Resumo: Estruturas de Controle em Java

# ✅ 1. Instruções em Java

São comandos que dizem ao computador o que fazer. Elas controlam o fluxo de execução, como decidir algo, repetir ações ou mexer em dados.

# 🧠 2. Estruturas Condicionais

Permitem que o programa tome decisões, dependendo de alguma condição (por exemplo, idade, nota, valor etc.).

# 🔹 if

Executa um bloco se a condição for verdadeira.

int idade = 18;

if (idade >= 18) {

System.out.println("Você é maior de idade.");

}

# 🔹 if - else

Tem duas opções: se a condição for verdadeira, faz uma coisa. Se for falsa, faz outra.

int idade = 16;

if (idade >= 18) {

System.out.println("Você é maior de idade.");

} else {

System.out.println("Você é menor de idade.");

}

# 🔹 if - else if - else

Verifica várias condições.

int nota = 85;

if (nota >= 90) {

System.out.println("Excelente!");

} else if (nota >= 70) {

System.out.println("Bom trabalho!");

} else {

System.out.println("Precisa melhorar.");

}

# 🔀 3. Estrutura de Seleção: switch

Substitui vários if-else quando você quer verificar vários valores possíveis de uma variável.

int diaDaSemana = 3;

switch (diaDaSemana) {

case 1: System.out.println("Domingo"); break;

case 2: System.out.println("Segunda-feira"); break;

case 3: System.out.println("Terça-feira"); break;

...

default: System.out.println("Dia inválido"); break;

}

✅ break serve para encerrar o caso.

✅ default é o "senão" do switch.

# 🔁 4. Instruções de Repetição

Usadas para repetir comandos automaticamente.

# 🔹 for

Repetição com número fixo de vezes.

for (int i = 1; i <= 5; i++) {

System.out.println(i);

}

# 🔹 while

Repetição enquanto uma condição for verdadeira.

int j = 1;

while (j <= 5) {

System.out.println(j);

j++;

}

# 🔹 do-while

Garante que o código será executado pelo menos uma vez.

int k = 1;

do {

System.out.println(k);

k++;

} while (k <= 5);

# 🧮 5. Vetores (Arrays)

São listas de valores do mesmo tipo.

int[] numeros = {1, 2, 3, 4, 5};

- Índices: numeros[0] = 1, numeros[1] = 2, etc.

Percorrer com for:

for (int i = 0; i < numeros.length; i++) {

System.out.println(numeros[i]);

}

Percorrer com for-each:

for (int numero : numeros) {

System.out.println(numero);

}

# 🧊 6. Matrizes (Arrays Bidimensionais)

É como uma tabela com linhas e colunas.

int[][] matriz = {

{1, 2, 3},

{4, 5, 6}

};

System.out.println(matriz[0][1]); // imprime 2

Percorrer:

for (int i = 0; i < matriz.length; i++) {

for (int j = 0; j < matriz[i].length; j++) {

System.out.print(matriz[i][j] + " ");

}

System.out.println();

}

# ⚙️ 7. Instruções de Controle: break e continue

🔹 break - Sai do laço imediatamente.

for (int i = 0; i < 10; i++) {

if (i == 5) break;

System.out.println("Iteração: " + i);

}

🔹 continue - Pula para a próxima repetição do laço.

for (int i = 0; i < 10; i++) {

if (i % 2 == 0) continue;

System.out.println("Iteração ímpar: " + i);

}