

POR DENTRO DO GAME:

CONDICIONAIS EM JAVA DESCOMPLICADAS



ENTENDENDO AS REGRAS DO GAME



Você já jogou um game onde o personagem só consegue passar por uma porta se tiver uma chave? Ou só ganha um item se tiver pontos suficientes? Isso é exatamente o que condicionais fazem na programação. Elas dizem ao seu código: "Se acontecer isso, faça aquilo." Neste eBook, vamos te mostrar como usar condicionais em Java como um verdadeiro mestre do jogo!

CAPÍTULO 1:



ABRINDO A
PORTA COM A
CHAVE CERTA

IF E ELSE SIMPLES:

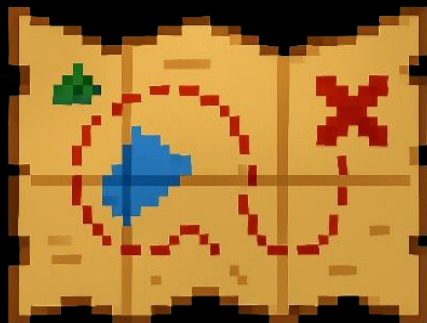
Imagine um jogo de aventura onde o jogador precisa ter uma chave para abrir uma porta. Com o if, você pode verificar se ele tem a chave. Se tiver, abre a porta. Se não tiver... nada acontece! Agora, se quiser dar um aviso tipo "Você precisa de uma chave!", aí entra o else. Em Java, isso fica assim:

```
destrancarPorta(temChave)

1 if (temChave) {
2     System.out.println("Porta aberta!");
3 } else {
4     System.out.println("Você precisa de uma chave!");
5 }
```

Usar if e else é como programar as reações do jogo às ações do jogador. Simples e direto, como qualquer bom tutorial inicial.

CAPÍTULO 2:



ESCOLHAS E
CAMINHOS NO
MAPA

IF, ELSE IF E ELSE:

Agora imagine um RPG onde o jogador tem vários caminhos: se escolher ser mago, ganha um cajado; se escolher guerreiro, ganha uma espada; qualquer outra coisa, ganha uma pedra (triste, mas justo). É aí que entra o else if, para criar várias possibilidades:

```
setItemsByClass(classe)

1 if (classe.equals("Mago")) {
2     System.out.println("Você recebeu um cajado mágico!");
3 } else if (classe.equals("Guerreiro")) {
4     System.out.println("Você recebeu uma espada afiada!");
5 } else {
6     System.out.println("Você recebeu uma pedra misteriosa...");
7 }
8
```

O else if é como um menu de seleção no jogo: cada escolha leva a um resultado diferente. Você cria decisões mais elaboradas, sem complicar demais.

CAPÍTULO 3:



0

INVENTARIO INTELIGENTE

SWITCH CASE:

Agora pensa num jogo de luta, onde o jogador pode apertar um botão e o personagem faz um ataque diferente dependendo da arma equipada. Se for espada, dá uma espadada; se for arco, atira flechas; se for machado, dá uma machadada. Em vez de vários if, você pode usar o switch:

```
attack(arma)

1 switch (arma) {
2     case "Espada":
3         System.out.println("Ataque com espada!");
4         break;
5     case "Arco":
6         System.out.println("Flecha lançada!");
7         break;
8     case "Machado":
9         System.out.println("Golpe de machado!");
10        break;
11    default:
12        System.out.println("Ataque com as mãos!");
13 }
```

O switch funciona como o inventário de um jogo: você olha qual item está selecionado e executa a ação correspondente. E com isso, o código fica limpo e fácil de entender, como uma interface bem feita.