

Nome: _____

Nº Mecanográfico: _____

1) Considere as entidades abaixo:

Banco, Casa, Mesa, Árvore, Prato

Crie uma classe para cada entidade, com 2 atributos e 1 método no mínimo. Sobrescreva o método *toString* para apresentar as informações sobre a classe. Crie um programa em Kotlin que crie 2 objetos para cada classe criada e apresente na tela todos eles.

Considerando a classe abaixo para as questões 2) e 3):

```
open class Produto(var nome: String, var marca: String, preco: Double) {
    var preco: Double = if (preco >= 0) preco else throw Exception("Valor menor que 0.")
    set(value) {
        field = if (value >= 0) value else throw Exception("Valor menor que 0.")
    }
    get() = field

    open fun imposto(valor: Double): Double {
        return preco * valor
    }

    open fun aplicarImposto(valor: Double): Double {
        return preco + imposto(valor)
    }
}
```

2) Crie uma lista de 5 produtos. Apresente uma entrada para o utilizador ao final para entrar com o nome de um produto e apresente toda a informação sobre esse produto.

3) Crie uma classe filha de *Produto*, *ProdutoDesconto*, e sobrescreva (*override*) a função *aplicarImposto* para diminuir o valor do produto em 10%.

4) Um sistema de entregas tem três Classes: *Pessoa*, *Entrega*, *Pacote*. A classe *Entrega* tem dois atributos *Pessoa*: Um remetente e um destinatário, assim como informação de entregue ou não. A classe *Pacote* tem um ID e o peso. Crie 8 pessoas e faça 200 entregas variando os atributos.

5) Escreva uma classe abstrata chamada *Compartimento*. Um *Compartimento* deve conter:

- Um nome
- Um tipo de piso
- Um tamanho em M²
- Um método para calcular o perímetro e um para área

Escreva as classes *Quarto*, *Cozinha* e *Sala*, subclasses de *Compartimento*

- *Quarto* contém um campo que indica o tipo de cama (String)
- *Cozinha* contém um campo que indica quantidade de geladeiras (Int)
- *Sala* contém um campo que indica quantos sofás tem na sala (Int)

Crie um programa com uma lista com 8 desses compartimentos, e crie uma função que dado uma lista de compartimentos retorne o compartimento com o maior tamanho dessa lista em área ou em perímetro.

Alguns computadores da escola aceitam somente a opção de baixo no caso de entrada do utilizador:

```
readln().toDouble()
readLine()!!.toDouble()
```

Para usar aleatórios:

```
import kotlin.random.Random

Random.nextInt(final)
Random.nextInt(comeco, final)
Random.nextDouble() // entre 0 e 1
Random.nextDouble(finalDouble)
Random.nextDouble(comecoDouble, finalDouble)
```