**Exercise 1: Mocking and Stubbing**

Src/main/java/org.example/

ExternalApi.java

package org.example;  
  
public class ExternalApi {  
 public String getData() {  
 return "Real data from external API";  
 }  
}

MyService.java

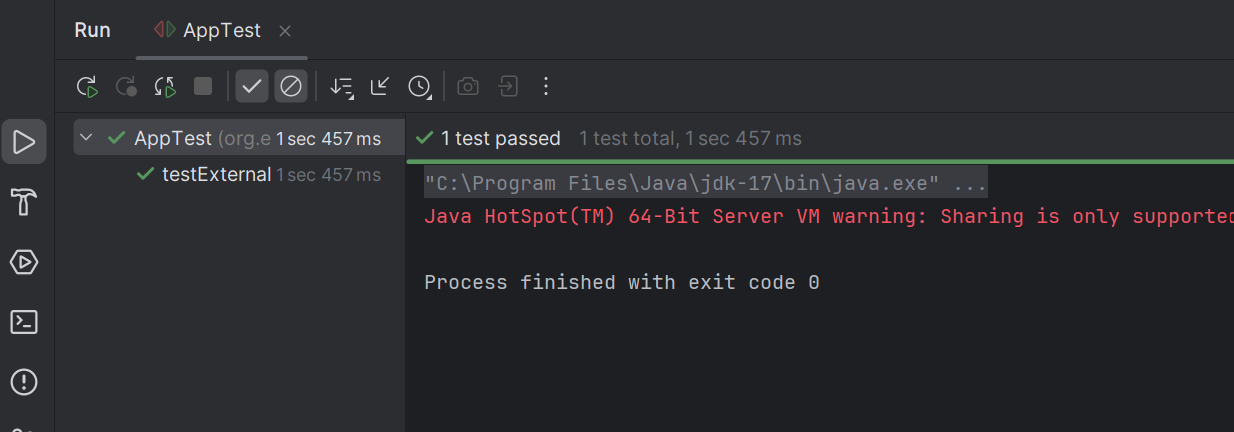
package org.example;  
  
public class MyService {  
 private ExternalApi api;  
  
 public MyService(ExternalApi api) {  
 this.api = api;  
 }  
  
 public String fetchData() {  
 return api.getData();  
 }  
}

Src/test/java/org.example/

AppTest.java

package org.example;  
  
import org.junit.Test;  
import static org.junit.Assert.\*;  
import static org.mockito.Mockito.\*;  
  
public class AppTest {  
  
 @Test  
 public void testExternalApiUsingMock() {  
 // Create a mock of ExternalApi  
 ExternalApi mockApi = *mock*(ExternalApi.class);  
  
 // Define behavior for the correct method  
 *when*(mockApi.getData()).thenReturn("Mocked Data");  
  
 // Use mock and assert result  
 String result = mockApi.getData();  
 *assertEquals*("Mocked Data", result);  
 }  
}

output:



**Exercise 2: Verifying Interactions**

Src/main/java/org.example/

ExternalApi.java

package org.example;  
  
public class ExternalApi {  
 public String getData() {  
 return "Real Data";  
 }  
}

MyService.java

package org.example;  
  
public class MyService {  
 private ExternalApi api;  
  
 public MyService(ExternalApi api) {  
 this.api = api;  
 }  
  
 public String fetchData() {  
 return api.getData();  
 }  
}

Src/test/java/org.example/

AppTest.java

package org.example;  
  
import org.junit.Test;  
import static org.junit.Assert.\*;  
import static org.mockito.Mockito.\*;  
  
public class AppTest {  
  
 @Test  
 public void testFetchDataFromMockedApi() {  
   
 ExternalApi mockApi = *mock*(ExternalApi.class);  
  
 *when*(mockApi.getData()).thenReturn("Mocked Data");  
   
 MyService service = new MyService(mockApi);  
   
 String result = service.fetchData();  
 *assertEquals*("Mocked Data", result);  
 }  
}

output

