

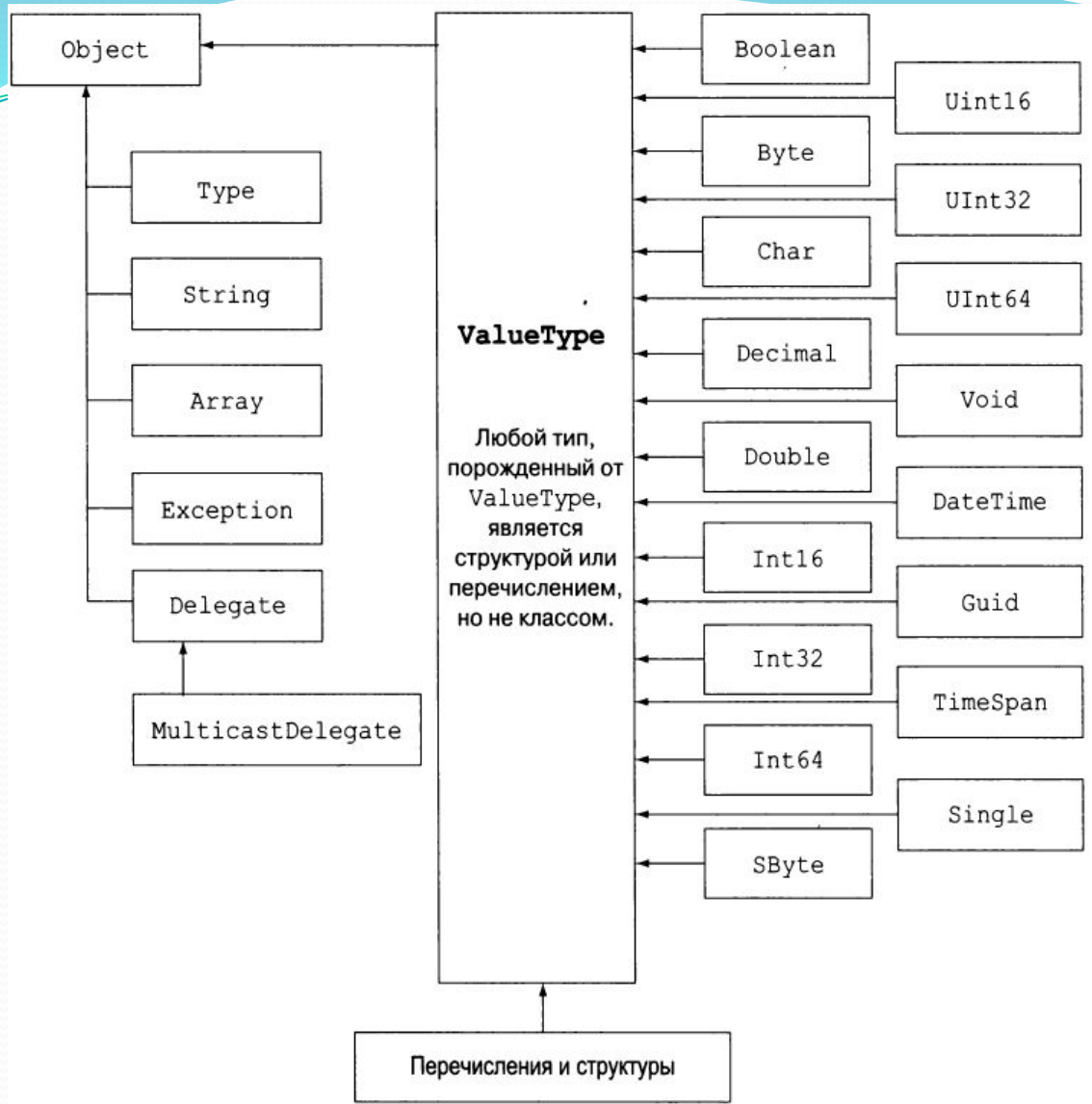
C#. Типы данных. Главные конструкции языка.

Типы данных

- Структурные
- Ссылочные

Внутренние типы

Сокращенное обозначение в C#	Совместимость с CLS	Системный тип	Диапазон	Описание
bool	Да	System.Boolean	true или false	Признак истинности или ложности
sbyte	Нет	System.SByte	от -128 до 127	8-битное число со знаком
byte	Да	System.Byte	от 0 до 255	8-битное число без знака
short	Да	System.Int16	от -32768 до 32767	16-битное число со знаком
ushort	Нет	System.UInt16	от 0 до 65535	16-битное число без знака
int	Да	System.Int32	от -2147483648 до 2147483647	32-битное число со знаком
uint	Нет	System.UInt32	от 0 до 4294967295	32-битное число без знака
long	Да	System.Int64	от -9223372036854775808 до 9223372036854775807	64-битное число со знаком
ulong	Нет	System.UInt64	от 0 до 18446744073709551615	64-битное число без знака
char	Да	System.Char	от U+0000 до U+ffff	Одиночный 16-битный символ Unicode
float	Да	System.Single	от -3.4×10^{38} до $+3.4 \times 10^{38}$	32-битное число с плавающей точкой
double	Да	System.Double	от $\pm 5.0 \times 10^{-324}$ до $\pm 1.7 \times 10^{308}$	64-битное число с плавающей точкой
decimal	Да	System.Decimal	(от -7.9×10^{28} до 7.9×10^{28}) / (10 от 0 до 28)	128-битное число со знаком
string	Да	System.String	Ограничен объемом системной памяти	Набор символов Unicode
object	Да	System.Object	В переменной object может храниться любой тип данных	Базовый класс для всех типов в мире .NET



Иерархия ТИПОВ данных

Рис. 3.2. Иерархия классов для системных типов

Строковый тип

Таблица 3.5. Избранные члены `System.String`

Член <code>System.String</code>	Описание
<code>Length</code>	Свойство, которое возвращает длину текущей строки
<code>Compare()</code>	Статический метод, который позволяет сравнить две строки
<code>Contains()</code>	Метод, который позволяет определить, содержится ли в строке определенная подстрока
<code>Equals()</code>	Метод, который позволяет проверить, содержатся ли в двух строковых объектах идентичные символьные данные
<code>Format()</code>	Статический метод, позволяющий сформатировать строку с использованием других элементарных типов данных (например, числовых данных или других строк) и нотации <code>{0}</code> , которая рассматривалась ранее в этой главе
<code>Insert()</code>	Метод, который позволяет вставить строку внутрь другой определенной строки
<code>PadLeft()</code> <code>PadRight()</code>	Методы, которые позволяют дополнить строку некоторыми символами
<code>Remove()</code> <code>Replace()</code>	Методы, которые позволяют получить копию строки с соответствующими изменениями (удалением или заменой символов)
<code>Split()</code>	Метод, который возвращает массив <code>string</code> , содержащий подстроки в этом экземпляре, которые разделяются элементами из указанного массива <code>char</code> или <code>string</code>
<code>Trim()</code>	Метод, который удаляет все вхождения набора указанных символов с начала и конца текущей строки
<code>ToUpper()</code> <code>ToLower()</code>	Методы, которые создают копию текущей строки в верхнем или нижнем регистре

Управляющие последовательности

Таблица 3.6. Управляющие последовательности в строковых литералах

Управляющая последовательность	Описание
\'	Вставляет в строковый литерал символ одинарной кавычки
\"	Вставляет в строковый литерал символ двойной кавычки
\\	Вставляет в строковый литерал символ обратной косой черты. Особенно полезна при определении путей к файлам и сетевым ресурсам
\a	Заставляет систему выдавать звуковой сигнал, который в консольных приложениях может служить аудио-подсказкой пользователю
\n	Вставляет символ новой строки (на платформах Windows)
\r	Вставляет символ возврата каретки
\t	Вставляет в строковый литерал символ горизонтальной табуляции

Приведение типов

- Неявное (Расширяющие преобразования)
- Явное (Сужающие преобразования)

Неявная типизация локальных переменных

- Ключевое слово `var`
- Применима только к локальным переменным
- Обязательно должно быть присвоено начальное значение (присвоение значения `Null` не допускается)

Операторы цикла

- for
- foreach
- while
- do/while

Конструкторы принятия решений

- If/else
- switch

Операции равенства и сравнения

Таблица 3.7. Операции равенства и сравнения в C#

Операция равенства/сравнения	Пример использования	Описание
==	<code>if (age == 30)</code>	Возвращает <code>true</code> , только если выражения одинаковы
!=	<code>if ("Foo" != myStr)</code>	Возвращает <code>true</code> , только если выражения разные
<	<code>if (bonus < 2000)</code>	Возвращает <code>true</code> , если выражение слева (<code>bonus</code>) меньше, больше, меньше или равно либо больше или равно выражению справа (<code>2000</code>)
>	<code>if (bonus > 2000)</code>	
<=	<code>if (bonus <= 2000)</code>	
>=	<code>if (bonus >= 2000)</code>	

Условные операции

Таблица 3.8. Условные операции C#

Операция	Пример	Описание
&&	<code>if (age == 30 && name == "Fred")</code>	Операция "И". Возвращает true, если все выражения дают true
	<code>if (age == 30 name == "Fred")</code>	Операция "ИЛИ". Возвращает true, если хотя бы одно из выражений дает true
!	<code>if (!myBool)</code>	Операция "НЕ". Возвращает true, если выражение дает false, или false, если выражение дает true



Методы