Pop.xml

```xml
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">Add commentMore actions
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.deep.skilling</groupId>
  <artifactId>mockito</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
    <!-- JUnit 5 -->
    <dependency>
        <groupId>org.junit.jupiter</groupId>
        <artifactId>junit-jupiter</artifactId>
        <version>5.10.0</version>
        <scope>test</scope>
    </dependency>

    <!-- Mockito Core -->
    <dependency>
        <groupId>org.mockito</groupId>
        <artifactId>mockito-core</artifactId>
        <version>5.11.0</version>
        <scope>test</scope>
    </dependency>
</dependencies>

</project>
```

external.API

```
package com.deep.skilling.mockito;Add commentMore actions


public interface ExternalApi {
   String getData();
}
```

Myservice

```
package com.deep.skilling.mockito;Add commentMore actions


public class MyService {
   private ExternalApi api;


   public MyService(ExternalApi api) {
      this.api = api;
   }


   public String fetchData() {
      return api.getData();
   }
}
```

Myservicetest

```
package com.deep.skilling.mockito;Add commentMore actions


import static org.junit.jupiter.api.Assertions.assertEquals;
import static org.mockito.Mockito.*;
```

```java
import org.junit.jupiter.api.Test;
import org.mockito.Mockito;

public class MyServiceTest {

    @Test
    public void testExternalApi() {
        // Step 1: Create mock
        ExternalApi mockApi = Mockito.mock(ExternalApi.class);

        // Step 2: Stub the method
        when(mockApi.getData()).thenReturn("Mock Data");

        // Step 3: Use mock in service
        MyService service = new MyService(mockApi);

        // Step 4: Call method and assert
        String result = service.fetchData();
        assertEquals("Mock Data", result);
    }
}
```