

Форматирование строк

1. Представление чисел

Для представления чисел используются следующие описатели формата:

Описатель формата	Описание
"C" или "c"	Представление валюты.
"D" или "d"	Представление целого числа.
"E" или "e"	Представление числа в экспоненциальном виде.
"F" или "f"	Представление числа в формате с плавающей точкой.
"P" или "p"	Представление процентов, выводит число умноженное на 100 со знаком процента.
"X" или "x"	Шестнадцатеричное представление.
"0"	Заместитель нуля.
"#"	Заместитель цифры.
"."	Разделитель целой и дробной части.

Примеры использования описателей формата для чисел:

```
Console.WriteLine("C symbol: {0:C}", 123); // 123,00 Р
Console.WriteLine("D symbol: {0:D5}", 123); // 00123
Console.WriteLine("E symbol: {0:E}", 123456789); // 1,234568E+008
Console.WriteLine("F symbol: {0:F2}", 123.4567); // 123,46
Console.WriteLine("P symbol: {0:P}", 0.123); // 123,46
Console.WriteLine("X symbol: 0x{0:X}", 567); // 0x237
Console.WriteLine("0 symbol: {0:000.00}", 12.6789); // 012,68
Console.WriteLine("# symbol: {0:##}", 14.6789); // 15
```

2. Представление даты и времени

Для представления даты и времени используются следующие описатели формата:

Описатель формата	Описание
"d"	Сокращенный формат даты
"D"	Полный формат даты
"f", "F"	Полный формат даты и времени с коротким (полным) форматом времени

"g", "G"	Общий формат даты и времени с коротким (полным) форматом времени
"t"	Короткий формат времени
"T"	Полный формат времени
"M", "m"	Шаблон дней месяца.
"Y", "y"	Шаблон месяца года.

Примеры использования описателей формата для даты и времени:

```
Console.WriteLine("d symbol: {0:d}", DateTime.Now);
Console.WriteLine("D symbol: {0:D}", DateTime.Now);
Console.WriteLine("f symbol: {0:f}", DateTime.Now);
Console.WriteLine("F symbol: {0:F}", DateTime.Now);
Console.WriteLine("g symbol: {0:g}", DateTime.Now);
Console.WriteLine("G symbol: {0:G}", DateTime.Now);
Console.WriteLine("t symbol: {0:t}", DateTime.Now);
Console.WriteLine("T symbol: {0:T}", DateTime.Now);
Console.WriteLine("{0:yyyy-MM-dd}", DateTime.Now);
Console.WriteLine("{0:dd/MM/yy}", DateTime.Now);
Console.WriteLine("{0:dd/MM/yy HH:mm:ss}", DateTime.Now);
```

3. \$ — интерполяция строк

Интерполированная строка содержит специальные выражения интерполяции, они похожи на элементы форматирования. Выражения интерполяции имеют следующий вид:

{<interpolationExpression>[,<alignment>][:<formatString>]}

где *interpolationExpression* – элемент, значение, которого будет интегрироваться в строку;

alignment – выравнивание;

formatString – формат (см. форматирование строк).

Примеры работы с интерполированной строкой:

```
int n1 = 45678;
double d1 = 123.34567;
bool b1 = true;
string sv = "test";
Console.WriteLine($"int val: {n1}, double val: {d1:#.###}");
Console.WriteLine($"bool val: {b1}, string val: {sv}");
```

4. Управляющие символы (литералы)

Управляющие символы позволяют вводить в текст команды управления кареткой и символы, которые имеют специальное назначение (одинарные и двойные кавычки). Ниже представлена таблица с управляющими символами.

Управляющий символ	Описание
<code>\a</code>	Звуковой сигнал
<code>\b</code>	Возврат на одну позицию
<code>\f</code>	Перевод страницы
<code>\n</code>	Новая строка
<code>\r</code>	Возврат каретки
<code>\t</code>	Горизонтальная табуляция
<code>\v</code>	Вертикальная табуляция
<code>\0</code>	Пустой символ
<code>\'</code>	Одинарная кавычка
<code>\"</code>	Двойная кавычка
<code>\\</code>	Обратная косая черта

Пример использования управляющих символов:

```
Console.WriteLine("\aName:\t\"John\"\\nAge:\t\"27\"");
```

@ – буквальный идентификатор

Еще один элемент, который можно использовать при создании строки – это буквальный идентификатор `@`. Если его поставить перед строкой, то она будет интерпретироваться буквально, в ней, *escape*-последовательности представляются без преобразования.

Пример работы с буквальным идентификатором:

```
Console.WriteLine(@"escape is not work: \a\t\n\x1234");
```