Check Your Test With a Text File

The doctest module also allows you to write interactive examples to a text file using ReStructuredText, which is a common mark-up language. Let's create a really simple example. Save the following in the same folder as you did for our add function above and be sure to name it **add.txt**:

```
====================
The ``add`` function
====================


------------------
How to use ``add``
------------------

The ``add`` function needs to be imported from the **mymath** module like this:

    >>> from mymath import add

Now you can use ``add`` to add various data types together, such as integers, floats, strings

    >>> add(1, 2)
    3
```

The example above explains what we are doing, which is basically importing the add function from our mymath module and then testing it out. Now we need to actually write a test script to run our test file:

```
import doctest

doctest.testfile('add.txt')
```

Once you have this code in your file, save it with the name **test_mymath.py**. Now we can run this code like so:

```
python test_mymath.py -v
```

We add the **-v** flag as before so that it runs the test in verbose mode. You should end up seeing the following output if you didn’t make any typos:

```
Trying:
    from mymath import add
Expecting nothing
ok
Trying:
    add(1, 2)
Expecting:
    3
ok
1 items passed all tests:
   2 tests in add.txt
2 tests in 1 items.
2 passed and 0 failed.
Test passed.
```

As you can see, we passed all our tests!