

 [Evgen959](#) / [Advanced_Backend](#) Public[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#)[Advanced_Backend](#) / [Lesen 043](#) / [code](#) / [d06_11_1](#) / [src](#) / [Main.java](#) 

Evgen959 newLes43

d71a156 · 9 hours ago



61 lines (48 loc) · 1.91 KB

Code

Blame

Raw



```
1  import java.util.ArrayList;
2  import java.util.List;
3  import java.util.function.Consumer;
4  import java.util.function.Function;
5  import java.util.function.UnaryOperator;
6
7  public class Main {
8      public static void main(String[] args) {
9          List<String> list = List.of("jack", "John", "Lena", "katy");
10         System.out.println(transformList1(list, s -> s.toUpperCase()));
11         System.out.println(transformList2(list, s -> s.toUpperCase()));
12
13         System.out.println(transformList2(List.of("qwe", "rtytgfg", "dgfgsasfg"), s -> s.
14
15         List<Person> people = List.of(
16             new Person("Jack", 10),
17             new Person("John", 17),
18             new Person("Ann", 8),
19             new Person("Mick", 13),
20             new Person("Oleg", 10)
21         );
22
23         List<String> list1 = transformList2(people, p -> p.getName());
24         System.out.println(list1);
25
26         System.out.println("-----");
27         people.forEach((p)-> System.out.println(p));
28
29         System.out.println("-----");
30         people.forEach(p -> p.agePlasYear()); // Consumer<E>
31         people.forEach((p)-> System.out.println(p));
32
33
34
35
36     }
```

```
37
38     /*public static List<String> transformList(List<String> list, Transformer transformer)
39         List<String> result = new ArrayList<>();
40         for (String str: list){
41             result.add(transformer.apply(str));
42         }
43         return result;
44     }*/
45
46     ✓ public static <E> List<E> transformList1(List<E> list, UnaryOperator<E> transformer){
47         List<E> result = new ArrayList<>();
48         for (E str: list){
49             result.add(transformer.apply(str));
50         }
51         return result;
52     }
53
54     ✓ public static <E,R> List<R> transformList2(List<E> list, Function<E, R> transformer){
55         List<R> result = new ArrayList<>();
56         for (E str: list){
57             result.add(transformer.apply(str));
58         }
59         return result;
60     }
61 }
```