



DigitalHouse>

# Igualdade e ordenamento



**Certified Tech  
Developer**

The Ultimate Degree

# Índice

1. Elementos iguais
2. Ordem entre elementos

# 1 | Elementos iguais

# Elementos iguais

As coleções **Set** (HashSet, LinkedHashSet, TreeSet) não aceitam valores repetidos, ou seja, iguais ou nulos, o mesmo ocorre com as chaves dos **Maps**. Mas como vamos, por exemplo, identificar que um objeto da classe Pessoa na coleção é igual a outra?



Pessoas
-nome: String -sobrenome: String -idade: int
+getNomeCompleto(): String +eMaiorIdade(): boolean

# Elementos iguais

Devemos estabelecer o critério pelo qual uma pessoa é igual a outra, se esse critério não for atendido, diremos que são diferentes.

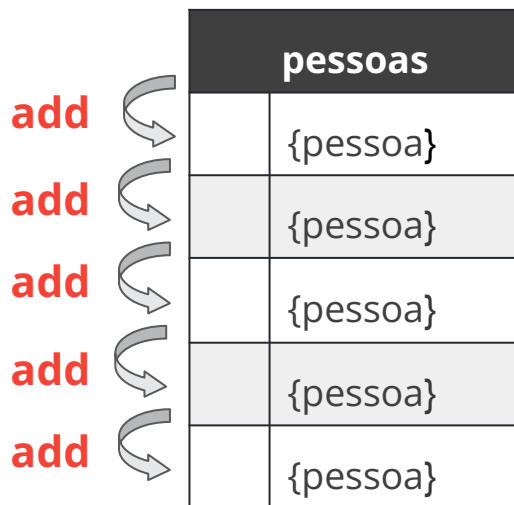


Em Java, para determinar se dois objetos são iguais, você deve substituir os métodos **equals()** e **hashCode()**.

Pessoas
-nome: String -sobrenome: String -idade: int
+getNomeCompleto(): String +eMaiorIdade(): boolean +equals(o: Object): boolean +hashCode(): int

# Elementos iguais

Dessa forma, substituindo os métodos equals e hashCode, as coleções podem determinar se o item que armazenam é igual a outro. No caso de Set, eles servirão para impedir sua inserção.



Pessoas
-nome: String -sobrenome: String -idade: int
+getNomeCompleto(): String +eMaiorIdade(): boolean <b>+equals(o: Object): boolean</b> <b>+hashCode(): int</b>

# 2 | Ordem entre elementos

# Ordem entre elementos

No caso das coleções **TreeSet e TreeMap**, os elementos são armazenados de maneira **ordenada**. **Não basta** determinarmos a igualdade, neste caso, devemos também compará-los, avaliando qual é maior, menor ou igual a outro.

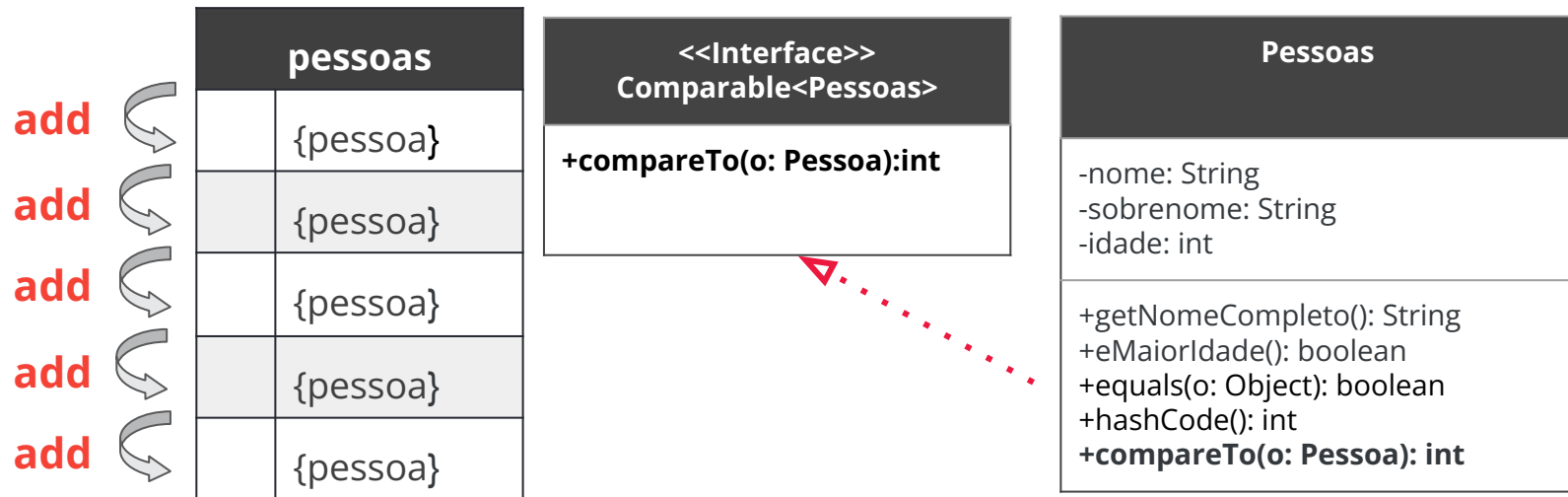


Em Java, para comparar objetos em coleções, devemos implementar a **interface Comparable**.



# Ordem entre elementos

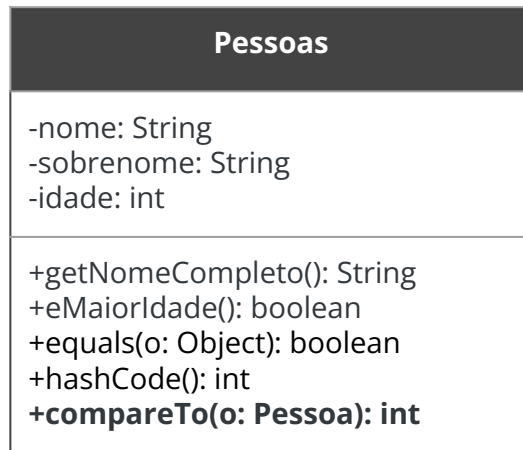
No caso das coleções TreeSet e TreeMap, os elementos são armazenados de forma ordenada e para isso devemos implementar a **interface Comparable**.



# Ordem entre elementos

Podemos utilizar a interface Comparable para ordenar os elementos de uma **List** (ArrayList ou LinkedList) invocando seu método **sort()**.

```
personas.sort(null);
```



DigitalHouse>  
Coding School