



## Fundamentos de linguagem JAVA

1. Escreva um programa que crie uma classe que possui uma propriedade “mensagem”, que tenha um método `get()`, um método `set()` (para se poder interagir com essa propriedade) e possua um construtor. Testar a classe criada.
2. Escreva um programa que crie uma classe que possui uma propriedade “mensagem”, com os respetivos `get()` e `set()`, um método que concatena o valor da mensagem com o nome que vai ser pedido ao utilizador e um método que apresenta a mensagem completa ao utilizador.
3. Escreva um programa que crie uma classe que calcule o fatorial de um número.
4. Escreva um programa que dado dois números, efetue as 4 operações básicas. Para o efeito crie uma classe que efetue essas operações.
5. Escreva um programa para gera um número aleatório entre 0 e 9, e que crie 3 jogadores que tentam adivinhar o número gerado. O programa será executado até pelo menos um dos jogadores adivinhar o número. Criar as classes `GuessGame`, `Player` e a classe de teste.
6. Escreva um programa que crie um array do tipo `Dog`, percorra esse array e para cada elemento invoque o método `bark()`. Criar uma classe `Dog`.
7. Escreva um programa que crie instancie 2 objetos do tipo `Dog`, que defina o tamanho de cada um desses objetos e invoque o seu método `bark()`. Nesse método se o tamanho for maior do que 60 deve imprimir “Woof! Woof!”, senão se for maior do que 40 deve imprimir “Ruff! Ruff!”, senão imprime “Yip! Yip!”.
8. Escreva um programa que faça o seguinte:
  - a. Crie uma classe `Boat`, que tem uma propriedade do tipo `int` chamada `length` (com os respetivos `gets` e `sets`) e o método **`move()`** que escreve na consola “Boat drift.”
  - b. Crie uma classe `Rowboat` que é uma subclasse de `Boat` e tenha um método **`rowTheWBoat()`** que escreve na consola “Rowboat stroke natasha”.
  - c. Crie uma classe `Sailboat` que é uma subclasse de `Boat` e tenha o método **`move()`** que escreve na consola “Sailboat hoist sail.”
  - d. Uma classe de teste que instancie as 3 classes invoque o método **`setLength()`** da instância da classe `Sailboat` e invoque o método **`move()`** das 3 instâncias.
9. Escreva um programa que faça o seguinte:
  - a. Crie uma interface com o nome `Nose` tenha um método **`iMethod()`** que devolve um `int`
  - b. Crie uma classe abstrata com o nome `Picasso`, que implemente a interface `Nose` e que tenha um override do método `iMethod()`
  - c. Crie uma classe com o nome `Clowns` que é uma subclasse de `Picasso`
  - d. Crie uma classe `Acts` que é uma subclasse de `Picasso` e tenha um override do método `iMethod()`
  - e. Crie uma classe de teste que crie um array do tipo `Nose` de tamanho 2 e insira na 1ª posição uma instância de `Acts` e na 2ª posição uma instância de `Clowns`
  - f. Na classe de teste, percorrer o array e imprimir e para cada elemento imprimir o método `iMethod()` e o método `getClass()` – este último método existe em todos os objetos em Java



10. Escreva um programa que faça o seguinte:
  - a. Crie uma classe Produto com duas propriedades: uma do tipo String chamada descrição e outra do tipo int chamada quantidade.
  - b. Nessa mesma classe criar um método que imprima a descrição e a quantidade de um produto
  - c. Crie uma classe Stock com uma propriedade do tipo ArrayList. Esse ArrayList terá como elementos objetos do tipo Produto.
  - d. Nessa mesma classe criar 3 métodos: um adicionar um Produto no ArrayList, outro método para obter um Produto do ArrayList através do seu índice e outro para listar todos os produtos
  - e. Criar uma classe de teste que pergunte ao utilizador quantos produtos quer inserir, para cada produto inserir a sua descrição e quantidade e no fim de inserir os produtos apresentar todos os produtos inseridos e respetiva informação
11. Escreva um programa que faça o seguinte:
  - a. Crie uma classe Pessoa com as propriedades: do tipo String nome, do tipo char sexo, do tipo int idade, e do tipo Cao cao
  - b. Essa classe vai ter 2 construtores: Um em que recebe todos os parâmetros, e outro em que recebe três parâmetros...que correspondem às três primeiras propriedades da classe
  - c. Essa classe vai ter um método que recebe um parâmetro um objeto Pessoa e um objeto Cao e vai imprimir uma mensagem a dizer que uma pessoa com um dado nome está a passear com um cão (imprimir o nome do cão)
  - d. Crie uma classe Cao com as propriedades: do tipo String nome, do tipo String raça, do tipo char sexo, do tipo int idade
  - e. Essa classe deverá ter um construtor em que os parâmetros do mesmo correspondem a todas as propriedades da classe
  - f. Criar um classe de teste que instancie um cão e uma pessoa e invoque o método passear()
12. Escreva um programa que faça o seguinte:
  - a. Crie uma classe com o nome “ContaBancaria” que possui os atributos: nomeCliente, numeroConta, saldo
  - b. Esta classe deverá ter os métodos: LevantarDinheiro (que subtrai valores ao saldo) e DepositarDinheiro (que soma valores ao saldo)
  - c. Esta classe deverá ter um construtor com dois parâmetros: nomeCliente e saldo
  - d. Esta classe ao ser instanciada deverá gerar um número aleatório entre zero e um milhão para definir como número da conta.
  - e. Esta classe ao ser instanciada deverá ficar se o saldo inicial é inferior a 150€. Nesse caso será necessário efetuar um depósito com esse valor.