

C# Introduction. Урок 1

Знакомство с языком C#



Структура программы



```
class имя_класса{  
    //Описание класса  
}
```

Структура программы



```
static void Main(){  
    //Код программы  
}
```

Структура программы



```
class имя_класса{  
    static void Main(){  
        //Код программы  
    }  
}
```

Первая программа



```
class HelloWorld{  
    static void Main(){  
        System.Console.WriteLine("Изучаем язык  
C#")  
    }  
}
```

Использование среды разработки



Создание проекта

Последние шаблоны проектов

- Консольное приложение (.NET Framework) C#
- Веб-приложение ASP.NET Core C#
- Общий проект C#
- Пустое приложение (универсальное приложение для Windows) C#
- Приложение Windows Forms IronPython Python
- Мастер классических приложений Windows C++
- Пустое веб-приложение Node.js JavaScript
- Простое приложение Node.js Express 4 JavaScript

С# Все платформы Все типы проектов

Облачная служба Azure (классическая)
Проект для создания масштабируемой службы, выполняющейся в Microsoft Azure.
C# Azure Облако

Группа ресурсов Azure
Этот шаблон создает проект развертывания для группы ресурсов Azure. Проект развертывания содержит артефакты, необходимые для подготовки ресурсов Azure с помощью диспетчера Azure Resource Manager, который создает среду для вашего приложения.
C# Azure Облако

Консольное приложение (.NET Framework)
Проект приложения для командной строки
C# Windows Консоль

Веб-приложение ASP.NET (.NET Framework)
Шаблоны проекта для создания приложений ASP.NET. Можно создавать приложения веб-форм, MVC или веб-API ASP.NET, а также добавлять множество других функций ASP.NET.
C# Windows Облако Веб

Библиотека классов (.NET Framework)
Проект для создания библиотеки классов C# (.dll)

Назад Далее

Использование среды разработки



Настроить новый проект

Консольное приложение (.NET Framework) C# Windows Консоль

Имя проекта

Расположение

 ...

Имя решения ⓘ

Поместить решение и проект в одном каталоге

Платформа

Назад Создать

Использование среды разработки



Скриншот среды разработки Visual Studio, демонстрирующий процесс отладки проекта MyFirstProject. В центре экрана открыт файл Program.cs с кодом на C#. Меню отладки (Отладка) открыто, и выделены пункты "Начать отладку" (F5) и "Запуск без отладки" (Ctrl+F5). В правой части экрана виден "Обозреватель решений" (Solution Explorer) с деревом проекта, включающим файлы Properties, Ссылки, App.config и Program.cs. В нижней части экрана открыто окно "Вывод" (Output), которое отображает сообщения о загрузке и выполнении программы.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace MyFirstProject
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
        }
    }
}
```

Вывод:

```
Показать выходные данные из: Отладка
"MyFirstProject.exe" (CLR v4.0.30319: DefaultDomain). Загружено "C:\Windows\Microsoft.Net\assembly\GAC_32\mscorlib\v4.0.4.0.0_b77a5c561934e089\mscorlib.dll". Загрузка символов пропущена. Модуль оптимизирован, включен параметр отладчика "Только мой код".
"MyFirstProject.exe" (CLR v4.0.30319: DefaultDomain). Загружено "C:\Users\User\source\repos\MyFirstProject\MyFirstProject\bin\Debug\MyFirstProject.exe". Символы загружены.
Программа "[13912] MyFirstProject.exe" завершилась с кодом 0 (0x0).
```


Использование среды разработки



```
C:\Windows\system32\cmd.exe
Изучаем язык C#
Для продолжения нажмите любую клавишу . . .
```

Пространство имен



```
using System;
```

```
class HelloWorld{  
    static void Main(){  
        System.Console.WriteLine("Изучаем язык C#")  
    }  
}
```

Консольный ввод



```
using System;
```

```
class InputConsoleDemo{  
    static void Main(){  
        string name;  
        Console.Title="Давайте знакомится";  
        Console.Write("Как вас зовут?");  
        name=Console.ReadLine();  
        string txt="Очень приятно, "+name+"!";  
        Console.Title="Знакомство состоялось";  
        Console.WriteLine(txt)  
    }  
}
```

Считывание чисел



```
using System;
using Microsoft.VisualBasic;
using System.Windows.Forms;

class EnteringInteger{
    static void Main(){
        string res, txt;
        int year=2019, age, born;
        Console.WriteLine("В каком году вы родились?", "Год рождения");
        res= Console.ReadLine();
        born=Int32.Parse(res);
        age=year-born;
        txt="Тогда вам "+age+" лет";
        Console.WriteLine(txt,"Возраст");
    }
}
```

Форматированный ввод



`Console.WriteLine()`

`Console.Write()`

Форматированный ввод



```
Console.WriteLine(«Значения {0}, {1} и {2}», A, B, C)
```

```
// Значения {0}, {1} и {2}
```

Форматированный ввод



{индекс, ширина: формат}

Форматированный ввод



{0, 20: #,###}

Форматированный ввод



X – шестнадцатеричный формат

E – экспоненциальный формат

N – десятичный формат

S – денежный формат