

## ATIVIDADE

### Assunto:

Visibilidade e encapsulamento.

### Orientações:

A atividade deve ser executada individualmente e entregue através do ambiente *Google Classroom*.

### Regras de criação dos programas:

Crie um novo projeto Java denominado **AtividadeVisibilidadeEncapsulamento**. As classes devem possuir os nomes informados nas questões. Ao final, o projeto deve ser exportado para um arquivo em formato ZIP.

### Nome completo:

**Débora de Lima Silva**

1. Crie uma classe pública denominada `Funcionario`. Esta classe deve possuir os atributos a seguir: `nome`, `dataAdmissao` e `salario`. Todos os atributos devem ser **privados**.
2. Crie os *getters* e *setters* necessários da sua classe `Funcionario`, conforme o exemplo a seguir:

```
class Funcionario {  
  
    private double salario;  
  
    //...  
  
    public double getSalario() {  
        return this.salario;  
    }  
  
    public void setSalario(double salario) {  
        this.salario = salario;  
    }  
}
```

**Não copie e cole!** Aproveite para praticar a sintaxe.

3. Crie uma classe pública denominada `Principal`, que contém um método `main`. Neste método, crie 3 instâncias da classe `Funcionario`, fornecendo os atributos através dos métodos de acesso.

Dica: use a classe `java.util.Date` para armazenar datas e a classe `java.text.SimpleDateFormat` para converter datas em texto para objetos `java.util.Date`.

Referência útil: <https://docs.oracle.com/javase/8/docs/api/java/text/SimpleDateFormat.html>

Ex:

```
SimpleDateFormat sdf = new SimpleDateFormat("dd/MM/yyyy");  
Date d = sdf.parse("01/02/2015");
```

4. Adicione um atributo na classe `Funcionario` de tipo `int` que se chama `identificador`. Esse identificador deve ter um valor único para cada instância do tipo `Funcionario`. O primeiro `Funcionario` instanciado deve ter o identificador 1, o segundo 2, e assim por diante. Isto deve ser

feito automaticamente na classe `Funcionario`, com o uso do construtor simples sem argumentos. Sugere-se utilizar um atributo auxiliar inteiro, privado e estático para armazenar o próximo identificador a ser utilizado neste processo de contagem.

Crie um *getter* para o identificador. Precisamos de um *setter*?

5. Adicione uma validação no método `setSalario`, de forma que não seja permitido adicionar um salário inferior a R\$ 1.100,00.
6. Adicione uma validação no método `setDataAdmissao`, de forma que não seja permitido adicionar uma data no futuro.

Boa sorte!

Prof. Igor.