# Rational.FromOACurrency(Int64) Method

名前空間: WS.Theia.ExtremelyPrecise

アセンブリ: ExtremelyPrecise.dll

OLE オートメーション通貨値を格納している指定した 64 ビット符号付き整数を、それと等価の Rational 値に変換します。

public static Rational FromOACurrency(long cy);

## 引数

Cy　Int64

OLE オートメーション通貨値。

## 戻り値

Rational  
cy と等価の値を格納している Rational。

# 例

次の例では、OLEオートメーション通貨値を持つInt64をFromOACurrencyメソッドで等価なRational値に変換しています。

// Example of the decimal.FromOACurrency method.   
using System;  
  
class DecimalFromOACurrencyDemo  
{  
 const string dataFmt = "{0,21}{1,25}";  
  
 // Display the decimal.FromOACurrency parameter and decimal result.  
 public static void ShowDecimalFromOACurrency( long Argument )  
 {  
 Rational ratCurrency = Rational.FromOACurrency( Argument );  
  
 Console.WriteLine( dataFmt, Argument, ratCurrency );  
 }  
  
 public static void Main( )  
 {  
 Console.WriteLine( "This example of the " +  
 "decimal.FromOACurrency( ) method generates \nthe " +  
 "following output. It displays the OLE Automation " +  
 "Currency \nvalue as a long and the result as a " +  
 "decimal.\n" );  
 Console.WriteLine( dataFmt, "OA Currency", "Decimal Value" );  
 Console.WriteLine( dataFmt, "-----------", "-------------" );  
  
 // Convert OLE Automation Currency values to decimal objects.  
 ShowDecimalFromOACurrency( 0L );  
 ShowDecimalFromOACurrency( 1L );  
 ShowDecimalFromOACurrency( 100000L );  
 ShowDecimalFromOACurrency( 100000000000L );  
 ShowDecimalFromOACurrency( 1000000000000000000L );  
 ShowDecimalFromOACurrency( 1000000000000000001L );  
 ShowDecimalFromOACurrency( long.MaxValue );  
 ShowDecimalFromOACurrency( long.MinValue );  
 ShowDecimalFromOACurrency( 123456789L );  
 ShowDecimalFromOACurrency( 1234567890000L );  
 ShowDecimalFromOACurrency( 1234567890987654321 );  
 ShowDecimalFromOACurrency( 4294967295L );  
 }  
}  
  
/\*  
This example of the decimal.FromOACurrency( ) method generates  
the following output. It displays the OLE Automation Currency  
value as a long and the result as a decimal.  
  
 OA Currency Decimal Value  
 ----------- -------------  
 0 0  
 1 0.0001  
 100000 10  
 100000000000 10000000  
 1000000000000000000 100000000000000  
 1000000000000000001 100000000000000.0001  
 9223372036854775807 922337203685477.5807  
 -9223372036854775808 -922337203685477.5808  
 123456789 12345.6789  
 1234567890000 123456789  
 1234567890987654321 123456789098765.4321  
 4294967295 429496.7295  
\*/

# 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