

PalindromPrueferTest.java

```

1 import static org.hamcrest.CoreMatchers.equalTo;
7
8 public class PalindromPrueferTest {
9     private String[] satzpalindromArray = {
10         //Quelle Wikipedia https://de.wikipedia.org/wiki/
        Liste_deutscher_Palindrome
11         "Ein Golf flog nie.                ",
12         "Ein Ledergurt trug Redel nie.      ",
13         "Ein Neger mit Gazelle zagt im Regen nie.    ",
14         "Ein Neger mit Gazette zagt im Regen nie.    ",
15         "Ein Regal mit Sirup pur ist im Lager nie.   ",
16         "Ein Siamese lese Mais nie.             ",
17         "Eine Blase salbe nie.                 ",
18         "Eine Gans nage nie.                  ",
19         "Eine güldne, gute Tugend: Lüge nie!      ",
20         "Eine Horde bedrohe nie!              "
21     };
22 };
23
24 private String[] keinStatzPalindromArray = {
25     "Dies ist ein Satzpalindrom Satz.",
26     "Das ist ein Test",
27 };
28 @Test
29 public void testIstPalindrom() {
30     // Erzeuge ein Objekt zum Testen.
31     PalindromPruefer einPalindromPruefer = new PalindromPruefer();
32
33     // Teste die null-Referenz.
34     Throwable ex = assertThrows(IllegalArgumentException.class, () ->
        {einPalindromPruefer.istPalindrom(null)});
35     assertThat("Null-Referenz wurde nicht korrekt verarbeitet.",
36         ex.getMessage(),
37         is(equalTo("Argument darf nicht null sein.")));
38
39     // Teste Zeichenkette der Laenge 0.
40     assertThat("String der Laenge 0 wurde nicht korrekt verarbeitet.",
41         einPalindromPruefer.istPalindrom(""),
42         is(equalTo(true)));
43     assertThat("String \"Radar\" wurde nicht korrekt verarbeitet.",
44         einPalindromPruefer.istPalindrom("Radar"),
45         is(equalTo(true)));
46
47     assertThat("String \"0tTo\" wurde nicht korrekt verarbeitet.",
48         einPalindromPruefer.istPalindrom("0tTo"),
49         is(equalTo(true)));
50     assertThat("String \"[{\" wurde nicht korrekt verarbeitet.", //
        // Ohne überprüfung des bereiches in lower() wüede hier true zurückgegeben
        // werden weil [ auf Z un { auf z folgt.
51         einPalindromPruefer.istPalindrom("[{"),
52         is(equalTo(false)));
53     assertThat("String \"Palindrom\" wurde nicht korrekt verarbeitet.",
54         einPalindromPruefer.istPalindrom("Palindrom"),

```

PalindromPrueferTest.java

```

55         is(equalTo(false)));
56
57
58     /*
59     * Tests hier ergaenzen.
60     */
61
62 }
63
64 @Test
65 void testIstSatzpalindrom() {
66
67     // Erzeuge ein Objekt zum Testen.
68     PalindromPruefer einPalindromPruefer = new PalindromPruefer();
69
70     // Teste die null-Referenz.
71     Throwable ex = assertThrows(IllegalArgumentException.class, () ->
{einPalindromPruefer.istSatzPalindrom(null)});
72     assertThat("Null-Referenz wurde nicht korrekt verarbeitet.",
73         ex.getMessage(),
74         is(equalTo("Argument darf nicht null sein.")));
75
76     for (String satzPalindrom : satzpalindromArray) {
77         assertThat(String.format("String \"%s\" wurde nicht korrekt
verarbeitet.", satzPalindrom),
78             einPalindromPruefer.istSatzPalindrom(satzPalindrom),
79             is(equalTo(true)));
80     }
81
82     for (String satzPalindrom : keinStatzPalindromArray) {
83         assertThat(String.format("String \"%s\" wurde nicht korrekt
verarbeitet.", satzPalindrom),
84             einPalindromPruefer.istSatzPalindrom(satzPalindrom),
85             is(equalTo(false)));
86     }
87 }
88
89 }
90

```