

01

## Introdução

### Transcrição

[00:00] Nesse curso, vamos desenvolver um projeto, um jogo, baseado no Pac-man, em que nosso herói foge dos fantasmas. Foge foge vai ser o nome. Preciso criar o código base e depois o mapa.

[00:35] Primeiro vou criar o arquivo, que vai ter boas-vindas. Mostramos uma mensagem, lemos o nome do jogador, imprimimos algumas linhas em branco, e dizemos que vamos começar o jogo. Esse cara é a interface com o usuário.

[01:20] Se vamos dar boas-vindas, precisamos da lógica que chama boas-vindas, que joga o jogo, etc. Então vamos criar o arquivo .rb, que vai ser relativo ao ui, vai ter a função joga, que recebe o nome, implementamos o jogo, e iniciamos o foge foge. Faltou criar o main.rb, que é o ponto inicial do jogo. O main vai dar um require\_relative foge foge e chamar o inicia\_fogefoge.

[02:20] Precisamos criar o mapa, uma estrutura para ele. Todos eles vão ter uma linha, outra linha, uma coluna, outra coluna. Um grid. Nos jogos tradicionais, temos um grid com um número de linhas e número de colunas, sendo que nosso personagem pode estar em cada um dos quadrados. Por exemplo, posso criar um mapa 4x4, com casinhas vazias.

[03:22] Queremos representar nesse mapa muros, então vou colocar um muro. Vou usar a letra X para representar. Tenho um mapa 4x4, com quatro linhas, quatro colunas, sendo que nas beiradas é tudo muro. Um personagem não consegue sair de lá dentro.

[04:10] Agora, preciso representar meu herói. Vou representar com a letra H. E o fantasma vai ser F. Mas esse mapa parece sem graça demais. Vamos aumentar. Vai ser o mapa1. Ele tem dez linhas, cinco colunas. Tenho espaço para o herói e para o fantasma andar.

[05:10] Teremos também um segundo mapa. Ele vai ter treze linhas e nove colunas. Vai ser o mapa2, com dois fantasmas. Quanto mais complicado o mapa, mais lógica vamos ter que escrever e mais bugs vamos ter, com certeza. Mas quanto mais corrigirmos, mais vamos aprender lógica de programação.