

Nomenclatura de código

a
(Axxx)

b
(Axxx)

Sandro Manuel Fernandes Duarte
(A94731)

21 de setembro de 2023

Conteúdo

1	Introdução	2
2	Funções	3
3	Variáveis	4
4	Classes e interfaces	5

Capítulo 1

Introdução

A documentação e comentários são escritos em Português. Todo o código é escrito em Inglês e segue as regras especificadas nas secções seguintes.

Capítulo 2

Funções

As funções estão declaradas seguindo o estilo **camelCase**.

```
1 public int getBar() { // ... }
2 public void setFooBar(int x) { // ... }
```

Listing 2.1: Delcaração de funções

O nome de todas as funções deve expressar inequivocamente a intenção das mesmas.

```
1 public void updateCollection(Object o) {
2     // atualiza a informacao da colecao
3 }
4
5 public static ObjectInfo getInfoFromObject(Object o) {
6     // nao deve alterar nada sobre o objeto e simplesmente retornar a
7     // informacao relativa ao objeto
8     return;
9 }
```

Listing 2.2: Intenção de funções deve ser clara

Funções que devolvam booleanos devem ser perguntas que respondam a questões de verdadeiro ou falso.

```
1 public boolean isOn() { // ... }
2 public static boolean shouldResetCache(Object o) { // ... }
```

Listing 2.3: Funções booleanas devem ser questões

Capítulo 3

Variáveis

As variáveis estão declaradas seguindo o estilo **camelCase**.

```
1 private int x;  
2 private int horsePower;
```

Listing 3.1: Declaracao de variáveis

As variáveis que tenham atributos especiais (globais, estáticas, constantes ou booleanas) devem ser precedidas de identificadores relevantes.

```
1 int x; // variavle normal  
2 boolean bResult; // b para booleano  
3 final int kWattage; // k para "konstant" (para evitar capslock)  
4 static int sCounter; // s para static  
5 int gWattage; // g para global
```

Listing 3.2: Identificador de variáveis

O nome das variáveis deve ser uma breve descrição das mesmas, sempre que possível.

```
1 int counter;  
2 Lst<int> intList;  
3 Map<int, Group> studentIdToGroupMap;
```

Listing 3.3: Nome de variáveis

Capítulo 4

Classes e interfaces

O nome das classes deve ser um nome que representa uma entidade no singular.

```
1  class Student    { // ... }
2  class Provider   { // ... }
3  class Connection { // ... }
```

Listing 4.1: Nome de classes

Uma classe deve seguir a seguinte estrutura (F. Mário, 2018):

```
1  class Foo {
2      // variaveis e constantes de classe
3      private static int    sCounter = 0;
4      private final double kTax      = 0.13;
5
6      // metodos de classe
7      public static double getPriceAfterTax(double price) {
8          // ...
9          return;
10     }
11
12     // variaveis de instancia
13     private String name;
14     private int    age;
15
16     // construtores
17     public Foo() {
18         this.name = "null";
19         this.age  = 0;
20     }
21
22     // metodos de instancia
23     // em primeiro lugar metodos interrogadores
24     // depois metodos modificadores
25     public String getNome() {
```

```

26         return this.name;
27     }
28
29     public void setAge(int age) {
30         this.age = age;
31     }
32 }

```

Listing 4.2: Estrutura de classes

Uma interface deve ser um adjetivo, ou em último caso, um nome precedido pela letra I.

```

1  interface Addable { // ... }
2  interface Powerable { // ... }
3  interface IDatabase { // ... }

```

Listing 4.3: Nome de interfaces

Bibliografia

F. Mário, M. (2018). *Java 8 POO + Construções Funcionais*. FCA - Editora de Informática, Lda.