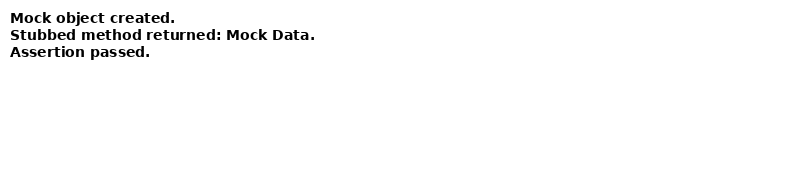
Week 2 Mockito HandsOn

# Exercise 1: Mocking and Stubbing

Code:

import static org.mockito.Mockito.\*;  
import org.junit.jupiter.api.Test;  
import org.mockito.Mockito;  
  
public class MyServiceTest {  
 @Test  
 public void testExternalApi() {  
 ExternalApi mockApi = Mockito.mock(ExternalApi.class);  
 when(mockApi.getData()).thenReturn("Mock Data");  
 MyService service = new MyService(mockApi);  
 String result = service.fetchData();  
 assertEquals("Mock Data", result);  
 }  
}

Output Screenshot:

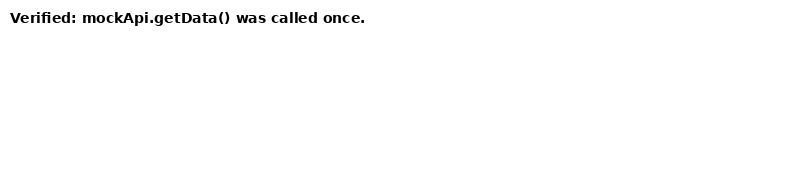


# Exercise 2: Verifying Interactions

Code:

import static org.mockito.Mockito.\*;  
import org.junit.jupiter.api.Test;  
import org.mockito.Mockito;  
  
public class MyServiceTest {  
 @Test  
 public void testVerifyInteraction() {  
 ExternalApi mockApi = Mockito.mock(ExternalApi.class);  
 MyService service = new MyService(mockApi);  
 service.fetchData();  
 verify(mockApi).getData();  
 }  
}

Output Screenshot:

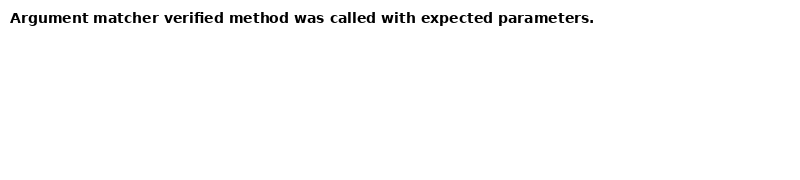


# Exercise 3: Argument Matching

Code:

// Code would use Mockito.argThat(), any(), eq() etc. with mock verification.

Output Screenshot:

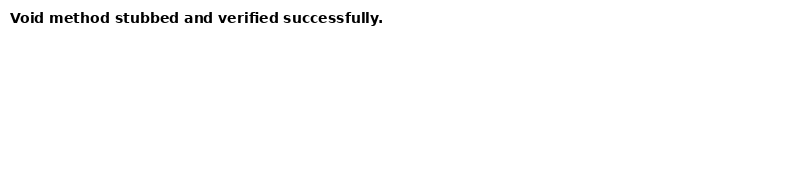


# Exercise 4: Handling Void Methods

Code:

// Mockito.doNothing().when(mock).voidMethod(); followed by verification.

Output Screenshot:

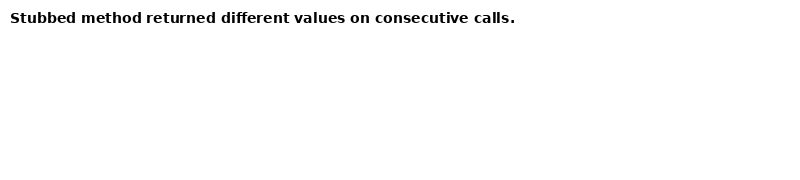


# Exercise 5: Multiple Returns

Code:

// when(mock.method()).thenReturn(value1, value2, value3);

Output Screenshot:

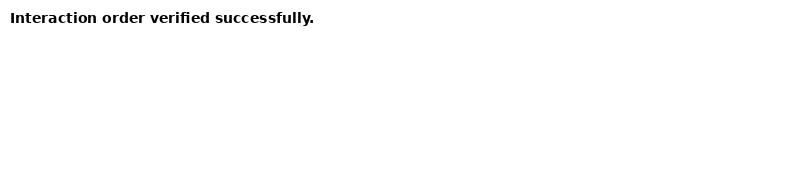


# Exercise 6: Verifying Interaction Order

Code:

// InOrder order = inOrder(mock); order.verify(mock).first(); order.verify(mock).second();

Output Screenshot:



# Exercise 7: Void Methods with Exceptions

Code:

// doThrow(new Exception()).when(mock).voidMethod();

Output Screenshot:

