



Controle de Fluxo em JavaScript

if, else, switch, for, while

Objetivos

-  Entender o que é **controle de fluxo**
-  Explorar estruturas condicionais
-  Explorar estruturas de repetição
-  Aplicar boas práticas

O que é Controle de Fluxo?

~ Define como e quando blocos de código são executados

~ Permite:

~ Tomar decisões (condições)

~ Repetir ações (loops)

~ Essencial em qualquer programa

✓ Estruturas Condicionais: **if...else**

```
if (condição) {  
    // executa se verdadeiro  
} else {  
    // executa se falso  
}
```

↗ Encadeamento com else if

↗ Sempre usar { } mesmo em blocos simples

Exemplo if...else

 Exemplo de IF else com operador `>=`

```
let nota = 85;

if (nota >= 90) {
  console.log("A");
} else if (nota >= 70) {
  console.log("B");
} else {
  console.log("C");
}
```



Estrutura switch

```
switch (expressão) {  
  case valor1:  
    // código  
    break;  
  case valor2:  
    // código  
    break;  
  default:  
    // código padrão  
}
```

~ Alternativa ao if...else

~ Usado quando há muitos casos baseados no mesmo valor

~ ⚠ Sem break → executa os próximos blocos (fall-through)

Exemplo switch

```
let dia = 3;
let nomeDia;

switch (dia) {
  case 1:
    nomeDia = "Segunda-feira";
    break;
  case 2:
    nomeDia = "Terça-feira";
    break;
  case 3:
    nomeDia = "Quarta-feira";
    break;
  case 4:
    nomeDia = "Quinta-feira";
    break;
  case 5:
    nomeDia = "Sexta-feira";
    break;
  default:
    nomeDia = "Fim de semana";
}

console.log("Hoje é " + nomeDia);
```

Loop for

Exemplo de uso do for

```
for (let i = 0; i < 5; i++) {  
  console.log(i);  
}
```

- Melhor para número conhecido de repetições
- Usado com índices e arrays



Loop while

Exemplo de laço while

```
let i = 0;
while (i < 5) {
  console.log(i);
  i++;
}
```

- ↗ Executa enquanto a condição for verdadeira
- ↗ Bom para repetições indefinidas
- ↗ ⚠ Cuidado com loops infinitos