

12

Encapsulando atributos da classe

Agora que temos a nossa classe `Funcionario` vamos adicionar alguns membros para ela, ou seja, adicionar atributos e encapsulá-los conforme fazemos em diversos modelos no nosso dia a dia.

Para essa classe vamos considerar os seguintes atributos:

- **Nome**
- **Matrícula**
- **Data de nascimento**

A nossa classe `Funcionario`, a princípio vai ficar assim:

```
public class Funcionario {

    String nome;
    int matricula;
    LocalDate dataNascimento;

}
```

O princípio da orientação de objetos é de fato fazer com que os atributos das classes sejam encapsulados para evitar o acesso direto, pois dessa forma evitamos comportamentos não desejados conforme é abordado nos primeiros cursos de Java.

Assim sendo, faremos com que esses atributos sejam encapsulados também! Para isso, precisamos deixar o modificador de acesso como `private` para cada um dos atributos em primeiro lugar.

Se escrevermos esse mesmo trecho `private` para cada um dos atributos, percebemos que tende a ser uma tarefa nada produtiva, pois para cada atributo que tivermos escreveremos um `private` ...

Alternando o modo de seleção para coluna

Sendo assim, vamos modificar o modo de seleção para coluna, pois dessa forma é possível selecionar uma coluna inteira e modificar todos pontos dela de uma vez. Para utilizar esse modo de seleção utilize o atalho `Alt + Shift + Insert / CMD + Shift + 8`.

Perceba na parte de baixo do IntelliJ apresentará a mensagem **Column**, isso indica que o modo de seleção por coluna foi ativado. Agora selecione com o mouse ou com as **setas + Shift** a coluna que deixa o cursor na frente de todos os atributos, algo próximo do exemplo abaixo:

```
public class Funcionario {
    String nome;
    int matricula;
    LocalDate dataNascimento;
```

Agora escreva `private` e note que o "private" será escrito para todos. Após realizar essa edição utilize o mesmo atalho para alternar o modo de seleção por linha que é o mais comum durante a edição de código de modo geral.

Criando getters e setters

Agora que todos os atributos estão privados, precisamos apenas adicionar os métodos de acesso, isto é, tanto os getters como também os setters.

Autocomplete

A princípio podemos escrever, por exemplo: "**setNome**" que o IntelliJ já mostra no autocomplete um template do setter para o atributo nome, ou seja, podemos fazer isso para os demais atributos, como também, poderia ser "**getNome**" e já completaria o getter...

Porém, mesmo sendo mais prático, ainda sim é uma tarefa manual ter que fazer para cada um dos atributos.

Generate

Considerando esse contexto, temos a possibilidade de agilizar esse processo com o generate (**Alt + Insert / CMD + N**). Para isso, dentro da classe `Funcionario` use o generate e escolha a opção **Getter and Setter**, então, selecione todos os atributos e tecle **Enter**.