Форматирование строк

Важной функцией класса String является функция Format. Эта функция формирует строку из нескольких составляющих и с определенным форматированием. Синтаксис функции Format следующий:

В методе Format могут использоваться различные спецификаторы и описатели, которые позволяют настроить вывод данных. Рассмотрим основные описатели.

Все используемые форматы:

|  |  |
| --- | --- |
| C / c | Задает формат денежной единицы, указывает количество десятичных разрядов после запятой |
| D / d | Целочисленный формат, указывает минимальное количество цифр |
| E / e | Экспоненциальное представление числа, указывает количество десятичных разрядов после запятой |
| F / f | Формат дробных чисел с фиксированной точкой, указывает количество десятичных разрядов после запятой |
| G / g | Задает более короткий из двух форматов: F или E |
| N / n | Также задает формат дробных чисел с фиксированной точкой, определяет количество разрядов после запятой |
| P / p | Задает отображения знака процентов рядом с число, указывает количество десятичных разрядов после запятой |
| X / x | Шестнадцатеричный формат числа |

Форматирование валюты

Для форматирования валюты используется описатель "C":

|  |  |
| --- | --- |
| 1  2  3  4  5 | double number = 23.7;  string result = String.Format("{0:C}", number);  Console.WriteLine(result); // $ 23.7  string result2 = String.Format("{0:C2}", number);  Console.WriteLine(result2); // $ 23.70 |

Число после описателя указывает, сколько чисел будет использоваться после разделителя между целой и дробной частью. При выводе также добавляется обозначение денежного знака для текущей культуры компьютера.

Форматирование целых чисел

Для форматирования целочисленных значение применяется описатель "d":

|  |  |
| --- | --- |
| 1  2  3  4  5 | int number = 23;  string result = String.Format("{0:d}", number);  Console.WriteLine(result); // 23  string result2 = String.Format("{0:d4}", number);  Console.WriteLine(result2); // 0023 |

Число после описателя указывает, сколько цифр будет в числовом значении. Если в исходном числе цифр меньше, то к нему добавляются нули.

Форматирование дробных чисел

Для форматирования дробны чисел используется описатель F, число после которого указывает, сколько знаков будет использоваться после разделителя между целой и дробной частью. Если исходное число - целое, то к нему добавляются разделитель и нули.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11 | int number = 23;  string result = String.Format("{0:f}", number);  Console.WriteLine(result); // 23,00    double number2 = 45.08;  string result2 = String.Format("{0:f4}", number2);  Console.WriteLine(result2); // 45,0800    double number3 = 25.07;  string result3 = String.Format("{0:f1}", number3);  Console.WriteLine(result2); // 25,1 |

Формат процентов

Описатель "P" задает отображение процентов. Используемый с ним числовой спецификатор указывает, сколько знаков будет после запятой:

|  |  |
| --- | --- |
| 1  2 | decimal number = 0.15345m;  Console.WriteLine("{0:P1}", number);// 15.3% |

Настраиваемые форматы

Используя знак #, можно настроить формат вывода. Например, нам надо вывести некоторое число в формате телефона +х (ххх)ххх-хх-хх:

|  |  |
| --- | --- |
| 1  2  3 | long number = 19876543210;  string result = String.Format("{0:+# (###) ###-##-##}", number);  Console.WriteLine(result); // +1 (987) 654-32-10 |

Метод ToString

Метод ToString() не только получает строковое описание объекта, но и может осуществлять форматирование. Он поддерживает те же описатели, что используются в методе Format:

Форматирование строк

|  |  |
| --- | --- |
| Функция | Результат |
| String.Format(«{0:c}», 12.53) | 12,53 р. |
| String. Format(«{0:е}», 12.53) | 1,253000е + 001 |
| String.Format(«{0:f}», 12.53) | 12,53 |
| String. Format(«{0:00.0000}», 12.53) | 12,5300 |
| String. Format(«{0:(#). ##}», 12.53) | (12),53 |
| String. Format(«{0:0,0}», 12.53) | 13 |
| String. Format(«{0:000.0}», 12.53) | 012,5 |
| String.Format(«{0:0,0%}», 12.53) | 1253% |
| String. Fortnat(«{0:$#, ##0.00;  ($#,##0.00);Zero}», 1245.53) | $1245,53 |
| String. Format(«{ 0:$#, ##0.00; ($#,##0.00);Zero}», 1245.53) | $1245,53 |
| String. Format(«{0:d}», DateTime. Now) | 13.07.2015 |
| String.Format(«{0:t}», DateTime.Now) | 13:00 |
| String. Format(«{0:g}», DateTime. Now) | 13.07.2015 13:00 |