



Linguagem Java

-

Estruturas de Controle

Estruturas de Controle



Estruturas de Controle são construções em Java que permitem que, de maneira condicional, façamos o controle da execução de uma condição ou sentença, repetições com e sem controle fixo, além de permitir o desvio no fluxo de execução de um programa.



An abstract graphic on the left side of the slide, consisting of several overlapping green shapes. There is a dark green L-shaped block at the top left, a large medium-green curved shape in the center, and a light green curved shape at the bottom left.

Estruturas de Controle

Estruturas de Controle

O IF e o ELSE

A sintaxe do if no Java é a seguinte:

```
if (condicaoBooleana) {  
    codigo;  
}
```

Uma condição booleana é qualquer expressão que retorne true ou false .
Para isso, você pode usar os operadores:

< , > , <= , >= e outros.

Um exemplo:

```
int idade = 15;  
if (idade < 18) {  
    System.out.println("Não pode entrar");  
}
```

Estruturas de Controle

O IF e o ELSE



Além disso, você pode usar a cláusula else para indicar o comportamento que deve ser executado no caso da expressão booleana ser falsa:

```
int idade = 15;
if (idade < 18) {
    System.out.println("Não pode entrar");
} else {
    System.out.println("Pode entrar");
}
```

Você pode concatenar expressões booleanas através dos operadores lógicos "E" e "OU". O "E" é representado pelo && e o "OU" é representado pelo || .

Um exemplo seria verificar se ele tem menos de 18 anos e se ele não é amigo do dono:

```
int idade = 15;
boolean amigoDoDono = true;
if (idade < 18 && amigoDoDono == false) {
    System.out.println("Não pode entrar");
} else {
    System.out.println("Pode entrar");
}
```

Estruturas de Controle

O IF e o ELSE

Esse código poderia ficar ainda mais legível, utilizando-se o operador de negação, o !. Esse operador transforma o resultado de uma expressão booleana de false para true e vice versa.

```
int idade = 15;
boolean amigoDoDono = true;
if (idade < 18 && !amigoDoDono) {
    System.out.println("Não pode entrar");
} else {
    System.out.println("Pode entrar");
}
```

Repare que o trecho `amigoDoDono == false` virou `!amigoDoDono`. Eles têm o mesmo valor.

Estruturas de Controle

O WHILE

O while é um comando usado para fazer um laço (loop), isto é, repetir um trecho de código algumas vezes. A ideia é que esse trecho de código seja repetido enquanto uma determinada condição permanecer verdadeira.

```
int idade = 15;
while (idade < 18) {
    System.out.println(idade);
    idade = idade + 1;
}
```

O trecho dentro do bloco do while será executado até o momento em que a condição `idade < 18` passe a ser falsa. E isso ocorrerá exatamente no momento em que `idade == 18`, o que não o fará imprimir 18.

Estruturas de Controle

O FOR

Outro comando de loop extremamente utilizado é o for . A ideia é a mesma do while : fazer um trecho de código ser repetido enquanto uma condição continuar verdadeira. Mas além disso, o for isola também um espaço para inicialização de variáveis e o modificador dessas variáveis. Isso faz com que fiquem mais legíveis, as variáveis que são relacionadas ao loop:

```
for (inicializacao; condicao; incremento) {  
    codigo;  
}
```

Um exemplo é o a seguir:

```
for (int i = 0; i < 10; i = i + 1) {  
    System.out.println("olá!");  
}
```

Repare que esse for poderia ser trocado por:

```
int i = 0;  
while (i < 10) {  
    System.out.println("olá!");  
    i = i + 1;  
}
```


FIM DO SEGUNDO EPISÓDIO

