One syntactic restriction not indicated by these productions is that whitespace is not allowed between the **[stringprefix](http://docs.python.org/py3k/reference/lexical_analysis.html?highlight=raw%20bytes" \l "grammar-token-stringprefix)** or **[bytesprefix](http://docs.python.org/py3k/reference/lexical_analysis.html?highlight=raw%20bytes" \l "grammar-token-bytesprefix)** and the rest of the literal. The source character set is defined by the encoding declaration; it is UTF-8 if no encoding declaration is given in the source file; see section [*Encoding declarations*](http://docs.python.org/py3k/reference/lexical_analysis.html?highlight=raw%20bytes#encodings).

In plain English: Both types of literals can be enclosed in matching single quotes (') or double quotes ("). They can also be enclosed in matching groups of three single or double quotes (these are generally referred to as *triple-quoted strings*). The backslash (\) character is used to escape characters that otherwise have a special meaning, such as newline, backslash itself, or the quote character.

Bytes literals are always prefixed with 'b' or 'B'; they produce an instance of the [**bytes**](http://docs.python.org/py3k/library/functions.html#bytes) type instead of the **[str](http://docs.python.org/py3k/library/functions.html" \l "str" \o "str)** type. They may only contain ASCII characters; bytes with a numeric value of 128 or greater must be expressed with escapes.

Both string and bytes literals may optionally be prefixed with a letter 'r' or 'R'; such strings are called *raw strings* and treat backslashes as literal characters. As a result, in string literals, '\U' and '\u' escapes in raw strings are not treated specially. When a byte literal is prefixed with both 'r' or 'R' and 'b' or 'B', the b must precede the r, as in: 'Br', 'bR', or 'BR' and not 'Rb', 'rB', or 'RB'.

In triple-quoted strings, unescaped newlines and quotes are allowed (and are retained), except that three unescaped quotes in a row terminate the string. (A “quote” is the character used to open the string, i.e. either ' or ".)

Unless an 'r' or 'R' prefix is present, escape sequences in strings are interpreted according to rules similar to those used by Standard C. The recognized escape sequences are: