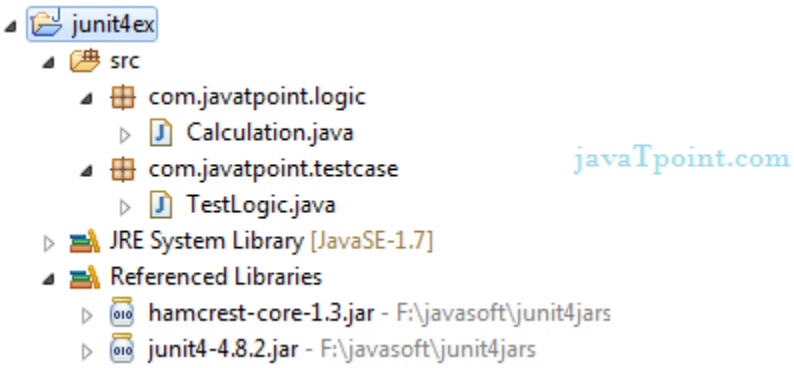
LAB EXERCISE 2

Aim: Unit Testing using Junit tool

JUnit tutorial provides basic and advanced concepts of **unit testing in java** with examples. Our junit tutorial is designed for beginners and professionals. It is an *open-source testing framework* for java programmers. The java programmer can create test cases and test his/her own code. It is one of the unit testing framework. Current version is junit 4. To perform unit testing, we need to create test cases. The **unit test case** is a code which ensures that the program logic works as expected. The **org.junit** package contains many interfaces and classes for junit testing such as Assert, Test, Before, After etc.

JUnit example in eclipse IDE

Let's see the directory structure of this example.



Write the program logic

Let's write the logic to find the maximum number for an array.

```
1. package com.javatpoint.logic;
2. public class Calculation {
3.
4.    public static int findMax(int arr[]){
5.       int max=0;
6.       for(int i=1;i<arr.length;i++){
7.          if(max<arr[i])
8.             max=arr[i];
```

```
9.        }
10.       return max;
11.   }
12. }
```

Write the test case

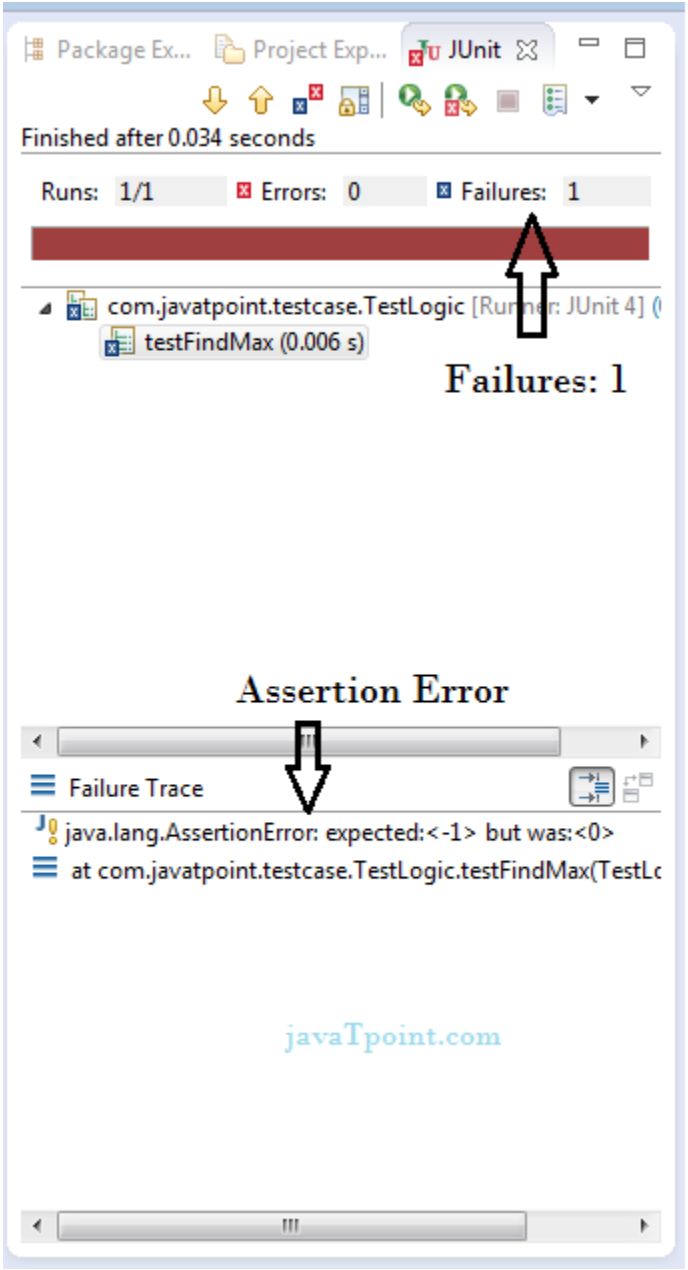
Here, we are using JUnit 4, so there is no need to inherit TestCase class. The main testing code is written in the testFindMax() method. But we can also perform some task before and after each test, as you can see in the given program.

```
1.  package com.javatpoint.testcase;
2.
3.  import static org.junit.Assert.*;
4.  import com.javatpoint.logic.*;
5.  import org.junit.Test;
6.
7.  public class TestLogic {
8.
9.      @Test
10.     public void testFindMax(){
11.         assertEquals(4,Calculation.findMax(new int[]{1,3,4,2}));
12.         assertEquals(-1,Calculation.findMax(new int[]{-12,-1,-3,-4,-2}));
13.     }
14. }
```

To run this example, **right click on TestLogic class -> Run As -> 1Junit Test**.

```
Output:Assertion Error
```

Let's see the output displayed in eclipse IDE.



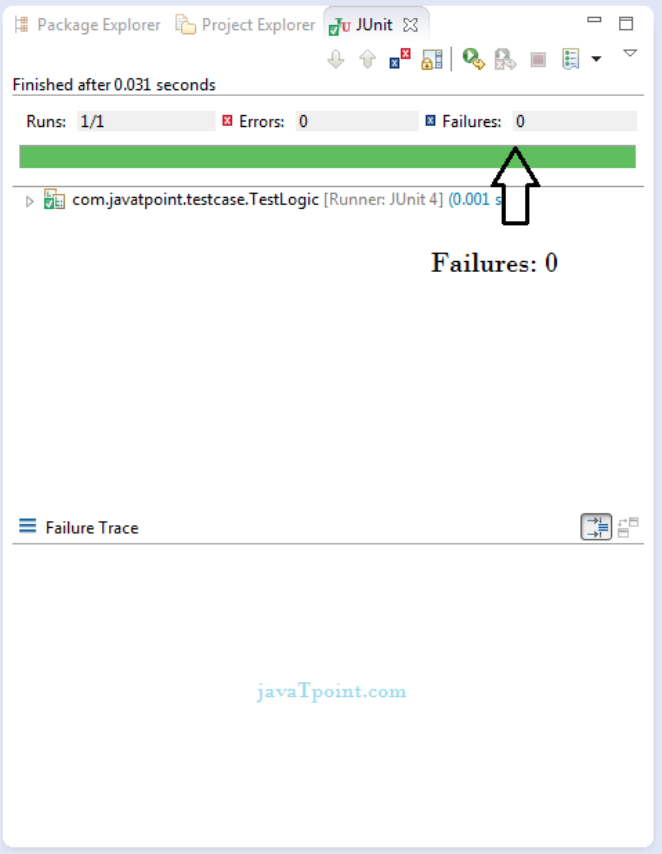
As you can see, when we pass the negative values, it throws AssertionError because second time findMax() method returns 0 instead of -1. It means our program logic is incorrect.

Correct program logic

As you can see, program logic to find the maximum number for the given array is not correct because it doesn't return -1 in case of negative values. The correct program logic is given below:

```
1.  package com.javatpoint.logic;
2.  public class Calculation {
3.
4.      public static int findMax(int arr[]){
5.          int max=arr[0];//arr[0] instead of 0
6.          for(int i=1;i<arr.length;i++){
7.              if(max<arr[i])
8.                  max=arr[i];
9.          }
10.         return max;
11.     }
12. }
```

If you run the junit program again, you will see the following output.



Q1. Write a program in java to find sum of two numbers and test this unit with JUnit tool.

Q2. Write a program in Java to find the reverse of a number and test it using JUnit tool.