Методы класса Char

GetNumericValue	Возвращает числовое значение символа, если он является цифрой, и -1 в противном случае.
GetUnicodeCategory	Возвращает категорию Unicode-символа. В Unicode символы разделены на категории, например цифры (DecimalDigitNumber), римские цифры (LetterNumber), разделители строк (LineSeparator), буквы в нижнем регистре (LowercaseLetter) и т.д.
IsControl	Возвращает true, если символ является управляющим.
IsDigit	Возвращает true, если символ является десятичной цифрой.
IsLetter	Возвращает true, если символ является буквой.
IsLetterOrDigit	Возвращает true, если символ является буквой или десятичной цифрой.
IsLower	Возвращает true, если символ задан в нижнем регистре.
IsNumber	Возвращает true, если символ является числом (десятичным или шестнадцатеричным).
IsPunctuation	Возвращает true, если символ является знаком препинания.
IsSeparator	Возвращает true, если символ является разделителем.
IsUpper	Возвращает true, если символ задан в верхнем регистре.
IsWhiteSpace	Возвращает true, если символ является пробельным (пробел, перевод строки, возврат каретки).
Parse	Преобразует строку в символ (строка должна состоять из одного символа).
ToLower	Преобразует символ в нижний регистр
ToUpper	Преобразует символ в верхний регистр

Работа со строками в C#

Length	Позволяет получить количество символов в строке.
Concat()	Позволяет соединить несколько строк или переменных типа object.
CompareTo()	Позволяет сравнить две строки. В случае равенства строк результат выполнения функции равен нулю. При положительном значении функции большей является строка, для которой вызывался метод.
Copy()	Создает новую копию существующей строки.
Format()	Применяется для форматирования строки с использованием различных примитивов (строк и числовых данных) и подстановочных выражений вида {0}.
Insert()	Позволяет вставить одну строку внутрь существующей.
Remove() **Replace()**	Удаляют или заменяют символы в строке.
ToUpper() **ToLower()**	Преобразуют все символы строки в строчные или прописные.
Chars	Позволяет получить символ, находящийся в определенной позиции строки.
Join()	Создает строку, соединяя заданные строки и разделяя их строкой-разделителем.
Replace()	Заменяет один символ строки другим.
Split()	Возвращает массив строк с элементами - подстроками основной строки, между которыми находятся символы-разделители.
Substring()	Позволяет получить подстроку основной строки, начинающуюся с определенного символа и имеющую заданную длину.
Trim()	Удаляет пробелы либо набор заданных символов в начале и конце основной строки.
ToCharArray()	Создает массив символов и помещает в него символы исходной строки.

Использование класса System.Text.StringBuilder

При работе со строками в C# необходимо учитывать то, что строки являются неизменяемыми. Все действия, направленные на изменение строк, на самом деле не изменяют исходный ее вариант. Они лишь возвращают измененную копию строки.

C# содержит специальный класс StringBuilder, используя который можно избежать создания копий строк при их обработке.

Append	Добавление заданной строки в конец строки объекта.
AppendFormat	Добавление заданной форматированной строки (строки, содержащей управляющие символы) в конец строки объекта.
CopyTo	Копирование символов заданного сегмента строки в заданные ячейки массива символов.
Insert	Добавление строки в заданную позицию строки объекта.
Remove	Удаление заданного количества символов из строки объекта
Replace	Замена заданного символа либо строки объекта на другой заданный символ либо строку.