## Annotations

Run the test using PowerMockito

`@RunWith(PowerMockRunner.class)`

Prepare to test static methods of a class

`@PrepareForTest(StaticClass.class)`

## Mocks

Create a mock

`PowerMockito.mock(ClassToMock.class)`

Do nothing when a method is called

`PowerMockito.doNothing().when(mock).someMethod()`

Return this value when a method is called

`PowerMockito.when(mock.someMethod()).thenReturn(value)`

Return this value when a method is called with any arg of type String

`PowerMockito.when(mock.someMethod(any(String.class)).thenReturn(value)`

## Verify

Method was called on mock once

`Mockito.verify(mock).someMethod()`

Method was called at least n times

`Mockito.verify(mock, times(n)).someMethod()`

Method was never called

`Mockito.verify(mock, never()).someMethod()`

Method was called atleast, atmost n times

`Mockito.verify(mock, atLeastOnce()).someMethod()`

`Mockito.verify(mock, atLeast(n)).someMethod()`

`Mockito.verify(mock, atMost(n)).someMethod()`

## Statics

Mock all static methods of a class

`PowerMockito.mockStatic(ClassWithStatics.class)`

Do nothing when static method is called

`PowerMockito.doNothing().when(ClassWithStatics.class)`

Prepare to verify

`PowerMockito.verifyStatic()`

## Private

Do nothing when a private method is called

`PowerMockito.doNothing().when(obj, "privateMethod", param1, param2)`

Return some value when a private method is called

`PowerMockito.when(obj, "privateMethod", param1, param2).thenReturn(value)`

Verify that a private method was called

`PowerMockito.verifyPrivate(obj).invoke("privateMethod", new Object[] { param1, param2 });`