

# 第05期-基本数值类型系统

2022年3月23日 9:56

## 整型数值类型

C#关键字	范围	大小	.NET 类型
sbyte	-128 到 127	8 位带符号整数	<a href="#">System.SByte</a>
byte	0 到 255	无符号的 8 位整数	<a href="#">System.Byte</a>
short	-32,768 到 32,767	有符号 16 位整数	<a href="#">System.Int16</a>
ushort	0 到 65,535	无符号 16 位整数	<a href="#">System.UInt16</a>
int	-2,147,483,648 到 2,147,483,647	带符号的 32 位整数	<a href="#">System.Int32</a>
uint	0 到 4,294,967,295	无符号的 32 位整数	<a href="#">System.UInt32</a>
long	-9,223,372,036,854,775,808 到 9,223,372,036,854,775,807	64 位带符号整数	<a href="#">System.Int64</a>
ulong	0 到 18,446,744,073,709,551,615	无符号 64 位整数	<a href="#">System.UInt64</a>
nint	取决于平台	带符号的 32 位或 64 位整数	<a href="#">System.IntPtr</a>
nuint	取决于平台	无符号的 32 位或 64 位整数	<a href="#">System.UIntPtr</a>

## 浮点型数值类型

C# 关键字	大致范围	精度	大小	.NET 类型
float	$\pm 1.5 \times 10^{-45}$ 至 $\pm 3.4 \times 10^{38}$	大约 6-9 位数字	4 个字节	<a href="#">System.Single</a>
double	$\pm 5.0 \times 10^{-324}$ 到 $\pm 1.7 \times 10^{308}$	大约 15-17 位数字	8 个字节	<a href="#">System.Double</a>
decimal	$\pm 1.0 \times 10^{-28}$ 至 $\pm 7.9228 \times 10^{28}$	28-29 位	16 个字节	<a href="#">System.Decimal</a>

## 静态常量

MinValue、MaxValue、Double.NaN、Double.NegativeInfinity 和 Double.PositiveInfinity

## 类型转换：

### 基本类型转换

```
public static int ToInt32 (string? value);
```

```
public static int Parse (string s);  
public static bool TryParse (string? s, out int result);
```

强制类型转换异常：FormatException 和 OverflowException

数值与字节：

BitConverter.GetBytes 和 BitConverter.ToInt32 方法

字节与字符串：

```
string str = System.Text.Encoding.Default.GetString( byteArray );  
byte[] byteArray = System.Text.Encoding.Default.GetBytes( str );
```

## 运算支持

加法、减法、除法、乘法、减法、位运算 AND 、 OR 和 XOR 左移和右移运算符。

数学类 Math 提供很多运算支持。

## 相等性比较

== 等号判断符号

```
public int CompareTo (int value);  
public bool Equals (int obj);  
public static bool ReferenceEquals (object? objA, object? objB);
```

## 标准数字格式字符串

<https://docs.microsoft.com/zh-cn/dotnet/standard/base-types/standard-numeric-format-strings>

## 进制表示和转换

十进制：不使用任何前缀

十六进制：使用 或 0X 前缀

二进制：使用 或 0B 前缀

```
var decimalLiteral = 42;  
var hexLiteral = 0x2A;  
var binaryLiteral = 0b_0010_1010;
```

## [数值转换为进制字符串](#)

复合格式字符串: { index[,alignment][:formatString]}

## 整数文本类型由后缀确定

没有后缀: int、uint、long、ulong

U 或 u 为后缀: uint、ulong

L 或 l 为后缀: long、ulong

UL、Ul、uL、ul、LU、Lu、lU 或 lu: ulong。

d 或 D 后缀: double

f 或 F 后缀: float

m 或 M 后缀: decimal

## 参考资料

[Convert.ToInt32\(\) 和 int.Parse\(\) 区别](#)