

LITERALS IN JAVA

Literals are numbers, characters, and string representations in your program. Literals can be classified as numeric literals, character literals and string literals.

Numeric literals

Numeric literals consist of digits with an optional sign and an optional decimal point. Numeric literals that contain a decimal point are by default double. Those prefixed with a 0x indicate the number is a hexadecimal number (base 16). Those that begin with a 0 are octal numbers (base 8). But usually they are base 10.

Integer literals are of type int by default. A suffix of f or F can be used to declare a floating point literal as of type float. Floating point literals are numbers that contain a decimal point or those that are written using scientific notation. E.g. 7.77 is double, 7e5 is double and 0.077F is float. A literal can be specified as type long by appending an L to the end of the literal. Either a lowercase or uppercase L can be used to designate an integer as type long. However, it is better to use an uppercase L to avoid confusing the letter with the numeric digit "1" (one).

Java 7 added the ability to use underscore characters (_) in numeric literals. The use of the underscore is to make the code more readable to the developer but it is ignored by the compiler.

```
long ssn = 111_22_3333L;
```

```
System.out.println(ssn);
```

The output will be:

```
111223333
```

Consecutive underscores are treated as one and also ignored by the compiler. Also, underscores cannot be placed:

- At the beginning or end of a number
- Adjacent to a decimal point

- Prior to the D, F, or L suffix

Character literals

Character literals are single characters enclosed in single quotes.

```
char letter = 'a';
```

```
letter = 'F';
```

Escape characters (also called escape sequences or escape codes) in general are used to signal an alternative interpretation of a series of characters. In Java, a character preceded by a backslash (\) is an escape sequence and has special meaning to the java compiler. For example, '\n' represents the carriage return line feed character. These special escape sequences represent certain special values. You can use escape sequences in character literals as well as in string literals. Read more about escape sequences at

String literals

String literals are a sequence of characters that are enclosed in a set of double quotes. String literals cannot be split across two lines.

```
String message = "Hello\tWorld\n";
```

Source : <http://javajee.com/literals-in-java>