



链滴

java 基础 | java 中注释的写法

作者: [xiaodaojava](#)

原文链接: <https://ld246.com/article/1576074535497>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



javaDEMO

本网站记录了最全的各种JavaDEMO,保证下载,复制就是可用的,包括基础的,集合的, spring的, Mybatis的等等各种,助力你从菜鸟到大牛,记得收藏哦~~

<https://www.javastudy.cloud>

java注释总述

注释不会被编译进class,只存在于源码中,供开发人员一代一代的传承下去,给后人以警醒. 通常是一段代码如果不写注释说明备注下,过个几个星期,自己再看的时候就是,我去,,这是什么鬼?这是谁写的?? 我当为什么要这么写???

注释相关的文档地址如下:

<https://www.oracle.com/technetwork/java/javase/documentation/index-137868.html>

总体来说,有以下位置需要注释:

- 1.类注释
- 2.类属性注释
- 2.方法注释
- 3.方法参数,返回值注释
- 4.代码内部逻辑注释
- 5.包的注释
- 6.注释中常用的注解

各种注释示例

常用注解

@author (classes and interfaces only, required)
@version
@param (methods and constructors only)
@return (methods only)
@exception (@throws is a synonym added in Javadoc 1.2)
@see
@since
@serial (or @serialField or @serialData)
@deprecated (see [How and When To Deprecate APIs])
@date(上面都是摘自于jdk文档. 这个是我自己加的)

类注释

如下所示,通常来说,类注释会加上@author和@date这两个注解,标注是谁在什么时间写的

```
/**
 * 这个类是用来演示注释怎么写的
 * @author lixiang
 * @date 2019/12/11
 */
public class DocDemo {
```

类属性注释

在全局变量上,推荐使用/***/的方法来写注释

```
/** 这是一个全局的变量 */
private String globalArg;
```

方法注释

方法上的注解,一般要写清入参,返回值,如果不再使用,要写替换的方法

```
/**
 * 这是第一个方法
 * @param arg1 参数1
 * @param arg2 参数2
 * @return 返回java
 */
public String method1(String arg1,String arg2){
    return "java";
}

/**
 * 这是第二个方法
 * @deprecated 不可再使用
 * @see #method1(String, String) 用这个方法
 * @param arg1 第一个参数
```

```
* @return 返回demo
*/
@Deprecated
public String method2(String arg1){
    return "demo";
}
```

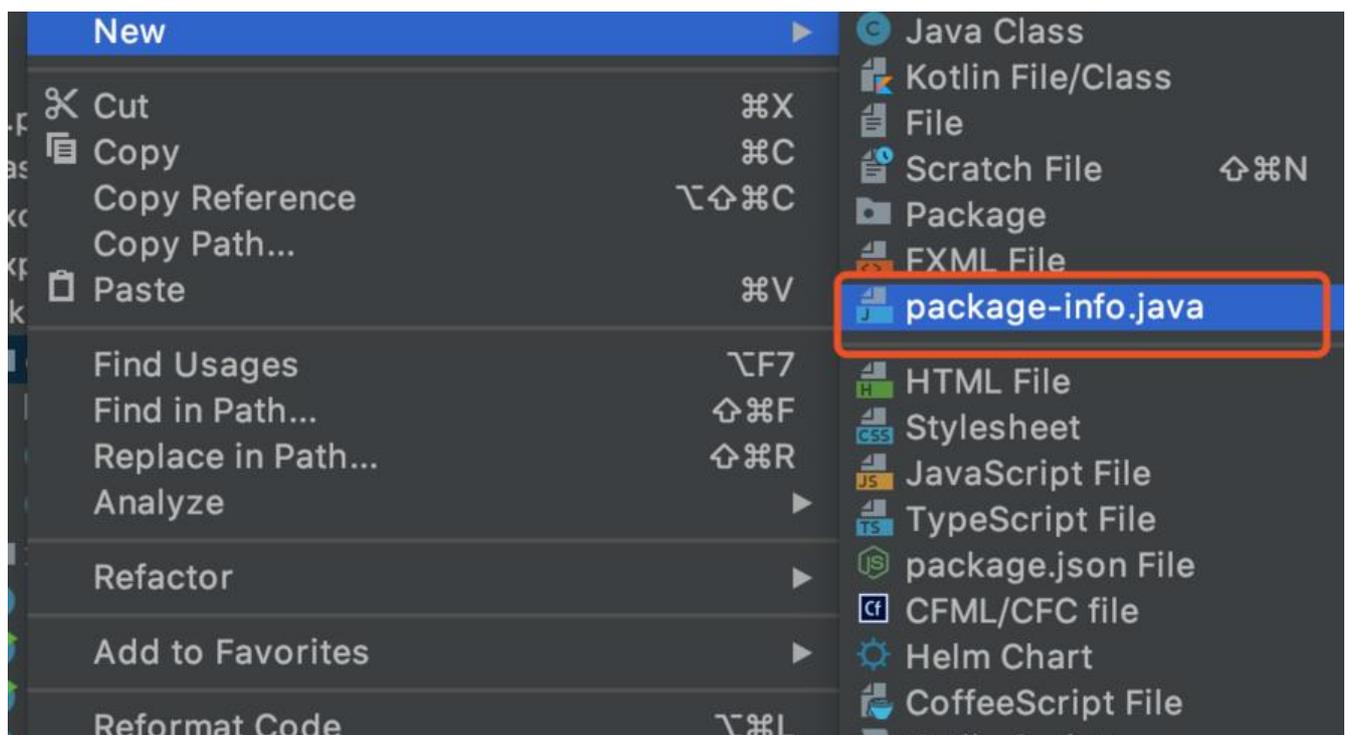
代码内部注释

要注意，尽量不写行尾注释,如有if,for等,尽量把注释写在{}内部

```
//输出Hello
System.out.println("hello");
if("t".equals(arg1)){
    //如果arg1是t的话,则输出t
    System.out.println("t");
}
```

包注释

这个有的项目有,有的没有,如要新建的话,可以对着包点右键,New->package-info.java



其注释的写法如下:

```
package-info.java x
1  /**
2   * 这是一个包的注释,这个包是做什么用的, 都写在这里
3   * @author lixiang
4   * @date 2019/12/11
5   */
6  package com.platf|form.tools.jdk.demo;
```

DEMO总评

注释是编程很重要的一个环节,如果一份代码不写注释,三天回过头就全不认识了.注释的写法没有准确标准,更重要的是团队统一,方便.在有统一的写法后,后面根据注释去自动生成文档就简单多了.