

Using the with Operator

Files in python can be handled in a simpler way using the with operator.

Python has a neat little builtin called **with** which you can use to simplify reading and writing files. The **with** operator creates what is known as a **context manager** in Python that will automatically close the file for you when you are done processing it. Let's see how this works:

```
with open("test.txt") as file_handler:  
    for line in file_handler:  
        print(line)
```



The syntax for the **with** operator is a little strange, but you'll pick it up pretty quickly. Basically what we're doing is replacing:

```
handle = open("test.txt")
```



with this:

```
with open("test.txt") as file_handler:
```



You can do all the usual file I/O operations that you would normally do as long as you are within the **with** code block. Once you leave that code block, the file handle will close and you won't be able to use it any more. Yes, you read that correctly. You no longer have to close the file handle explicitly as the **with** operator does it automatically! See if you can change some of the earlier examples from this chapter so that they use the **with** method too.

