

Aula 16

Estruturas Condicionais
switch...case

Slide 1/4: Revisão da Aula 6

Revisão da Aula 6: Múltiplos Caminhos com `else if`

- **A Cadeia** `if...else if...else` :
 - Aprendemos a criar uma sequência de verificações para lidar com mais de dois resultados possíveis.
 - Vimos que o JavaScript executa o bloco da **primeira condição verdadeira** que encontrar e ignora o resto.
- **A Importância da Ordem:**
 - Entendemos que a ordem das condições é crucial, especialmente ao trabalhar com faixas de valores (ex: notas, preços).
- **Código Limpo:**
 - O `else if` nos ajudou a evitar "ninhos" de `if` s, tornando nosso código muito mais legível e organizado.

Slide 2/4: Uma Nova Ferramenta de Decisão

Aula 7: Estrutura Condicional de Escolha `switch...case`

- **Objetivo de Hoje:**
 - Aprender uma alternativa ao `if...else if` para casos onde comparamos uma **única variável** com **vários valores específicos**.
 - **O que é o `switch...case` ?**
 - É a estrutura "ESCOLHA...CASO" do VisualG.
 - Pense nela como um seletor de opções em um menu.
 - **Sintaxe Intuitiva:**
 - `switch (variavel)` : Inicia a verificação da variável.
 - `case valor:` : Compara se a `variavel` é igual ao `valor`.
-

Slide 3/4: As Peças do `switch`

Entendendo a Sintaxe: `case`, `break` e `default`

- `case valor:`
 - Cada `case` representa um caminho possível. A comparação é **estrita** (`===`).
- `break;` **(A Palavra-Chave Essencial!)**
 - O `break` **interrompe** a execução e sai do bloco `switch`.
 - **Se você esquecer o `break`**, o código continuará executando os `case` s abaixo (efeito "fall-through"), o que geralmente é um erro!
- `default:`
 - É o bloco "padrão", executado se **nenhum** `case` corresponder ao valor da variável.
 - Funciona como o `else` final e é opcional, mas muito recomendado.


Slide 4/4: Quando Usar Cada Estrutura?

`if...else if` vs. `switch...case`

- Use `if...else if` para:
 - Verificar **faixas de valores** (ex: `idade >= 18`).
 - Condições complexas com **múltiplas variáveis** (ex: `nota >= 7 && frequencia >= 0.75`).
 - Qualquer comparação que não seja de igualdade estrita.
- Use `switch...case` para:
 - Verificar uma **única variável ou expressão**.
 - Comparar com uma lista de **valores específicos e exatos** (ex: `diaDaSemana === "segunda"` , `opcaoMenu === 1`).
- **Meta da Aula:**
 - Saber escolher a ferramenta de decisão mais limpa e adequada para cada problema.

Prática

<> index.html X JS script.js

<> index.html >  html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9      <h1>Estruturas Condicionais switch...case</h1>
10     <script src="script.js"></script>
11 </body>
12 </html>
```

```
6 console.log("--- Exemplo 1: Menu com switch ---");
7 let opcao = prompt("Escolha uma opção:\n1. Ver Notícias\n2. Jogar\n3. Ouvir Música");
8 opcao = Number(opcao);
9
10 switch (opcao) {
11     case 1: // Compara se opcao === 1
12         console.log("Mostrando as últimas notícias do dia.");
13         alert("Exibindo as notícias!");
14         break; // O 'break' é ESSENCIAL. Ele sai do switch.
15
16     case 2: // Compara se opcao === 2
17         console.log("Iniciando o jogo...");
18         alert("Que os jogos comecem!");
19         break;
20
21     case 3: // Compara se opcao === 3
22         console.log("Tocando sua playlist favorita.");
23         alert("Conectando ao Spotify...");
24         break;
25
26     default: // Se nenhuma das opções acima corresponder.
27         console.log("Opção digitada é inválida.");
28         alert("Opção inválida! Por favor, escolha 1, 2 ou 3.");
29         break;
30 }
```



```
32 // =====
33 // 2. O PERIGO DE ESQUECER O `break` (FALL-THROUGH)
34 // =====
35 // Se o `break` é omitido, o JavaScript executa o código do `case` correspondente
36 // E CONTINUA executando os códigos dos `case`s abaixo, até encontrar um `break`.
37
38 console.log("\n--- Exemplo 2: O efeito 'fall-through' sem o break ---");
39 let codigoProduto = "A";
40
41 switch (codigoProduto) {
42     case "A":
43         console.log("Produto A selecionado."); // Este código executa.
44         // SEM BREAK AQUI!
45     case "B":
46         console.log("Produto B também é processado!"); // Este código TAMBÉM executa.
47         break; // Agora para.
48     case "C":
49         console.log("Produto C não é afetado.");
50         break;
51 }
```

```
53 // =====
54 // 3. AGRUPANDO `case`s (USO INTENCIONAL DO FALL-THROUGH)
55 // =====
56 // Podemos omitir o `break` de propósito para que vários `case`s executem o mesmo bloco de código.
57
58 console.log("\n--- Exemplo 3: Agrupando dias da semana ---");
59 let dia = prompt("Digite um dia da semana (ex: segunda):").toLowerCase(); // toLowerCase() para facilitar
60
61 switch (dia) {
62     case "segunda":
63     case "terça":
64     case "quarta":
65     case "quinta":
66     case "sexta":
67         console.log(dia + " é um dia útil.");
68         alert("É um dia útil. Foco no trabalho/estudo!");
69         break; // Para aqui se for um dia útil.
70
71     case "sábado":
72     case "domingo":
73         console.log(dia + " é fim de semana!");
74         alert("É fim de semana! Hora de descansar!");
75         break;
76
77     default:
78         console.log("Dia não reconhecido.");
79         alert("Isso não parece ser um dia da semana válido.");
80         break;
81 }
82
```

ATIVIDADE

Faça cada exercício em um JS diferente

1. **Estações do Ano:**

Peça ao usuário para digitar uma estação do ano ("Verão", "Outono", "Inverno", "Primavera"). Use um `switch` para exibir uma mensagem característica de cada estação. Ex: para "Inverno", "Que frio! Época de chocolate quente.". Inclua um `default` para entradas inválidas.

Faça cada exercício em um JS diferente

2. Código de Acesso:

Peça um código de acesso (numérico). Use `switch` para verificar os seguintes códigos:

- `1111` : "Acesso total permitido."
- `2222` : "Acesso limitado a relatórios."
- `3333` : "Acesso de convidado, apenas visualização."
- `default` : "Código incorreto. Acesso negado."

Faça cada exercício em um JS diferente

3. Tipos de Assinatura:

Peça ao usuário o tipo de sua assinatura ("basic", "premium", "gold"). Use um `switch` para listar os benefícios:

- `"basic"`: "Acesso a filmes e séries em qualidade padrão."
- `"premium"`: "Acesso a filmes e séries em HD e 2 telas simultâneas."
- `"gold"`: "Acesso a filmes e séries em 4K, 4 telas e conteúdo exclusivo."
- `default`: "Tipo de assinatura não reconhecido."

Faça cada exercício em um JS diferente

4. Avaliação de Atendimento:

Peça uma nota de 1 a 5 para um atendimento. Use um `switch` com `case` s agrupados:

- `case 1:` e `case 2:` -> "Que pena! Vamos trabalhar para melhorar."
- `case 3:` -> "Obrigado pelo feedback. Estamos sempre buscando evoluir."
- `case 4:` e `case 5:` -> "Que ótimo! Ficamos felizes com a sua satisfação."
- `default:` "Nota inválida."

Faça cada exercício em um JS diferente

Use o exercício 3 e crie um sistema com interface utilizando IA