

Mãos na massa: equals

Chegou o momento de replicar o que foi feito em vídeo.

- 1) Renomeie a classe **Teste** para **TesteArrayList** e crie uma cópia dessa classe com o nome **TesteArrayListEquals**.
- 2) Remova parte do código da classe **TesteArrayListEquals**, ficando apenas:

```
public static void main(String[] args){

    ArrayList<Conta> lista = new ArrayList<Conta>();

    Conta cc1 = new ContaCorrente(22, 11)
    Conta cc2 = new ContaCorrente(22, 22);

    lista.add(cc1);

    for(Conta conta : lista){
        System.out.println(conta);
    }
}
```

- 3) Teste se sua lista contém uma conta, usando o método `contains` :

```
public static void main(String[] args){

    ArrayList<Conta> lista = new ArrayList<Conta>();

    Conta cc1 = new ContaCorrente(22, 11);
    Conta cc2 = new ContaCorrente(22, 22);

    lista.add(cc1);

    boolean existe = lista.contains(cc2); //novo

    System.out.println("Já existe? " + existe);

    for(Conta conta : lista){
        System.out.println(conta);
    }
}
```

Repare que a saída é **false**, mas não é isso que queremos pois os dois objetos representam a mesma conta.

- 4) Na classe **Conta** adicione a funcionalidade que verificará se uma conta é igual a outra. Veja abaixo:

```
public boolean ehIgual(Conta outra){

    if(this.agencia != outra.agencia){
        return false;
    }
}
```

```
}

    if(this.numero != outra.numero){
        return false;
    }

    return true;
}
```

5) Na classe **TesteArrayListEquals** teste o método através do código abaixo:

```
//criação das contas omitidas

boolean igual = cc1.ehIgual(cc2);
System.out.println(igual); //deve imprimir true
```

6) Faça o mesmo teste agora com as contas tendo valores diferentes de numero e agência.

7) Modifique se método **ehIgual** para sobrescrever o método **equals** lembre-se de usar o **@Override**. Renomeia o método, veja o código abaixo:

```
@Override
public boolean equals(Object ref){

    Conta outra = (Conta) ref;

    if(this.agencia != outra.agencia){
        return false;
    }

    if(this.numero != outra.numero){
        return false;
    }

    return true;
}
```

8) Conserte o código na classe teste e execute. Veja que agora nossa saída é do método `contains` é `true` !