

第13章 bool型とif文



この章では値型のひとつであるbool型と、if文について解説します。

値型のデータ型については、第5章でも解説しています。

			C#型	ビット長	.NET型
数 値 型	整数型	符号付き	sbyte	8ビット	System.SByte
			short	16ビット	System.Int16
			int	32ビット	System.Int32
			long	64ビット	System.Int64
		符号なし	byte	8ビット	System.Byte
			ushort	16ビット	System.UInt16
			uint	32ビット	System.UInt32
			ulong	64ビット	System.UInt64
	実数型	浮動小数点数型	float	32ビット	System.Single
			double	64ビット	System.Double
10進データ型		decimal	128ビット	System.Decimal	
論理型			bool	8ビット	System.Boolean
文字型			char	16ビット	System.Char

bool型はtrue(真)かfalse(偽)のいずれかを表します。C/C++のように0をfalseのかわりに使うことは許されていません。

今までのプログラムでは、書いたものはすべて順番に実行されました。これでは、困ることもあります。条件によりある部分を実行したりしなかったりできると、プログラムらしくなります。これには、if文を使います。

```
if (条件式)
{
    ...
}
```

条件式が真なら、...が実行され、偽なら実行されません。また、...の部分が一つの文であるときは、{}は省略できます。また、次のようなelse文とセットで使うこともよく行われます。

```
if (条件式)
{
    ... (1)
}
else
{
    ... (2)
}
```

条件式が真の時は...(1)が実行され、偽の時は...(2)が実行されます。それぞれの文が1つのみの時はやはり、{}は省略可能です。

また、次のような使い方もできます。

```
if (条件式1)
{
    ... (1)
}
else if (条件式2)
{
    ... (2)
}
else if (条件式3)
{
    ... (3)
}
...
else if (条件式n)
{
    ... (n)
}
else
{
    ...
}
```

条件式1が真なら...(1)が実行され、残りは無視されます。

条件式1が偽なら条件式2が評価されます。真なら...(2)が実行され残りは無視されます。

条件式1からnまですべて偽ならelseのところの...が実行されます。

さて、条件式によく使われる演算子は次のようなものがあります。

$a > b$	aがbより大きいならこの式は真、そうでないなら偽
$a \geq b$	aがbより大きいか等しい時この式は真、そうでない時は偽
$a < b$	aがbより小さいならこの式は真、そうでないときは偽
$a \leq b$	aがbより小さいか等しい時はこの式は真、そうでない時は偽
$a == b$	aとbが等しいならこの式は真、そうでない時は偽
$a != b$	aとbが等しくないならこの式は真、そうでない時は偽
$!a$	aが真なら偽、偽なら真

また、条件式を「または」とか「かつ」で結合することもできます。「または」に相当する演算子が「&&」で、「かつ」に相当する演算子が「||」です。(注:C#では、結果として&や|でも同じ意味に使えますが、C/C++の名残で&&と||を使うことが圧倒的に多いようです。詳しくは後の章で解説します。)

では、簡単なサンプルを見てみましょう。

```
// bool01.cs

using System;

class bool01
{
    public static void Main()
    {
        int n;

        Console.WriteLine("0から9までの数字を入力---");
        string strNum = Console.ReadLine();

        if (strNum.Length >= 2)
        {
            Console.WriteLine("1文字だけの入力です");
            return;
        }

        if (!Char.IsNumber(strNum, 0))
        {
            Console.WriteLine("数字を入力します");
            return;
        }
        else
            n = Int32.Parse(strNum);

        if (n >= 5)
            Console.WriteLine("あなたの入力した数字は5以上です");
        else
            Console.WriteLine("あなたの入力した数字は4以下です");
    }
}
```

ユーザーに、0から9までの整数値を入力してもらいます。もし、ユーザーが2文字以上入力した場合はstrNum.Lengthの値が2以上となります。

```
if (strNum.Length >= 2)
{
    Console.WriteLine("1文字だけの入力です");
    return;
}
```

strNum.Lengthが2以上なら、{} 内が実行されます。注意を表示してreturn文が実行されます。Main関数内でreturn文を実行するとプログラムが終了します。

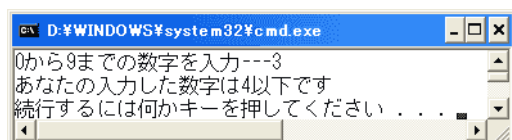
次にstrNum文字列の最初の文字が、数字かどうかを調べます。ユーザーがaとかbとか数字以外の文字を入力するとChar.IsNumber(strNum, 0)がfalseになります。これに、!がついてるのでfalseの場合はtrueになります。つまり、このif文は数字以外の場合、{} 内が実行されます。数字以外の場合は、注意を促して、プログラムは終了します。

数字ならば、

```
n = Int32.Parse(strNum);
```

で、strNum文字列をint型に変換してnに代入します。

このnについて、5以上なら(if(n >= 5))、「あなたの入力した数字は5以上です」と表示します。そうでない場合は、「あなたの入力した数字は4以下です」と表示します。



if文を使って、メニューのようなものも作れます。

```
// bool02.cs

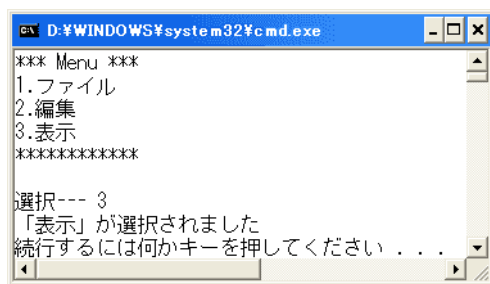
using System;

class bool02
{
    public static void Main()
    {
        Console.WriteLine("*** Menu ***");
        Console.WriteLine("1.ファイル");
        Console.WriteLine("2.編集");
        Console.WriteLine("3.表示");
        Console.WriteLine("*****");
        Console.Write("Yn");
        Console.Write("選択--- ");

        string selNo = Console.ReadLine();
        if (selNo.Length >= 2)
        {
            Console.WriteLine("入力桁数が多すぎます");
            return;
        }
        if (!char.IsNumber(selNo[0]))
        {
            Console.WriteLine("数字を入力します");
            return;
        }

        int no = int.Parse(selNo);
        if (no <= 0 || no >= 4)
        {
            Console.WriteLine("番号が不正です");
            return;
        }
        if (no == 1)
            Console.WriteLine("「ファイル」が選択されました");
        else if (no == 2)
            Console.WriteLine("「編集」が選択されました");
        else
            Console.WriteLine("「表示」が選択されました");
    }
}
```

実行すると、次のようになります。



[\[C# Index\]](#) [\[総合Index\]](#) [\[Previous Chapter\]](#) [\[Next Chapter\]](#)

Update 20/Aug/2006 By Y.Kumei

当ホームページの一部または全部を無断で複写、複製、転載あるいはコンピュータ等のファイルに保存することを禁じます。