# Rational.ToDouble Method

名前空間: WS.Theia.ExtremelyPrecise

アセンブリ: ExtremelyPrecise.dll

指定した Rational の値を、それと等価の倍精度浮動小数点数に変換します。

public static double ToDouble(Rational value);

## パラメーター

value　Rational

変換するRational。

## 戻り値

value　Double  
valueと等価の倍精度浮動小数点数。

## 例外

OverflowException

ValueがMinValueより小さいかMaxValueより大きい場合。

# 例

次の例ではToDouble(Rational)メソッドを使って倍精度浮動小数値に変換しています。

// Example of the Rational.ToSingle and Rational.ToDouble methods.  
using System;  
  
class RationalToSgl\_DblDemo  
{  
 static string formatter = "{0,30}{1,17}{2,23}";  
 // Convert the decimal argument; no exceptions are thrown.  
 public static void RationalToSgl\_Dbl( Rational argument )  
 {  
 object SingleValue;  
 object DoubleValue;  
  
 // Convert the argument to a float value.  
 SingleValue = Rational.ToSingle( argument );  
  
 // Convert the argument to a double value.  
 DoubleValue = Rational.ToDouble( argument );

Console.WriteLine( formatter, argument,   
 SingleValue, DoubleValue );  
 }  
  
 public static void Main( )  
 {  
 Console.WriteLine( "This example of the \n" +  
 " Rational.ToSingle( Rational ) and \n" +  
 " Rational.ToDouble( Rational ) \nmethods " +  
 "generates the following output. It \ndisplays " +  
 "several converted decimal values.\n" );  
 Console.WriteLine( formatter, "decimal argument",   
 "float", "double" );  
 Console.WriteLine( formatter, "----------------",   
 "-----", "------" );  
  
 // Convert decimal values and display the results.  
 RationalToSgl\_Dbl( 0.0000000000000000000000000001M );  
 RationalToSgl\_Dbl( 0.0000000000123456789123456789M );  
 RationalToSgl\_Dbl( 123M );  
 RationalToSgl\_Dbl( new decimal( 123000000, 0, 0, false, 6 ) );  
 RationalToSgl\_Dbl( 123456789.123456789M );  
 RationalToSgl\_Dbl( 123456789123456789123456789M );  
 RationalToSgl\_Dbl( decimal.MinValue );  
 RationalToSgl\_Dbl( decimal.MaxValue );  
 }  
}  
  
/\*  
This example of the  
 Rational.ToSingle( Rational ) and  
 Rational.ToDouble( Rational )  
methods generates the following output. It  
displays several converted decimal values.  
  
 decimal argument float double  
 ---------------- ----- ------  
0.0000000000000000000000000001 1E-28 1E-28  
0.0000000000123456789123456789 1.234568E-11 1.23456789123457E-11  
 123 123 123  
 123.000000 123 123  
 123456789.123456789 1.234568E+08 123456789.123457  
 123456789123456789123456789 1.234568E+26 1.23456789123457E+26  
-79228162514264337593543950335 -7.922816E+28 -7.92281625142643E+28  
 79228162514264337593543950335 7.922816E+28 7.92281625142643E+28

\*/

# 注釈

この操作は倍精度浮動小数の有効桁数が少ない為、丸めによる誤差が発生します。変換結果はDouble型への明示的なキャストと等価です。

# 適用対象

### .NET Core

2.0

### .NET Framework

4.6.1

### .NET Standard

2.0

### UWP

10.0.16299

### Xamarin.Android

8.0

### Xamarin.iOS

10.14

### Xamarin.Mac

3.8