

プログラムの流れを変えよう

● プログラムの実行順序を変えるには？

これまでのプログラムは実行順序が上から順番だったり、イベントが発生した順番だったりしましたが、順番に実行しては、単純な動作しかできません。状況によっては、「演算結果によっては処理を終わりにしたい」とか「入力されたデータによって動作を変えたい」ということもありますよね？ そういうときはif文を利用して条件分岐します。

● if文

if文は条件式を評価して結果が真 (true) だったら、処理を実行します。「もし条件を満たしていたら、処理をする」というイメージですね。記述方法は以下のとおりです。

```
if (条件式)
{
    条件式を満たした場合に実行する処理
}
```

条件式には以下のような比較演算子を用いて比較式を書くことがほとんどです。

比較演算子	内容
==	等しい
<	小なり
>	大なり
<=	以下
>=	以上
!=	等しくない

等しい場合の演算子が「=」ではなく、「==」となっている点に注意してください。C#の場合「=」は代入の意味ですので、書き間違えないようにしましょう。

比較式は計算式と同様に結果を返します。計算式の場合は計算結果でしたが、比較式の場合は真か偽か (trueまたは false) を返します。この真か偽かを扱うデータ型を「**bool型**」と言いますので、こちらも合わせて覚えておいてください。

では、if文の例を挙げてみます。

```
if (x < 15)
{
    // xが15未満の場合
    Console.WriteLine("xは15未満です。");
}

if (20 <= x)
{
    // xが20以上の場合
    Console.WriteLine("xは20以上です。");
}

if (x == y)
{
    // xとyが等しい場合
    Console.WriteLine("xとyは等しいです。");
}
```

本書での比較式は数直線と合わせて、比較演算子の左側の値より右側の値の方が大きいように記述しています

● if~else 文

if文は条件式を満たした場合の処理だけを記述しましたが、満たしていない場合の処理も記述したい場合はif~else文を使用します。「もし条件を満たしていたら処理1をして、満たさなければ処理2をする」というイメージですね。記述方法は以下のとおりです。

```

if (条件式)
{
    条件式を満たした場合に実行する処理
}
else
{
    条件式を満たしていない場合に実行する処理
}
    
```

では、if~else文の例を挙げてみます。キーボードから数値を入力させ、奇数なら「奇数です」、偶数なら「偶数です」と表示するプログラムです。

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace if文の使い方
8  {
9      class Program
10     {
11         static void Main(string[] args)
12         {
13             Console.WriteLine("整数を入力して、Enterキーを押してください。");
14
15             // string型変数の宣言と入力結果の代入
16             string inputNumber = Console.ReadLine();
17
18             // int型変数の宣言と入力結果をint型に変換して代入
19             int number = int.Parse(inputNumber);
20
21             // 入力された整数で条件分岐
22             if (number % 2 == 0)
23             {
24                 // 偶数の場合
25                 Console.WriteLine("偶数です。");
26             }
27             else
28             {
29                 // 奇数の場合
30                 Console.WriteLine("奇数です。");
31             }
32
33             // Enterキーが押されるまで待機
34             Console.ReadLine();
35         }
36     }
37 }
    
```

「2で割った余りが0になる」
場合を偶数と判断しています

「2で割った余りが0にならない」場合を奇数と判断しています

実行結果は次のようになります。

「12」を入力した場合

```

整数を入力して、Enterキーを押してください。
12
偶数です。
    
```

「33」を入力した場合

```

整数を入力して、Enterキーを押してください。
33
奇数です。
    
```

🖥️ 練習1

以下のようなプロジェクトを作成してください。

```
ソリューション名：身長チェッカー
プロジェクト名：身長チェッカー
プロジェクトテンプレート：コンソール アプリケーション
```

よく遊園地のジェットコースターで身長制限があったりしますよね？ この身長制限のプログラムを作ってみましょう。キーボードから身長（cm単位で）を入力させて、入力された身長によって「ジェットコースターに乗れます」「ジェットコースターに乗れません」と表示される結果が切り替わるようなプログラムを作成してください。ジェットコースターに乗れる身長は「120cm以上」とします。

実行結果

```
「120」を入力した場合
身長を入力して、Enterキーを押してください。
120
ジェットコースターに乗れます。

「119」を入力した場合
身長を入力して、Enterキーを押してください。
119
ジェットコースターに乗れません。
```

● if～else if～else 文

if文もif～else文も条件式が1つでしたが、複数の条件式のどれにあてはまるかで処理を切り替えたい場合はif～else if～else文を使用します。

「もし条件1を満たしているなら処理1をして、満たさなければ処理2をする」というイメージですね。記述方法は以下のとおりです。

```
if (条件式 1)
{
    条件式 1 を満たした場合に実行する処理
}
else if (条件式 2)
{
    条件式 1 は満たさず、条件式 2 を満たした場合に実行する処理
}
else
{
    条件式 1、条件式 2 ともに満たしていない場合に実行する処理
}
```

else ifの部分はif～else if～else if～else if …… elseのように条件式の分だけ増やすことができます。

では、if～else if～else文の例を挙げてみます。キーボードから年齢を入力させ、15歳以下なら子供料金、16歳以上～19歳未満なら高校生料金、19歳以上なら大人料金と表示するプログラムです。

```

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace 年齢チェッカー
8  {
9      class Program
10     {
11         static void Main(string[] args)
12         {
13             // string型変数の宣言と入力結果の代入
14             Console.WriteLine("年齢を入力して、Enterキーを押してください。");
15             string inputAge = Console.ReadLine();
16
17             // int型変数の宣言とint型に変換した入力結果の代入
18             int age = int.Parse(inputAge);
19
20             // 入力された年齢で条件分岐
21             if (age <= 15)
22             {
23                 // 15歳以下の場合
24                 Console.WriteLine("子供料金");
25             }
26             else if (age < 19)
27             {
28                 // 16歳以上19歳未満の場合
29                 Console.WriteLine("高校生料金");
30             }
31             else
32             {
33                 // 19歳以上の場合
34                 Console.WriteLine("大人料金");
35             }
36
37             // Enterキーが押されるまで待機
38             Console.ReadLine();
39         }
40     }
41 }

```

15 以下の場合は if 文で条件が一致するので、else if 文で 19 未満とすれば、16 から 18 が条件に一致します

実行結果は次のようになります。

「15」を入力した場合

```

年齢を入力して、Enterキーを押してください。
15
子供料金

```

「18」を入力した場合

```

年齢を入力して、Enterキーを押してください。
18
高校生料金

```

「19」を入力した場合

```

年齢を入力して、Enterキーを押してください。
19
大人料金

```

 **練習2**

以下のようなプロジェクトを作成してください。

ソリューション名：サービスチケット
プロジェクト名：サービスチケット
プロジェクトテンプレート：コンソール アプリケーション

お店の来店回数でサービスが変わるプログラムを作ってみましょう。来店回数をキーボードから入力させ、その来店回数で表示されるサービスが変わるプログラムを作成してください。

来店回数	サービス
1	サービスチケット1枚
2～3	サービスチケット2枚
4以上	サービスチケット3枚

実行結果

「1」を入力した場合

```
お店の来店回数を入力して、Enterキーを押してください。  
1  
サービスチケット1枚
```

「2」を入力した場合

```
お店の来店回数を入力して、Enterキーを押してください。  
2  
サービスチケット2枚
```

「4」を入力した場合

```
お店の来店回数を入力して、Enterキーを押してください。  
4  
サービスチケット3枚
```