

## 7 Multiple Decorators

### Multiple Decorators

Patch decorators are stackable. The Python runtime runs decorators from bottom to top. The callable signature matches the bottom up order of the decorators.

```
1 from unittest.mock import patch
2 def greeter():
3     name = input("What's your name? ")
4     print(f'Hello, {name}')
5 @patch(f'builtins.print')
6 @patch(f'builtins.input')
7 def test_stacks(input_mock, print_mock):
8     # The patch decorators pass mocked objects as input parameters
9     # Python runs decorators from the bottom up.
10    # The order of the input parameters match.
11    input_mock.return_value = 'World'
12    # Running greeter will call input and print.
13    greeter()
14    # Verify the calls
15    input_mock.assert_called_once()
16    print_mock.assert_called_with('Hello, World')
17 test_stacks()
18 #####
19 print('No assertion errors')
20
```

### Instructions

1. Arrange the **playground.py** file to match the code above.
2. Run **playground.py** in the IDEs **terminal** pane.

```
1 python3 cloudacademy/playground.py
```

No assertion errors

< Back

Start check ↻

