**JSDoc注释使用**

JSDoc通过JS文件中的一些特殊JSDoc标记，解析文档。下面列出了可以创建HTML文档的一些特殊JSDoc标记。如果你曾在Java代码中编写过javadoc注释，应该对这些标记并不陌生。如果要在最后生成的文档中包含某个注释块，所有这些注释块都必须以/\*\*开头，并以\*/结束。

**JSDoc 命令属性**  
  
命令名                描述   
@param  
@argument           指定参数名和说明来描述一个函数参数。   
@return  
@returns              描述函数的返回值。   
@author               指示代码的作者。   
@deprecated     指示一个函数已经废弃，不建议使用，而且在将来版本的代码中可能会彻底删除。要避免使用这段代码。   
@see                    创建一个HTML链接指向指定类的描述。   
@version              指定发布版本。   
@requires             创建一个HTML链接，指向这个类所需的指定类。   
@throws  
@exception           描述函数可能抛出的异常的类型。   
{@link}                创建一个HTML链接，指向指定的类。这与@see很类似，但{@link}能嵌在注释文本中。   
@author                指示代码的作者。（译者注：这个标记前面已经出现过，建议去掉）   
@fileoverview     这是一个特殊的标记，如果在文件的第一个文档块中使用这个标记，则指定该文档块的余下部分将用来提供文件的一个概述。   
@class                  提供类的有关信息，用在构造函数的文档中。   
@constructor        明确一个函数是某个类的构造函数。   
@type                  指定函数的返回类型。   
@extends             指示一个类派生了另一个类。通常JSDoc自己就可以检测出这种信息，不过，在某些情况下则必须使用这个标记。   
@private              指示一个类或函数是私有的。私有类和函数不会出现在HTML文档中，除非运行JSDoc时提供了—private命令行选项。   
@final                  指示一个值是常量值。要记住JavaScript无法真正保证一个值是常量。   
@ignore JSDoc    会忽略有这个标记的函数。   
  
  
JSDoc 发布包中包括一个名为test.js的文件，这是一个很好的参考例子，可以从中了解如何使用JSDoc。应该记得，第一次测试JSDoc安装是否成功时就是根据这个文件来创建文档文件（译者注：原文称“这就是第一次测试JSDoc安装是否成功时创建的文档文件”，这种理解有误，test.js显然不是文档文件，所得到的文档文件应当是js\_docs\_out的目录下的index.html）。如果对如何使用JSDoc标记还有疑问，可以参考这个文件。  
以下是一个小例子，这里展示了JSDoc的用法。jsDocExample.js定义了两个类：Person和Employee。Person类有一个属性name，还有一个方法getName。Employee类继承自Person，并增加了title和salary属性，另外还增加了一个方法 getDescription。