```java
import static org.junit.Assert.assertEquals;

import org.junit.Test;

import components.naturalnumber.NaturalNumber;
import components.naturalnumber.NaturalNumber2;

public class NNtoStringWithCommas1Test {

    /**
     *  * Calls the method under test.  *  * @param n  *            the number
     * to pass to the method under test  * @return the {@code String} returned
     * by the method under test  * @ensures <pre>
      * redirectToMethodUnderTest = [String returned by the method under test]
      * </pre>
     */
    private static String redirectToMethodUnderTest(NaturalNumber n) {
        return NNtoStringWithCommas1.toStringWithCommas(n);
    }

    /**
     * Test toStringWithCommas with input 0. boundary case as it tests at zero,
     * the smallest value allowed.
     */
    @Test
    public void testToStringWithCommas0() {
        NaturalNumber n = new NaturalNumber2(0);
        NaturalNumber expected = new NaturalNumber2(0);
        String actualOutput = redirectToMethodUnderTest(n);
        String expectedOutput = "0";

        assertEquals(expectedOutput, actualOutput);
        assertEquals(expected, n);
    }

    /**
     * Test toStringWithCommas with input 1. routine
     */
    @Test
    public void testToStringWithCommas1() {
        NaturalNumber n = new NaturalNumber2(1);
        NaturalNumber expected = new NaturalNumber2(1);
        String actualOutput = redirectToMethodUnderTest(n);
        String expectedOutput = "1";

        assertEquals(expectedOutput, actualOutput);
        assertEquals(expected, n);
    }

    /**
     * Test toStringWithCommas with input 100. routine
     */
    @Test
    public void testToStringWithCommas100() {
        NaturalNumber n = new NaturalNumber2(100);
        NaturalNumber expected = new NaturalNumber2(100);
        String actualOutput = redirectToMethodUnderTest(n);
```

```
58          String expectedOutput = "100";
59
60          assertEquals(expectedOutput, actualOutput);
61          assertEquals(expected, n);
62      }
63
64      /**
65       * Test toStringWithCommas with input 1000. boundary, first number to have a
66       * comma
67       */
68      @Test
69      public void testToStringWithCommas1000() {
70          NaturalNumber n = new NaturalNumber2(1000);
71          NaturalNumber expected = new NaturalNumber2(1000);
72          String actualOutput = redirectToMethodUnderTest(n);
73          String expectedOutput = "1,000";
74
75          assertEquals(expectedOutput, actualOutput);
76          assertEquals(expected, n);
77      }
78
79      /**
80       * Test toStringWithCommas with input 1234567. routine
81       */
82      @Test
83      public void testToStringWithCommas1234567() {
84          NaturalNumber n = new NaturalNumber2(1234567);
85          NaturalNumber expected = new NaturalNumber2(1234567);
86          String actualOutput = redirectToMethodUnderTest(n);
87          String expectedOutput = "1,234,567";
88
89          assertEquals(expectedOutput, actualOutput);
90          assertEquals(expected, n);
91
92      }
93
94  }
95
```