I tried this in Ruby 2.1.4, 2.1.5 and 2.2.0, but in none of the versions it seems correct to me.

```
irb(main):041:0* amount1 = BigDecimal("241.3")
=> #
irb(main):042:0> amount2 = BigDecimal("1800")
=> #
irb(main):043:0> rate = amount1 / amount2
=> #
irb(main):044:0> rate * amount2 #should return amount1 = 241.3 in BigDecimal, but it does not
=> #
```

My guess is that there is a bug in the BigDecimal implementation.

The result of 241.3 / 1800 is a recurring decimal 0.1340555555....., which can't be represented exactly by a BigDecimal.
Please use Rational if you need an exact representation of 241.3 / 1800.

Thanks for the answer Shugo,
however I am not sure if that is easily possible for me in that case. I use BigDecimal as a field type in a Rails application and don't see how I can use Rational here :-(
Any suggestions or tips or references? would be very helpful.

#3 - 03/20/2015 06:18 AM - shugo (Shugo Maeda)
Elyasin Shaladi wrote:

Thanks for the answer Shugo,

however I am not sure if that is easily possible for me in that case. I use BigDecimal as a field type in a Rails application and don't see how I can use Rational here :-(
Any suggestions or tips or references? would be very helpful.

You can use BigDecimal#to_r to convert a BigDecimal to a Rational.

#4 - 04/27/2015 10:15 AM - Elyasin (Elyasin Shaladi)

Maybe you have an advice for the following I am thinking to use.

I could convert BigDecimal to Rational, do the calculation in Rational numbers and then convert that to back to BigDecimal.

Do you see any caveats with this approach?

Shugo Maeda wrote:

Elyasin Shaladi wrote:

Thanks for the answer Shugo,

however I am not sure if that is easily possible for me in that case. I use BigDecimal as a field type in a Rails application and don't see how I can use Rational here :-(
Any suggestions or tips or references? would be very helpful.

You can use BigDecimal#to_r to convert a BigDecimal to a Rational.