

```
1 import static org.junit.Assert.assertEquals;
11
12 /**
13  * JUnit test fixture for {@code CountPrimitiveCalls}'s static method
14  * countOfPrimitiveCalls.
15  *
16  * @author Put your name here
17  *
18  */
19 public final class CountPrimitiveCallsTest {
20
21     /**
22      * Creates and returns a {@code Statement} constructed from a given input
23      * file.
24      *
25      * @param fileName
26      *      the name of the file containing the statement
27      * @param block
28      *      flag to indicate whether to read an entire BLOCK (sequence of
29      *      statements) or a single statement
30      * @return the constructed statement
31      * @requires <pre>
32      * [fileName is the name of a file containing zero, one, or more
33      *  valid BL statements]
34      * </pre>
35      * @ensures createFromArgs = [statement(s) from file fileName]
36      */
37     private Statement createFromArgs(String fileName, boolean block) {
38         SimpleReader in = new SimpleReader1L(fileName);
39         Queue<String> tokens = Tokenizer.tokens(in);
40         in.close();
41         Statement s = new Statement1();
42         if (block) {
43             s.parseBlock(tokens);
44         } else {
45             s.parse(tokens);
46         }
47         return s;
48     }
49
50     @Test
51     public void test1true() {
52         Statement s1 = this.createFromArgs("data/test1.bl", true);
53         Statement s2 = this.createFromArgs("data/test1.bl", true);
54         int count = CountPrimitiveCalls.countOfPrimitiveCalls(s1);
55         assertEquals(25, count);
56         assertEquals(s2, s1);
57     }
58
59     @Test
60     public void test2true() {
61         Statement s1 = this.createFromArgs("data/test2.bl", true);
62         Statement s2 = this.createFromArgs("data/test2.bl", true);
63         int count = CountPrimitiveCalls.countOfPrimitiveCalls(s1);
64         assertEquals(2, count);
65         assertEquals(s2, s1);
66     }
}
```

CountPrimitiveCallsTest.java

Thursday, October 19, 2023, 1:01 PM

```
67  
68 }  
69
```