

## [“Non-terminating decimal expansion” Kenapa bisa terjadi?](#)

Untuk pembaca yang sudah terbiasa dengan Java mungkin pernah mendengar atau mengalami kejadian ini. Dengan sedikit bereksperimen saya mencoba melakukan perhitungan angka menggunakan variable bertipe BigDecimal, jika kita akan melakukan proses penambahan, pengurangan, perkalian dan pembagian dengan menggunakan BigDecimal kita dapat melakukannya dengan perintah “add()”, “subtract()”, “multiply()” dan “divide()”, semua proses tersebut dapat berjalan dengan lancar seperti yang diharapkan, namun akan menghasilkan error jika kita melakukan operasi pembagian menggunakan divide yang menghasilkan nilai pecahan atau ada nilai dibelakang koma, dan tampilan error yang dihasilkan jika kita gunakan perintah “printStackTrace” akan seperti berikut.

```
java.lang.ArithmeticException: Non-terminating decimal expansion; no exact representable decimal result.
    at java.math.BigDecimal.divide(BigDecimal.java:1514)
...
...
...
...
...
    at java.lang.Thread.run(Thread.java:619)
```

Perlu diketahui jika kita menggunakan framework apapun itu, maka isi dari titik – titik tersebut merupakan error message dari framework tersebut. Hal ini bisa terjadi jika dalam melakukan operasi pembagian, kita dapat mengambil contoh perhitungan dengan menggunakan source seperti berikut:

```
public class SuatuClass {
    static BigDecimal bilAwal = BigDecimal.valueOf(10);
    static BigDecimal bilAkhir = BigDecimal.valueOf(15);
    static BigDecimal hasilBagi = BigDecimal.valueOf(0);

    public static void main(String[] args) {
        divideOperation();
    }

    public static void divideOperation() {
        hasilBagi = bilAwal.divide(bilAkhir);

        System.out.println("Print Hasil Akhir "+ hasilBagi);
    }
}
```

Sekarang akan kita pusatkan perhatian kita kepada baris perintah berikut “bilAwal.divide(bilAkhir);” jika hasil dari perhitungan merupakan pecahan maka akan menghasilkan error seperti diatas, bagaimana cara mengatasinya.? Gampang.. Ketik reg spasi manjur kirim ke..... Hehe.. Becanda, Jika hasil yang ingin ditampilkan harus bilangan bulat maka baris perintah tersebut berubah menjadi “bilAwal.divide(bilAkhir, RoundingMode.HALF\_UP);” property HALF\_UP akan menghasilkan nilai yang dibulatkan keatas

jika nilai dibelakang koma  $\geq 0.5$ , adapun untuk lebih jelas berikut merupakan sekilas tentang beberapa properti dan kegunaannya dari class RoundingMode:

Input Number	Nilai yang dihasilkan dari input dengan beberapa properti RoundingMode							UNNECESSARY
	UP	DOWN	CEILING	FLOOR	HALF_UP	HALF_DOWN	HALF_EVEN	
5.5	6	5	6	5	6	5	6	throw ArithmeticException
2.5	3	2	3	2	3	2	2	throw ArithmeticException
1.6	2	1	2	1	2	2	2	throw ArithmeticException
1.1	2	1	2	1	1	1	1	throw ArithmeticException
1.0	1	1	1	1	1	1	1	1
-1.0	-1	-1	-1	-1	-1	-1	-1	-1
-1.1	-2	-1	-1	-2	-1	-1	-1	throw ArithmeticException
-1.6	-2	-1	-1	-2	-2	-2	-2	throw ArithmeticException
-2.5	-3	-2	-2	-3	-3	-2	-2	throw ArithmeticException
-5.5	-6	-5	-5	-6	-6	-5	-6	throw ArithmeticException

Untuk lebih jelas mengenai penggunaan property RoundingMode dapat dilihat dalam link berikut <http://java.sun.com/j2se/1.5.0/docs/api/java/math/RoundingMode.html>.

Jika kita tetap menginginkan nilai dibelakang koma tetap ditampilkan maka baris perintah seperti diatas perlu mengalami sedikit perubahan dengan menggunakan perintah divide yang lain hingga seperti berikut `bilAwal.divide(bilAakhir, 3, RoundingMode.HALF_UP);` nilai numeric yang sekarang sudah disisipkan ditengah operasi divide tersebut untuk menentukan berapa digit nilai dibelakang koma yang hendak ditampilkan, contoh diatas jika kita menginginkan tiga digit dibelakang koma ditampilkan.

Untuk lebih jelas mengenai penggunaan property divide dalam class BigDecimal dapat dilihat dalam link berikut <http://java.sun.com/j2se/1.5.0/docs/api/java/math/BigDecimal.html>.

Mungkin saat ini hanya sekian ilmu yang bisa dibagi dengan pembaca semoga bermanfaat, jika ada kekurangan atau masukan yang dapat meningkatkan kemampuan jangan sungkan untuk memberikan komentar.