# Rational.Implicit Operator

名前空間: WS.Theia.ExtremelyPrecise

アセンブリ: ExtremelyPrecise.dll

# オーバーロード

|  |  |
| --- | --- |
| Implicit(Byte to Rational) | Byte 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(SByte to Rational) | SByte 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(Int32 to Rational) | Int32 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(UInt32 to Rational) | UInt32 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(Int16 to Rational) | Int16 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(UInt16 to Rational) | UInt16 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(Int64 to Rational) | Int64 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(UInt64 to Rational) | UInt64 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(Single to Rational) | Single 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(Double to Rational) | Double 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(Boolean to Rational) | Boolean 値から Rational 値への暗黙的な変換を定義します。 |
| Implicit(Decimal to Rational) | Decimal 値から Rational 値への暗黙的な変換を定義します。 |

# Implicit(Byte to Rational)

Byte 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(byte value);

## パラメータ

value　Byte  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、Byte型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Int32)メソッドを使用します。

byte byteValue = 254;  
Rational number = byteValue;  
number = Rational.Add(number, byteValue);  
Console.WriteLine(number > byteValue); // Displays True

# Implicit(SByte to Rational)

⚠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.Implicit(Int32 to Rational)

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(sbyte value);

SByte 値から Rational 値への暗黙的な変換を定義します。

## パラメータ

value　SByte  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、SByte型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Int32)メソッドを使用します。

sbyte sByteValue = -12;  
Rational number = Math.Pow(sByteValue, 3);  
Console.WriteLine(number < sByteValue); // Displays True

# Implicit(Int32 to Rational)

Int32 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(int value);

Int32 値から Rational 値への暗黙的な変換を定義します。

## パラメータ

value　Int32  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、Int32型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Int32)メソッドを使用します。

int intValue = 65000;  
Rational number = intValue;  
number = Rational.Multiply(number, intValue);  
Console.WriteLine(number == intValue); // Displays False

# Implicit(UInt32 to Rational)

⚠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.Implicit(Int64 to Rational)

UInt32 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(uint value);

## パラメータ

value　UInt32  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、UInt32型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Int64)メソッドを使用します。

uint uIntValue = 65000;  
Rational number = uIntValue;  
number = Rational.Multiply(number, uIntValue);  
Console.WriteLine(number == uIntValue); // Displays False

# Implicit(Int16 to Rational)

Int16 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(short value);

## パラメータ

value　Int16  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、Int16型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Int32)メソッドを使用します。

short shortValue = 25064;  
Rational number = shortValue;  
number += shortValue;  
Console.WriteLine(number < shortValue); // Displays False

# Implicit(UInt16 to Rational)

⚠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.Implicit(Int32 to Rational)

UInt16 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(ushort value);

## パラメータ

value　UInt16  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、UInt16型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Int32)メソッドを使用します。

ushort uShortValue = 25064;  
Rational number = uShortValue;  
number += uShortValue;  
Console.WriteLine(number < uShortValue); // Displays False

# Implicit(Int64 to Rational)

Int64 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(long value);

## パラメータ

value　Int64  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、Int64型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Int64)メソッドを使用します。

long longValue = 1358754982;  
Rational number = longValue;  
number = number + (longValue / 2);  
Console.WriteLine(number \* longValue / longValue); // Displays 2038132473

# Implicit(UInt64 to Rational)

⚠重要  
この API は CLS 準拠ではありません。  
CLS 準拠の代替　System.Double

UInt64 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(ulong value);

## パラメータ

value　UInt64  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、UInt64型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(UInt64)メソッドを使用します。

ulong uLongValue = 1358754982;  
Rational number = uLongValue;  
number = number \* 2 - uLongValue;  
Console.WriteLine(number \* uLongValue / uLongValue); // Displays 1358754982

# Implicit(Single to Rational)

Single 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(float value);

## パラメータ

value　Single  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、Single型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Single)メソッドを使用します。

float floatValue = 135875498.25f;  
Rational number = floatValue;  
number = number \* 2 - floatValue;  
Console.WriteLine(number \* floatValue / floatValue); // Displays 135875498.25

# Implicit(Double to Rational)

Double 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(double value);

## パラメータ

value　Double  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、Double型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Double)メソッドを使用します。

double doubleValue = 135875498.25;  
Rational number = doubleValue;  
number = number \* 2 - doubleValue;  
Console.WriteLine(number \* doubleValue / doubleValue); // Displays 135875498.25

# Implicit(Boolean to Rational)

Boolean 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(bool value);

## パラメータ

value　Boolean  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、Boolean型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Boolean)メソッドを使用します。

bool boolValue = true;  
Rational number = boolValue;  
number = number \* 2 - boolValue;  
Console.WriteLine(number \* boolValue / boolValue); // Displays 1

# Implicit(Decimal to Rational)

Decimal 値から Rational 値への暗黙的な変換を定義します。

public static implicit operator WS.Theia.ExtremelyPrecise.Rational(decimal value);

## パラメータ

value　Decimal  
Rational へと変換する値。

## 戻り値

Rational  
valueパラメータと等価なRational値。

## 注釈

次の例では、Decimal型をRational型に変換する方法示します。暗黙的な変換演算子をサポートしない言語では、代わりにRational.Rational(Decimal)メソッドを使用します。

decimal decimalValue = 135875498.2m;  
Rational number = decimalValue;  
number = number \* 2 - decimalValue;  
Console.WriteLine(number \* decimalValue / decimalValue); // Displays 135875498.2

# 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