Decimal

# 1、Decimal(m,n)

## ****解释：decimal(m,n)  m总共几位，n小数点后面保留几位 ，先保证小数点，再保证总位数****

**select CONVERT(decimal(5,3),1.2226)  ---1.223**  
**select CONVERT(decimal(4,2),1.2226)  ---1.22**  
**select CONVERT(decimal(8,4),1.2226)  ---1.2226**

# ****2、加减乘除****

## 进行相应的计算后，我们可能需要将BigDecimal对象转换成相应的基本数据类型的变量，可以使用floatValue()，doubleValue()等方法。

1. **public** **class** Arith {
2. /\*\*
3. \* 提供精确加法计算的add方法
4. \* @param value1 被加数
5. \* @param value2 加数
6. \* @return 两个参数的和
7. \*/
8. **public** **static** **double** add(**double** value1,**double** value2){
9. BigDecimal b1 = **new** BigDecimal(Double.valueOf(value1));
10. BigDecimal b2 = **new** BigDecimal(Double.valueOf(value2));
11. **return** b1.add(b2).doubleValue();
12. }
14. /\*\*
15. \* 提供精确减法运算的sub方法
16. \* @param value1 被减数
17. \* @param value2 减数
18. \* @return 两个参数的差
19. \*/
20. **public** **static** **double** sub(**double** value1,**double** value2){
21. BigDecimal b1 = **new** BigDecimal(Double.valueOf(value1));
22. BigDecimal b2 = **new** BigDecimal(Double.valueOf(value2));
23. **return** b1.subtract(b2).doubleValue();
24. }
26. /\*\*
27. \* 提供精确乘法运算的mul方法
28. \* @param value1 被乘数
29. \* @param value2 乘数
30. \* @return 两个参数的积
31. \*/
32. **public** **static** **double** mul(**double** value1,**double** value2){
33. BigDecimal b1 = **new** BigDecimal(Double.valueOf(value1));
34. BigDecimal b2 = **new** BigDecimal(Double.valueOf(value2));
35. **return** b1.multiply(b2).doubleValue();
36. }
38. /\*\*
39. \* 提供精确的除法运算方法div
40. \* @param value1 被除数
41. \* @param value2 除数
42. \* @param scale 精确范围
43. \* @return 两个参数的商
44. \* @throws IllegalAccessException
45. \*/
46. **public** **static** **double** div(**double** value1,**double** value2,**int** scale) **throws** IllegalAccessException{
47. //如果精确范围小于0，抛出异常信息
48. **if**(scale<0){
49. **throw** **new** IllegalAccessException("精确度不能小于0");
50. }
51. BigDecimal b1 = **new** BigDecimal(Double.valueOf(value1));
52. BigDecimal b2 = **new** BigDecimal(Double.valueOf(value2));
53. **return** b1.divide(b2, scale).doubleValue();
54. }
55. }