5.Write a program to demonstrate dynamic tests.

pom.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
http://maven.apache.org/xsd/maven-4.0.0.xsd"
>
  <modelVersion>4.0.0</modelVersion>
  <groupId>UsingJUnit</groupId>
  <artifactId>UsingJUnit</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
    <dependency>
      <groupId>org.junit.jupiter</groupId>
      <artifactId>junit-jupiter-engine</artifactId>
      <version>5.4.2</version>
    </dependency>
    <dependency>
      <groupId>org.junit.platform</groupId>
      <artifactId>junit-platform-launcher</artifactId>
      <version>1.2.0</version>
    </dependency>
  </dependencies>
</project>
```

DynamicTests.java

```
package com.ecommerce.tests;

import java.util.Arrays;
import java.util.Collection;
import java.util.List;
import org.junit.jupiter.api.*;
import org.junit.jupiter.api.AfterAll;
import org.junit.jupiter.api.BeforeAll;
import org.junit.jupiter.api.Test;
import org.junit.jupiter.api.condition.DisabledIf;
import org.junit.jupiter.api.condition.EnabledOnOs;
import org.junit.jupiter.api.condition.OS;
import org.junit.jupiter.api.function.Executable;
import org.junit.platform.runner.JUnitPlatform;
import org.junit.runner.RunWith;
import org.junit.jupiter.api.DynamicTest.*;
```

```
import org.junit.jupiter.api.TestFactory;

@DisplayName("JUnit 5 Dynamic Tests Example")
@RunWith(JUnitPlatform.class)
public class DynamicTests {
    @TestFactory
    Collection<DynamicTest> dynamicTests() {
        return Arrays.asList(
                dynamicTest("simple dynamic test", () -> assertTrue(true)),
                dynamicTest("My Executable Class", new MyExecutable()),
                dynamicTest("Exception Executable", () -> {
                    throw new Exception("Exception Example");
                }),
                dynamicTest("simple dynamic test-2", () -> assertTrue(true)));
    }
}

class MyExecutable implements Executable {
    @Override
    public void execute() throws Throwable {
        System.out.println("Hello World!");
    }
}
```