



### Interaction

## Tools

## Languages

 Edit links

Article | **Talk**

Read Edit View history

Search Wikipedia



From Wikipedia, the free encyclopedia



## Contents

- ### Definition [\[edit\]](#)

Control character [\[edit\]](#)

In many programming languages, an escape character also forms some escape sequences which are referred to control characters. For example, `line break` has an escape sequence of `\n`.

## Examples [\[edit\]](#)

### JavaScript [\[edit\]](#)

JavaScript uses the `\` (backslash) as an escape character for:<sup>[1][2]</sup>

- `'` single quote
- `"` double quote
- `\\` backslash
- `\n` new line
- `\r` carriage return
- `\t` tab
- `\b` backspace
- `\f` form feed
- `\v` vertical tab (IE < 9 treats `'\v'` as `'v'` instead of a vertical tab (`'\x0B'`). If cross-browser compatibility is a concern, use `\x0B` instead of `\v`.)
- `\0` null character (U+0000 NULL) (only if the next character is not a decimal digit; else it is an octal escape sequence)

Note that the `\v` and `\0` escapes are not allowed in JSON strings.

### ASCII escape character [\[edit\]](#)

The ASCII "escape" character (**octal**: `\033`, **hexadecimal**: `\x1B`, or `^["`, or, in decimal, 27) is used in many output devices to start a series of characters called a control sequence or escape sequence. Typically, the escape character was sent first in such a sequence to alert the device that the following characters were to be interpreted as a control sequence rather than as plain characters, then one or more characters would follow to specify some detailed action, after which the device would go back to interpreting characters normally. For example, the sequence of `^["`, followed by the printable characters `[2;10H`, would cause a **DEC VT102** terminal to move its **cursor** to the 10th cell of the 2nd line of the screen. This was later developed to **ANSI escape codes** covered by the **ANSI X3.64** standard. The escape character also starts each command sequence in the **Hewlett Packard Printer Command Language**.

Early reference to the term "escape character" is found in **Bob Bemer**'s IBM technical publications, who is credited with inventing this mechanism during his work on the **ASCII** character set.<sup>[3]</sup>

The **Escape key** is usually found on standard PC keyboards. However it is commonly absent from keyboards for PDAs and other devices not designed primarily for ASCII communications. The **DEC VT220** series was one of the few popular keyboards that did not have a dedicated Esc key, instead using one of the keys above the main keypad. In **user interfaces** of the 1970s–1980s it was not uncommon to use this key as an escape character, but in modern desktop computers such use is dropped. Sometimes the key was identified with **AltMode** (for alternative mode). Even with no dedicated key, the escape character code could be generated by typing '[' while simultaneously holding down the **Control key**, 'Ctrl'.

### Programming and data formats [\[edit\]](#)

Many modern **programming languages** specify the doublequote character (`"`) as a **delimiter** for a **string literal**. The **backslash** (`\`) escape character typically provides two ways to include doublequotes inside a string literal, either by modifying the meaning of the doublequote character embedded in the string (`\"` becomes `"`), or by modifying the meaning of a sequence of characters including the hexadecimal value of a doublequote character (`\x22` becomes `"`).

**C**, **C++**, **Java**, and **Ruby** all allow exactly the same two backslash escape styles. The **PostScript** language and Microsoft **Rich Text Format** also use backslash escapes. The **quoted-printable** encoding uses the **equals sign** as an escape character.

**URL** and **URI** use **%-escapes** to quote characters with a special meaning, as for non-ASCII characters. The **ampersand** (`&`) character may be considered as an escape character in **SGML** and derived formats such as **HTML** and **XML**.

Some programming languages also provide other ways to represent special characters in literals, without requiring an escape character (see e.g. **delimiter collision**).

### Communication protocols [\[edit\]](#)

The **Point-to-Point Protocol** uses the **0x7D octet** (`\175`, or ASCII: `}`) as an escape character. The octet

immediately following should be **XORed** by 0x20 before being passed to a higher level protocol. This is applied to both 0x7D itself and the control character 0x7E (which is used in PPP to mark the beginning and end of a frame) when those octets need to be transmitted by a higher level protocol encapsulated by PPP, as well as other octets negotiated when the link is established. That is, when a higher level protocol wishes to transmit 0x7D, it is transmitted as the sequence 0x7D 0x5D, and 0x7E is transmitted as 0x7D 0x5E.

## Bourne shell [\[edit\]](#)

In **Bourne shell** (sh), the **asterisk** ( `*` ) and **question mark** ( `?` ) characters are **wildcard characters** expanded via **globbing**. Without a preceding escape character, an `*` will expand to the names of all files in the **working directory** that do not start with a period **iff** there are such files, otherwise `*` remains unexpanded. So to refer to a file literally called `"*"`, the shell must be told not to interpret it in this way, by preceding it with a backslash ( `\` ). This modifies the interpretation of the asterisk ( `*` ). Compare:

```
rm *      # delete all files in the current directory

rm \*     # delete the file named *
```

## Windows Command Prompt [\[edit\]](#)

The **Windows command-line interpreter** uses a **caret** character ( `^` ) to escape reserved characters that have special meanings (in particular: `& | ( ) < > ^`).<sup>[4]</sup> The **DOS command-line interpreter**, though it supports similar syntax, does not support this.

For example, on the Windows Command Prompt, this will result in a syntax error.

```
echo <hello world>
```

whereas this will output the string: `<hello world>`

```
echo ^<hello world^>
```

## Windows PowerShell [\[edit\]](#)

In **Windows** backslash is used as path separator, therefore it generally can't be used as an escape character. **PowerShell** uses **backtick**<sup>[5]</sup> ( ``` ) instead.

For example, the following command:

```
echo "`tFirst line`nNew line"
```

Will output:

```
First line
New line
```

## Others [\[edit\]](#)

- **Quoted-printable**, which encodes 8-bit data into 7-bit data of limited line lengths, uses the **equals sign** `"=`" as an escape character.

## See also [\[edit\]](#)

- **Escape sequences in C**
- **Leaning toothpick syndrome**
- **Stropping (syntax)** – in some conventions a leading character (such as an apostrophe) functions as an escape character

Not to be confused with:

- [Escape sequences](#) used for communication between devices
- [Esc key](#) on a keyboard

## References [\[edit\]](#)

- ↑ "JavaScript character escape sequences" [Mathias Bynens](#). Mathiasbynens.be. Retrieved 2014-06-30.
- ↑ "Special Characters (JavaScript)" [Msdn.microsoft.com](#). 2014-06-20. Retrieved 2014-06-30.
- ↑ Bemer, Bob. "How Bob Bemer Invented the ESCAPE Sequence and Key" [www.bobbemer.com](#). Retrieved 22 March 2018.
- ↑ Tim Hill (1998). "The Windows NT Command Shell" [MacMillan Technical Publishing](#). Retrieved 2010-01-13.
- ↑ "about\_Escape\_Characters" [Msdn.microsoft.com](#). 2014-05-08. Retrieved 2016-11-24.

## External links [\[edit\]](#)

- [Escape characters in 9 languages](#)
- [That Powerful ESCAPE Character -- Key and Sequences](#) – Bob Bemer

This article incorporates [public domain material](#) from the [General Services Administration](#) document "[Federal Standard 1037C](#)".

Categories: [Pattern matching](#) | [Control characters](#)

This page was last edited on 12 April 2018, at 08:09.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Cookie statement](#) [Mobile view](#)

