

03

Vida do Jogador

Transcrição

[00:01] Vamos continuar a adicionar funcionalidades aqui no nosso jogo. Então, vou dar um play aqui para a gente ver onde a gente parou no curso passado. Nossa personagem está andando, olhando para todos os lados e atirando; e o zumbi persegue o personagem e quando ele ataca, eu dou game over.

[00:16] O jogo já está bem dinâmico, só que tem uma coisa aqui que me incomoda bastante e já deve estar incomodando você também, que é o seguinte: não seria mais interessante jogador ter uma vida, por exemplo, e quando o zumbi ataca ele, ele perde um pouco de vida e aí à medida que essa vida acaba, aí sim o jogo dá game over? Porque quando eu clico aqui para reiniciar, aí eu vou, tento fugir, aí zumbi e me ataca, game over.

[00:39] Então não tem dinamismo no nosso jogo. Não tem como eu aprender melhor a jogar esse jogo dessa forma. Então, vou tirar o play aqui, vamos fazer exatamente isso, vamos criar uma vida para o nosso jogador. Como que a gente vai fazer isso então? Clica no jogador, vai no script de, controla jogador, abre ele. E aí a gente tem que começar então a criar a vida.

[01:01] Onde que a gente vai guardar a vida? Como que a gente vai fazer isso? Como que a gente vai guardar um valor para nossa vida? Vamos utilizar as variáveis que a gente já aprendeu no outro curso. Então, vou declarar uma variável aqui como public e aí pensa, a vida é o seguinte, a vida, quando você pensa em quantidade de vida você já pensa: Vai ser um número, mas vai ser um número quebrado, vai ser um número inteiro?

[01:23] Então você tem que pensar: faz sentido a vida ter um valor tipo 49.5, ou faz mais sentido à vida ter um valor 49 e um valor 50? Faz mais sentido a vida ter valores inteiros, até facilita na hora de lidar com ela. Então, vamos colocar do tipo int, que é integer, números inteiros que não tem casas decimais, não tem números após a vírgula. Então, estou colocando a vida aqui como int, vou dar o nome de vida nessa variável e vou colocar um valor inicial aqui de 100, 100 de vida.

[01:57] Salva isso aqui, vamos dar uma olhada aqui no espectro o que está acontecendo, a vida já está aqui, se eu der um play aqui o que vai acontecer quando o zumbi me bater? Nada, porque eu não mexi com a vida. A gente na verdade estava mexendo com a variável booleana que chamava vivo aqui, que é a variável que a gente está usando para fazer o game over.

[02:17] Eu ainda não mexi com a vida. Eu tenho que começar a mexer com ela. Da forma que a gente fez o código, se eu vir aqui no inimigo, eu vou abrir aqui pelo lado, no visual studio, se você tiver usando um monodevelop você pode abrir pelo menu aqui na lateral também. No visual studio fica desse lado aqui, vou clicar duas para abrir o script do inimigo.

[02:37] Do jeito que a gente fez a variável de vivo, a gente veio e falou: olha, quando zumbi atacar o jogador, esse método é chamado e aí eu faço a variável vivo ficar falsa. Só que a variável vivo é uma variável que está lá no jogador, e eu estou utilizando ela e trocando o valor dela aqui no inimigo. Nesse curso a gente vai focar em dar nome aos bois, falar: olha, isso aqui é do jogador e isso aqui é do inimigo.

[03:08] A variável vivo na verdade ela é do jogador. Então, esse código não deveria estar aqui, ele deveria estar lá no jogador, que é exatamente o que a gente vai fazer com a nossa nova variável de vida, isso é conhecido como encapsulamento.

[03:22] A gente está fechando funções, encapsulando, isso aqui é do jogador, o código do jogador tem que ficar no script do jogador, o código do inimigo, tem que ficar script do inimigo, se não daqui a pouco você está trocando coisas em

diversos lugares e você não entende, onde que eu troquei o valor de vivo para fazer isso?

[03:40] Aí você não entende onde que está esse código, onde que está essa troca de valor. Então, o código ele tem que ser escrito aqui, ele pode ser usado em outros locais mas ele tem que ser escrito aqui. Então, vamos fazer isso com a variável de vida. Só que como que a gente vai criar uma forma da gente vir aqui e trocar o valor da vida. Fazer o código no jogador e usar esse código lá no inimigo, como que a gente pode fazer isso?

[04:06] No start? No update? No fixo de update? Ou a gente pode descer mais um pouco e criar um método nosso. Lembra quando a gente fez esse método aqui? Método é uma forma da gente organizar o nosso código, fazer várias linhas de comando acontecerem em sequência e a gente organizar pequenos comportamentos, que é exatamente o que eu vou fazer aqui.

[04:27] Eu vou fazer um script, um método, que vai tirar a vida do meu jogador aqui no meu jogador, para eu utilizar ele lá no inimigo. Vamos começar a criar um método parecido com esse aqui de atacar o jogador, só que ele vai chamar "tomardano". Criei um método aqui do tipo void, que eu dei o nome "tomardano", método sempre tem que abrir e fechar parêntesis, abrir e fechar chaves.

[05:04] Criei esse método aqui. Por que que é do tipo void void ? Void é um método que eu executo as linhas de código que eu colocar dentro dele, e não faço mais nada, ele executa e acabou. O nome dele, eu que dei esse nome, não era o nome da Unity, eu que dei, abre e fecha parêntesis, abre e fecha chaves. Aí eu vou começar a escrever códigos aqui dentro, aqui dentro um código que vai tirar a vida, que vai dar dano ao meu jogador, que vai tirar a vida dele.

[05:32] Vou pegar a variável de vida e vou subtrair um valor , vamos supor 30. Então, criei o método "tomardano", peguei a vida e subtraí 30 e depois joguei o resultado dessa conta de volta no valor da vida, para isso que serve o menos igual, lembra dele? A gente utilizou bastante o mais igual no outro curso.

[05:53] Então pega a vida, subtrai 30, deu 70, jogo o valor de volta com a variável vida, então a variável vida agora vale 70, para isso que serve o mnos igual. Vamos salvar isso aqui, vamos testar, o que que vai acontecer aqui agora. A variável vida mudou? Não, por quê? Porque a gente ainda não utilizou esse método, esse método a gente que fez, ele não é igual start, igual update, que a Unity define. Olha, o update roda uma vez por frame, o start roda no começo do jogo.

[06:24] Esse método aqui, a gente tem que utilizar ele, a gente que vai dizer onde que a Unity tem que rodar esse método. E onde que a gente quer rodar esse método? no inimigo. Então, agora eu quero fazer o seguinte, eu quero cancelar essas três linhas, por quê? Porque se eu utilizar o "tomardano" aqui agora, nada vai acontecer, porque o jogo vai pausar, eu vou imprimir o game over, eu vou rodar a variável de vivo.

[06:50] Vou pausar o jogo todo e não vou ver nada acontecendo, então o que eu vou fazer? Eu quero cancelar elas, então eu vou comentar essas linhas.

[06:58] Para comentar uma linha é só colocar barra barra no começo da linha. O que é o comentário aqui? O comentário, ele faz com que essa linha seja ignorada pela Unity. Ela está aqui no código mas ela não vai acontecer, ela só está aqui para eu visualizar. Depois eu posso tirar o barra barra e utilizar essa linha mas por enquanto eu não quero que ela aconteça, e eu vou aqui no começo colocar, começar a utilizar esse método "tomardano".

[07:29] Então eu quero utilizar o "tomardano" lá do jogador aqui no inimigo. Eu posso utilizar métodos de um script no outro? Posso, só que o código está no jogador, é isso que é importante, você dizer onde que o código está sendo criado. O código de "tomardano" no jogador tem que estar no jogador, mas o inimigo pode pegar e utilizar, chamar este método.

[07:56] Como que a gente chama esse método, porque ele é diferente do start e do update, a gente que tem que chamar ele aqui, como que a gente chama um método em outro script? É simples, é como utilizar uma variável, eu venho aqui,

puxo o componente de controla jogador e dou um ponto "tomardano". Só que ele não apareceu aqui, por que? Porque um método, igual uma variável, ele tem que ser público para a gente achar ele em um outro script.

[08:28] Então vou vir aqui e vou agora ponto "tomardano", ponto e vírgula aqui. Salvo isso aqui, agora quando o meu inimigo ataca o meu jogador, eu estou chamando método de "tomardano". Salvo isso aqui, vamos dar um play e olha lá, toda vez que o meu inimigo ataca eu perco 30 de vida aqui.

[08:52] Isso faz com que o nosso script vá ficar com dinamismo bem legal porque eu estou escrevendo códigos, encapsulando comportamentos nos seus lugares corretos . Agora falta a gente utilizar essa vida para gerar o game over.