

# Rational.ToInt32 Method

名前空間: WS.Theia.ExtremelyPrecise

アセンブリ: ExtremelyPrecise.dll

## ⚠ 重要

この API は CLS 準拠ではありません。

CLS 準拠の代替

WS.Theia.ExtremelyPrecise.Rational.ToInt64(Rational)

指定した Rational の値を、等価の 32 ビット符号なし整数に変換します。

---

```
public static int ToInt32(Rational value);
```

---

## パラメーター

value Rational

変換する Rational。

## 戻り値

value UInt32

value と等価の 32 ビット符号なし整数。

## 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 ToUInt32(Rational) メソッドを使って UInt32 値に変換しています。

---

```
using System;
```

```
class Example
```

```
{
    public static void Main( )
    {
        Rational[] values = { 123m, new decimal(123000, 0, 0, false, 3),
                               123.999m, 4294967295m, 4294967295.001m,
                               4294967296m, 2147483647m, 2147483647.001m,
                               -0.999m, -1m, -2147483648m, -2147483648.001m };
        foreach (var value in values) {
            try {
                uint number = Rational.ToUInt32(value);
                Console.WriteLine("{0} --> {1}", value, number);
            }
            catch (OverflowException e)
            {
                Console.WriteLine("{0}: {1}", e.GetType().Name, value);
            }
        }
    }
}

// The example displays the following output:
//      123 --> 123
//      123.000 --> 123
//      123.999 --> 123
//      4294967295 --> 4294967295
//      OverflowException: 4294967295.001
//      OverflowException: 4294967296
//      2147483647 --> 2147483647
//      2147483647.001 --> 2147483647
//      OverflowException: -0.999
//      OverflowException: -1
//      OverflowException: -2147483648
//      OverflowException: -2147483648.001
```

---

# 注釈

`value` パラメーターの値から小数点以下の値を切り捨てた値に変換されます。変換結果は `UInt32` 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**