



## 05. O poder dos sets

📅 Date	@26/10/2022
📁 Categoria	Java
📖 Curso	Java Collections: Dominando Listas Sets e Mapas

### Tópicos

- O poder dos sets
- Características de conjuntos
- Interface mãe
- Imprimindo conjunto
- Elementos duplicados
- Velocidade de busca das listas e conjuntos

### Características de conjuntos

Sobre o conjunto `Set`, podemos afirmar que:

- Um conjunto não aceita elementos repetidos
  - Um `Set` garante que nenhum elemento se repete. Por isso o método `add(..)` devolve um booleano que indica o sucesso da inserção.
- Há métodos que fazem busca por um objeto, como `contains(..)` ou `remove(..)`, que são muito mais rápidos se comparado com uma lista.
  - Isso é uma grande vantagem. Devido a implementação do `Set` (aquela tabela de espalhamento), o acesso aos elementos é muito rápido.

- Um conjunto não garante a ordem de inserção.
  - Diferentemente de uma lista, um conjunto não define uma ordem. Isso significa que não recebemos os elementos na mesma ordem da inserção.

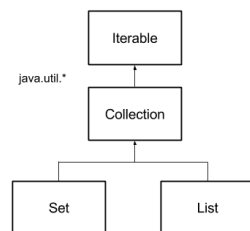
---

## Interface mão

Ambos, `List` e `Set` estendem a interface `Collection`.

Sabendo disso, podemos usar a interface como tipo do nosso conjunto:

```
//Set<String> alunos = new HashSet<>();  
Collection<String> alunos = new HashSet<>();  
alunos.add("Rodrigo Turini");  
alunos.add("Alberto Souza");  
alunos.add("Nico Steppat");
```



Há mais uma interface na hierarquia, mais genérica ainda :

Por causa dessa interface `Iterable` podemos iterar com qualquer coleção. Em outras palavras, podemos usar uma coleção no *foreach*.

---