

Сортиране на списъци в Java

В този урок ще покажем как се сортират списъци.

Подготовка

За урока ще ни е нужен един списък, който ще направим отново като превърнем масив от низове в списък.

```
String[] crap = {"apple", "lemons", "bacon", "youtube"};
List<String> list1 = new LinkedList<String>(Arrays.asList(crap));
```

Сортиране

След като вече имаме списък с различни низове в него, нека да го сортираме. Това става чрез командата **sort()**, която се намира в класа **Collections**. За първи и единствен параметър подаваме нашият списък.

```
Collections.sort(list1);
```

Един още по-лесен начин за принтиране на списъци е просто като напишем форматирано принтиране на низ, ни като втори параметър подадем списък.

```
System.out.printf("%s\n", list1);
```

След тези две команди на конзолата получаваме подреден списък.

```
>>> [apple, bacon, lemons, youtube]
```

Сортиране с опции

Методът **sort()** може да бъде използван и като се подаде и втори параметър, който показва как да бъдат сортирани данните. Ще използваме **reverseOrder()** от класа **Collections**, който ще укаже на методът **sort()** да сортира в обратен ред.

```
Collections.sort(list1, Collections.reverseOrder());
```

След като принтираме на конзолата получаваме нашият списък, но подреден обратно на азбучният ред.

```
System.out.printf("%s\n", list1);
```

Ето го и резултата

```
>>> [youtube, lemons, bacon, apple]
```

Краен резултат

Цялата програма изглежда така:

```
import java.util.*;
class Demo {
    public static void main(String[] args){

        String[] crap = {"apple", "lemons", "bacon", "youtube"};
        List<String> list1 = new LinkedList<String>(Arrays.asList(crap));

        Collections.sort(list1);
        System.out.printf("%s\n", list1);

        Collections.sort(list1, Collections.reverseOrder());
        System.out.printf("%s\n", list1);

    }
}
```