

Mãos na massa: Ordenação de listas

- 1) Começaremos criando a classe `Teste` dentro do nosso pacote `util`. Não esqueça de colocar o método `main` !
- 2) Agora para economizarmos tempo, podemos copiar o seguinte código disponibilizado dentro de nosso `main` :

```
Conta cc1 = new ContaCorrente(22, 33);
cc1.deposita(333.0);

Conta cc2 = new ContaPoupanca(22, 44);
cc2.deposita(444.0);

Conta cc3 = new ContaCorrente(22, 11);
cc3.deposita(111.0);

Conta cc4 = new ContaPoupanca(22, 22);
cc4.deposita(222.0);

List<Conta> lista = new ArrayList<>();
lista.add(cc1);
lista.add(cc2);
lista.add(cc3);
lista.add(cc4);
```

Não esqueça de importar as classes!

- 3) Como vimos, precisamos criar uma classe que implementa a interface `Comparator`, faremos isso dentro do mesmo arquivo `Teste.java` do seguinte modo:

```
class NumeroDaContaComparator implements Comparator<Conta> {

    @Override
    public int compare(Conta c1, Conta c2) {

    }

}
```

- 4) Olhando o retorno do método `compare()`, aprendemos como devemos fazer nossa comparação, que ficará assim:

```
class NumeroDaContaComparator implements Comparator<Conta> {

    @Override
    public int compare(Conta c1, Conta c2) {
        return Integer.compare(c1.getNumero(), c2.getNumero());
    }

}
```

- 5) Agora, na classe `Teste`, no final do método `main` crie um objeto da classe `NumeroDaContaComparator` e passe o comparador para o método `sort` :

```
NumeroDaContaComparator comparator = new NumeroDaContaComparator();  
lista.sort(comparator);
```

6) Vamos mostrar na tela o resultado antes e depois da ordenação da seguinte maneira:

```
for (Conta conta : lista) {  
    System.out.println(conta);  
}  
  
NumeroDaContaComparator comparator = new NumeroDaContaComparator();  
lista.sort(comparator);  
  
System.out.println("-----");  
  
for (Conta conta : lista) {  
    System.out.println(conta);  
}
```

7) Fique atento aos erros de compilação. Uma vez funcionando rode a classe `Teste` e verifique o resultado.