

 Evgen959 / Advanced\_Backend Public[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#)[Advanced\\_Backend](#) / [Lesen 031](#) / [code](#) / [hw\\_na\\_les030\\_1](#) / [test](#) / MainTest.java 

...



Evgen959 newLes31

709f570 · 3 days ago



85 lines (68 loc) · 2.29 KB

[Code](#) [Blame](#)[Raw](#)

```
1  import org.junit.jupiter.api.Assertions;
2  import org.junit.jupiter.api.Disabled;
3  import org.junit.jupiter.api.DisplayName;
4  import org.junit.jupiter.api.Test;
5
6  import java.util.ArrayList;
7  import java.util.Collections;
8  import java.util.List;
9
10 import static org.junit.jupiter.api.Assertions.*;
11
12 class MainTest {
13
14
15     @Test
16     void parsePersons() {
17         List<String> data = List.of("John,18",
18             "Jack,32",
19             "Anna,21",
20             "Olga,25",
21             "Gektar,21"
22         );
23         List<Person> expectedResult = List.of(
24             new Person("John",18),
25             new Person("Jack",32),
26             new Person("Anna",21),
27             new Person("Olga",25),
28             new Person("Gektar",21)
29         );
30
31         List<Person> actualResult = Main.parsePersons(data);
32
33         Assertions.assertEquals(expectedResult, actualResult);
34     }
35
36     @Test
37     @DisplayName("input empty list")
38     void testParsePersons() {
39         //List<String> data = Collections.EMPTY_LIST;
40         List<String> data = new ArrayList<>();
41         List<Person> expectedResult = new ArrayList<>();
42
43         List<Person> actualResult = Main.parsePersons(data);
44
45         Assertions.assertEquals(expectedResult, actualResult);
46
47     },
```

```
47     }
48
49     @Test
50     @DisplayName("input null")
51     void parsePersons_List_Null() {
52         //List<String> data = Collections.EMPTY_LIST;
53         List<String> data = null;
54         List<Person> expectedResult = new ArrayList<>();
55
56         List<Person> actualResult = Main.parsePersons(data);
57
58         Assertions.assertEquals(expectedResult, actualResult);
59     }
60
61     @Test
62     void parsePersons_If_listWithNullAndEmptyValues() {
63         List<String> data = new ArrayList<>(List.of(
64             "John,18",
65             "",
66             "Jack,32",
67             "Anna,21",
68             "Olga,25",
69             "Jack32",
70             "Gektar,21"
71         ));
72         data.add(null);
73         List<Person> expectedResult = List.of(
74             new Person("John",18),
75             new Person("Jack",32),
76             new Person("Anna",21),
77             new Person("Olga",25),
78             new Person("Gektar",21)
79         );
80
81         List<Person> actualResult = Main.parsePersons(data);
82
83         Assertions.assertEquals(expectedResult, actualResult);
84     }
85 }
```