Advanced_Backend / Lesen 039 / code / hw_na_les038_1 / src / Main.java

Evgen959 newLes39 82fdf81 · 6 minutes ago

76 lines (63 loc) · 3.22 KB

Code Blame Raw

```java
/* 1
Предположим, вы пишете программу учета студентов компьютерных курсов.
У вас есть списки студентов нескольких групп (т.е. несколько List<Student>).
Некоторые студенты посещают занятия в нескольких группах.
Ваша задача получить список (List) всех студентов школы.*/
/* 2
Чуть сложнее. У вас есть 3 списка студентов:
    - список студентов прослушавших курс бэкенд
    - список студентов прослушавших курс фронтенд
    - список студентов прослушавших курс qa
Ваша задача получить список студентов прослушавших все три курса.*/


import java.util.*;

public class Main {
    public static void main(String[] args) {


        System.out.println("Hello world!");
    }

    public static List<Student> getAllStudents(List<Student>...lists){
        Set<Student> set = new HashSet<>();
        for (List<Student> list : lists){
            set.addAll(list);
        }
        return new ArrayList<>(set);
    }

    public static List<Student> getAllCoursesStudents1(List<Student> backEnd,
                                                       List<Student> frontEnd,
                                                       List<Student> qa){
        Set<Student> set1 = new HashSet<>(backEnd);
        Set<Student> set2 = new HashSet<>(frontEnd);
        Set<Student> set3 = new HashSet<>(qa);

```

```
38              set1.retainAll(set2); // в set1 останустя только повторяющиеся студенты
39              set1.retainAll(set3); // в set1 останустя только повторяющиеся студенты
40
41              return new ArrayList<>(set1);
42          }
43
44          public static List<Student> getAllCoursesStudents2(List<Student> backEnd,
45                                                             List<Student> frontEnd,
46                                                             List<Student> qa){
47              Set<Student> set1 = new HashSet<>(backEnd);
48              Set<Student> set2 = new HashSet<>(frontEnd);
49              Set<Student> set3 = new HashSet<>(qa);
50
51              Iterator<Student> iterator = set1.iterator();
52              while (iterator.hasNext()){
53                  Student student = iterator.next();
54                  if (!set2.contains(student) || !set3.contains(student)){
55                      iterator.remove();
56                  }
57              }
58              return new ArrayList<>(set1);
59          }
60
61          public static List<Student> getAllCoursesStudents(List<Student> backEnd,
62                                                            List<Student> frontEnd,
63                                                            List<Student> qa){
64              Set<Student> set1 = new HashSet<>(backEnd);
65              Set<Student> set2 = new HashSet<>(frontEnd);
66              Set<Student> set3 = new HashSet<>(qa);
67              ArrayList<Student> result = new ArrayList<>();
68
69              for (Student student: set1){
70                  if (set2.contains(student) && set3.contains(student)){
71                      result.add(student);
72                  }
73              }
74              return result;
75          }
76      }
```