

### Exercício Impostos com Interfaces e Equals

Considerando o exercício Impostos, faça o seguinte:

1. Na classe principal, crie um contentor do tipo *ArrayList<Contribuinte>* chamado *contribuintes*.
2. Adicione contribuintes ao contentor, usando o seguinte código:

```
contribuintes.add(new Desempregado("Lito Vidal", "Lisboa", 230, 1));
contribuintes.add(new Reformado("Valter Cubilhas", "Olival", 600, 400));
contribuintes.add(new Desempregado("Paulo Santos", "Braga", 150, 4));
contribuintes.add(new TrabalhadorContaOutrem("Ana", "Ovar", 1800, 300, "CMM"));
contribuintes.add(new Reformado("Jaime Magalhães", "Porto", 1500, 100));
contribuintes.add(new TrabalhadorContaOutrem("Salvio", "Seixal", 8000, 1000, "SSB"));
contribuintes.add(new TrabalhadorContaPropria("Mário", "Guarda", 1500, 300, "Eletricista"));
contribuintes.add(new Reformado("Aníbal Silva", "Coimbra", 1000, 500));
contribuintes.add(new TrabalhadorContaOutrem("João", "Lisboa", 800, 100, "CML"));
contribuintes.add(new TrabalhadorContaPropria("Carla", "Porto", 15000, 1000, "Advogado"));
contribuintes.add(new Desempregado("Maria Sá", "Afife", 230, 8));
```
3. Programe uma listagem dos contribuintes do contentor.
4. Programe a eliminação do desempregado "Lito Vidal" do contentor. Para o efeito use o método *remove(Object o)* do contentor e, passando por parâmetro, uma nova instância equivalente à que se pretende eliminar.
5. Programe uma nova listagem dos contribuintes do contentor e verifique que a eliminação do desempregado do contentor não foi efetuada.
6. Reescreva o método *equals* nas classes da hierarquia de contribuintes.
7. Invoque novamente a eliminação do desempregado e verifique que foi realizada.
8. Invoque o método *equals* verificando a equivalência de:
  - a) Duas instâncias com características iguais.
  - b) Duas instâncias do mesmo tipo com características diferentes.
  - c) Duas instâncias de tipos diferentes.
9. Programe as seguintes listagens dos contribuintes armazenados no contentor:
  - a) Ordem alfabética inversa dos tipos de contribuintes.
  - b) Ordem alfabética dos tipos de contribuintes e, em cada tipo de contribuinte, os contribuintes apresentados por ordem alfabética dos seus nomes.
10. Crie testes unitários para testar os métodos de cálculo do valor do imposto.
11. Analise a cobertura de testes através do plugin Maven JaCoCo (Java Code Coverage).