# C# Introduction. Урок 2

# Базовые типы и операторы

# Базовые типы языка C#

| Тип | Память в битах | Диапазон значений | Описание | Структура |
|---|---|---|---|---|
| byte | 8 | от 0 до 255 | Целое неотрицательное число | Byte |
| sbyte | 8 | от -128 до 127 | Целое число | SByte |
| short | 16 | от -32768 до 32767 | Целое число | Int16 |
| ushort | 16 | от 0 до 65535 | Целое неотрицательное число | UInt16 |
| int | 32 | от -2147483648 до 2147483647 | Целое число | Int32 |
| uint | 32 | от 0 до 4294967295 | Целое неотрицательное число | UInt32 |
| long | 64 | от -9223372036854775808 до 9223372036854775807 | Целое число | Int64 |
| ulong | 64 | от 0 до 18446744073709551615 | Целое неотрицательное число | UInt64 |
| float | 32 | от 1.5E-45 до 3.4E+38 | Действительное число(с плавающей точкой) | Single |
| double | 64 | от 5E-324 до 1.7E+308 | Действительное число(с плавающей точкой) двойной точности | Double |
| decimal | 128 | от 1E-28 до 7.9E+28 | Действительное число для выполнения особо точных(финансовых) расчетов | Decimal |
| char | 16 | от 0 до 65535 | Символьное значение | Char |
| bool | 8 | Значения true и false | Логическое значение | Boolean |

# Управляющие символы

\n – переход к новой строке
\t – горизонтальная табуляция
\b – перемещение курсора на одну позицию назад
\r – перемещение курсора в начало текущей строки
@ - литерал становится «буквальным»

# Управляющие символы

«Изучаем\n язык C#»

@«Изучаем\n язык C#»

# Объявление переменной

```
static void Main(){
    int A;
    //команды
    {
        int B;
       //команды
    }
    //команды
}
```

# Операторы

- Арифметические операторы
- Логические операторы
- Операторы сравнения
- Побитовые операторы

# Арифметические операторы

| Оператор | Описание |
|---|---|
| + | Оператор сложения. Значением выражения вида A+B является сумма значений переменных A и B |
| - | Оператор вычитания. Значением выражения вида A-B является разность значений переменных A и B |
| * | Оператор умножения. Значением выражения вида A*B является произведение значений переменных A и B |
| / | Оператор деления. Значением выражения вида A/B является частное значений переменных A и B. Если оба операнда A и B – целочисленные, то деление выполняется нацело. Для выполнения обычного деления с целочисленными операндами перед выражением указывают инструкцию (double) |
| % | Оператор вычисления остатка от деления. Значением выражения вида A%B является остаток от целочисленного деления переменных A и B. |
| ++ | Оператор инкремента. При выполнении команды вида A++ или ++A значение переменной A увеличивается на 1 |
| -- | Оператор декремента. При выполнении команды вида A-- или --A значение переменной A уменьшается на 1 |

# Операторы сравнения

| Оператор | Описание |
|---|---|
| < | Оператор «меньше». Значение выражения вида A<B равно true, если значение операнда A меньше значения операнда B. В противном случае выражение A<B равно false |
| <= | Оператор «меньше или равно». Значение выражения вида A<=B равно true, если значение операнда A меньше или равно значению операнда B. В противном случае выражение A<=B равно false |
| > | Оператор «больше». Значение выражения вида A>B равно true, если значение операнда A больше значения операнда B. В противном случае выражение A>B равно false |
| >= | Оператор «больше или равно». Значение выражения вида A>=B равно true, если значение операнда A больше или равно значению операнда B. В противном случае выражение A>=B равно false |
| = | Оператор «равно». Значение выражения вида A==B равно true, если значение операнда A равно значению операнда B. В противном случае выражение A==B равно false |
| != | Оператор «не равно». Значение выражения вида A!=B равно true, если значение операнда A не равно значению операнда B. В противном случае выражение A!=B равно false |

# Логические операторы

| Оператор | Описание |
|---|---|
| & | Оператор «логическое и». Результатом выражения A&B является значение true, если оба операнда A и B равны true. Если хотя бы один из операндов равен false, то результатом выражения A&B является false |
| && | Оператор «логическое и» (упрощенная форма). Значение выражения A&&B равно true, если оба операнда A и B равны true. В противном случае результат выражения равен false. Если при вычислении первого операнда A оказалось, что он равен false, то значение второго операнда B не вычисляется. |
| \| | Оператор «логическое или». Результатом выражения A\|B является значение true, если хотя бы один из операндов A или B равен true. Если оба операнда равны false, то результатом выражения A\|B является false |
| \|\| | Оператор «логическое или» (упрощенная форма). Результатом выражения A\|\|B является значение true, если хотя бы один из операндов A или B равен true. В противном случае результат выражения равен false. Если при вычислении первого операнда A оказалось, что он равен true, то значение второго операнда B не вычисляется. |
| ^ | Оператор «логическое исключающее или». Результатом выражения A^B является значение true, если один из операндов A или B равен true, а другой равен false. Если боа операнда равны false или оба операнда равны true, то результатом выражения A^B является значение false |
| ! | Оператор «логическое отрицание». Значение выражения вида !A равно true, если значение операнда A равно false. Если значение операнда A равно true, то значение выражения !A равно false |

# Логические операторы

A&B или A&&B

| B \ A | true | false |
|---|---|---|
| true | true | false |
| false | false | false |

A|B или A||B

| B \ A | true | false |
|---|---|---|
| true | true | true |
| false | true | false |

# Логические операторы

A^B

| B \ A | true | false |
|---|---|---|
| true | false | true |
| false | true | false |

!A

| A | true | false |
|---|---|---|
| !A | false | true |

# Побитовые операторы и двоичные коды

0+0 = 0

1+0 = 1

0+1 = 1

1+1 = 10

# Побитовые операторы и двоичные коды

13 = 1101

9 = 1001

22 = 10110

$$\begin{array}{r} 1101 \\ +\ 1001 \\ \hline 10110 \end{array}$$

# Побитовые операторы

| Оператор | Описание |
|---|---|
| & | «Побитовое и». Результатом выражения «A&B» является число, которое получается сравнением соответствующих битов в представлении чисел A и B. На выходе получается единичный бит, если оба сравниваемых бита единичные. Если хотя бы один из сравниваемых битов нулевой, в рез-те получаем нулевой бит |
| \| | «Побитовое или». Результатом выражения «A\|B» является число, которое получается сравнением соответствующих битов в представлении чисел A и B. На выходе получается единичный бит, если хотя бы один из сравниваемых битов единичный. Если оба сравниваемых бита нулевые, в рез-те получаем нулевой бит |
| ^ | «Побитовое исключающее или». Результатом выражения «A^B» является число, которое получается сравнением соответствующих битов в представлении чисел A и B. На выходе получается единичный бит, если сравниваемые бита различны. Если биты одинаковы, в рез-те получаем нулевой бит |
| ~ | «Побитовое инверсия». Значение выражения вида ~A есть число, битовое представление которого получается заменой в битовом представлении числа A нулей на единицы и единиц на нулей. Значение операнда A при этом не меняется |
| << | «Сдвиг налево». Результатом выражения A<<n является число, которое получается путем сдвига бинарного представления числа A на n позиций влево. Младшие биты при этом заменяются нулями, а старшие биты теряются. Значение операнда A при этом не меняется |
| >> | «Сдвиг направо». Результатом выражения A>>n является число, которое получается путем сдвига бинарного представления числа A на n позиций вправо. Младшие биты при этом теряются, а старшие биты заполняются значением старшего бита. Значение операнда A при этом не меняется |

# Тернарный оператор

условие?значение:значение

num = (x<0)&1:2

# Проверка на четность/нечетность

```
using System;

class OddEvenDemo{
    static void Main(){
        int number, reminder;
        Console.WriteLine(«Введите целое число: »);
        number=Int32.Parse(Console.ReadLine());
        reminder=number%2;
        string txt=“Вы ввели ”;
        txt+=(reminder==0?«четное»:«нечетное»)+« число!»;
        Console.WriteLine(txt);
    }
}
```