



Declare e use um ArrayList - Iterator e o enhanced for

A interface `Iterator` define uma maneira de percorrer coleções. Isso é necessário porque, em coleções diferentes de `List`, não possuímos métodos para pegar o enésimo elemento. Como, então, percorrer todos os elementos de uma coleção?

- `hasNext` : retorna um booleano indicando se ainda há elementos a serem percorridos por esse iterador;
- `next` : pula para o próximo elemento, devolvendo-o;
- `remove` : remove o elemento atual da coleção.

O código que costuma aparecer para percorrer uma coleção é o seguinte:

```
Collection<String> strings = new ArrayList<String>();
Iterator<String> iterator = strings.iterator();
while (iterator.hasNext()) {
    String atual = iterator.next();
    System.out.println(atual);
}
```

COPIAR CÓDIGO

O `enhanced-for` também pode ser usado nesse caso:

```
Collection<String> strings = new ArrayList<String>();
for (String atual : strings) {
```

```
        System.out.println(atual);  
    }
```

[COPIAR CÓDIGO](#)

O método equals em coleções

A maioria absoluta das coleções usa o método `equals` na hora de buscar por elementos, como nos métodos `contains` e `remove`. Se você deseja ser capaz de remover ou buscar elementos, terá que provavelmente sobrescrever o método `equals` para refletir o conceito de igualdade em que está interessado, e não somente a igualdade de referência (implementação padrão do método).

Cuidado ao tentar sobrescrever o método `equals`, se você escrevê-lo recebendo um tipo específico em vez de `Object`, não o estará sobrescrevendo, e o `ArrayList` continuará invocando o código antigo, a implementação padrão de `equals` !

ArrayList e referências

Vale lembrar que Java sempre trabalha com referências para objetos, e não cria cópias de objetos cada vez que os atribuímos a uma variável ou referência:

```
Cliente guilherme = new Cliente();  
guilherme.setNome("guilherme");
```

```
ArrayList<Cliente> clientes = new ArrayList<Cliente>();  
clientes.add(guilherme);
```

```
System.out.println(guilherme.getNome()); // guilherme  
System.out.println(clientes.get(0).getNome()); // guilherme
```

```
guilherme.setNome("Silveira");
```

```
System.out.println(guilherme.getNome()); // Silveira
```

```
System.out.println(clientes.get(0).getNome()); // Silveira
```

[COPIAR CÓDIGO](#)