

O DEV SOMBRIO E A DUNGEON DO BACKEND



KAUAN SILVA

📖 O Dev Sombrio e os Seletores da Lógica

No coração da Dungeon do Backend, você — o Dev Sombrio — caminha entre bugs adormecidos e sistemas ancestrais. Diante de você, um grande portão se ergue. Gravado em seu arco: “A Lógica é a Chave”. Para atravessar, você deverá dominar os Seletores Java — artefatos sagrados que moldam o destino do seu código.

```
// Um artefato foi encontrado, mas... vazio!
String pergaminhoMagico = null;

// O Dev tenta lançar a magia... mas algo dá errado!
int comprimento = pergaminhoMagico.length(); // 💀 ERRO FATAL AQUI!
```



CAPÍTULO 1

A ESPADA do if: A PRIMEIRA ESCOLHA

✂️ ☐ Todo herói precisa fazer escolhas.

E o if é sua primeira lâmina. Ele corta a escuridão da incerteza e executa ações apenas quando a condição é verdadeira.

```
int energia = 70;

if (energia > 50) {
    System.out.println("Você está forte o suficiente para entrar na sala do boss!");
}
```

Uso real: Validação de entrada, permissões de usuário, lógicas de autenticação.



CAPÍTULO 2

O ESCUDO DO
ELSE:

QUANDO TUDO
DÁ ERRADO

❓ Às vezes, a espada falha.

O caminho que você esperava... não se revela. É nesse momento que entra o else, seu escudo encantado, o plano B que impede o código de cair no vazio.

```
boolean acessoLiberado = false;

if (acessoLiberado) {
    System.out.println("Entrada permitida.");
} else {
    System.out.println("Acesso negado. Complete a missão primeiro.");
}
```

Uso prático: Retornos padrão, respostas para falhas, rotas alternativas em APIs REST.



CAPÍTULO 3

else if:

As Trilhas
Ramificadas

➤ Em certas encruzilhadas, há mais de dois caminhos.

Com `else if`, você pode testar várias condições e guiar o herói por trilhas únicas — cada uma com suas próprias consequências.

```
int nivel = 5;

if (nivel < 5) {
    System.out.println("Apenas um novato...");
} else if (nivel == 5) {
    System.out.println("Cavaleiro intermediário.");
} else {
    System.out.println("Veterano da Guilda.");
}
```

Uso comum: Fluxos condicionais múltiplos, status de usuários, escalonamento de permissões.



CAPÍTULO 4

switch:

O Menu
da Taverna

Você entra na taverna. O taverneiro te oferece opções:

INICIAR, SALVAR, SAIR. Aqui, o switch é o menu mágico que interpreta comandos e responde com ações precisas.

```
String comando = "SAIR";

switch (comando) {
    case "INICIAR":
        System.out.println("Iniciando aventura...");
        break;
    case "SALVAR":
        System.out.println("Progresso salvo.");
        break;
    case "SAIR":
        System.out.println("Saindo da dungeon...");
        break;
    default:
        System.out.println("Comando desconhecido.");
}
```

Aplicação real: Menus de sistema, comandos de terminal, mapeamento de ações.

CAPÍTULO 5

o ternário:
O Flecha
a Rápida da
Decisão

Quando não há tempo para longas batalhas, o Dev Sombrio lança uma flecha de decisão:

o operador ternário. Preciso. Letal. Em uma única linha, ele resolve o dilema.

```
int vida = 0;

String status = (vida > 0) ? "Vivo" : "Morto";
System.out.println("Status: " + status);
```





CAPÍTULO 6

AS RUNAS DE COMPARAÇÃO

🗡 Cada decisão exige comparação.

Essas são as runas mágicas, símbolos arcanos usados para testar valores e forjar os caminhos da lógica:

Operador	Significado	Exemplo
-----	-----	-----
<code>'=='</code>	Igual	<code>`if (x == 10)`</code>
<code>'!='</code>	Diferente	<code>`if (x != 10)`</code>
<code>'>'</code>	Maior que	<code>`if (pontos > 100)`</code>
<code>'<'</code>	Menor que	<code>`if (tempo < 30)`</code>
<code>'>='</code>	Maior ou igual	<code>`if (nivel >= 5)`</code>
<code>'<='</code>	Menor ou igual	<code>`if (moedas <= 3)`</code>

Dica do mago: ao comparar Strings, use `.equals()` ao invés de `==` — evite maldições silenciosas.

