

# String.Format in VB.NET

Wednesday, 12 November 2008

Last Updated Tuesday, 24 November 2009

## Syntax

The syntax for String.Format is

```
String.Format("{parameter no:format character}", thingtoformat, anotherthing to format, ....)
```

where :-

parameter number is the parameter number starting at 0

format character tells .net how to format the parameter (see below for formats)

thingtoformat, anotherthing to format - parameters to be formatted

## Format Characters

### integer formats

- C or c: The number is converted to a string that represents a currency amount.
- D or d: The number is converted to a string of decimal digits (0-9), prefixed by a minus sign if the number is negative. The precision specifier indicates the minimum number of digits desired in the resulting string. If required, the number is padded with zeros to its left to produce the number of digits given by the precision specifier.
- E or e: The number is converted to a string of the form "-d.ddd...E+ddd" or "-d.ddd...e+ddd", where each 'd' indicates a digit (0-9). The string starts with a minus sign if the number is negative. One digit always precedes the decimal point. The precision specifier indicates the desired number of digits after the decimal point.
- F or f: The number is converted to a string of the form "-ddd.ddd..." where each 'd' indicates a digit (0-9).
- G or g: The number is converted to the most compact of either fixed-point or scientific notation, depending on the type of the number and whether a precision specifier is present.
- N or n: The number is converted to a string of the form "-d,ddd,ddd.ddd...", where each 'd' indicates a digit (0-9).
- P or p: The number is converted to a string that represents a percent.
- X or x: The number is converted to a string of hexadecimal digits.

### date formats

- d: The value is converted to the short date pattern.
- D: The value is converted to the long date pattern.
- t: The value is converted to the short time pattern.
- T: The value is converted to the long time pattern.
- f: The value is converted to the full date-time pattern, including short time.
- F: The value is converted to the full date-time pattern, including long time.
- g: The value is converted to the short date and short time.
- G: The value is converted to the long date and long time.
- M or m: The value is converted to the month day pattern.

## Examples

String.Format("Todays date is {0:D}",DateTime.now()) would return Todays date is January 25, 2008

String.Format("{0:c}", 10) would return £10.00

String.Format("\${0:0.00}"), 10.99) would return \$10.99

## Acknowledgements

Thanks to builderau.com for this gem - you can see the full article at

<http://www.builderau.com.au/program/dotnet/soa/Easily-format-string-output-with-String->

Format/0,339028399,339177160,00.htm