

```
import types
import unittest

class TestFunctions(unittest.TestCase):
    def test_functions(self):
        """
        A basic introduction to functions
        """
        # Define a function called ``add_2`` that returns 2 more than
        # what is passed into it. Normally function are global to a
        # module, but they can also be defined in the scope of another
        # function. You can either put it below, or outside of the
        # TestFunctions class.
        # =====
        def add_2(num):
            return num + 2
        self.assertEqual(isinstance(add_2, types.FunctionType))
        self.assertEqual(add_2(4), 6)

        # Write a function ``add_3`` that has the following docstring:
        # "Adds 3 to the input"
        # =====
        def add_3(num):
            "Adds 3 to the input"
            return num + 3
        self.assertEqual(add_3.__doc__, 'Adds 3 to the input')

        # Default Parameters.
        # Write a function ``mul_n`` that takes one or two parameters (the second
        parameter named ``x``).
        # If it has 2 parameters it multiplies them. If it takes one
        # parameter, it multiplies it by 5
        # =====
        def mul_n(num, x=5):
            return num*x

        self.assertEqual(mul_n(5, 1), 5)
        self.assertEqual(mul_n(5), 25)
        self.assertEqual(mul_n.func_defaults, (5,))

if __name__ == '__main__':
    unittest.main()
```