

# Платформа .NET и язык программирования C#

Основные типы данных  
и конструкции

# Платформа .NET

.NET появилась в 2002 г. как инициативный проект фирмы Microsoft, включающий в себя комплекс технологий, программных средств, стандартов и средств разработки, направленный на обеспечение создания единого информационного пространства в Интернете и соединяющий или согласующий между собой современную вычислительную технику и программное обеспечение.

Платформа .NET — это совокупность программных средств, обеспечивающих разработку приложений на основе промежуточного кода и их выполнение в специализированной операционной среде .NET Framework.

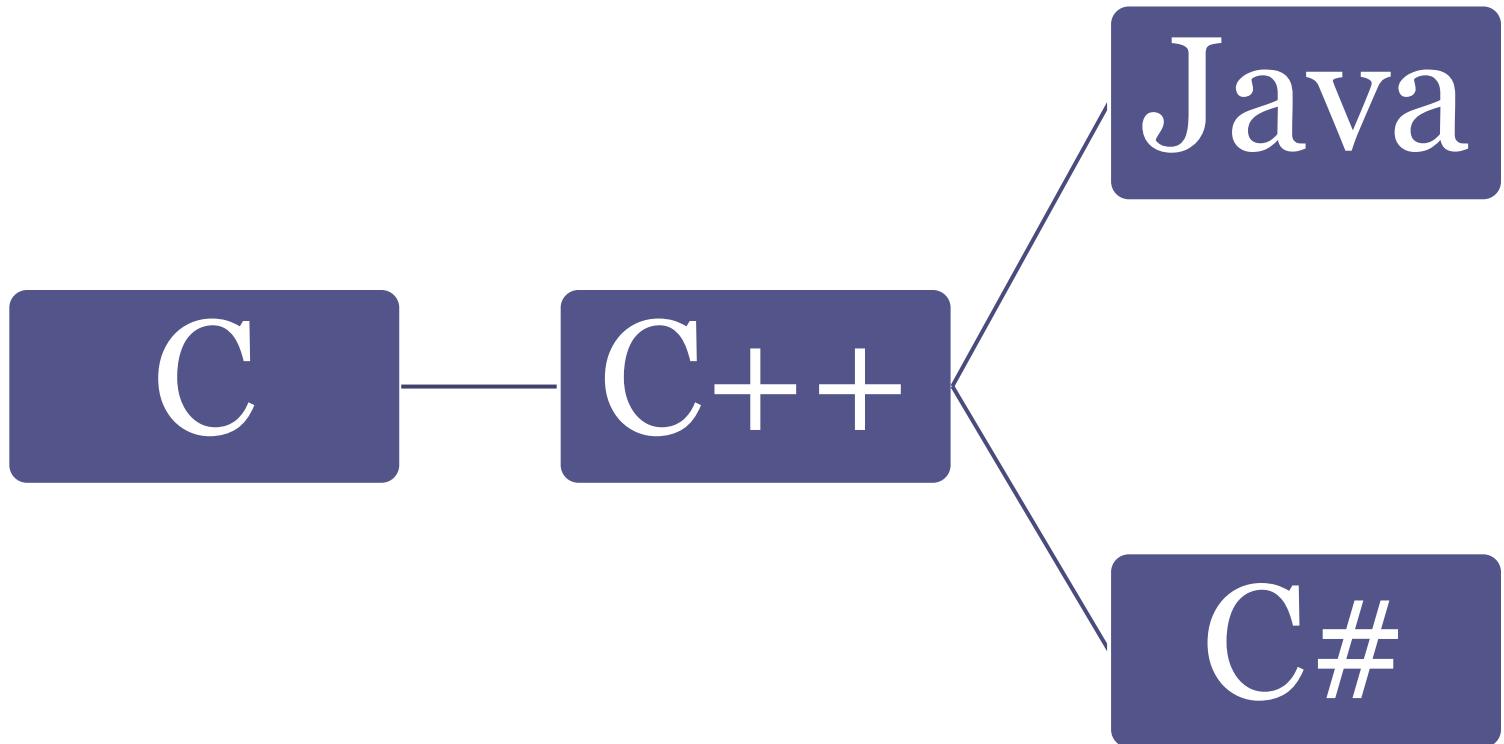
# Происхождение C#

Язык C# был разработан корпорацией Microsoft в конце 90-х годов как часть общей стратегии .NET. Впервые он был выпущен в виде альфа-версии в середине 2000 года.

Главным разработчиком C# был Андерс Хейльсберг — один из ведущих в мире специалистов по языкам программирования.

C# спроектирован и разработан специально для применения с .NET Framework.

Язык C# непосредственно связан с C, C++ и Java.



# .NET Framework

**Назначение .NET Framework** – служить средой для поддержки разработки и выполнения сильно распределенных компонентных приложений.

.NET Framework обеспечивает совместное использование разных языков программирования, а также безопасность, переносимость программ и общую модель программирования для платформы Windows.

## **Базовые функциональные возможности платформы .NET:**

- обеспечение взаимодействия с существующим программным кодом
- поддержка многочисленных языков программирования
- полная интеграция языков программирования
- усовершенствованная поддержка для создания динамических веб-страниц (ASP.NET)
- эффективный доступ к данным (ADO.NET)
- Универсальная среда разработки Visual Studio

# Структура платформы .NET

.NET состоит из пяти компонентов:

Среда разработки Visual Studio .NET

.NET Enterprise  
Servers

.NET Framework  
.NET Core

.NET Building Block  
Services

Операционная система (Windows для .NET Framework)

# Структура .NET Framework

Web Services

Web Forms

Windows Forms

Классы для работы с данными и XML  
ADO .NET, SQL, XML и т. д.

Базовые классы среды  
Ввод/вывод, сеть, безопасность, строки, коллекции и т. д.

Common Language Runtime (CLR)  
Отладка, исключения, проверка типов, JIT-компиляторы

# Пространство имен .NET

Платформа .NET основана на использовании классов. Чтобы избежать пересечения имен классов .NET использует пространство имен.

**Пространство имен** — метод, который применяется для создания иерархической структуры всех классов с целью избежать ситуации пересечения имен классов.

Допускается существование разных классов с одинаковыми именами, но только *в разных пространствах имен*.

Часто используемые пространства имен:

Категория	Имя	Назначение
Базовое пространство имен .NET	System	Классы для обработка типов данных, исключительных ситуаций, сборки мусора и т. д.

# Пространство имен .NET

Категория	Имя	Назначение
Обычные задачи	System.Collections System.Collections.Generic System.IO System.Text	Классы, поддерживающие коллекции объектов (списки, словари и др.) Классы для чтения и записи данных из потоков и файлов Классы для работы с текстом и строками
Графический интерфейс (GUI)	System.Drawing System.Windows.Forms	Классы для доступа к GDI (Graphic Dynamic Interface) и рисования двумерных картинок Классы для создания приложений Windows, которые используют интерфейс пользователя в стиле Microsoft Windows

# Пространство имен .NET

Категория	Имя	Назначение
Данные	System.Data	Классы для доступа к базам данных и источникам данных (ADO .NET)
	System.XML	Классы для поддержки XML
Сервисы среды	System.Diagnostics	Классы для отладки приложений и отслеживания выполнения программ
	System.Timers	Классы для обработки событий таймера
Сеть	System.Net	Классы для обмена данными по компьютерным сетям с использованием различных протоколов

# Основные типы данных

- 1. Числовые типы
- 2. Логический
- 3. Символьный
- 4. Строковый
- 5. Дата-время
- 6. Массив
- 7. Структура
- 8. Класс

# Типы данных в C#

Тип	Описание	Диапазон	Тип в CTS
<b>bool</b>	Логический	true или false	System.Boolean
<b>int</b>	4-байтовое целое число со знаком	-2 147 483 648 ... 2 147 483 647	System.Int32
<b>uint</b>	4-байтовое целое число без знака	0...4 294 967 295	System.UInt32
<b>byte</b>	1-байтовое число	0 ... 255	System.Byte
<b>short</b>	2-байтовое целое число со знаком	-32 768 ... 32 767	System.Int16
<b>ushort</b>	2-байтовое целое число без знака	0 ... 65 535	System.UInt16
<b>long</b>	8-байтовое целое число со знаком	Примерно от $-9 \times 10^{18}$ до $9 \times 10^{18}$	System.Int64
<b>ulong</b>	8-байтовое целое число без знака	От 0 до примерно $18 \times 10^{18}$	System.UInt64

# Типы данных в C#

Тип	Описание	Диапазон	Тип в CTS
<b>double</b>	8-байтовое вещественное число	Отрицательные: примерно от $-1.79 \cdot 10^{308}$ до $-4.94 \cdot 10^{-324}$ Положительные: примерно от $4.94 \cdot 10^{-324}$ до $1.79 \cdot 10^{308}$	System.Double
<b>float</b>	4-байтовое вещественное число	Отрицательные: примерно от $-3.40 \cdot 10^{38}$ до $-1.40 \cdot 10^{-45}$ Положительные: примерно от $1.40 \cdot 10^{-45}$ до $3.40 \cdot 10^{38}$	System.Single
<b>decimal</b>	16-байтовое десятичное число ( $B = 10$ ), 28 знаков после запятой	Отрицательные: примерно от $-7.92 \cdot 10^{28}$ до $-1.0 \cdot 10^{-28}$ Положительные: примерно от $1.0 \cdot 10^{-28}$ до $7.92 \cdot 10^{28}$	System.Decimal
<b>char</b>	Символ в формате Unicode	0...65 535	System.Char
<b>string</b>	Строка	Примерно до 2 млн. символов	System.String
<b>object</b>	Объект	Любой объект	System.Object