It’s handy when you have two related operations which you’d like to execute as a pair, with a block of code in between. The classic example is opening a file, manipulating the file, then closing it:

with open('output.txt', 'w') as f:

f.write('Hi there!')

The above with statement will automatically close the file after the nested block of code. (Continue reading to see exactly how the close occurs.) The advantage of using a with statement is that it is guaranteed to close the file no matter how the nested block exits. If an exception occurs before the end of the block, it will close the file before the exception is caught by an outer exception handler. If the nested block were to contain a return statement, or a continue or break statement, the with statement would automatically close the file in those cases, too.