

Test cases:

Input.	Converts.
String "good".	"good".
String "GOOD".	"GOOD".
One character "a".	"A".
Special character "?".	"?".

Test case 1:

eclipse-workspace - Lab9/src/mystringTest.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit X

Finished after 0.021 seconds

Runs: 1/1 Errors: 0 Failures: 1

mystringTest [Runner: JUnit 4] (0.000 s)

test (0.000 s)

```
1 import static org.junit.Assert.*;
4
5 public class mystringTest {
6
7     @Test
8     public void test() {
9         mystring st = new mystring();
10
11         st.setString("good");
12         assertEquals(" test case 1:", "good", st.invert());
13     }
14 }
15
16 }
17
```

Test case 2:

eclipse-workspace - Lab9/src/mystringTest.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer JUnit X

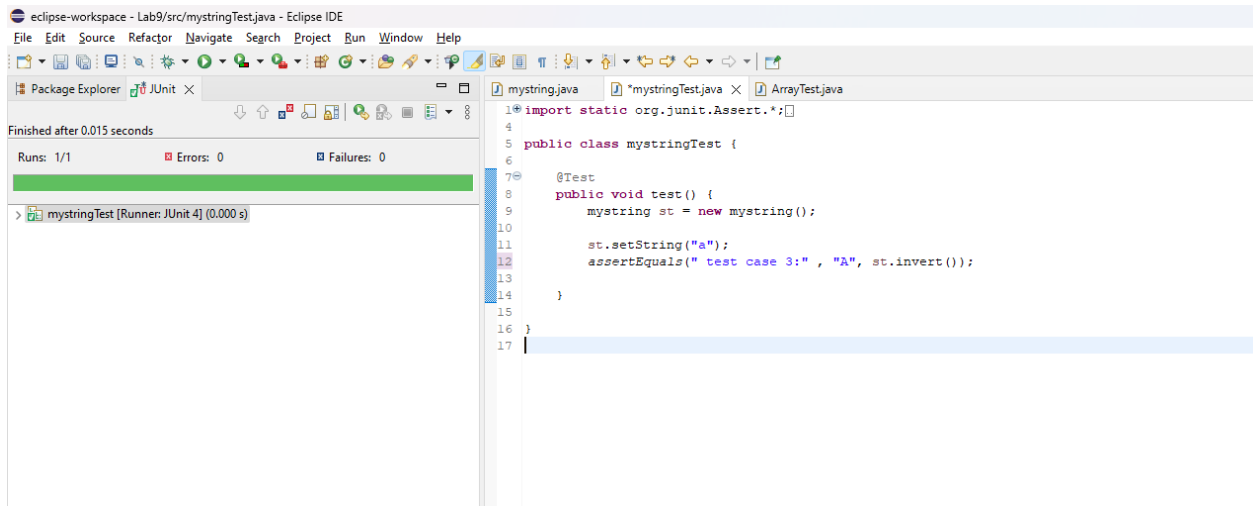
Finished after 0.014 seconds

Runs: 1/1 Errors: 0 Failures: 0

mystringTest [Runner: JUnit 4] (0.000 s)

```
1 import static org.junit.Assert.*;
4
5 public class mystringTest {
6
7     @Test
8     public void test() {
9         mystring st = new mystring();
10
11         st.setString("good");
12         assertEquals(" test case 2:", "GOOD", st.invert());
13     }
14 }
15
16 }
17
```

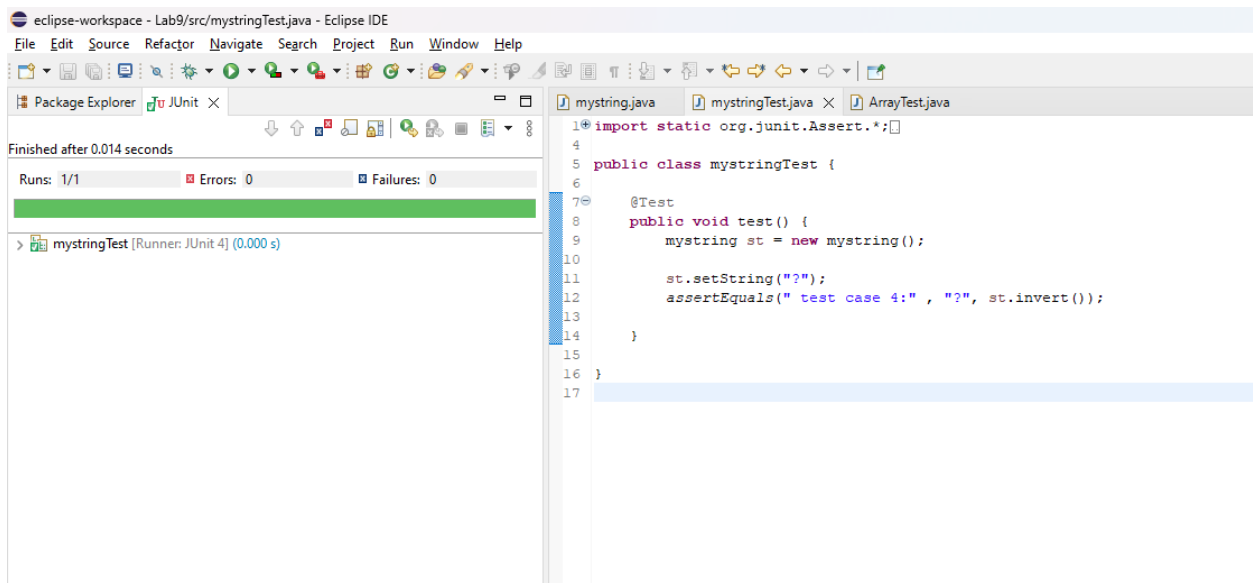
### Test case 3:



The screenshot shows the Eclipse IDE interface. The top toolbar includes icons for File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The Package Explorer on the left shows the project structure. The JUnit runner window displays the results of the test execution: "Finished after 0.015 seconds", "Runs: 1/1", "Errors: 0", and "Failures: 0". The console window shows the command: `> mystringTest [Runner: JUnit 4] (0.000 s)`. The main editor window shows the source code for `mystringTest.java`, which includes the following code:

```
1 import static org.junit.Assert.*;
4
5 public class mystringTest {
6
7     @Test
8     public void test() {
9         mystring st = new mystring();
10
11         st.setString("a");
12         assertEquals(" test case 3:" , "A", st.invert());
13     }
14
15 }
16
17
```

### Test case 4:



The screenshot shows the Eclipse IDE interface. The top toolbar includes icons for File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The Package Explorer on the left shows the project structure. The JUnit runner window displays the results of the test execution: "Finished after 0.014 seconds", "Runs: 1/1", "Errors: 0", and "Failures: 0". The console window shows the command: `> mystringTest [Runner: JUnit 4] (0.000 s)`. The main editor window shows the source code for `mystringTest.java`, which includes the following code:

```
1 import static org.junit.Assert.*;
4
5 public class mystringTest {
6
7     @Test
8     public void test() {
9         mystring st = new mystring();
10
11         st.setString("?");
12         assertEquals(" test case 4:" , "?", st.invert());
13     }
14
15 }
16
17
```