

```

1 import philatelist.Philatelist;
2 import philatelist.PostStamp;
3
4 import java.util.ArrayList;
5 import java.util.List;
6
7 public class PhilatelistImpl implements Philatelist {
8
9     private final List<PostStamp> postStamps;
10
11     // public PhilatelistImpl(List<PostStamp> postStamps) { //netinkamas nors ir sugeneruotas
12     //     this.postStamps = postStamps;
13     // }
14
15     public PhilatelistImpl() {
16         this.postStamps = new ArrayList<>();
17     }
18
19     @Override
20     public void addToCollection(PostStamp postStamp) {
21         if (postStamp == null || postStamp.getName() == null || postStamp.getName().equals("")) { //
            patikrinimas ar turi duomenis + ar ne empty
22             throw new IllegalArgumentException("PostStamp doesnt not exist.. Nothing to addd");
23         }
24
25         if (!postStamps.contains(postStamp)) { // patikrinimas ar jau nera
26             postStamps.add(postStamp);
27         }
28     }
29
30     @Override
31     public PostStamp getTheMostExpensivePostStamByMarketValue() {
32         // netikrinam nei null nei dubliavimosi nes metodas nepriima joki argumentu
33

```

```
34 PostStamp mostExpensiveStamp = postStamps.get(0); //apriori pirmas pats brangiausias
35 for (int i = 0; i < postStamps.size(); i++) { //lyginam per sarasa ar kuris nera brangeesnos
36     if (mostExpensiveStamp.getMarketPrice() < postStamps.get(i).getMarketPrice()) {
37         mostExpensiveStamp = postStamps.get(i); // jei itasis brangesnis tai jis ir tampa maksimaliu
38     }
39 }
40 return mostExpensiveStamp;
41 }
42
43 @Override
44 public double getAveragePostStampPrice() {
45     double sum = 0;
46     for (PostStamp postStamp : postStamps) { //iteruojam per sarasa ir kiekvieno zenklo kaina pridedam
47         sum += postStamp.getMarketPrice();
48     }
49     double average = sum / postStamps.size(); // nustatom kainu vidurki
50     return average;
51 }
52
53 @Override
54 public int getNumberOfPostStampsInCollection() {
55     int count = postStamps.size();
56     return count;
57 }
58 }
59
```