# Java一些细节点

## 一、Java基础：

### 1、数字0的计算：

* 正整数除以0结果为无穷大；
* 0/0结果为NaN；
* 负数的平方根结果为NaN；
* 所有“非数值”的值都认为是不相同的，即NaN != NaN，可以使用Double.isNaN方法来检测NaN值；

### 取模运算：

对于取模运算，数学家们几百年来都知道这样一个最优规则：余数总是要>=0；

但是在Java中，余数却可以为负数：

在数学中“-10 % 3”一定是正2，但是在Java中使用不同的方法，却得出不同的结果：

1）普遍的计算：

|  |
| --- |
| int a = -10 % 3; // -1 |

这种方法，得出的结果是 -1，是**错误的结果**。

1. BigInteger计算：

|  |
| --- |
| BigInteger.valueOf(-10L).mod(BigInteger.valueOf(3L)); // 2 |

这种方法，得出的结果是正2，是正确的结果。

之所以会出现这种情况，只能是因为Java语言设计之初的缺陷。