

# Chapter 10: Verbatim Strings

## Section 10.1: Interpolated Verbatim Strings

Verbatim strings can be combined with the new String interpolation features found in C#6.

```
Console.WriteLine($"@\"Testing \n 1 2 {5 - 2}  
New line\"");
```

### Output:

```
Testing \n 1 2 3  
New line
```

[Live Demo on .NET Fiddle](#)

As expected from a verbatim string, the backslashes are ignored as escape characters. And as expected from an interpolated string, any expression inside curly braces is evaluated before being inserted into the string at that position.

## Section 10.2: Escaping Double Quotes

Double Quotes inside verbatim strings can be escaped by using 2 sequential double quotes "" to represent one double quote " in the resulting string.

```
var str = @"""I don't think so,"" he said."";  
Console.WriteLine(str);
```

### Output:

```
"I don't think so," he said.
```

[Live Demo on .NET Fiddle](#)

## Section 10.3: Verbatim strings instruct the compiler to not use character escapes

In a normal string, the backslash character is the escape character, which instructs the compiler to look at the next character(s) to determine the actual character in the string. (Full list of character escapes)

In verbatim strings, there are no character escapes (except for "" which is turned into a "). To use a verbatim string, just prepend a @ before the starting quotes.

This verbatim string

```
var filename = @"c:\temp\newfile.txt"
```

### Output:

```
c:\temp\newfile.txt
```

As opposed to using an ordinary (non-verbatim) string:

```
var filename = "c:\temp\newfile.txt"
```

that will output:

```
c:      emp
ewfile.txt
```

using character escaping. (The `\t` is replaced with a tab character and the `\n` is replaced with a newline.)

[Live Demo on .NET Fiddle](#)

## Section 10.4: Multiline Strings

```
var multiline = @"This is a
multiline paragraph";
```

**Output:**

```
This is a
multiline paragraph
```

[Live Demo on .NET Fiddle](#)

Multi-line strings that contain double-quotes can also be escaped just as they were on a single line, because they are verbatim strings.

```
var multilineWithDoubleQuotes = @"I went to a city named
    ""San Diego""
    during summer vacation.";
```

[Live Demo on .NET Fiddle](#)

*It should be noted that the spaces/tabulations at the start of lines 2 and 3 here are actually present in the value of the variable; check [this question](#) for possible solutions.*