

## Métodos

### Transcrição

[00:00] Se voltarmos para o nosso herói, veremos como temos um assessor, uma escrita e a leitura de linha e coluna, ele está armazenando esses dois valores como atributos do meu herói. Posso chamar um método chamado linha que devolve o valor do atributo linha. Posso chamar um método chamado coluna que devolve um valor do atributo coluna.

[00:26] Mas e se eu não quisesse chamar esse método? Quisesse acessar diretamente o atributo do meu objeto? Escrevo com um arroba na frente. Assim, digo para ele pegar, por exemplo, o valor do atributo meu. Quando uso o assessor, tanto faz se vai com arroba ou sem, não vai fazer diferença. Vamos manter um método direto, que devolve o arroba, porque afinal de contas, o método faz exatamente isso. Quando chamo o método linha, ele devolve o valor do linha. É isso que o método faz.

[01:11] Se eu quisesse só o método de leitura, eu poderia usar o attr reader. Aí eu só teria o linha para ler o valor. Não teria como falar linha igual a tanto. E se eu quisesse, poderá fazer só escrita. Aí eu teria linha, espaço, igual a um valor. Não teria como fazer linha para ler o valor da linha.

[01:52] Além disso, vimos que eu poderia acessar qualquer atributo de um objeto onde eu estou, do próprio objeto, se dentro de um método eu usar o arroba linha, ou o arroba coluna, ou o arroba nome do herói. No nosso caso, como temos o reader, estamos usando direto pelo attr.

[02:20] Fechamos nosso código aqui. Agora o herói passou a ser um objeto e meu código está um pouco mais organizado, porque encapsulei o comportamento, escondi para dentro do herói. Somente dentro do herói você pode mandar ele se mover, não é aleatório, não é voando. O herói pode ser transformado numa array, pode ser colocado no mapa, retirado do mapa. é o herói que pode fazer essas coisas. Organizamos melhor nosso código colocando juntos alguns atributos e comportamentos ligados a esses atributos.