

Testify Mock

- [How To](#)
- [Sample Test Case](#)
- [Sample Mock Class](#)

How To

We will start by creating a mock object. It must implement all the functions of the object we are trying to mock.

In the struct, we will add testify's mock object.

Add "mock.Mock" to your type:

```
import (
    "github.com/stretchr/testify/mock"
)

type MyMock struct{
    mock.Mock
    ...
}

func (m *MyMock) ListTopics() ([]string, error){
    args := m.Called()
    return args.Get(0).([]string), args.Error(1)
}
...
```

In our Test Case

```
kafkaTopics := []string{"topic1", "topic2", "topic3"}

mock := mock.NewMockClusterAdmin()
mock.On("ListTopics").Return(kafkaTopics, nil).Once()
```

By adding .Once() at the end of the mock.On call, we can call mock.On again.

Sample Test Case

```

package consumer

import (
    "errors"
    "github.com/Shopify/sarama"
    "github.com/stretchr/testify/assert"
    "kafka-azure-sink/src/internal/mock"
    "slices"
    "testing"
)

func TestSuccessAdditionOfKafkaTopics(t *testing.T) {

    baseTopics := []string{"ncyd_kubernetes", "etlflow_*"}
    kafkaTopics := []string{"ncyd_test1", "etlflow_ssh", "etlflow_nrd"}

    initialKafkaTopics := buildTopicMap(kafkaTopics)

    mock := mock.NewMockClusterAdmin()
    mock.On("ListTopics").Return(initialKafkaTopics, nil).Once()

    topicManager, err := NewTopicManager(mock, baseTopics, 1)
    assert.Nil(t, err)

    topics := topicManager.GetTopicList()

    assert.True(t, slices.Contains(topics, baseTopics[0]))
    assert.True(t, slices.Contains(topics, kafkaTopics[1]))
    assert.True(t, slices.Contains(topics, kafkaTopics[2]))

    //add a kafka topic
    kafkaTopics = append(kafkaTopics, "etlflow_newTopic")
    revisedKafkaTopics := buildTopicMap(kafkaTopics)
    mock.On("ListTopics").Return(revisedKafkaTopics, nil).Once()

    //scan for new topic
    newTopic, err := topicManager.scanForNewTopics()
    assert.Nil(t, err)
    assert.True(t, newTopic)
}

func TestFailureInitializationError(t *testing.T) {

    baseTopics := []string{"ncyd_kubernetes", "etlflow_*"}

    emptyMap := buildTopicMap([]string{})

    mock := mock.NewMockClusterAdmin()
    mock.On("ListTopics").Return(emptyMap, errors.New("error")).Once()

    _, err := NewTopicManager(mock, baseTopics, 1)
    assert.NotNil(t, err)
}

func buildTopicMap(kafkaTopics []string) map[string]sarama.TopicDetail {

    topicMap := make(map[string]sarama.TopicDetail)
    for _, topic := range kafkaTopics {
        tp := sarama.TopicDetail{}
        topicMap[topic] = tp
    }
    return topicMap
}
...

```

Sample Mock Class

```
package mock

import (
    "github.com/Shopify/sarama"
    "github.com/stretchr/testify/mock"
)

type MockClusterAdmin struct{
    mock.Mock
    topics []string
}

func NewMockClusterAdmin() *MockClusterAdmin{
    mockClusterAdmin := &MockClusterAdmin{}
    return mockClusterAdmin
}

func (m *MockClusterAdmin) ListTopics() (map[string]sarama.TopicDetail, error){

    args := m.Called()
    return args.Get(0).(map[string]sarama.TopicDetail), args.Error(1)
}

...
```