

Example 1 - Stub the Stubbable

```
class BatchTests(unittest.TestCase):
    def setUp(self):
        self.consumer = mock.Mock()
        self.consumer.batch_size_overhead = 0
        self.consumer.get_item_size = lambda item: len(item)
        self.consumer.batch_size_limit = 5
        self.batcher = publisher.Batcher(self.consumer)

    def test_flush_empty_does_nothing(self):
        self.batcher.flush()
        self.assertEqual(self.consumer.consume_batch.called, False)

    def test_manual_flush(self):
        self.batcher.add("a")
        self.batcher.add("b")
        self.batcher.flush()
        self.consumer.consume_batch.assert_called_once_with(["a", "b"])
```

- Use case is testing a message/event batcher
 - Does it flush? 🚽
- `self.consumer`
 - Our mock object!
 - **Prevents actual API call**
 - **Let's us test how it was called by the code!**
- ***Let's look at this. —>***

Example 1 - Stub the Stubbable

```
class BatcherTests(unittest.TestCase):
    def setUp(self):
        self.consumer = mock.Mock()
        self.consumer.batch_size_overhead = 0
        self.consumer.get_item_size = lambda item: len(item)
        self.consumer.batch_size_limit = 5
        self.batcher = publisher.Batcher(self.consumer)

    def test_flush_empty_does_nothing(self):
        self.batcher.flush()
        self.assertEqual(self.consumer.consume_batch.called, False)

    def test_manual_flush(self):
        self.batcher.add("a")
        self.batcher.add("b")
        self.batcher.flush()
        self.consumer.consume_batch.assert_called_once_with(["a", "b"])
```

- **Notice a few things:**
 - You can set attributes on Mocks
 - You can see if they Mock was called
 - You can see how many times it was called
 - And what it was called with!!!