

### Базовые типы данных

bool	Логическая переменная. Значения 0 или 1 (истина/true или ложь/false).
int	Целое число. Значения от -2 147 483 648 до 2 147 483 647.
long	Целое число. Как int но больше. Значения от -9 223 372 036 854 775 808 до 9 223 372 036 854 775 807.
double	Дробное число. Хранит числа с плавающей точкой (3.14). Значения от $-3.4^{324}$ до $3.4^{324}$ .
char	Символ. Хранит единственный символ.
string	Строка. хранит текстовые данные. По факту string это набор символов (char-ов).

Типов данных много но самые ходовые это int (для целых чисел), double (для дробей), char (для символов), string (для строк) и bool (для логических значений).

### Функции

```
// главная программа
void main() {
    // вызов функции
    string res = FullName("Alex", "Man");
}
// функция
string FullName(string
firstName, string lastName) {
    string fullName = firstName
+ " " + lastName;
    return fullName;
}
/////////
```

Для объявления функции нужно указать *тип возвращаемого значения* (то что вернётся в **return**), потом *имя функции*, в круглых скобках идёт перечень параметров функции - перечисляются тип и имя параметра (string name, int age)

Из примера:  
string - тип функции. Он означает что функция должна вернуть строку  
FullName - имя функции. При помощи имени идёт обращение к функции (см. 4ю строку) (string firstName, string lastName) - список параметров. Они используются в теле функции.

### Создание переменной

```
void main() {
    // целое число
    int number = 50;
    // дробь
    double fractional = 4.21;
    // символ
    char literal = 'K';
    // строка
    string text = "Hello world!";
    // логическое значение
    bool logic = false;
}
```

тип название = значение;

### Создание класса

```
class Dog {
    // свойства класса
    private string name;
    private int age;
    // конструктор
    public Dog(string name, int age) {
        this->name = name;
        this->age =age;
    }
    // методы
    public string GetName() {
        return this->name;
    }
    // методы
    public int GetAge() {
        return this->age;
    }
}
```



### Дополнительные типы данных

byte	Целое число. Хранит значения от 0 до 255.
sbyte	Целое число. Диапазон тот же (те же 256 значений) но есть отрицательные значения. Диапазон от -128 до 127.
short	Целое число. Урезанный int. Диапазон от -32 768 до 32 767.
ushort	Целое число. Урезанный int + нет отрицательных значений (на это указывает префикс <b>u</b> ). Значения от 0 до 65 535.
uint	Целое число. Обычный int + нет отрицательных значений. Значения от 0 до 4 294 967 295.
ulong	Целое число. Как int но больше + нет отрицательных значений. Диапазон от 0 до 18 446 744 073 709 551 615.

### Дополнительные типы данных (cont)

float	Дробное число. Хранит числа с плавающей точкой + урезанный double. Диапазон от $-3.14^{38}$ до $3.14^{38}$ .
decimal	Дробное число. Хранит дроби как десятичные числа (5/10).
object	Универсальный тип. Может хранить значения любого типа.
var	Универсальный тип. Может хранить значения любого типа + тип вычисляется в момент компиляции в зависимости от того какое значение присваивается переменной.

C

By **Freeman**

[cheatography.com/freeman/](https://cheatography.com/freeman/)

Not published yet.

Last updated 5th April, 2020.

Page 2 of 2.

Sponsored by **ApolloPad.com**

Everyone has a novel in them. Finish Yours!

<https://apollopad.com>