



ОСНОВЫ ПРОГРАММИРОВАНИЯ

ТИПЫ ДАННЫХ:

- ▶ `bool`: хранит значение `true` или `false`

```
bool alive = true;
```

```
bool isDead = false;
```



- ▶ `sbyte`: хранит целое число от -128 до 127 и занимает 1 байт.
- ▶ `sbyte bit1 = -101;`
- ▶ `sbyte bit2 = 102;`

- ▶ `short`: хранит целое число от -32768 до 32767 и занимает 2 байта.
- ▶ `short n1 = 1;`
- ▶ `short n2 = 102;`



- ▶ `ushort`: хранит целое число от 0 до 65535 и занимает 2 байта.
 - ▶ `ushort n1 = 1;`
 - ▶ `ushort n2 = 102;`
-
- ▶ `ushort`: хранит целое число от 0 до 65535 и занимает 2 байта.
 - ▶ `ushort n1 = 1;`
 - ▶ `ushort n2 = 102;`

- ▶ `int`: хранит целое число от -2147483648 до 2147483647 и занимает 4 байта.
- ▶ `int a = 10;`
- ▶ `int b = -11;`
- ▶ `uint`: хранит целое число от 0 до 4294967295 и занимает 4 байта.
- ▶ `uint a = 10;`
- ▶ `uint b = 0;`



- ▶ `long`: хранит целое число от $-9\,223\,372\,036\,854\,775\,808$ до $9\,223\,372\,036\,854\,775\,807$ и занимает 8 байт.
- ▶ `long a = -10;`
- ▶ `long b = 10;`
- ▶ `ulong`: хранит целое число от 0 до $18\,446\,744\,073\,709\,551\,615$ и занимает 8 байт. Представлен системным типом `System.UInt64`
- ▶ `ulong a = 10;`
- ▶ `ulong b = -10;`



- ▶ `float`: хранит число с плавающей точкой от $-3.4 \cdot 10^{38}$ до $3.4 \cdot 10^{38}$ и занимает 4 байта. 7 символов
- ▶ `double`: хранит число с плавающей точкой от $\pm 5.0 \cdot 10^{-324}$ до $\pm 1.7 \cdot 10^{308}$ и занимает 8 байта.
- ▶ `char`: хранит одиночный символ в кодировке Unicode и занимает 2 байта. Представлен системным типом `System.Char`. Этому типу соответствуют символьные литералы:
 - ▶ `1 char a = 'A';`



- ▶ `string`: хранит набор символов Unicode.
- ▶ `string n = "Строка";`
- ▶ `object`: может хранить значение любого типа данных и занимает 4 байта на 32-разрядной платформе и 8 байт на 64-разрядной платформе.
- ▶ `object a = 22;`
- ▶ `object b = 3.14;`
- ▶ `object c = "hello code";`