)

[Gradle 2](#_Toc503819552)

[*Archivo de configuración: build.gradle* 2](#_Toc503819553)

[*Clase principal EjemploGradle.java* 3](#_Toc503819554)

[*Archivo de configuración log (log4j2.xml)* 3](#_Toc503819555)

[Notas: códigos fuentes 4](#_Toc503819556)

[src/main/java/b*uild.gradle* 4](#_Toc503819557)

[src/main/java/EjemploGradle.java 4](#_Toc503819558)

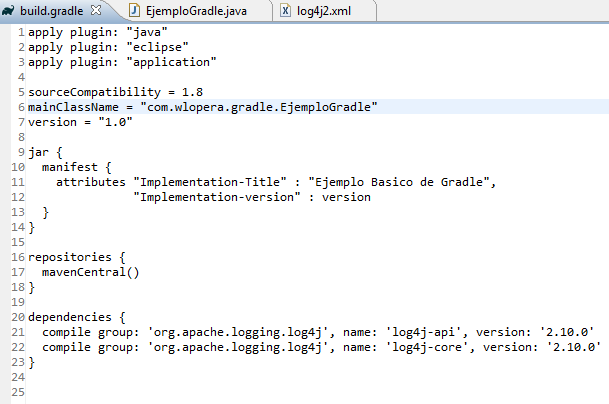
[src/main/resources/log4j2.xml 4](#_Toc503819559)

[</Configuration> 4](#_Toc503819560)

Gradle

* Herramienta para gestionar la automatización de los builds.
* Es una herramienta de automatización de la construcción de nuestro código.

### *Archivo de configuración: build.gradle*



Como podemos ver se trata de un DSL (***Domain Specified Language***) que es bastante compacto. En las primeras líneas registramos tres plugins que nos permitirán abordar diversas tareas.

**Java Plugin**: Se encarga de poner a nuestra disposición tareas del mundo java como por ejemplo la tarea “jar” de empaquetado que usamos en la línea 9.

**Eclipse Plugin :**Se encarga de integrar gradle dentro de eclipse y configurar un classpath basado en las dependencias que asignamos en la línea 20

**Application Plugin:** Se encarga de añadir las tareas de ejecución de tal forma que desde el propio gradle podamos ejecutar nuestro proyecto definiendo en la línea 6 la clase main.

Una vez tenemos definido el fichero podemos ver el contenido de la clase Principal que deseamos ejecutar.

### *Clase principal EjemploGradle.java*

### 

En este caso se trata de una clase que usa el **API de log4j (versión2) para imprimir un mensaje de error**. Para ello deberemos definir **en la carpeta de resources el fichero log4j2.xml que nos configura el appender de consola.**

### *Archivo de configuración log (log4j2.xml)*

### 

### 

Notas: códigos fuentes

### src/main/java/b*uild.gradle*

apply plugin: "java"

apply plugin: "eclipse"

apply plugin: "application"

sourceCompatibility = 1.8

mainClassName = "com.wlopera.gradle.EjemploGradle"

version = "1.0"

jar {

manifest {

attributes "Implementation-Title" : "Ejemplo Basico de Gradle",

"Implementation-version" : version

}

}

repositories {

mavenCentral()

}

dependencies {

compile group: 'org.apache.logging.log4j', name: 'log4j-api', version: '2.10.0'

compile group: 'org.apache.logging.log4j', name: 'log4j-core', version: '2.10.0'

}

### src/main/java/EjemploGradle.java

**package** com.wlopera.gradle;

**import** org.apache.logging.log4j.LogManager;

**import** org.apache.logging.log4j.Logger;

**public** **class** EjemploGradle {

**private** **static** **final** Logger ***logger*** = LogManager.*getLogger*(EjemploGradle.**class**);

**public** **static** **void** main(String[] args) {

***logger***.info("##=> Ejemplo basico de Gradle ...!");

}

}

### src/main/resources/log4j2.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<Configuration status=*"INFO"*>

<Appenders>

<Console name=*"Console"* target=*"SYSTEM\_OUT"*>

<PatternLayout pattern=*"%d{HH:mm:ss.SSS} [%t] %-5level %logger{36} - %msg%n"* />

</Console>

</Appenders>

<Loggers>

<Root level=*"info"*>

<AppenderRef ref=*"Console"* />

</Root>

</Loggers>

**</Configuration>**