

conta_bancaria.dart

```
conta_bancaria.dart > ...
1  class ContaBancaria {
2      String titular;
3      double saldo;
4
5      ContaBancaria(this.titular, this.saldo);
6
7      void mostrarSaldo() {
8          print("Titular: $titular |Saldo :R\$ ${saldo.toStringAsFixed(2)} ");
9      }
10 }
11
```

acesso_conta.dart

```
acesso_conta.dart > main
1  import 'conta_bancaria.dart';
2
3  class ContaCorrente extends ContaBancaria {
4      double limite;
5      ContaCorrente(String titular, double saldoInicial, this.limite)
6          : super(titular, saldoInicial);
7  }
8
9  Run | Debug
10 void main() {
11     ContaCorrente conta_diego = ContaCorrente("Diego", 50000.00, 350000.00);
12     conta_diego.mostrarSaldo();
13     conta_diego.saldo = 100000;
14     conta_diego.mostrarSaldo();
15 }
```

Alterando o “conta_bancaria.dart”

```
conta_bancaria.dart > ...
1  class ContaBancaria {
2      String titular;
3      double _saldo;
4
5      ContaBancaria(this.titular, this._saldo);
6
7      void mostrarSaldo() {
8          print("Titular: $titular |Saldo :R\$ ${_saldo.toStringAsFixed(2)} ");
9      }
10 }
11
```

Vamos ver nosso arquivo “acesso_conta.dart”

```
acesso_conta.dart > main
1  import 'conta_bancaria.dart';
2
3  class ContaCorrente extends ContaBancaria {
4      double limite;
5      ContaCorrente(String titular, double saldoInicial, this.limite)
6          : super(titular, saldoInicial);
7  }
8
Run | Debug
9  void main() {
10     ContaCorrente conta_diego = ContaCorrente("Diego", 50000.00, 350000.00);
11     conta_diego.mostrarSaldo();
12     conta_diego._saldo = 1000000;
13     conta_diego.mostrarSaldo();
14 }
15
```

```
PS C:\Dart_modulo02> dart ./acesso_conta.dart
acesso_conta.dart:12:15: Error: The setter '_saldo' isn't defined for the type 'ContaCorrente'.
- 'ContaCorrente' is from 'acesso_conta.dart'.
Try correcting the name to the name of an existing setter, or defining a setter or field named '_saldo'.
conta_diego._saldo = 1000000;
                ^^^^^^
```

VAMOS MODIFICAR NOVAMENTE NOSSOS ARQUIVOS

```
conta_bancaria.dart > ContaBancaria > saldo
1  class ContaBancaria {
2      String titular;
3      double _saldo;
4
5      ContaBancaria(this.titular, this._saldo);
6
7      double get saldo {
8          return _saldo;
9      }
10
11     set saldo(double novoSaldo) {
12         _saldo = novoSaldo;
13     }
14
15     void mostrarSaldo() {
16         print("Titular: $titular | Saldo :R\$ ${_saldo.toStringAsFixed(2)} ");
17     }
18 }
```

Arquivo “acesso_conta.dart”

```

acesso_conta.dart > ...
1  import 'conta_bancaria.dart';
2
3  class ContaCorrente extends ContaBancaria {
4      double limite;
5      ContaCorrente(String titular, double saldoInicial, this.limite)
6          : super(titular, saldoInicial);
7  }
8
Run | Debug
9  void main() {
10     ContaCorrente conta_diego = ContaCorrente("Diego", 50000.00, 350000.00);
11     conta_diego.mostrarSaldo();
12     conta_diego.saldo = 100000.00;
13     conta_diego.mostrarSaldo();
14 }
15

```

Vamos fazer mais um exercicio novo.

```

personagem.dart > Personagem
1  class Personagem {
2      String _nome;
3      int _vida;
4      int _forca;
5
6      // Construtor
7      Personagem(this._nome, this._vida, this._forca);
8
9      // ---Criando os GETTERS---
10     String get nome {
11         | return _nome;
12     }
13
14     int get vida {
15         | return _vida;
16     }
17
18     int get forca {
19         | return _forca;
20     }
21
22     // --- SETTERS ---
23     set nome(String novoNome) {
24         | _nome = novoNome;
25     }
26
27     set vida(int novaVida) {
28         | _vida = novaVida;
29     }
30
31     set forca(int novaForca) {
32         | _forca = novaForca;
33     }
34
35     // Método público para mostrar o status do personagem
36     void mostrarStatus() {
37         | print('$_nome | Vida: $_vida | Força: $_forca');
38     }
39 }

```

Criando "jogo.dart"

```
jogo.dart > ...
1 // jogo.dart
2 import 'personagem.dart';
3
4 // Classe filha que aplica a Herança (extends)
5 class Guerreiro extends Personagem {
6     int escudo; // Atributo específico do Guerreiro (Público)
7
8     // Construtor que repassa os 3 atributos para o 'super' (pai)
9     Guerreiro(String nomeInicial, int vidaInicial, int forcaInicial, this.escudo)
10        : super(nomeInicial, vidaInicial, forcaInicial);
11
12    // Método na classe filha que altera os status do guerreiro
13 }
14 Run | Debug
15 void main() {
16     print('--- INICIANDO O RPG ---\n');
17
18     // Criando o guerreiro com os 3 atributos do pai + o atributo do filho
19     Guerreiro jogador1 = Guerreiro('Aragorn', 100, 15, 50);
20
21     jogador1.mostrarStatus();
22     print('Defesa do Escudo: ${jogador1.escudo}');
23
24     // Executando a melhoria (que vai rodar os Setters por trás dos panos)
25     jogador1.vida = 150;
26     jogador1.forca = 200;
27     jogador1.escudo = 300;
28     // Mostrando o status atualizado para comprovar que os Setters funcionaram
29     jogador1.mostrarStatus();
30     print('Defesa do Escudo atualizada: ${jogador1.escudo}');
31 }
32 }
33
```

Exercicios

1. Como é definida uma classe em Dart que herda de outra classe existente?

A. class Carro implements Veiculo {}

B. class Carro inherits Veiculo {}

C. class Carro extends Veiculo {}

D. class Carro : Veiculo {}

2. Considere que a superclasse 'Animal' possui um construtor que recebe o parâmetro 'nome'. Qual é a forma correta de criar o construtor da subclasse 'Cachorro' repassando esse parâmetro?

- A. Cachorro(String nome) { super(nome); }
- B. Cachorro(this.nome);
- C. Cachorro(String nome) extends super(nome);
- D. Cachorro(String nome) : super(nome);

6. O que acontece se tentarmos acessar um membro privado (com prefixo '_') de uma superclasse a partir de uma subclasse que está em um arquivo/biblioteca diferente?

- A. O acesso é permitido normalmente através da herança.
- B. Ocorre um erro de compilação, pois a privacidade em Dart é baseada no nível de biblioteca (arquivo).
- C. O acesso é permitido apenas se usarmos a palavra-chave 'super._membro'.
- D. O programa executa, mas lança uma exceção em tempo de execução (Runtime Error).


```
// main.dart
import 'conta_bancaria.dart';

class ContaCorrente extends ContaBancaria {
  double limite;

  ContaCorrente(String titular, double saldoInicial, this.limite)
    : super(titular, saldoInicial);
}

void main() {
  ContaCorrente minhaConta = ContaCorrente('Professor Dart', 1000.00,
500.00);

  // ALTERANDO O SALDO:
  // O Dart intercepta esse '=' e roda o método 'set saldo(999999.00)'
dos bastidores
  //minhaConta.saldo = 999999.00;

  // Confirmando que o valor foi alterado usando o método da própria
classe
  minhaConta.mostrarSaldo();
}
```