

Manipulator	Function	Example	Output
<i>endl</i>	Prints a newline and flushes output buffer	<code>cout << 5 << endl << 10 << endl;</code>	5 10
<i>setw()</i>	Assigns the width of the next item to be displayed	<code>cout << setw(4) << 5;</code> <code>cout << setw(4) << 10;</code>	5 10
<i>setprecision()</i>	Assigns the number of digits printed.	<code>cout << setprecision(2) << 3.14159;</code>	3.1
<i>scientific</i>	Prints floating-point numbers in scientific format.	<code>cout << scientific << 3.14;</code>	3.14E+000
<i>fixed</i>	Prints floating-point numbers in fixed format	<code>cout << fixed << 1.61e1;</code>	16.1
<i>showpoint</i>	Prints the decimal point for floating-point numbers even if there are no decimal digits.	<code>cout << showpoint << 1.6e1;</code>	16.
<i>left</i>	Left-justifies output	<code>cout << setw(4) << left << 16;</code>	16
<i>right</i>	Right-justifies output	<code>cout << setw(4) << right << 16;</code>	16
<i>dec</i>	Assigns integer base to be base 10	<code>cout << dec << 16;</code>	16
<i>oct</i>	Assigns integer base to be base 8	<code>cout << oct << 16;</code>	20
<i>hex</i>	Assigns integer base to be base 16	<code>cout << hex << 16;</code>	10
<i>uppercase</i>	Coerce the exponent e to E	<code>cout << scientific << setprecision(3) << uppercase << 1.602e-19;</code>	1.602E-19
<i>nouppercase</i>	Coerce the exponent E to e	<code>cout << scientific << setprecision(3) << nouppercase << 1.602E-19;</code>	1.602e-19
<i>boolalpha</i>	Prints true as true	<code>cout << boolalpha << true</code>	true
<i>noboolalpha</i>	Prints true as 1	<code>cout << noboolalpha << true</code>	1
<i>setfill()</i>	Assigns the padding character	<code>cout << setw(5) << setfill('0') << 34;</code>	00034

#include <iomanip>