15.05.2018 Алгоритмы для работы с коллекциями 3

package edu.javacourse.collection;

import java.util.ArrayList;

import java.util.Collections;

import java.util.List;

/\*\*

\* Демонстрация различных операций над списком

\*/

public class Main {

    public static void main(String[] args) {

        List<MyClass> list = new ArrayList<MyClass>();

        list.add(new MyClass("Василий"));

        list.add(new MyClass("Павел"));

        list.add(new MyClass("Андрей"));

        list.add(new MyClass("Андрей"));

        list.add(new MyClass("Петр"));

        list.add(new MyClass("Анжелика"));

        printCollection("Оригинал", list);

        // Смешивание

        Collections.shuffle(list);

        printCollection("Смешивание", list);

        // Обратный порядок

        Collections.reverse(list);

        printCollection("Обратный порядок", list);

        // "Проворачивание" на определенное количество

        Collections.rotate(list, 2); // Число может быть отрицательным - тогда порядок будет обратный

        printCollection("Проворачивание", list);

        // Обмен элементов

        Collections.swap(list, 0, list.size()-1);

        printCollection("Обмен элементов", list);

        // Замена

        Collections.replaceAll(list, new MyClass("Андрей"), new MyClass("Алексей"));

        printCollection("Замена", list);

        // Копирование - здесь обязательно надо иметь нужные размеры

        List<MyClass> list2 = new ArrayList<MyClass>(list.size());

        // Поэтому заполняем список. Хоть чем-нибудь.

        for(MyClass mc : list) {

            list2.add(null);

        }

        // Компируем из правого аргумента в левый

        Collections.copy(list2, list);

        printCollection("Копирование", list2);

        // Полная замена

        Collections.fill(list2, new MyClass("Антон"));

        printCollection("Полная замена", list2);

    }

    private static void printCollection(String title, List<MyClass> list) {

        System.out.println(title);

        for(MyClass mc : list) {

            System.out.println("Item:" + mc);

        }

        System.out.println();

    }

}

package edu.javacourse.collection;

/\*\*

\* пример произвольного класса

\*/

public class MyClass {

    private String name;

    public MyClass(String name) {

        this.name = name;

    }

    @Override

    public String toString() {

        return name;

    }

    // Без методов equals и hashCode не будет замены в списках

    @Override

    public boolean equals(Object obj) {

        if (obj == null) {

            return false;

        }

        if (getClass() != obj.getClass()) {

            return false;

        }

        final MyClass other = (MyClass) obj;

        if ((this.name == null) ? (other.name != null) : !this.name.equals(other.name)) {

            return false;

        }

        return true;

    }

    @Override

    public int hashCode() {

        int hash = 7;

        hash = 29 \* hash + (this.name != null ? this.name.hashCode() : 0);

        return hash;

    }

}

Оригинал

Item:Василий

Item:Павел

Item:Андрей

Item:Андрей

Item:Петр

Item:Анжелика

Смешивание

Item:Петр

Item:Павел

Item:Василий

Item:Андрей

Item:Анжелика

Item:Андрей

Обратный порядок

Item:Андрей

Item:Анжелика

Item:Андрей

Item:Василий

Item:Павел

Item:Петр

Проворачивание

Item:Павел

Item:Петр

Item:Андрей

Item:Анжелика

Item:Андрей

Item:Василий

Обмен элементов

Item:Василий

Item:Петр

Item:Андрей

Item:Анжелика

Item:Андрей

Item:Павел

Замена

Item:Василий

Item:Петр

Item:Алексей

Item:Анжелика

Item:Алексей

Item:Павел

Копирование

Item:Василий

Item:Петр

Item:Алексей

Item:Анжелика

Item:Алексей

Item:Павел

Полная замена

Item:Антон

Item:Антон

Item:Антон

Item:Антон

Item:Антон

Item:Антон

СБОРКА УСПЕШНО ЗАВЕРШЕНА (общее время: 0 секунд)