Lombok项目是一个Java库，它能够自动嵌入到你的IDE编辑器和编译工具中。有了它，你再也不用写getter或者equals方法了，还有val特性等。

* **安装插件和配置依赖**

Lombok支持所有主流的IDE开发工具及编译构建工具。在IDEA中安装Lombok插件，这样在使用Lombok的时候就不会编译报错。

下面是Maven的依赖配置，在用Maven打包的时候也能自动生成需要的代码。

<!-- https://mvnrepository.com/artifact/org.projectlombok/lombok -->

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<version>1.16.20</version>

<scope>provided</scope>

</dependency>

如果使用了Spring Boot，可以不用带版本号，在Spring Bootspring - boot - dependencies.pom 这个配置文件里面定义了 Lombok 依赖。

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<scope>provided</scope>

</dependency>

* **使用注解简化代码**

从IDEA中的Lombok插件特性可以看到支持的注解。

**Features**

**@Getter and @Setter**

**@FieldNameConstants**

**@ToString**

**@EqualsAndHashCode**

**@AllArgsConstructor, @RequiredArgsConstructor and @NoArgsConstructor**

**@Log, @Log4j, @Log4j2, @Slf4j, @XSlf4j, @CommonsLog, @JBossLog, @Flogger**

**@Data**

**@Builder**

**@Singular**

**@Delegate**

**@Value**

**@Accessors**

**@Wither**

**@SneakyThrows**

**from Intellij 14.1 @val**

**from Intellij 15.0.2 @var**

**from Intellij 14.1 @var**

**from Intellij 2016.2 @UtilityClass**

**Lombok config system**

**Code inspections**

**Refactoring actions (lombok and delombok)**

* **@Getter和@Setter**

添加@Getter和@Setter注解用在Java Bean类上面，无需生成get/set方法，会**自动生成所有的get/set方法及一个默认的构造方法**。

@Getter和Setter注解也可以单独使用在字段上面，，同样也会生成对应的get/set方法及一个默认的构造方法。

* **@ToString**

使用在类上，**默认生成所有非静态字段**，以下面的格式输出，如：

@Override  
**public** String toString() {  
 **return "Student{"** +  
 **"id="** + **id** +  
 **", stuNo='"** + **stuNo** + **'\''** +  
 **", name='"** + **name** + **'\''** +  
 **'}'**;  
}

* **@NoArgsConstructor**

用在类上，用来生成一个默认的无参构造方法。

* **@RequiredArgsConstructor**

用在类上，**使用类中所有带有@NonNull注解和final类型的字段生成对应的构造方法**。

* **@AllArgsConstructor**

用在类上，生成一个所有参数的构造方法，默认不提供无参构造方法。

* **@Data**

用在类上，等同于下面这几个注解的合集。

@Getter

@Setter

@RequiredArgsConstructor

@ToString

@EqualsAndHashCode

* **@Value**

用在类上，等同于下面这几个注解合集。

@Getter

@FieldDefaults(makeFinal=true, level=AccessLevel.PRIVATE)

@AllArgsConstructor

@ToString @EqualsAndHashCode}

* **@NonNull**

用在属性上，用于字段的非空检查，如果传入到set方法中的值为空，则抛出空指针异常，该注解也会生成一个默认的构造方法。