Test Driven Development

Using JUnit to Test DVD.java

Produced by: Dr. Siobhán Drohan
Mairead Meagher

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics
http://www.wit.ie/

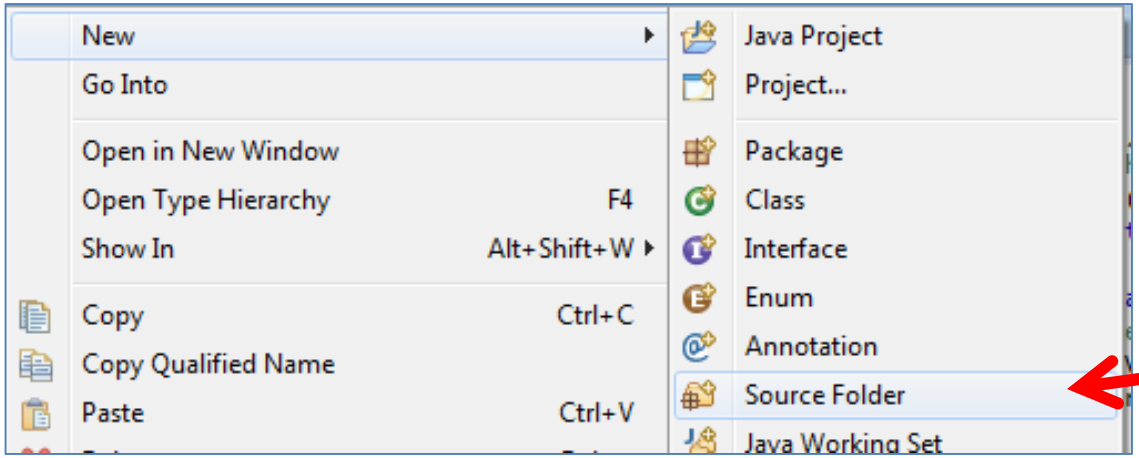
Let’s first write a complete test class and then we will discuss the theory behind it.

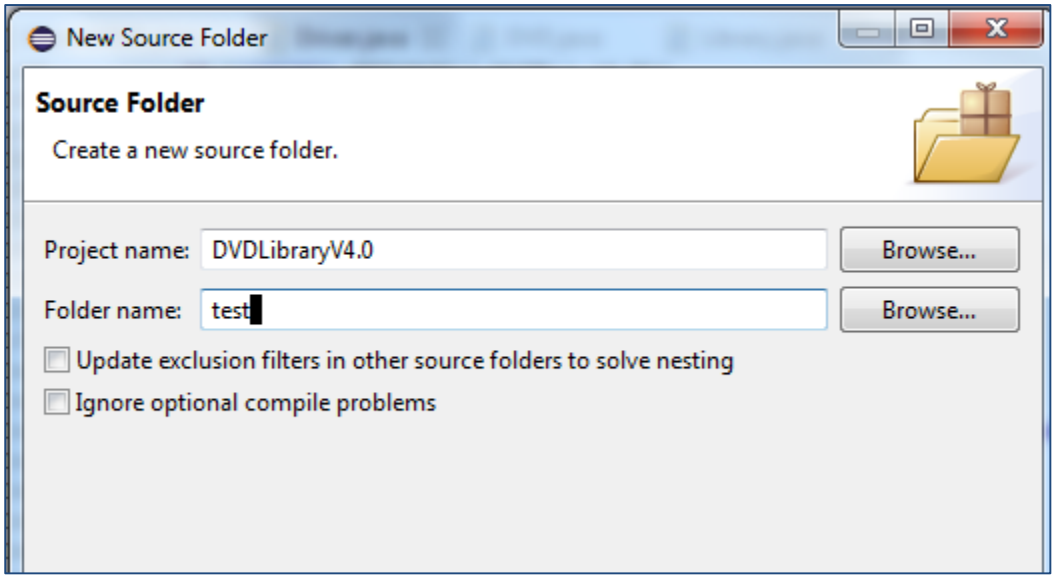
Testing DVD.java

DVD.java

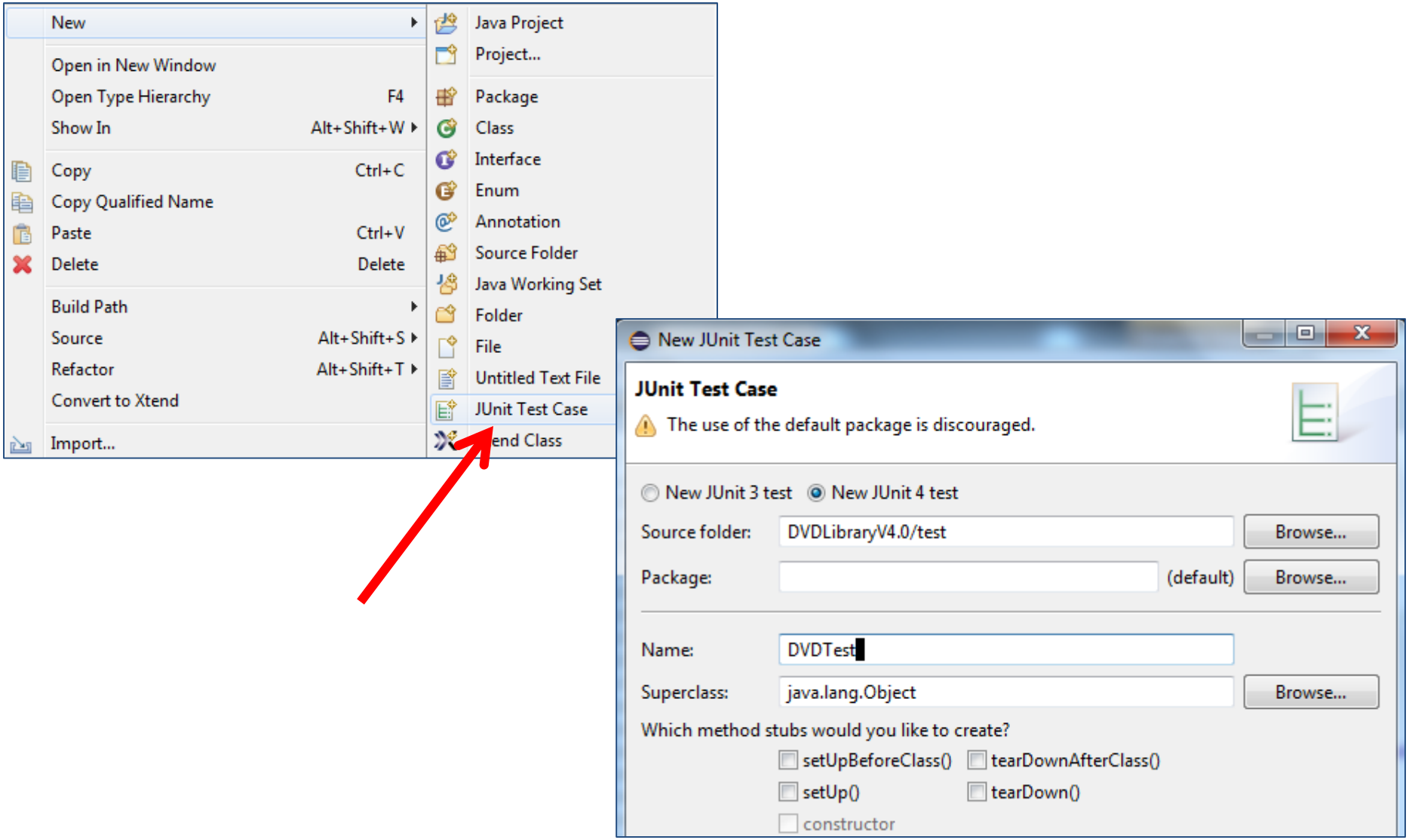
```
public class DVD
{
   private String title;

   public DVD(String title){
         setTitle(title);
   }

   public DVD(){
   }

  public String getTitle(){
        return title;
   }

   public void setTitle(String title){
       if (title.length() <= 20){
          this.title = title
       }
       else{
          this.title = title.substring(0,20);
       }
   }

   public String toString(){
       return "DVD Title is: " + title;
   }
}
```

In Eclipse, you should create a new "test" folder to hold your test classes

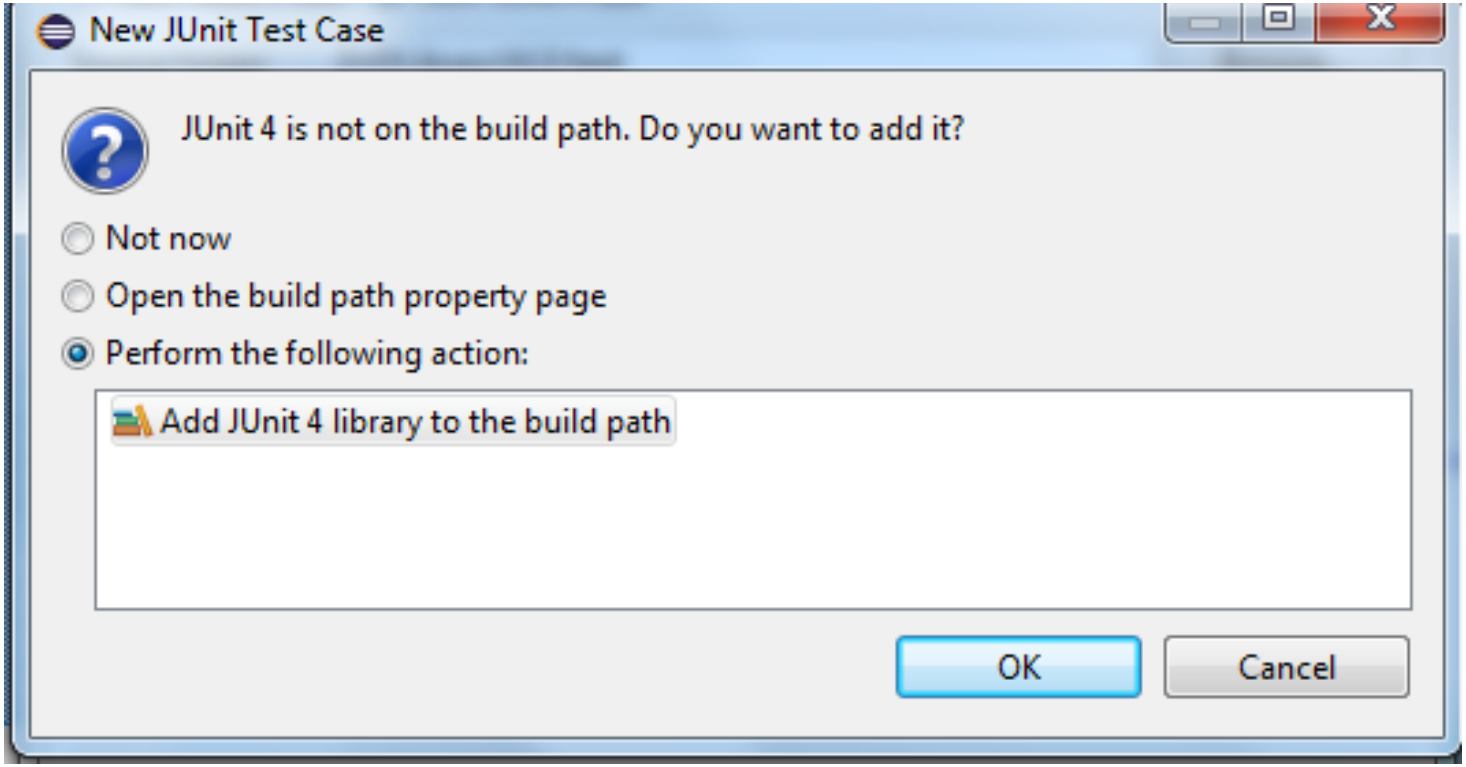




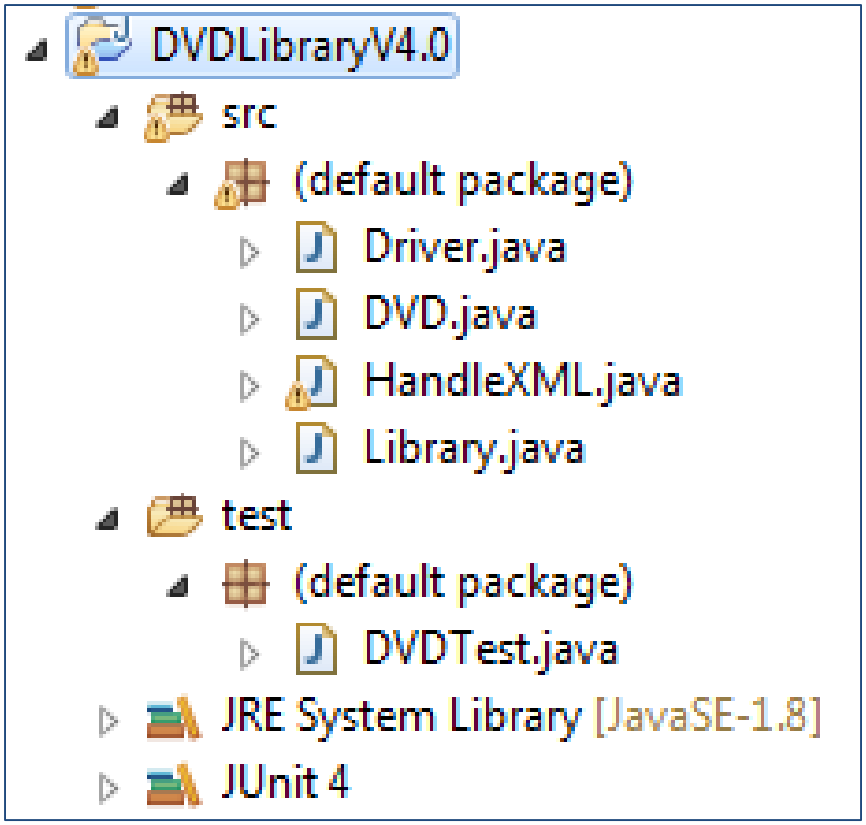
Within this “test” folder, create a new JUnit Test Case



You may be asked to add JUnit 4 to your build path → Click OK.



Revised Folder Structure

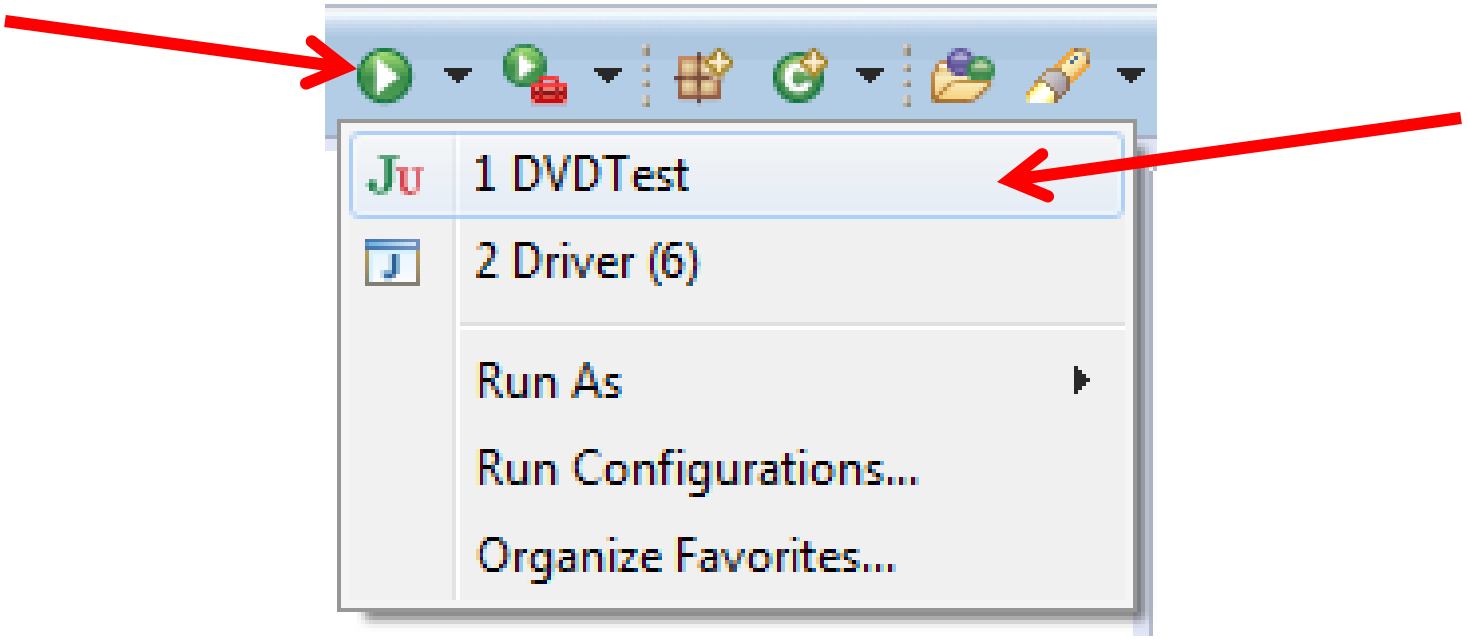


The generated DVDTest.java

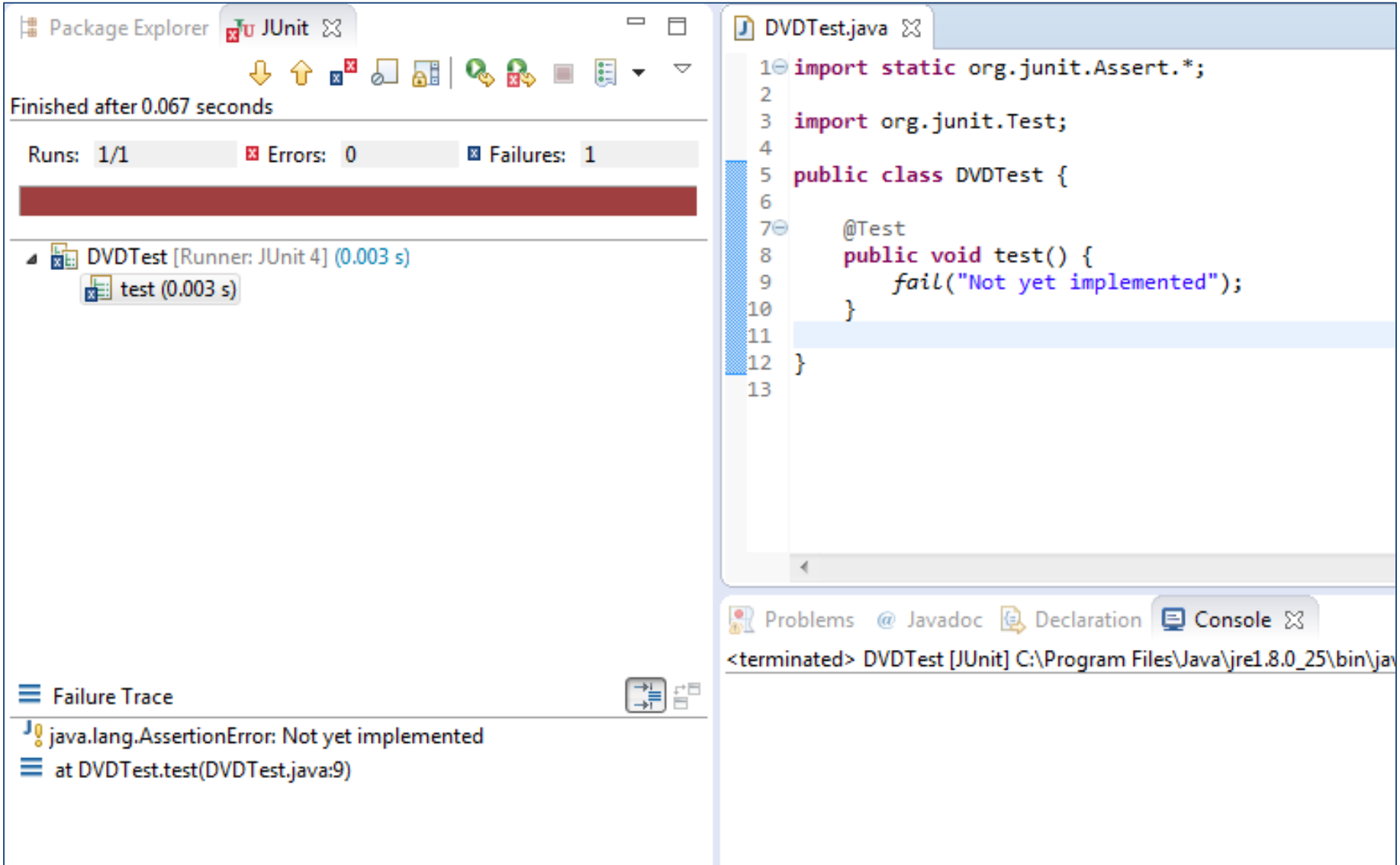
DVDTest.java

```
1  import static org.junit.Assert.*;
2
3  import org.junit.Test;
4
5  public class DVDTest {
6
7      @Test
8      public void test() {
9          fail("Not yet implemented");
10     }
11
12 }
13
```

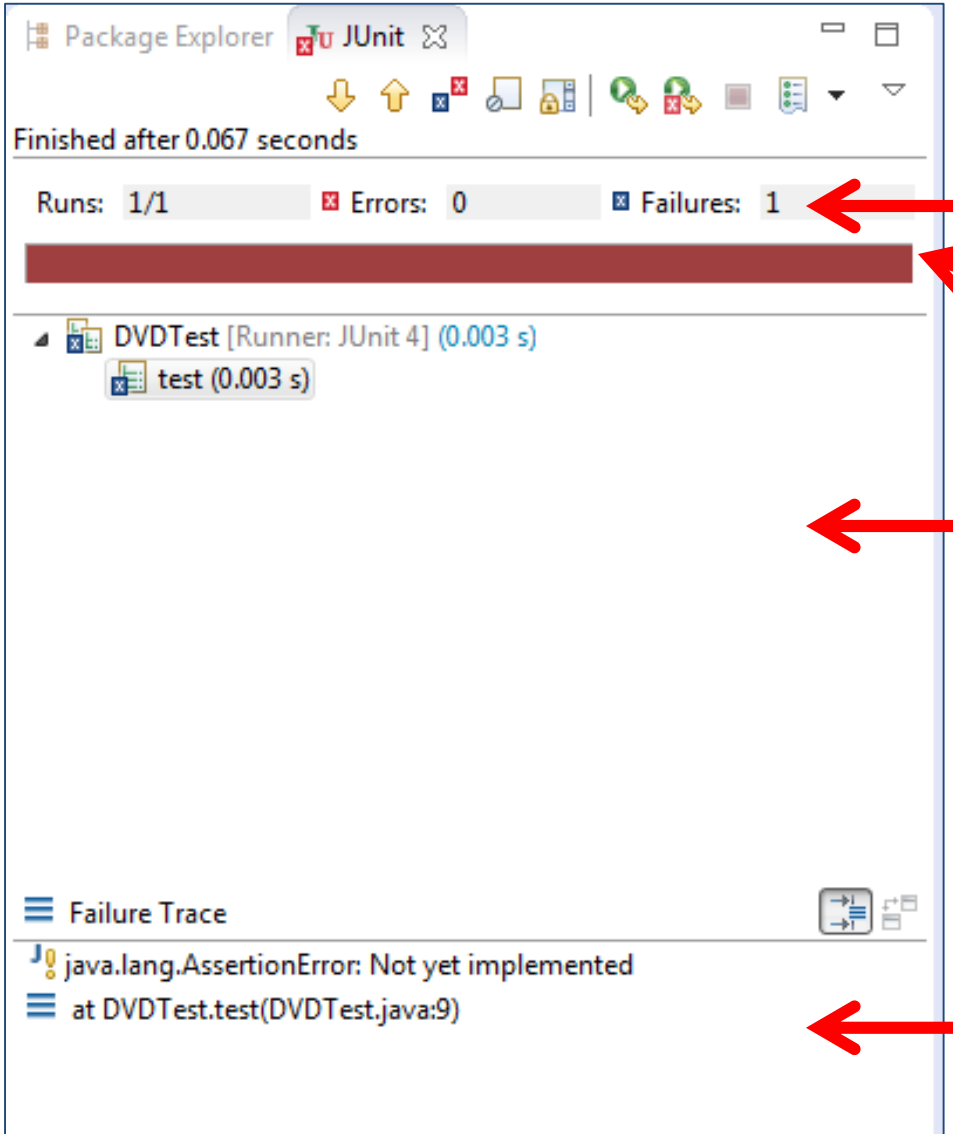
Running DVDTest.java



JUnit Results Pane (1)



JUnit Results Pane (2)



Number of tests run. Also the number that have errors (e.g. compile) and the number that failed.

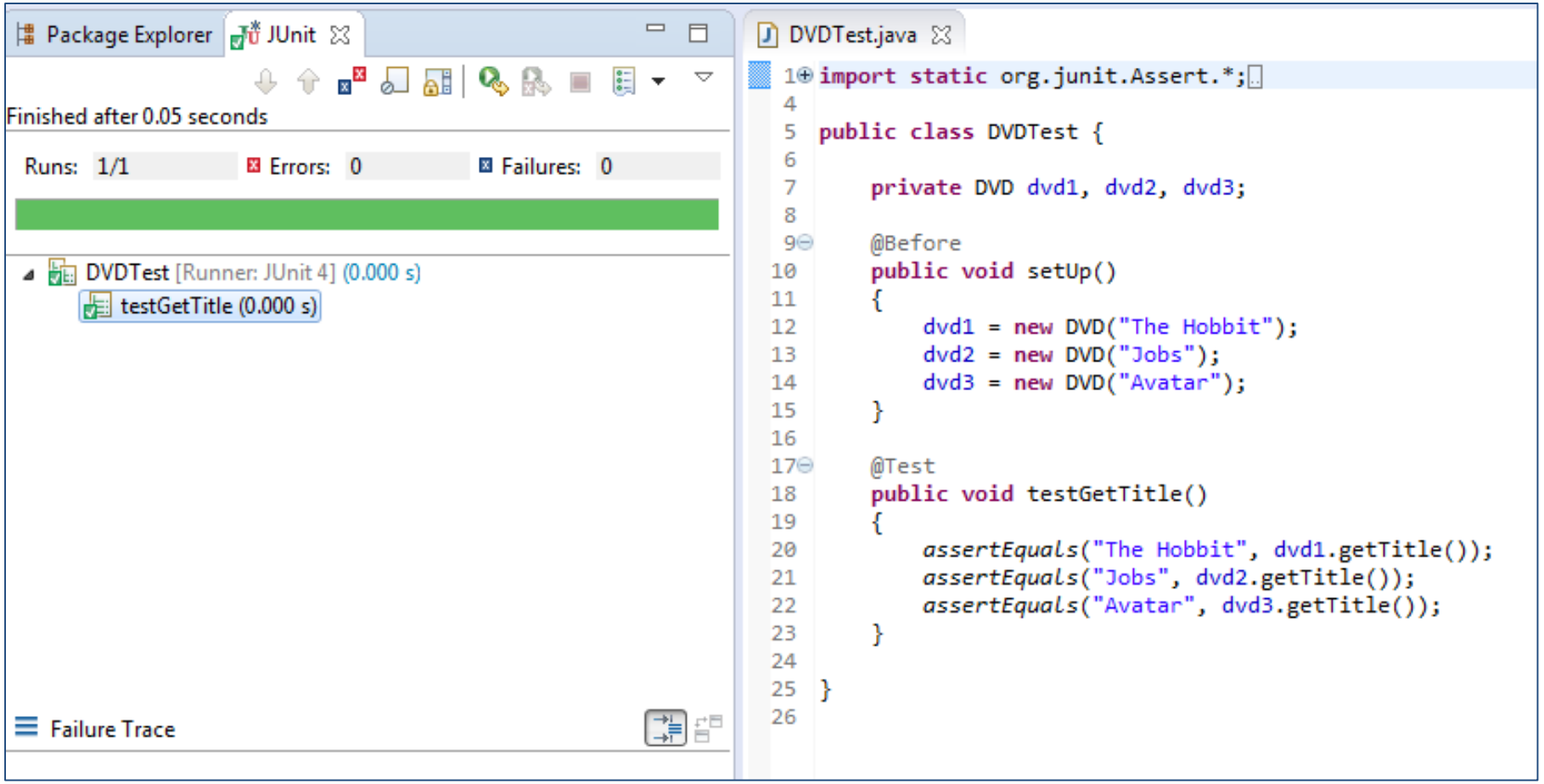
Test Bar indicating failure.

Itemised tests and their individual result.

When a failed test is selected in the above pane, the reason for failure appears here. Can be double clicked on for more information.

Testing getTitle()

```
public String getTitle() {
    return title;
}
```

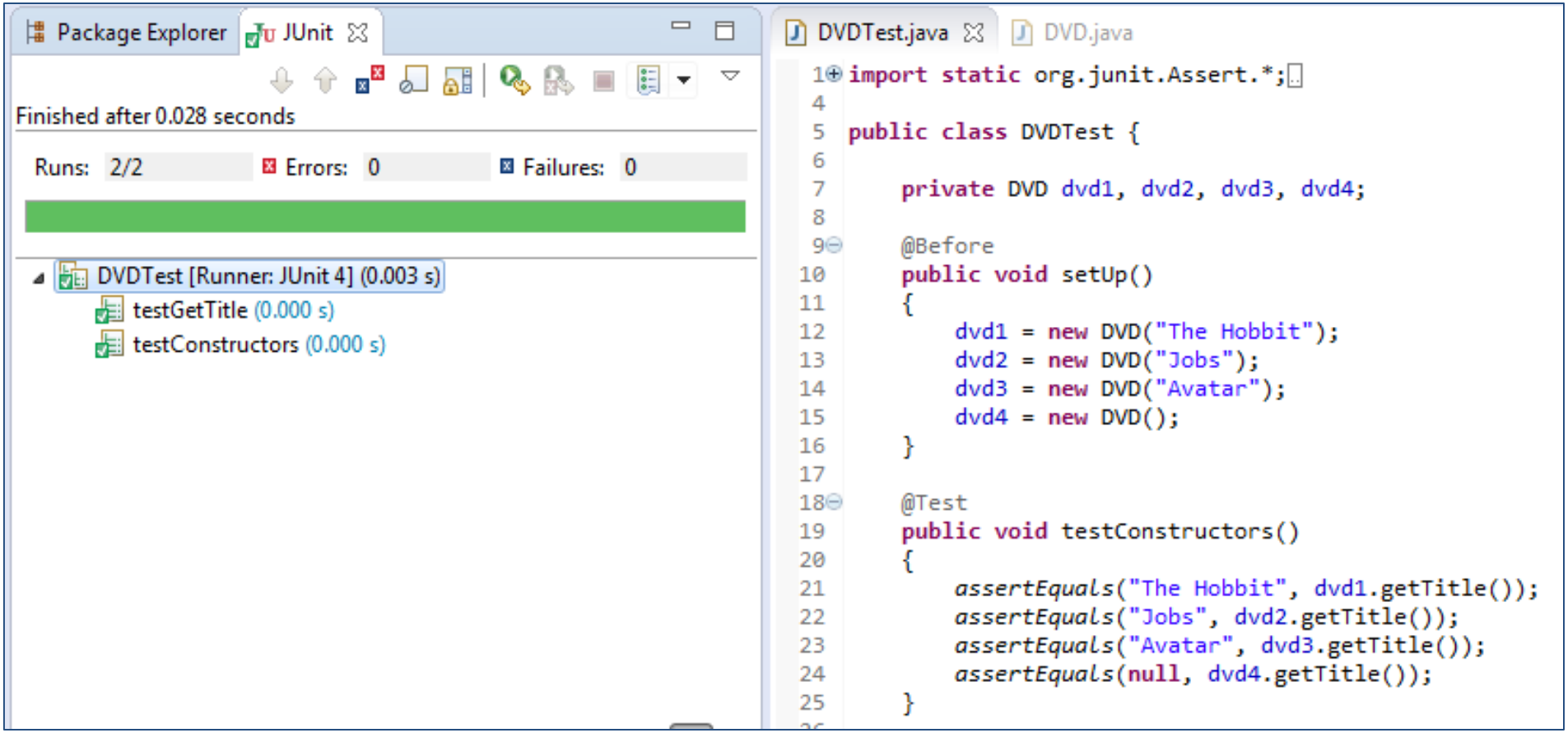


Testing Constructors (1)

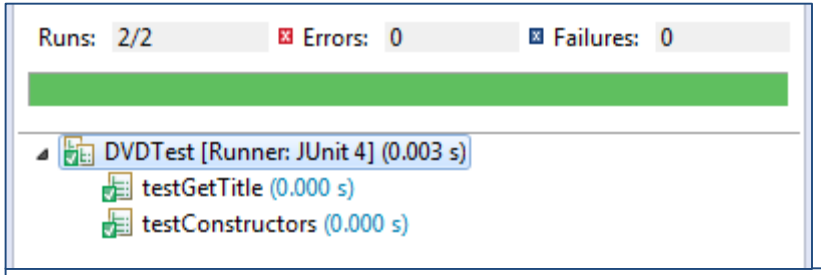
```
/**
 * Constructors for objects of class DVD
 */
public DVD(String title)
{
    setTitle(title);
}
```

```
public DVD(String title)
{
    if (title.length() <= 20){
        this.title = title;
    }
    else{
        this.title = title.substring(0,20);
    }
}

public DVD(){
}
```

Testing Constructors (2)



Testing Constructors (3)




```
@Before
public void setUp()
{
    dvd1 = new DVD("The Hobbit");
    dvd2 = new DVD("Jobs");
    dvd3 = new DVD("Avatar");
    dvd4 = new DVD();
}

@Test
public void testConstructors()
{
    assertEquals("The Hobbit", dvd1.getTitle());
    assertEquals("Jobs", dvd2.getTitle());
    assertEquals("Avatar", dvd3.getTitle());
    assertEquals(null, dvd4.getTitle());
}
```

```
/**
 * Constructors for objects of class DVD
 */
public DVD(String title)
{
    setTitle(title);
}
```

```
public DVD(String title)
{
    if (title.length() <= 20){
        this.title = title;
    }
    else{
        this.title = title.substring(0,20);
    }
}

public DVD(){
}
```

Our tests are successful, BUT have we tested every possible path in the constructors?

Testing Constructors (4)

DVDTest.java | DVD.java

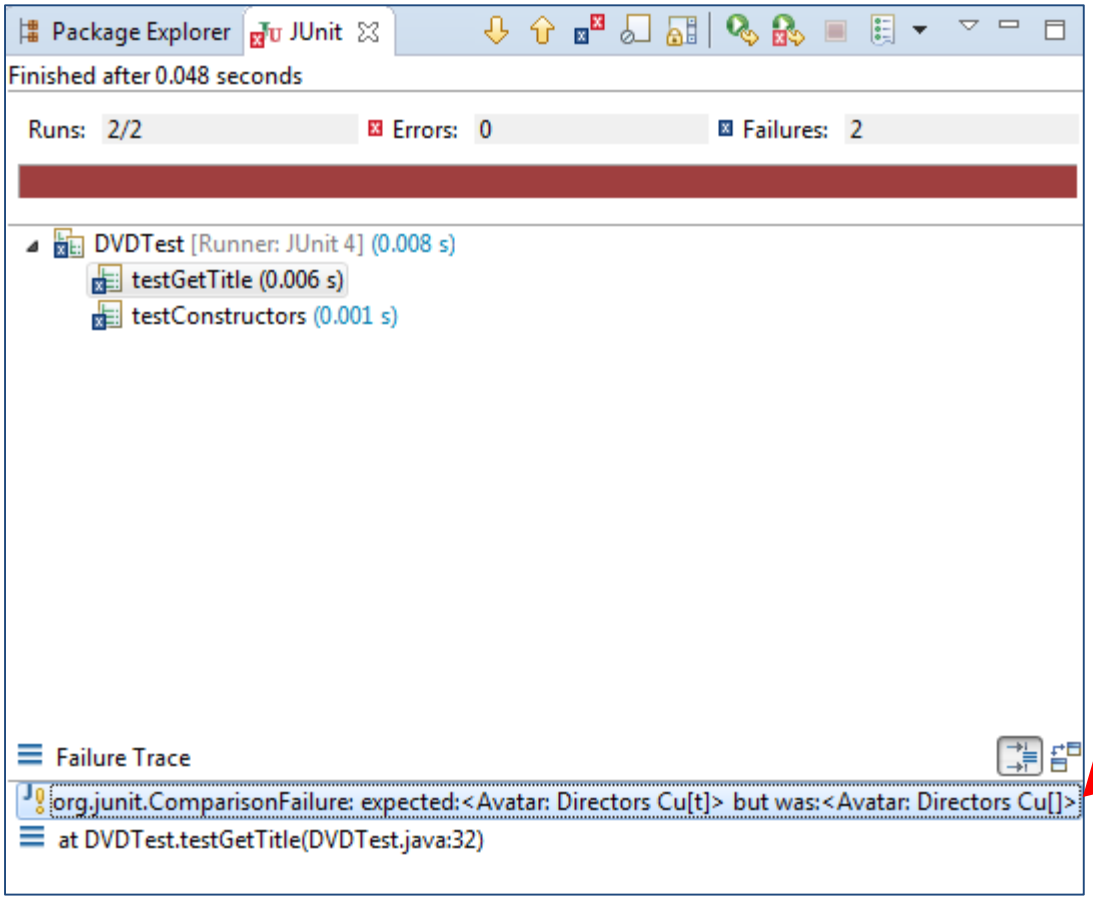
```
1  import static org.junit.Assert.*;
4
5  public class DVDTest {
6
7      private DVD dvd1, dvd2, dvd3, dvd4;
8
9      @Before
10     public void setUp()
11     {
12         dvd1 = new DVD("The Hobbit(Director)");  //title with 20 characters
13         dvd2 = new DVD("The Steve Jobs Film");   //title with 19 characters
14         dvd3 = new DVD("Avatar: Directors Cut"); //title with 21 characters
15         dvd4 = new DVD();
16     }
17
18     @Test
19     public void testConstructors()
20     {
21         assertEquals("The Hobbit(Director)", dvd1.getTitle());
22         assertEquals("The Steve Jobs Film", dvd2.getTitle());
23         assertEquals("Avatar: Directors Cut", dvd3.getTitle());
24         assertEquals(null, dvd4.getTitle());
25     }
26
```

What do you think of this approach?

Better?

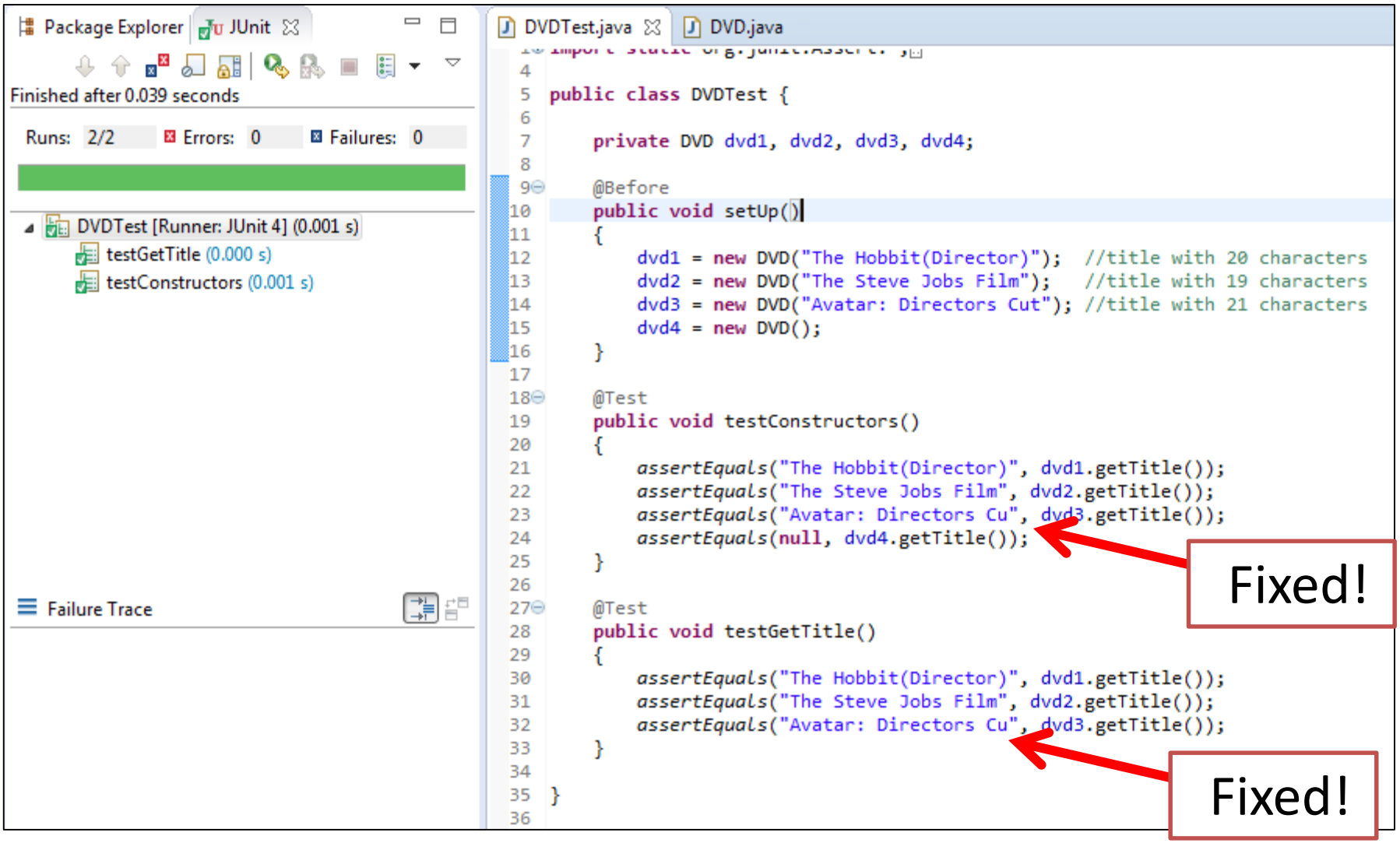
Will this test be successful?

Testing Constructors (5)



Can you see what went wrong?

Testing Constructors (6)

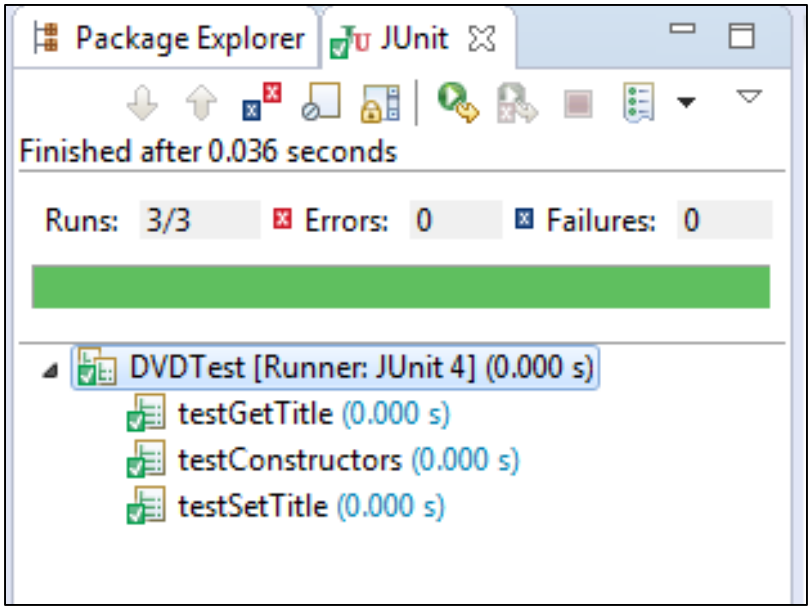


Testing setTitle method (1)

DVDTest.java | LibraryTest.java | DVD.java

```
1  import static org.junit.Assert.*;
5
6  public class DVDTest {
7
8      private DVD dvd1, dvd2, dvd3;
9
10     @Before
11     public void setUp(){
12         dvd1 = new DVD("The Hobbit(Director)");  //title with 20 characters
13         dvd2 = new DVD("The Steve Jobs Film");   //title with 19 characters
14         dvd3 = new DVD("Avatar: Directors Cut"); //title with 21 characters
15     }
16
18     public void tearDown(){
20
22     public void testConstructors(){
27
29     public void testGetTitle(){
34
35     @Test
36     public void testSetTitle(){
37         dvd1.setTitle("The Hobbit");
38         assertEquals ("The Hobbit", dvd1.getTitle());
39         dvd1.setTitle("The Hobbit (Director)");  //attempting to set title to 21 characters
40         assertEquals ("The Hobbit (Director", dvd1.getTitle());
41         dvd1.setTitle("The Hobbit(Director)");  //attempting to set title to 20 characters
42         assertEquals ("The Hobbit(Director)", dvd1.getTitle());
43         dvd1.setTitle("The Hobbit:Director");  //attempting to set title to 20 characters
44         assertEquals ("The Hobbit:Director", dvd1.getTitle());
45     }
46
48     public void testToString(){
53
54 }
```

Testing setTitle method (2)




```
35    @Test
36    public void testSetTitle(){
37        dvd1.setTitle("The Hobbit");
38        assertEquals ("The Hobbit", dvd1.getTitle());
39        dvd1.setTitle("The Hobbit (Director)");  //attempting to set title to 21 characters
40        assertEquals ("The Hobbit (Director", dvd1.getTitle());
41        dvd1.setTitle("The Hobbit(Director)");  //attempting to set title to 20 characters
42        assertEquals ("The Hobbit(Director)", dvd1.getTitle());
43        dvd1.setTitle("The Hobbit:Director");  //attempting to set title to 20 characters
44        assertEquals ("The Hobbit:Director", dvd1.getTitle());
45    }
46
```

Testing toString method (1)

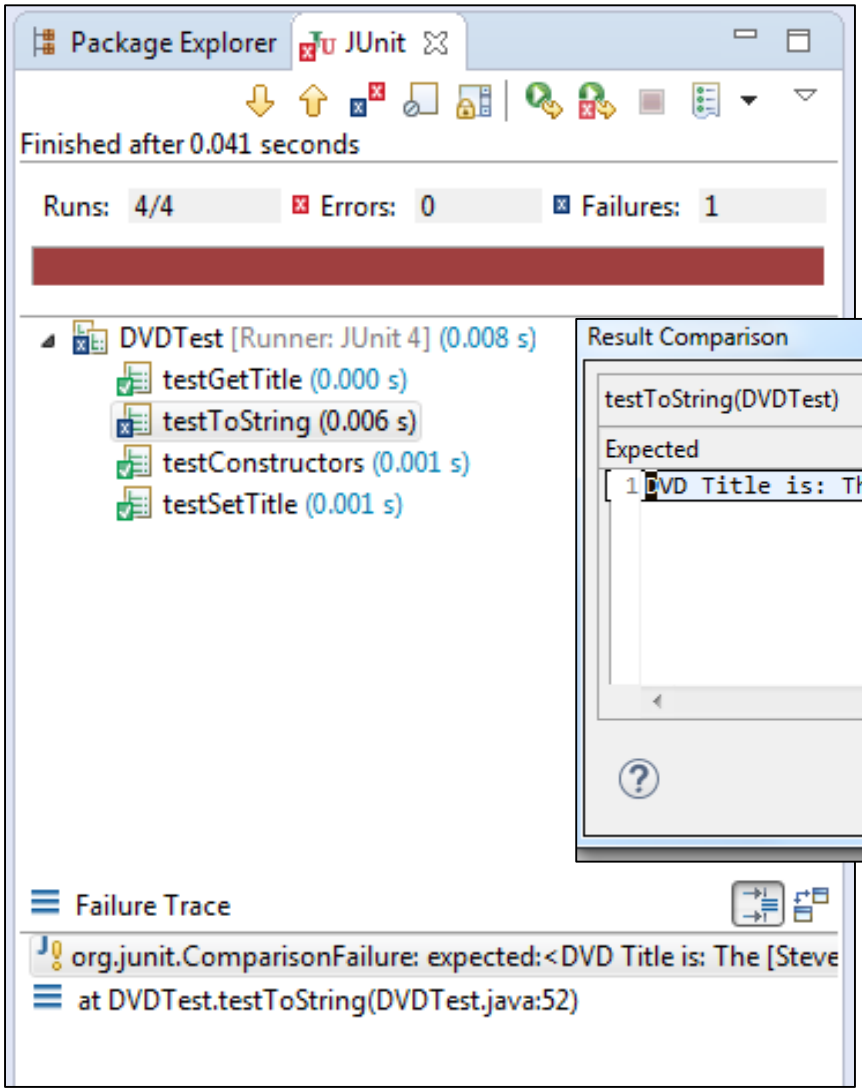
```
    /**
     * toString() - This method returns the title of the DVD.
     *
     * @return     the title of the DVD entered by the user
    */
    public String toString()
    {
      return "DVD Title is: " + title;
    }
}
```

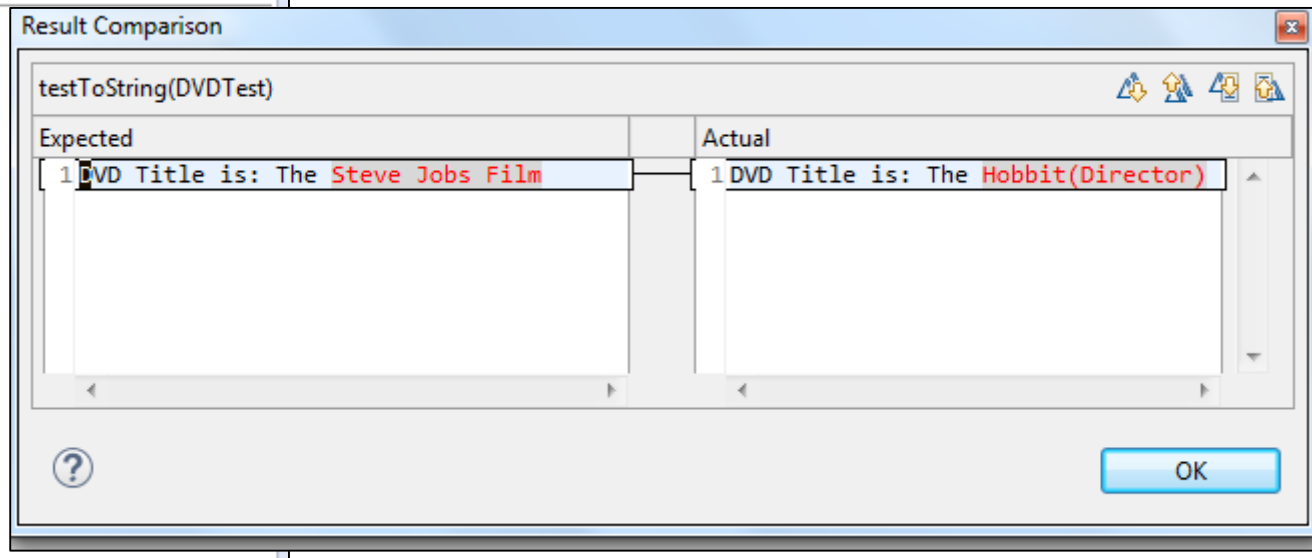
DVDTest.java | DVD.java

```
1  import static org.junit.Assert.*;
4
5  public class DVDTest {
6
7      private DVD dvd1, dvd2, dvd3, dvd4;
8
9      @Before
10     public void setUp()
11     {
12         dvd1 = new DVD("The Hobbit(Director)");  //title with 20 characters
13         dvd2 = new DVD("The Steve Jobs Film");   //title with 19 characters
14         dvd3 = new DVD("Avatar: Directors Cut"); //title with 21 characters
15         dvd4 = new DVD();
16     }
17
19     public void testConstructors()
26
28     public void testGetTitle()
34
36     public void testSetTitle()
47
48     @Test
49     public void testToString()
50     {
51         assertEquals("DVD Title is: The Hobbit(Director)", dvd1.toString());
52         assertEquals("DVD Title is: The Steve Jobs Film", dvd1.toString());
53         assertEquals("DVD Title is: Avatar: Directors Cu", dvd1.toString());
54     }
55
56 }
57
```

Does this test look right?
Will it run successfully?

Testing toString method (3)





DVDTest.java | DVD.java

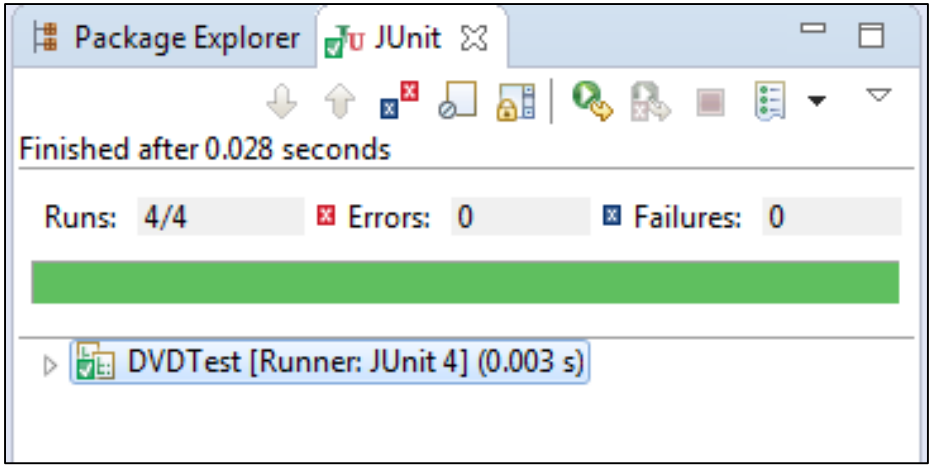
```
1  import static org.junit.Assert.*;
4
5  public class DVDTest {
6
7      private DVD dvd1, dvd2, dvd3, dvd4;
8
9      @Before
10     public void setUp()
11     {
12         dvd1 = new DVD("The Hobbit(Director)");  //title with 20 characters
13         dvd2 = new DVD("The Steve Jobs Film");   //title with 19 characters
14         dvd3 = new DVD("Avatar: Directors Cut"); //title with 21 characters
15         dvd4 = new DVD();
16     }
17
19     public void testConstructors()
26
28     public void testGetTitle()
34
36     public void testSetTitle()
47
48     @Test
49     public void testToString()
50     {
51         assertEquals("DVD Title is: The Hobbit(Director)", dvd1.toString());
52         assertEquals("DVD Title is: The Steve Jobs Film", dvd1.toString());
53         assertEquals("DVD Title is: Avatar: Directors Cu", dvd1.toString());
54     }
55
56 }
57
```

We are referencing the wrong dvd objects in the last two method calls.

Testing toString method (5)

```
48    @Test
49    public void testToString()
50    {
51        assertEquals("DVD Title is: The Hobbit(Director)", dvd1.toString());
52        assertEquals("DVD Title is: The Steve Jobs Film", dvd2.toString());
53        assertEquals("DVD Title is: Avatar: Directors Cu", dvd3.toString());
54    }
55
56 }
57
```

We are now referencing the correct dvd objects, so our test is now successful.



We have now successfully automated the testing of the entire DVD class

```
public class DVD
{
   private String title;

   public DVD(String title){
         setTitle(title);
   }

   public DVD(){
   }

  public String getTitle(){
        return title;
   }
```

```
   public void setTitle(String title){
       if (title.length() <= 20){
          this.title = title
       }
       else{
          this.title = title.substring(0,20);
       }
   }

   public String toString(){
       return "DVD Title is: " + title;
   }
}
```

Any Questions?

Except where otherwise noted, this content is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

For more information, please see http://creativecommons.org/licenses/by-nc/3.0/

Waterford Institute *of* Technology
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics
http://www.wit.ie/