

`java.math` 包中有两个很有用的类：`BigInteger` 和 `BigDecimal`。`BigInteger` 类实现任意精度的整数运算，`BigDecimal` 类实现任意精度的浮点数运算。

下面列出这两个类的常用 API。

`BigInteger` 类：

```
/* java.math.BigInteger */

// 返回这个大整数与 other 的和
BigInteger add(BigInteger other)

// 返回这个大整数与 other 的差
BigInteger subtract(BigInteger other)

// 返回这个大整数与 other 的积
BigInteger multiply(BigInteger other)

// 返回这个大整数与 other 的商
BigInteger divide(BigInteger other)

// 返回这个大整数与 other 的余数
BigInteger mod(BigInteger other)

// 得到这个大整数的平方根
BigInteger sqrt()

// 如果这个大整数与 other 相等，返回 0；如果小于 other，返回负数；如果大于 other，返回正数
int compareTo(BigInteger other)

// 将普通正数转换为大整数
static BigInteger valueOf(long x)

// 将一个字符串转换为大整数
BigInteger(String val)

// 常量 0
BigInteger.ZERO

// 常量 1
BigInteger.ONE

// 常量 10
BigInteger.TEN

// 常量 2, Java 9 引入
BigInteger.TWO
```

`BigDecimal` 类：

```
/* java.math.BigDecimal */

// 返回这个大实数与 other 的和
BigDecimal add(BigDecimal other)

// 返回这个大实数与 other 的差
BigDecimal subtract(BigDecimal other)

// 返回这个大实数与 other 的积
BigDecimal multiply(BigDecimal other)

// 返回这个大实数与 other 的商。如果商是无限循环小数，会抛出异常
BigDecimal divide(BigDecimal other)

// 得到一个舍入结果，第二个参数使用 RoundingMode.HALF_UP 采用四舍五入方式
BigDecimal divide(BigDecimal other, RoundingMode mode)

// 如果这个大实数与 other 相等，返回 0；如果小于 other，返回负数；如果大于 other，返回正数
int compareTo(BigDecimal other)

// 将 long 类型整数转换为大实数
static BigDecimal valueOf(long x)

// 返回值等于  $x/(10^{\text{scale}})$  的大实数
static BigDecimal valueOf(long x, int scale)
```

Java 不支持运算符重载，因此不能用算术运算符处理大数，只能使用这两个类的 API 来进行运算。