**一、语法形式**

  Java在1.4中新增了一个关键字：assert。在程序开发过程中使用它创建一个断言(assertion)，它的语法形式有如下所示的两种形式：

1·assert condition;

     这里condition是一个布尔表达式。如果表达式的结果为true，那么断言为真，则无任何行动。如果表达式结果为false，则断言失败，则会抛出一个AssertionError。这个AssertionError继承于Error对象，而Error继承于Throwable，Error是和Exception并列的一个错误对象，通常用于表达系统级运行错误。

2·asser condition:expr;

     这里condition是和上面一样的，冒号后的expr是一个表达式，通常用于断言失败后的提示信息，说白了，它是一个传到AssertionError构造函数的值，如果断言失败，该值被转化为它对应的字符串，并显示出来。

**二、使用示例**

下面是一个使用assert的例子：

|  |
| --- |
| **public** **class** TestAssert {  **public** **static** **void** main(String[] args) {  String name = "abc";  // name = null;  **assert** (name != **null**) : "变量name为空null";  System.*out*.println(name);  }  } |

**三、让断言生效**

要想让断言起效用，即让断言语句在运行时确实检查，在运行含有assert的程序时，必须指定-ea的jvm参数，如：java -ea TestAssert。

**四、注意事项**

理解断言最重要的一点是必须不依赖它们完成任何程序实际所需的行为。理由是正常发布的代码都会设置让断言不生效，即正常发布的代码中断言语句都不会执行（或不起作用），如果不小心错误的使用断言，如：

|  |
| --- |
| **public** **class** TestPerson {  **private** String name = **null**;  **public** **void** setName(String nameStr) {  **this**.name = nameStr;  }  **public** String getName() {  **return** **this**.name;  }  **public** TestPerson(String name) {  **this**.name = name;  }  **public** **static** **void** main(String[] args) {  TestPerson personObj = **new** TestPerson("abc");  String personName = **null**;  // 这里对personName的赋值被安排在assert语句中，尽管断言生效时程序可以很好地运行  // 但如果断言失效，则程序会在运行时在personName.length()处报空指针错误。  // 因为断言无效时，personName=personObj.getName()一句永远不会执行！  **assert** ((personName = personObj.getName()) != **null**);  System.*out*.println(personName.length());  }  } |

gai正确的使用assert：

|  |
| --- |
| //String personName = null;  //assert ((personName = personObj.getName()) != null);  String personName = personObj.getName();  **assert** personName != **null**; |

**五、其它选项**

当执行代码时，可以使用-ea使断言有效，也可以使用-da使断言无效（默认为无效）。另外，也可以通过在-ea或-da后面指定包名来使**一个包的断言有效或无效**，例如：

|  |  |
| --- | --- |
| -da:com.test | 使com.test包中的断言无效 |
| -ea:com.test... | 使一个包中的所有子包中的断言能够有效或无效，在包名后加上三个点 |

**ref:**

理解和正确使用Java中的断言(assert): https://blog.csdn.net/AbnerChai/article/details/87379