

Lesson 01 Demo 05

Maven Surefire Plugin

Objective: To explore Maven Surefire plugin by defining test cases and adding different configurations for test cases

Tools Required: Visual Studio Code and Maven

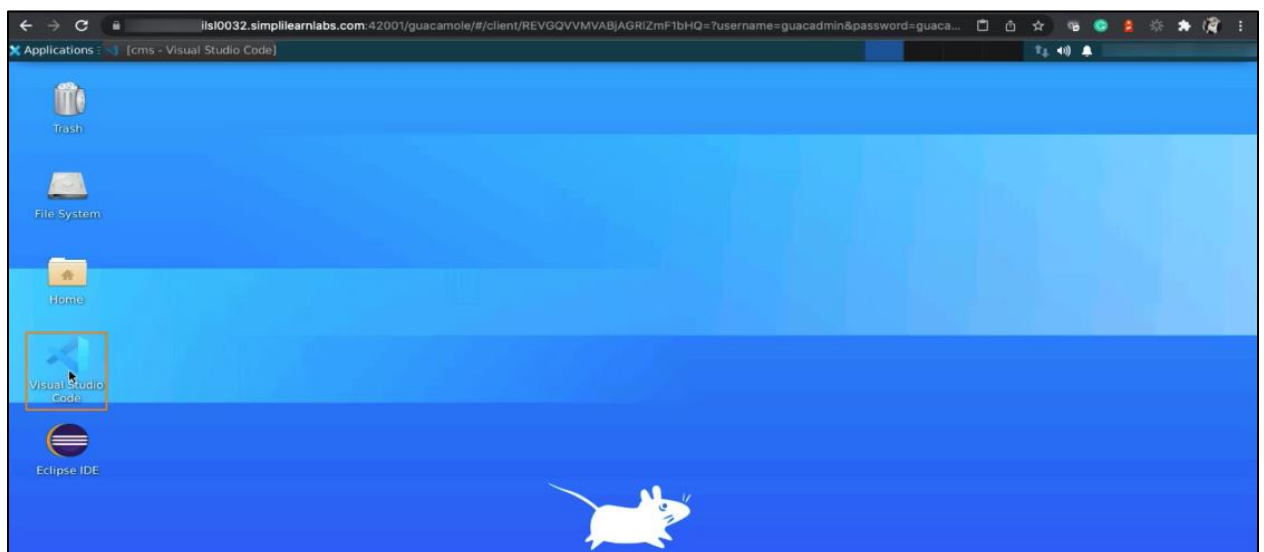
Prerequisites: None

Steps to be followed:

1. Check Maven Surefire plugin
2. Define test cases
3. Test defined test cases
4. Add different configurations for test cases
5. Work with the security manager

Step 1: Check Maven Surefire plugin

1.1 Open Visual Studio Code



1.2 Under the CMS project, open the **pom.xml** file

```

80      <maxmem>512m</maxmem>
81
82      <compilerArgs>
83        <arg>-verbose</arg>
84        <arg>-Xlint:all</arg>
85      </compilerArgs>
86
87    </configuration>
88  </plugin>
89  <plugin>
90    <artifactId>maven-surefire-plugin</artifactId>
91    <version>2.22.1</version>
92  </plugin>
93  <plugin>
94    <artifactId>maven-jar-plugin</artifactId>
95    <version>3.0.2</version>
96  </plugin>
97  <plugin>
98    <artifactId>maven-install-plugin</artifactId>
99    <version>2.5.2</version>
100  </plugin>
101  <plugin>
102    <artifactId>maven-deploy-plugin</artifactId>
103    <version>2.8.2</version>
104  </plugin>
105  <!-- site lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#site_Lifecycle -->
106  <plugin>
107    <artifactId>maven-site-plugin</artifactId>
108    <version>3.7.1</version>
109  </plugin>
110  <plugin>
111    <artifactId>maven-project-info-reports-plugin</artifactId>
112    <version>3.0.0</version>

```

Note: Please refer to the previous demo on how to create the CMS project

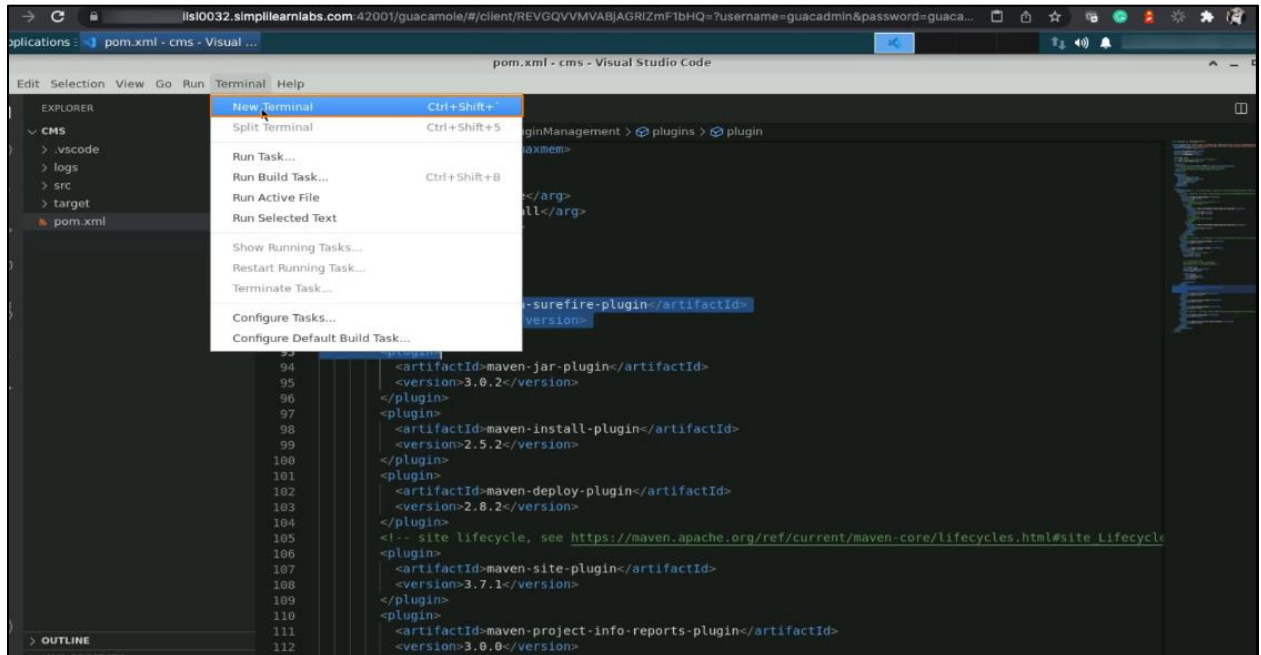
You can see the plugin as **maven-surefire-plugin**:

```

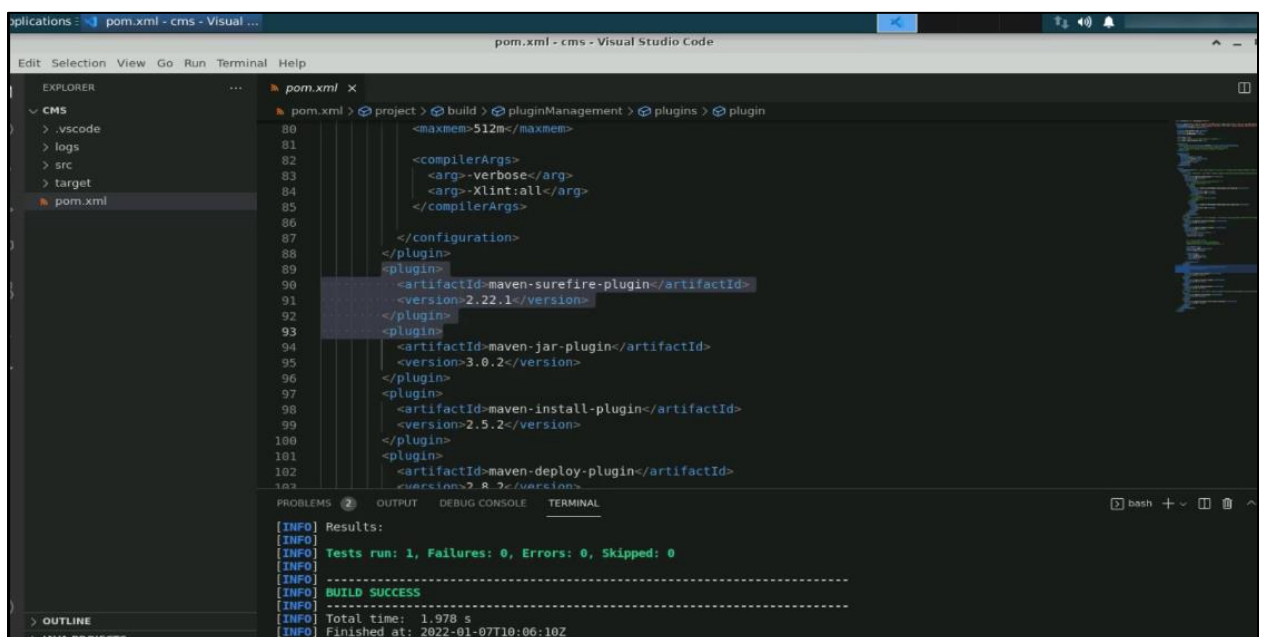
80      <maxmem>512m</maxmem>
81
82      <compilerArgs>
83        <arg>-verbose</arg>
84        <arg>-Xlint:all</arg>
85      </compilerArgs>
86
87    </configuration>
88  </plugin>
89  <plugin>
90    <artifactId>maven-surefire-plugin</artifactId>
91    <version>2.22.1</version>
92  </plugin>
93  <plugin>
94    <artifactId>maven-jar-plugin</artifactId>
95    <version>3.0.2</version>
96  </plugin>
97  <plugin>
98    <artifactId>maven-install-plugin</artifactId>
99    <version>2.5.2</version>
100  </plugin>
101  <plugin>
102    <artifactId>maven-deploy-plugin</artifactId>
103    <version>2.8.2</version>
104  </plugin>
105  <!-- site lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#site_Lifecycle -->
106  <plugin>
107    <artifactId>maven-site-plugin</artifactId>
108    <version>3.7.1</version>
109  </plugin>

```

1.3 Go to the **Terminal** tab and click on **New Terminal**



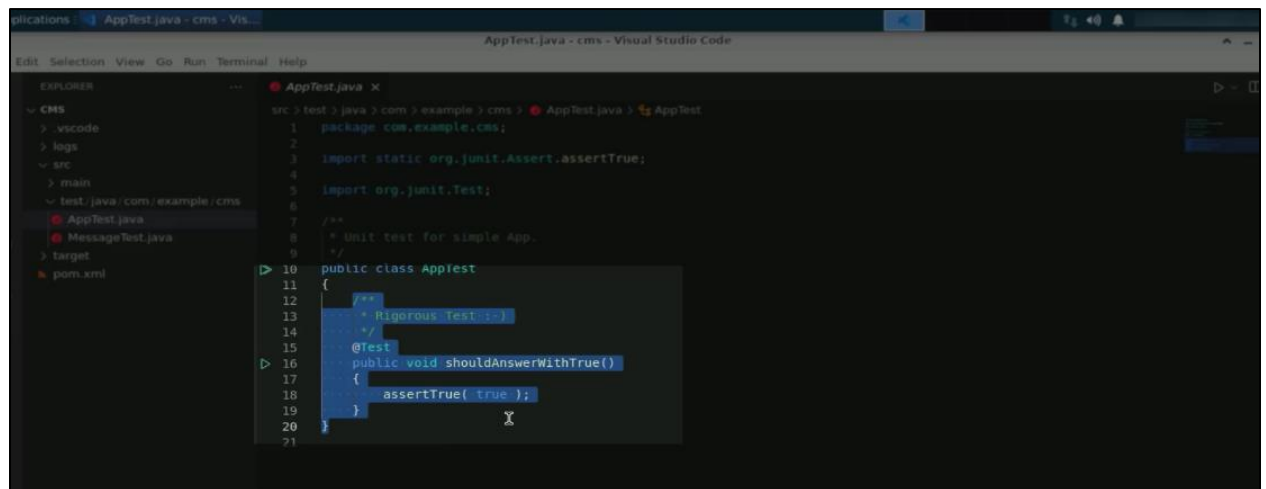
1.4 Run the below command to check Surefire plugin usages and run the test cases: **mvn test**



The command **mvn test** is used in Maven to execute the unit tests for a Java project. It compiles the source code and runs all the test cases in the project.

Step 2: Define test cases

2.1 Under src, open the **AppTest.java** from the **test** directory



The screenshot shows the Visual Studio Code editor with the **AppTest.java** file open. The file is located in the **src > test > java > com > example > cms >** directory. The code in the file is as follows:

```
src > test > java > com > example > cms > AppTest.java
1 package com.example.cms;
2
3 import static org.junit.Assert.assertTrue;
4
5 import org.junit.Test;
6
7 /**
8  * Unit test for simple App.
9  */
10 public class AppTest
11 {
12     /**
13      * Rigorous Test :-
14      */
15     @Test
16     public void shouldAnswerWithTrue()
17     {
18         assertTrue(true);
19     }
20
21 }
```

2.2 Go to **Message.java** and create a few methods in it

```
public static String getMessage(int code){
}
```

```

src > main > java > com > example > cms > Message.java > Message > gteMessage(int)
1 package com.example.cms;
2
3 public class Message {
4     public static String gteMessage(int code){
5     }
6 }
7
8
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] BUILD SUCCESS
[INFO]

```

The code snippet declares a public static method named **getMessage** which takes an integer parameter named **code**. The method is expected to return a string value.

2.3 Add the if condition and the return statement

```

if(code == 101){
    return "Customer Registered Successfully";
}

```

```

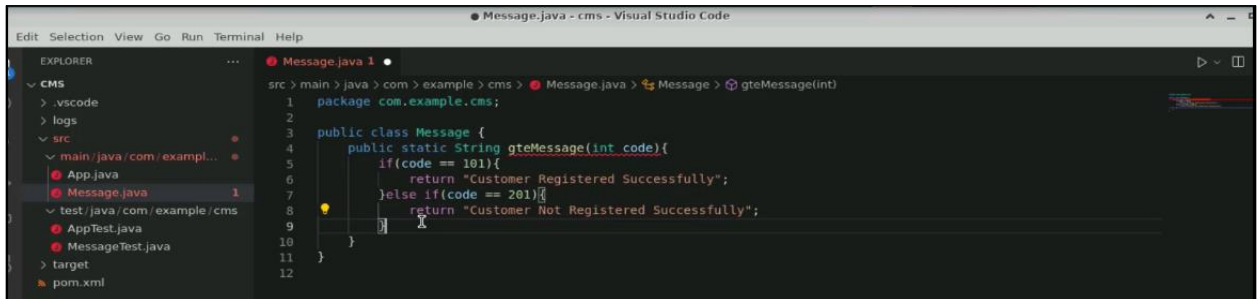
src > main > java > com > example > cms > Message.java > Message > gteMessage(int)
1 package com.example.cms;
2
3 public class Message {
4     public static String gteMessage(int code){
5         if(code == 101){
6             return "Customer Registered Successfully";
7         }
8     }
9 }
10
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] BUILD SUCCESS
[INFO]

```

This code snippet checks if the value of the variable **code** is **101**, and then the function returns the string **Customer Registered Successfully**.

2.4 Add the else if case and return statement

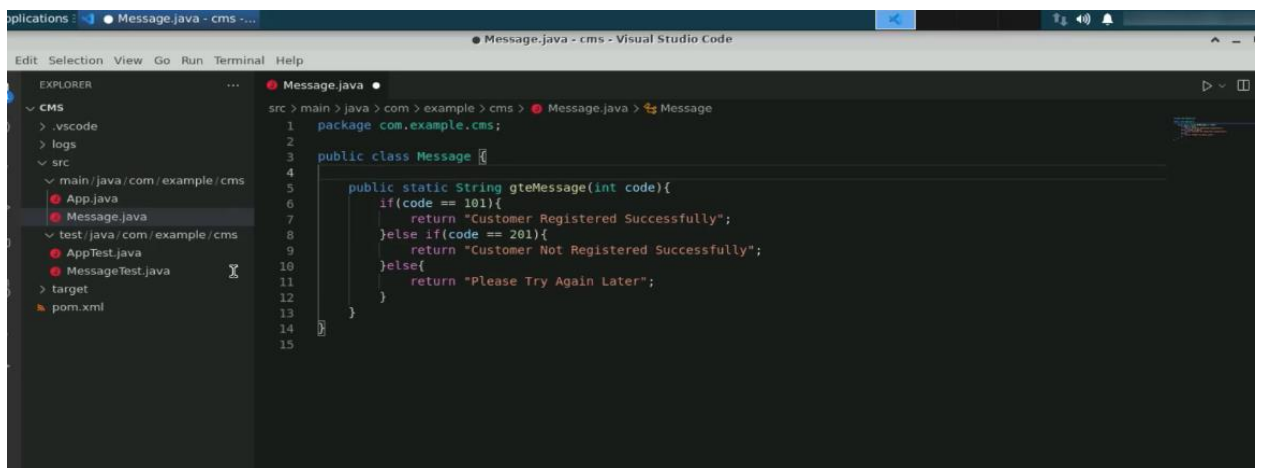
```
else if(code == 201){  
    return "Customer Not Registered Successfully";  
}
```



These two lines of code use the conditional statement **else if** to check if the variable **code** is **201**, and then the function returns a string indicating that the customer was not registered successfully.

2.5 Add else case and the return statement

```
else{  
    return "Please Try Again Later";  
}
```



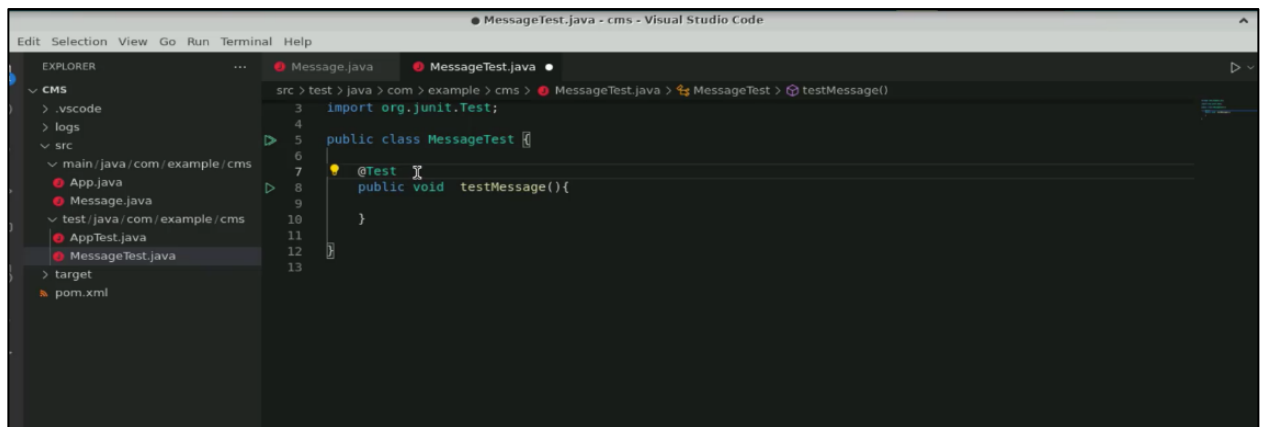
This code snippet uses the **else** keyword to define what should happen if a certain condition is not met in an `if` statement. In this case, if the condition is not met, the function will return the string **Please Try Again Later**.

Step 3: Test defined test cases

3.1 Create a **test** method in the **MessageTest.java** file to test the cases

@Test

```
public void testMessage(){  
}
```



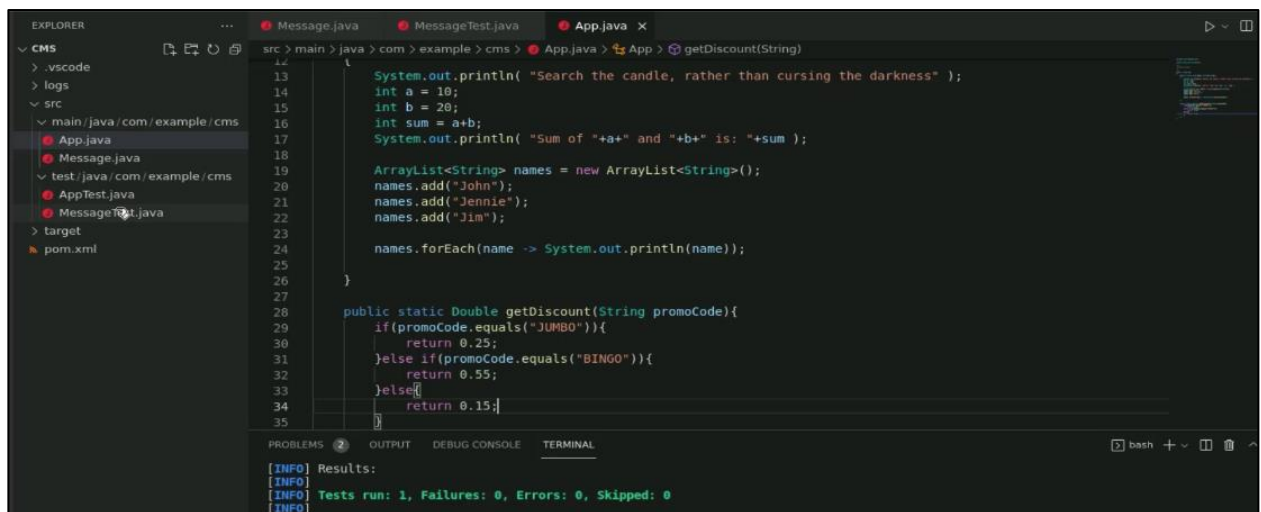
3.2 Define a scenario for the test and add the expected and the actual output for it
assertEquals("Customer Registered Successfully", Message.getMessage(101));



The given code is a test method that verifies if the message returned by the method `getMessage()` with the argument **101** is equal to the expected string **Customer Registered Successfully**. The `assertEquals()` method is used to compare the actual and expected values.

3.3 Add a method in **App.java** to define the discount

```
public static Double getDiscount (String promoCode) {
    if(promoCode.equals("JUMBO")){
        return 0.25;
    }else if (promoCode.equals("BINGO")){
        return 0.55;
    }else{
        return 0.15;
    }
}
```



```
src > main > java > com > example > cms > App.java > App > getDiscount(String)
13
14 System.out.println( "Search the candle, rather than cursing the darkness" );
15
16 int a = 10;
17 int b = 20;
18 int sum = a+b;
19 System.out.println( "Sum of "+a+" and "+b+" is: "+sum );
20
21 ArrayList<String> names = new ArrayList<String>();
22 names.add("John");
23 names.add("Jennie");
24 names.add("Jim");
25
26 names.forEach(name -> System.out.println(name));
27
28 }
29
30 public static Double getDiscount(String promoCode){
31     if(promoCode.equals("JUMBO")){
32         return 0.25;
33     }else if (promoCode.equals("BINGO")){
34         return 0.55;
35     }else{
36         return 0.15;
37     }
38 }
39 }
```

```
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
```

This code defines a static method named **getDiscount** that takes a String argument named **promoCode**. If the **promoCode** is equal to **JUMBO**, the method returns a discount of **25%**; if it is equal to **BINGO**, the method returns a discount of **55%**; and if it is anything else, the method returns a discount of **15%**.

3.4 Write test promo code in the **AppTest.java** file and add the **assertEqual** statement, including the actual and expected output


```

src > test > java > com > example > cms > AppTest.java > %AppTest> testPromoCode()
1 package com.example.cms;
2 import static org.junit.Assert.assertEquals;
3
4 import org.junit.Test;
5
6 /**
7  * Unit test for simple App.
8  */
9 public class AppTest
10 {
11     /**
12      * Rigorous Test :-
13      */
14     @Test
15     public void testPromoCode()
16     {
17         assertEquals(0.55, App.getDiscount("BINGO"));
18     }
19 }
20
[INFO] Results:
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO] BUILD SUCCESS

```

3.5 Change the expected output value to a double data type value

```

src > test > java > com > example > cms > AppTest.java > %AppTest> testPromoCode()
1 package com.example.cms;
2 import static org.junit.Assert.assertEquals;
3
4 import org.junit.Test;
5
6 /**
7  * Unit test for simple App.
8  */
9 public class AppTest
10 {
11     /**
12      * Rigorous Test :-
13      */
14     @Test
15     public void testPromoCode()
16     {
17         assertEquals(Double.valueOf(0.55), App.getDiscount("BINGO"));
18     }
19 }
20
[INFO] Results:
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO] BUILD SUCCESS
[INFO] Total time: 1.978 s
[INFO] Finished at: 2022-01-07T10:06:10Z
[INFO]

```

3.6 Use the **mvn test** again to run the test cases

```

src > test > java > com > example > cms > MessageTest.java > ...
1 package com.example.cms;
2
3 import static org.junit.Assert.assertEquals;
4
5 import org.junit.Test;
6
7 public class MessageTest {
8
9     @Test
10    public void testMessage(){
11        assertEquals("Customer Registered Successfully", Message.getMessage(101));
12    }
13
14 }
15

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

[INFO] Finished at: 2022-01-07T10:06:10Z
[INFO] -----
erishant@gmail@ip-172-31-17-157:~/Downloads/cms$ mvn test
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/guice.jar) to m
od java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
[INFO] -----
[INFO] Building cms 1.0-SNAPSHOT
[INFO] -----[ jar ]-----

```

3.7 Change the promo code in the **AppTest.java** file to **Hello55**

```

src > test > java > com > example > cms > AppTest.java > AppTest > testPromoCode()
4 import org.junit.Test;
5
6 /**
7  * Unit test for simple App.
8  */
9 public class AppTest
10 {
11     /**
12     * Rigorous Test :-))
13     */
14     @Test
15     public void testPromoCode()
16     {
17         assertEquals(Double.valueOf(0.55), App.getDiscount("HELL055"));
18     }
19 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```

[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running com.example.cms.MessageTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.032 s - in com.example.cms.MessageTest
[INFO] Running com.example.cms.AppTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.001 s - in com.example.cms.AppTest
[INFO] Results:
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
[INFO] -----

```

3.8 Run the **mvn test** again to run the test cases

```

AppTest.java - cms - Visual Studio Code
Edit Selection View Go Run Terminal Help

EXPLORER
  CMS
  > .vscode
  > logs
  > src
    > main/java/com/example/cms
      App.java
      Message.java
    > test/java/com/example/cms
      AppTest.java
      MessageTest.java
  > target
  > pom.xml

src > test > java > com > example > cms > AppTest.java > AppTest > testPromoCode()
  4 import org.junit.Test;
  5
  6 /**
  7  * Unit test for simple App.
  8  */
  9 public class AppTest
 10 {
 11     /**
 12     * Rigorous Test :-)
 13     */
 14     @Test
 15     public void testPromoCode()
 16     {
 17         assertEquals(Double.valueOf(0.55), App.getDiscount("HELLO55"));
 18     }
 19 }

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.001 s - in com.example.cms.AppTest
[INFO] Results:
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0
[INFO] BUILD SUCCESS
[INFO] Total time: 2.179 s
[INFO] Finished at: 2022-01-07T10:20:56Z
[INFO] -----
erishantgmail@ip-172-31-17-157:~/Downloads/cms$ mvn test
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/guice.jar) to
od java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release

```

Step 4: Add different configurations for test cases

4.1 Add configuration in the pom.xml file

<configuration></configuration>

```

EXPLORER
  CMS
  > .vscode
  > logs
  > src
    > main/java/com/example/cms
      App.java
      Message.java
    > test/java/com/example/cms
      AppTest.java
      MessageTest.java
  > target
  > pom.xml

pom.xml
  80 <project>
  81 <build>
  82 <pluginManagement>
  83 <plugins>
  84 <plugin>
  85 <groupId>org.apache.maven.plugins</groupId>
  86 <artifactId>maven-surefire-plugin</artifactId>
  87 <configuration>
  88 </configuration>
  89 </plugin>
  90 </plugins>
  91 </build>
  92 </project>

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL
[INFO] TESTS
[INFO] -----
[INFO] Running com.example.cms.MessageTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.038 s - in com.example.cms.MessageTest
[INFO] Running com.example.cms.AppTest
[ERROR] Tests run: 1, Failures: 1, Errors: 0, Skipped: 0, Time elapsed: 0.007 s <<< FAILURE! - in com.example.cms.AppTest
[ERROR] testPromoCode(com.example.cms.AppTest) Time elapsed: 0 s <<< FAILURE!
java.lang.AssertionError: expected:<0.55> but was:<0.15>
    at com.example.cms.AppTest.testPromoCode(AppTest.java:17)
[INFO] Results:
[INFO]

```

4.2 Under configurations, add SkipTest and pass the value as true

<skipTests>true</skipTests>

```

pom.xml
80 <maxmem>512m</maxmem>
81
82 <compilerArgs>
83   <arg>-verbose</arg>
84   <arg>-Xlint:all</arg>
85 </compilerArgs>
86
87 </configuration>
88 </plugin>
89 <plugin>
90   <artifactId>maven-surefire-plugin</artifactId>
91   <version>2.22.1</version>
92   <configuration>
93     <skipTests>true</skipTests>
94   </configuration>
95 </plugin>

```

```

[INFO] T E S T S
[INFO] -----
[INFO] Running com.example.cms.MessageTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.038 s - in com.example.cms.MessageTest
[INFO] Running com.example.cms.AppTest
[ERROR] Tests run: 1, Failures: 1, Errors: 0, Skipped: 0, Time elapsed: 0.007 s <<< FAILURE! - in com.example.cms.AppTest
[ERROR] testPromoCode(com.example.cms.AppTest) Time elapsed: 0 s <<< FAILURE!
java.lang.AssertionError: expected:<0.55> but was:<0.15>
    at com.example.cms.AppTest.testPromoCode(AppTest.java:17)
[INFO]
[INFO] Results:
[INFO]
[ERROR] Failures:

```

4.3 Run the **mvn clean** command to remove the **target** directory

```

pom.xml
80 <maxmem>512m</maxmem>
81
82 <compilerArgs>
83   <arg>-verbose</arg>
84   <arg>-Xlint:all</arg>
85 </compilerArgs>
86
87 </configuration>
88 </plugin>
89 <plugin>
90   <artifactId>maven-surefire-plugin</artifactId>
91   <version>2.22.1</version>
92   <configuration>
93     <skipTests>true</skipTests>
94   </configuration>
95 </plugin>

```

```

erishantgmail@ip-172-31-17-157:~/Downloads/cms$ mvn clean

```

4.4 Type the **mvn test** command and run the test cases again to check if it will work or not

```

pom.xml
<maxmem>512m</maxmem>
<compilerArgs>
  <arg>-verbose</arg>
  <arg>-Xlint:all</arg>
</compilerArgs>
</configuration>
</plugin>
<plugin>
  <artifactId>maven-surefire-plugin</artifactId>
  <version>2.22.1</version>
  <configuration>
    <skipTests>true</skipTests>
  </configuration>
</plugin>

```

```

[INFO] --- maven-clean-plugin:3.1.0:clean (default-clean) @ cms ---
[INFO] Deleting /home/erishantgmail/Downloads/cms/target
[INFO] Deleting /home/erishantgmail/Downloads/cms/logs/dev (includes = [**/*.log, **/*.tmp], excludes = [])
[INFO] Deleting /home/erishantgmail/Downloads/cms/logs/prod (includes = [**/*.log], excludes = [])
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 0.463 s
[INFO] Finished at: 2022-01-07T10:22:36Z
[INFO] -----
erishantgmail@ip-172-31-17-157:~/Downloads/cms$ mvn test
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/guice.jar) to
method java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
[INFO] Scanning for projects...

```

4.5 To directly use the skip test without writing in the **pom.xml** file, use the below command:

mvn test -DskipTest=true

```

pom.xml
<maxmem>512m</maxmem>
<compilerArgs>
  <arg>-verbose</arg>
  <arg>-Xlint:all</arg>
</compilerArgs>
</configuration>
</plugin>
<plugin>
  <artifactId>maven-surefire-plugin</artifactId>
  <version>2.22.1</version>
  <configuration>
    <skipTests>true</skipTests>
  </configuration>
</plugin>

```

```

[INFO] [loading /modules/java.base/java/lang/Double.class]
[INFO] [loading /modules/java.base/java/lang/String.class]
[INFO] [loading /modules/java.base/java/lang/Comparable.class]
[INFO] [loading /modules/java.base/java/lang/Number.class]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/App.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/AppTest.class]
[INFO] [checking com.example.cms.MessageTest]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/Message.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/MessageTest.class]
[INFO] [total 268ms]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cms ---
[INFO] Tests are skipped.
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2.739 s
[INFO] Finished at: 2022-01-07T10:22:46Z
[INFO] -----
erishantgmail@ip-172-31-17-157:~/Downloads/cms$ mvn test -DskipTest=true

```

4.6 Add excludes in the configuration

<excludes></excludes>


```

pom.xml | @project > @build > @pluginManagement > @plugins > @plugin > @configuration > @excludes
81
82     <compilerArgs>
83       <arg>-verbose</arg>
84       <arg>-Xlint:all</arg>
85     </compilerArgs>
86
87   </configuration>
88 </plugin>
89 <plugin>
90   <artifactId>maven-surefire-plugin</artifactId>
91   <version>2.22.1</version>
92   <configuration>
93     <!-- <skipTests>true</skipTests> -->
94   </configuration>
95 </plugin>
96 </excludes>

```

```

[INFO] [loading /modules/java.base/java/lang/Double.class]
[INFO] [loading /modules/java.base/java/lang/String.class]
[INFO] [loading /modules/java.base/java/lang/Comparable.class]
[INFO] [loading /modules/java.base/java/lang/Number.class]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/App.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/AppTest.class]
[INFO] [checking com.example.cms.MessageTest]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/Message.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/MessageTest.class]
[INFO] [total 268ms]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cms ---
[INFO] Tests are skipped.
[INFO] BUILD SUCCESS
[INFO] Total time: 2.739 s
[INFO] Finished at: 2022-01-07T18:22:46Z
[INFO]
erishantgmail@ip-172-31-17-157: ~$ cd Downloads/cms &

```

4.7 Under the excludes tag, add the Java file in which you want to exclude
`<exclude>AppTest.java</exclude>`

```

pom.xml | cms - Visual Studio Code
Edit Selection View Go Run Terminal Help
pom.xml x
pom.xml | @project > @build > @pluginManagement > @plugins > @plugin > @configuration > @excludes
87
88     <configuration>
89       <plugin>
90         <artifactId>maven-surefire-plugin</artifactId>
91         <version>2.22.1</version>
92         <configuration>
93           <!-- <skipTests>true</skipTests> -->
94           <excludes>
95             <exclude>AppTest.java</exclude>
96           </excludes>
97           <includes>
98             <include>MessageTest.java</include>
99           </includes>
100         </configuration>
101       </plugin>

```

```

[INFO] [loading /modules/java.base/java/lang/Double.class]
[INFO] [loading /modules/java.base/java/lang/String.class]
[INFO] [loading /modules/java.base/java/lang/Comparable.class]
[INFO] [loading /modules/java.base/java/lang/Number.class]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/App.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/AppTest.class]
[INFO] [checking com.example.cms.MessageTest]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/Message.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/MessageTest.class]
[INFO] [total 268ms]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cms ---
[INFO] Tests are skipped.

```

4.8 Create includes and add a file in it
`<includes>`
`<include>MessageTest.java</include>`
`</includes>`

```

pom.xml - cms - Visual Studio Code

pom.xml | project | build | pluginManagement | plugins | plugin | configuration | excludes

87 </configuration>
88 </plugin>
89 <plugin>
90 <artifactId>maven-surefire-plugin</artifactId>
91 <version>2.22.1</version>
92 <configuration>
93 <!-- <skipTests>true</skipTests> -->
94 <excludes>
95 <exclude>AppTest.java</exclude>
96 </excludes>
97 <includes>
98 <include>MessageTest.java</include>
99 </includes>
100 </configuration>
101 </plugin>

[INFO] [loading /modules/java.base/java/lang/Double.class]
[INFO] [loading /modules/java.base/java/lang/String.class]
[INFO] [loading /modules/java.base/java/lang/Comparable.class]
[INFO] [loading /modules/java.base/java/lang/Number.class]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/App.class]
[INFO] [checking com.example.cms.MessageTest]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/Message.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/AppTest.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/MessageTest.class]
[INFO] [total 268ms]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cms ---
[INFO] Tests are skipped.
[INFO] BUILD SUCCESS
[INFO] Total time: 2.739 s
[INFO] Finished at: 2022-01-07T10:22:46Z
[INFO]
erishantgmail@ip-172-31-17-157:~/Downloads/cms$

```

4.9 Write **mvn clean** in terminal

```

pom.xml - cms - Visual Studio Code

pom.xml | project | build | pluginManagement | plugins | plugin | configuration | includes | include

87 </configuration>
88 </plugin>
89 <plugin>
90 <artifactId>maven-surefire-plugin</artifactId>
91 <version>2.22.1</version>
92 <configuration>
93 <!-- <skipTests>true</skipTests> -->
94 <excludes>
95 <exclude>AppTest.java</exclude>
96 </excludes>
97 <includes>
98 <include>MessageTest.java</include>
99 </includes>
100 </configuration>
101 </plugin>

[INFO] [loading /modules/java.base/java/lang/String.class]
[INFO] [loading /modules/java.base/java/lang/Comparable.class]
[INFO] [loading /modules/java.base/java/lang/Number.class]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/App.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/AppTest.class]
[INFO] [checking com.example.cms.MessageTest]
[INFO] [loading /home/erishantgmail/Downloads/cms/target/classes/com/example/cms/Message.class]
[INFO] [wrote /home/erishantgmail/Downloads/cms/target/test-classes/com/example/cms/MessageTest.class]
[INFO] [total 268ms]
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cms ---
[INFO] Tests are skipped.
[INFO] BUILD SUCCESS
[INFO] Total time: 2.739 s
[INFO] Finished at: 2022-01-07T10:22:46Z
[INFO]
erishantgmail@ip-172-31-17-157:~/Downloads/cms$ mvn clean

```

4.10 Run the test using the **mvn test** command

The screenshot shows the VS Code interface with the `pom.xml` file open. The `<configuration>` tag inside the `maven-surefire-plugin` is highlighted. The terminal output shows the build process:

```
[INFO] -----[ jar ]-----
[INFO] --- maven-clean-plugin:3.1.0:clean (default-clean) @ cms ---
[INFO] Deleting /home/erishantmail/Downloads/cms/target
[INFO] Deleting /home/erishantmail/Downloads/cms/logs/dev (includes = [**/*.log, **/*.tmp], excludes = [])
[INFO] Deleting /home/erishantmail/Downloads/cms/logs/prod (includes = [**/*.log], excludes = [])
[INFO] BUILD SUCCESS
[INFO] Total time: 0.523 s
[INFO] Finished at: 2022-01-07T10:24:54Z
[INFO] -----
erishantmail@ip-172-31-17-157:~/Downloads/cms$ mvn test
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by com.google.inject.internal.cglib.core.$ReflectUtils$1 (file:/usr/share/maven/lib/guice.jar) to
od java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain)
WARNING: Please consider reporting this to the maintainers of com.google.inject.internal.cglib.core.$ReflectUtils$1
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
```

4.11 Comment the code inside the `<configuration>` tag

The screenshot shows the VS Code interface with the `pom.xml` file open. The `<configuration>` tag inside the `maven-surefire-plugin` is highlighted. The terminal output shows the build process:

```
[INFO] -----[ test ]-----
[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cms ---
[INFO] T E S T S
[INFO] Running com.example.cms.MessageTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.046 s - in com.example.cms.MessageTest
[INFO] Results:
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO] BUILD SUCCESS
[INFO] Total time: 3.331 s
[INFO] Finished at: 2022-01-07T10:25:02Z
[INFO] -----
erishantmail@ip-172-31-17-157:~/Downloads/cms$
```

4.12 To run a single test class **MessageTest** through the command prompt, use the below command to execute the **MessageTest**:

mvn test -Dtest=MessageTest


```

pom.xml > project > build > pluginManagement > plugins > plugin > configuration
87 </configuration>
88 </plugin>
89 <plugin>
90 <artifactId>maven-surefire-plugin</artifactId>
91 <version>2.22.1</version>
92 <configuration>
93 <!-- <skipTests>true</skipTests> -->
94 <!-- <excludes>
95 <exclude>AppTest.java</exclude>
96 </excludes>
97 <includes>
98 <include>MessageTest.java</include>
99 </includes> -->
100 </configuration>
101 </plugin>

[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cms ---
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running com.example.cms.MessageTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.046 s - in com.example.cms.MessageTest
[INFO] Results:
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 3.331 s
[INFO] Finished at: 2022-01-07T10:25:02Z
[INFO] -----
erishantmail@ip-172-31-17-157:~/Downloads/cms$ mvn test -Dtest=MessageTest

```

4.13 To run a single test class **MessageTest** through the command prompt, use the below command:

mvn test -Dtest=MessageTest

```

pom.xml > project > build > pluginManagement > plugins > plugin > configuration
87 </configuration>
88 </plugin>
89 <plugin>
90 <artifactId>maven-surefire-plugin</artifactId>
91 <version>2.22.1</version>
92 <configuration>
93 <!-- <skipTests>true</skipTests> -->
94 <!-- <excludes>
95 <exclude>AppTest.java</exclude>
96 </excludes>
97 <includes>
98 <include>MessageTest.java</include>
99 </includes> -->
100 </configuration>
101 </plugin>

[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cms ---
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running com.example.cms.AppTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.028 s - in com.example.cms.AppTest
[INFO] Results:
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.936 s
[INFO] Finished at: 2022-01-07T10:26:02Z
[INFO] -----
erishantmail@ip-172-31-17-157:~/Downloads/cms$ mvn test -Dtest=AppTest

```

4.14 Go to the **AppTest.java** file and write one more test case. Copy-paste the above test case and make changes as shown below:

Change name to **testMyPromoCode**

Change value to **0.25** and discount coupon name to **JUMBO**

```

src > test > java > com > example > cms > AppTest.java > AppTest > testMyPromoCode()
10 {
11     /**
12      * Rigorous Test :-)
13      */
14     @Test
15     public void testPromoCode()
16     {
17         assertEquals(Double.valueOf(0.55), App.getDiscount("HELL055"));
18     }
19
20     @Test
21     public void testMyPromoCode()
22     {
23         assertEquals(Double.valueOf(0.25), App.getDiscount("JUMB0"));
24     }
25 }

```

```

[INFO] -----
[INFO] BUILD FAILURE
[INFO] -----
[INFO] Total time: 2.023 s
[INFO] Finished at: 2022-01-07T10:26:31Z
[INFO] -----
[ERROR] Failed to execute goal org.apache.maven.plugins:maven-surefire-plugin:2.22.1:test (default-test) on project cms: There are test failures.
[ERROR] Please refer to /home/erishantgmail/Downloads/cms/target/surefire-reports for the individual test results.
[ERROR] Please refer to dump files (if any exist) [date].dump, [date]-jvmRun[N].dump and [date].dumpstream.
[ERROR] -> [Help 1]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]

```

4.15 Use **mvn test** with method name in it, that is, **-Dtest**, and specify the promo code
mvn test -Dtest=AppTest#testMyPromoCode

```

src > test > java > com > example > cms > AppTest.java > AppTest > testMyPromoCode()
10 {
11     /**
12      * Rigorous Test :-)
13      */
14     @Test
15     public void testPromoCode()
16     {
17         assertEquals(Double.valueOf(0.55), App.getDiscount("HELL055"));
18     }
19
20     @Test
21     public void testMyPromoCode()
22     {
23         assertEquals(Double.valueOf(0.25), App.getDiscount("JUMB0"));
24     }
25 }

```

```

[INFO] Results:
[INFO] Tests run: 0, Failures: 0, Errors: 0, Skipped: 0
[INFO] -----
[INFO] BUILD FAILURE
[INFO] -----
[INFO] Total time: 2.032 s
[INFO] Finished at: 2022-01-07T10:31:29Z
[INFO] -----
[ERROR] Failed to execute goal org.apache.maven.plugins:maven-surefire-plugin:2.22.1:test (default-test) on project cms: No tests were executed.
[ERROR] (Set -DfailIfNoTests=false to ignore this error.) -> [Help 1]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]
[ERROR] For more information about the errors and possible solutions, please read the following articles:
[ERROR] [Help 1] http://cwiki.apache.org/confluence/display/MAVEN/MojoFailureException
erishantgmail@ip-172-31-17-157: ~/Downloads/cms$ mvn test -Dtest=AppTest#testMyPromoCode

```

4.16 To test multiple classes at once, add multiple test names separated by a comma
mvn test -Dtest=MessageTest, AppTest

```

src > test > java > com > example > cms > AppTest.java > AppTest > testMyPromoCode()
10 {
11     /**
12      * Rigorous Test :-
13      */
14     @Test
15     public void testPromoCode()
16     {
17         assertEquals(Double.valueOf(0.55), App.getDiscount("HELLO55"));
18     }
19
20     @Test
21     public void testMyPromoCode()
22     {
23         assertEquals(Double.valueOf(0.25), App.getDiscount("JUMBO"));
24     }
25 }

[INFO] --- maven-surefire-plugin:2.22.1:test (default-test) @ cms ---
[INFO] T E S T S
[INFO] Running com.example.cms.AppTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.014 s - in com.example.cms.AppTest
[INFO] Results:
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO] BUILD SUCCESS
[INFO] Total time: 1.994 s
[INFO] Finished at: 2022-01-07T10:31:51Z
[INFO]
erishant@gmailip-172-31-17-157:~/Downloads/cms$ mvn test -Dtest=MessageTest, AppTest

```

Step 5: Work with the security manager

5.1 In the **pom.xml** file, add system property variables

```

pom.xml > project > build > pluginManagement > plugins > plugin > configuration > systemPropertyVariables
87     </configuration>
88     </plugin>
89     <plugin>
90         <artifactId>maven-surefire-plugin</artifactId>
91         <version>2.22.1</version>
92         <configuration>
93             <!-- <skipTests>true</skipTests> -->
94             <!-- <excludes>
95                 <exclude>AppTest.java</exclude>
96             </excludes>
97             <includes>
98                 <include>MessageTest.java</include>
99             </includes> -->
100             <systemPropertyVariables>
101             </systemPropertyVariables>
102         </configuration>
103     </plugin>
104 </plugins>
105 </configuration>
106 </project>

```

```

[INFO] Building cms 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO] BUILD FAILURE
[INFO] Total time: 0.160 s
[INFO] Finished at: 2022-01-07T10:32:27Z
[INFO]
[ERROR] Unknown lifecycle phase "AppTest". You must specify a valid lifecycle phase or a goal in the format <plugin-prefix>:<goal> or <plugin-group-id>:<plugin-artifact-id>[:<plugin-version>]:<goal>. Available lifecycle phases are: validate, initialize, generate-sources, process-sources, generate-resources, process-resources, compile, process-classes, generate-test-sources, process-test-sources, generate-test-resources, process-test-resources, test-compile, process-test-classes, test, prepare-package, package, pre-integration-test, integration-test, post-integration-test, verify, install, deploy, pre-clean, clean, post-clean, pre-site, site, post-site, site-deploy. -> [Help]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.

```

5.2 Use the **surefire.security.manager** tag to work with older versions of unit tests:

<surefire.security.manager></surefire.security.manager>

```

pom.xml
<!-- configuration -->
</configuration>
<plugin>
<artifactId>maven-surefire-plugin</artifactId>
<version>2.22.1</version>
<configuration>
<!-- <skipTests>true</skipTests> -->
<!-- <excludes>
<exclude>AppTest.java</exclude>
</excludes>
<includes>
<include>MessageTest.java</include>
</includes> -->
<systemPropertyVariables>
<surefire.security.manager>[redacted]
</systemPropertyVariables>

```

```

[INFO] Building cms 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] BUILD FAILURE
[INFO]
[INFO] Total time: 0.160 s
[INFO] Finished at: 2022-01-07T10:32:27Z
[INFO]
[ERROR] Unknown lifecycle phase "AppTest". You must specify a valid lifecycle phase or a goal in the format <plugin-prefix>:<goal> or <plugin-group-id>:<plugin-artifact-id>[:<plugin-version>]:<goal>. Available lifecycle phases are: validate, initialize, generate-sources, process-sources, generate-resources, process-resources, compile, process-classes, generate-test-sources, process-test-sources, generate-test-resources, process-test-resources, test-compile, process-test-classes, test, prepare-package, package, pre-integration-test, integration-test, post-integration-test, verify, install, deploy, pre-clean, clean, post-clean, pre-site, site, post-site, site-deploy. -> [Help]

```

5.3 Add a security manager in the App.java file

```

src > main > java > com > example > cms > App.java > App > main(String[])
3 import java.util.ArrayList;
4
5 /**
6  * Hello world!
7  */
8
9 public class App
10 {
11
12     Run | Debug
13     public static void main( String[] args )
14     {
15         SecurityManager
16
17         System.out.println( "Search the candle, rather than cursing the darkness" );
18         int a = 10;
19         int h = 20;
20     }
21 }

```

```

[INFO] Building cms 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] BUILD FAILURE
[INFO]
[INFO] Total time: 0.160 s
[INFO] Finished at: 2022-01-07T10:32:27Z
[INFO]
[ERROR] Unknown lifecycle phase "AppTest". You must specify a valid lifecycle phase or a goal in the format <plugin-prefix>:<goal> or <plugin-group-id>:<plugin-artifact-id>[:<plugin-version>]:<goal>. Available lifecycle phases are: validate, initialize, generate-sources, process-sources, generate-resources, process-resources, compile, process-classes, generate-test-sources, process-test-sources, generate-test-resources, process-test-resources, test-compile, process-test-classes, test, prepare-package, package, pre-integration-test, integration-test, post-integration-test, verify, install, deploy, pre-clean, clean, post-clean, pre-site, site, post-site, site-deploy. -> [Help]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]

```

5.4 Go to the pom.xml file and configure the API for JUnit 3, which is: java.lang.SecurityManager


```

90 <artifactId>maven-surefire-plugin</artifactId>
91 <version>2.22.1</version>
92 <configuration>
93 <!-- <skipTests>true</skipTests> -->
94 <!-- <excludes>
95 <exclude>AppTest.java</exclude>
96 </excludes>
97 <includes>
98 <include>MessageTest.java</include>
99 </includes> -->
100 <systemPropertyVariables>
101 <surefire.security.manager>java.lang.SecurityManager</surefire.security.manager>
102 </systemPropertyVariables>
103 </configuration>
104 </plugin>
105 </plugins>

```

```

[INFO] Building cms 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO] BUILD FAILURE
[INFO] Total time: 0.160 s
[INFO] Finished at: 2022-01-07T10:32:27Z
[ERROR] Unknown lifecycle phase "AppTest". You must specify a valid lifecycle phase or a goal in the format <plugin-prefix>:<goal> or <plugin-group-id>:<plugin-artifact-id>[:<plugin-version>]:<goal>. Available lifecycle phases are: validate, initialize, generate-sources, process-sources, generate-resources, process-resources, compile, process-classes, generate-test-sources, process-test-sources, generate-test-resources, process-test-resources, test-compile, process-test-classes, test, prepare-package, package, pre-integration-test, integration-test, post-integration-test, verify, install, deploy, pre-clean, clean, post-clean, pre-site, site, post-site, site-deploy. -> [Help]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.

```

5.5 Use the **parallel** tag in configuration to execute the tests in parallel and have unlimited threads to run the configuration
<parallel></parallel>

```

105 <parallel></parallel>

```

```

[INFO] Building cms 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO] BUILD FAILURE
[INFO] Total time: 0.160 s
[INFO] Finished at: 2022-01-07T10:32:27Z
[ERROR] Unknown lifecycle phase "AppTest". You must specify a valid lifecycle phase or a goal in the format <plugin-prefix>:<goal> or <plugin-group-id>:<plugin-artifact-id>[:<plugin-version>]:<goal>. Available lifecycle phases are: validate, initialize, generate-sources, process-sources, generate-resources, process-resources, compile, process-classes, generate-test-sources, process-test-sources, generate-test-resources, process-test-resources, test-compile, process-test-classes, test, prepare-package, package, pre-integration-test, integration-test, post-integration-test, verify, install, deploy, pre-clean, clean, post-clean, pre-site, site, post-site, site-deploy. -> [Help]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.

```

5.6 Add **methods** in the parallel tag and provide value for the **threadCount** as **10** to execute ten different tests
<parallel>methods</parallel>
<threadCount>10</threadCount>

The screenshot shows an IDE with the following components:

- EXPLORER:** A file tree on the left showing a project structure with folders like `src`, `test`, and `target`. Files include `App.java`, `Message.java`, `AppTest.java`, `MessageTest.java`, and `pom.xml`.
- Editor:** The `pom.xml` file is open, showing XML configuration for the Surefire plugin. The configuration includes:


```

      <configuration>
      <!-- <skipTests>true</skipTests> -->
      <!-- <excludes>
      | <exclude>AppTest.java</exclude>
      </excludes>
      <includes>
      | <include>MessageTest.java</include>
      </includes> -->
      <!-- <systemPropertyVariables>
      | <surefire.security.manager>java.lang.SecurityManager</surefire.security.manager>
      </systemPropertyVariables> -->
      <parallel>methods</parallel>
      <threadCount>10</threadCount>
      </configuration>
      
```
- PROBLEMS:** A panel showing build output and errors. The output includes:


```

      [INFO] Building cms 1.0-SNAPSHOT
      [INFO] -----[ jar ]-----
      [INFO] BUILD FAILURE
      [INFO] -----
      [INFO] Total time: 0.160 s
      [INFO] Finished at: 2022-01-07T10:32:27Z
      [INFO] -----
      [ERROR] Unknown lifecycle phase "AppTest". You must specify a valid lifecycle phase or a goal in the format <plugin-prefix>:<goal> or <plugin-group-id>:<plugin-artifact-id>[:<plugin-version>]:<goal>. Available lifecycle phases are: validate, initialize, generate-sources, process-sources, generate-resources, process-resources, compile, process-classes, generate-test-sources, process-test-sources, generate-test-resources, process-test-resources, test-compile, process-test-classes, test, prepare-package, package, pre-integration-test, integration-test, post-integration-test, verify, install, deploy, pre-clean, clean, post-clean, pre-site, site, post-site, site-deploy. -> [Help 1]
      [ERROR] To see the full stack trace of the error, go up to the top of the output.
      
```
- TERMINAL:** A terminal window at the bottom showing a bash prompt.

By following these steps, you have successfully configured the Surefire plugin for various roles. You have achieved the goal of running unit tests for your application using the test goal.