

# Перечисления

Перечисления – это способ задания набора именованных констант, который можно назначить переменной

```
enum Days { Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday }  
enum Months : byte { Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec };
```

перечислитель  
и

Объявление переменной типа перечисление:

```
Days d;  
Months m
```

Обращение к конкретным параметрам

```
Days.Sunday,  
Months.Feb
```

**По умолчанию базовым типом  
каждого элемента перечисления  
является int**

```
enum Days { Sat = 1, Sun, Mon, Tue, Wed, Thu, Fri };
```

Для перечисления утверждены следующие типы:

- byte,
- sbyte,
- short,
- ushort,
- int,
- uint,
- long
- ulong.

**Имя перечислителя не может содержать пробелов.**

Базовый тип указывает размер памяти, выделенный для каждого перечислителя

```
int x = (int)Days.Sun;
```

**Пример 1.** В этом примере объявляется перечисление Days. Два перечислителя явно преобразуются в целые числа и назначаются целочисленным переменным.

```
public class EnumTest
{
    enum Days { Sun, Mon, Tue, Wed, Thu, Fri, Sat };

    static void Main()
    {
        int x = (int)Days.Sun; int y = (int)Days.Fri;
        Console.WriteLine("Sun = {0}", x); Console.WriteLine("Fri = {0}", y);
    }
}
/* Результат:
   Sun = 0
   Fri = 5
*/
```

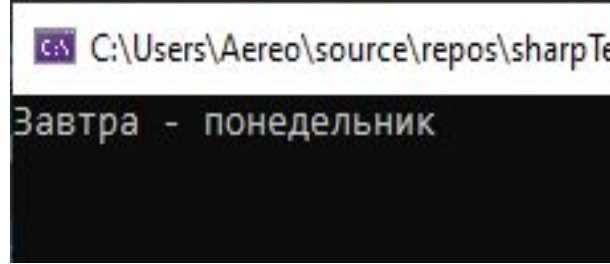
**Пример 2.** В данном примере для объявления enum, члены которого имеют тип long, используется параметр базового типа.

```
public class EnumTest2
{
    enum Range : long { Max = 2147483648L, Min = 255L };
    static void Main()
    {
        long x = (long)Range.Max; long y = (long)Range.Min;
        Console.WriteLine("Max = {0}", x);
        Console.WriteLine("Min = {0}", y);
    }
}
/* Результат:
Max = 2147483648
Min = 255
*/
```

**Пример 3.** Пример программы, в которой перечисление используется в операторе switch,

```
using System;
namespace app19_enum
{
    class Program
    {
        public enum DaysOfWeek // Объявление типа "перечисление"
        {
            Monday = 0, Tuesday = 1, Wednesday = 2,
            Thursday = 3, Friday = 4,
            Saturday = 5,
            Sunday = 6
        }
        static void Main()
        {
            WriteText(DaysOfWeek.Sunday); // Задание значения
                                           // параметра
            Console.ReadLine();
        }
    }
}
```

```
static void WriteText(DaysOfWeek days)
{
    switch (days)           // Обработка значений переменной days
    {
        case DaysOfWeek.Monday: Console.WriteLine("Понедельник");
            break;
        case DaysOfWeek.Tuesday: Console.WriteLine("Вторник");
            break;
        case DaysOfWeek.Wednesday: Console.WriteLine("Среда!");
            break;
        case DaysOfWeek.Thursday: Console.WriteLine("Четверг");
            break;
        case DaysOfWeek.Friday: Console.WriteLine("Пятница");
            break;
        case DaysOfWeek.Saturday: Console.WriteLine("Суббота");
            break;
        case DaysOfWeek.Sunday: Console.WriteLine("Завтра – понедельник");
            break;
    }
}
```



```
using System;

namespace ConsoleApp
{
    class Program
    {
        enum TrafficLight
        {
            Red,
            Yellow,
            Green
        }
        static void Main(string[] args)
        {
            var t = TrafficLight.Green;
            Console.WriteLine(t);
            Console.ReadLine();
        }
    }
}
```



## Типы перечислений как битовые флаги

бит, отражающий некоторое состояние, называют **флагом**

```
using System;
namespace app20_flags
{
    class Program
    {
        [Flags()]
        public enum MilitaryType    // Тип используемого
                                    // вооружения
        {
            Marine = 1,
            Land = 2,
            Cosmic = 4,
            AllType = Marine | Land | Cosmic
        }
    }
}
```

```
public static void Main(string[] args)
{
    // Установка флажков Land и Cosmic в type0:
    MilitaryType type0 = MilitaryType.Land |
                        MilitaryType.Cosmic;
    // Console.WriteLine(type0);
    // Выведется "Land, Cosmic"
    Console.WriteLine("Установлены флажки {0}", type0);

    // Установка флажка Marine в type0:
    type0 |= MilitaryType.Marine;
    // Проверка флажка Land в type0: он должен быть
    // установлен в предыдущих операторах
    bool b=System.Convert.ToBoolean(type0 & MilitaryType.Land);
    if(b) Console.WriteLine("Флажок Land в type0 "+"установлен"); // Выведется
        //"Land, Cosmic"
}
}
```

C:\Users\Aereo\source\repos\sharpTest\sharp

Установлены флажки Land, Cosmic  
Флажок Land в type0 установлен