15.05.2018 Коллекции начало Сортировка коллекции 4

package edu.javacourse.collection;

import java.util.ArrayList;

import java.util.Collections;

import java.util.List;

public class Main

{

    public static void main(String[] args) {

        System.out.println("Вариант сортировки через Comparator");

        List<MyClassCompare> listSort = new ArrayList<MyClassCompare>();

        listSort.add(new MyClassCompare("Василий"));

        listSort.add(new MyClassCompare("Павел"));

        listSort.add(new MyClassCompare("Андрей"));

        listSort.add(new MyClassCompare("Андрей"));

        listSort.add(new MyClassCompare("Петр"));

        listSort.add(new MyClassCompare("Анжелика"));

        // Сортировка через Comparable

        printCollection1("Без сортировки", listSort);

        Collections.sort(listSort);

        printCollection1("После сортировки", listSort);

        System.out.println("Вариант сортировки через Comparator");

        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"));

        // Сортировка с классом Comparator

        printCollection2("Без сортировки", list);

        Collections.sort(list, new MyClassComparator());

        printCollection2("После сортировки", list);

    }

    private static void printCollection1(String title, List<MyClassCompare> list) {

        System.out.println(title);

        for (MyClassCompare mc : list) {

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

        }

        System.out.println();

    }

    private static void printCollection2(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;

public class MyClass

{

    private String name;

    public MyClass(String name) {

        this.name = name;

    }

    @Override

    public String toString() {

        return name;

    }

}

package edu.javacourse.collection;

public class MyClassCompare implements Comparable<MyClassCompare>

{

    private String name;

    public MyClassCompare(String name) {

        this.name = name;

    }

    // этот метод как раз и сравнивает текущий объект с другим,

    // который передается в качестве аргумента

    public int compareTo(MyClassCompare o) {

        return name.compareTo(o.name);

    }

    @Override

    public String toString() {

        return name;

    }

}

package edu.javacourse.collection;

import java.util.Comparator;

public class MyClassComparator implements Comparator

{

    public int compare(Object o1, Object o2) {

        return o1.toString().compareTo(o2.toString());

    }

}

run:

Вариант сортировки через Comparator

Без сортировки

Item:Василий

Item:Павел

Item:Андрей

Item:Андрей

Item:Петр

Item:Анжелика

После сортировки

Item:Андрей

Item:Андрей

Item:Анжелика

Item:Василий

Item:Павел

Item:Петр

Вариант сортировки через Comparator

Без сортировки

Item:Василий

Item:Павел

Item:Андрей

Item:Андрей

Item:Петр

Item:Анжелика

После сортировки

Item:Андрей

Item:Андрей

Item:Анжелика

Item:Василий

Item:Павел

Item:Петр

СБОРКА УСПЕШНО ЗАВЕРШЕНА (общее время: 0 секунд)