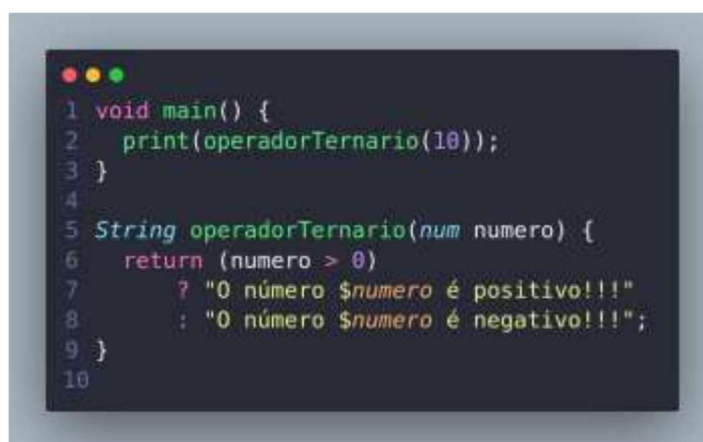


Operador ternário

Um operador ternário é uma forma de declaração de uma estrutura *IF/Else*, de forma a simplificar sua sintaxe.

Essa abordagem usa uma sintaxe alternativa para cada bloco da estrutura do *IF*, onde o “*IF*” é substituído por um carácter “?”, o qual pode ser lido como “Caso verdadeiro”, e o “*Else*” é substituído por um carácter “:”, o qual pode ser lido como “Caso falso”.



```
1 void main() {  
2     print(operadorTernario(10));  
3 }  
4  
5 String operadorTernario(num numero) {  
6     return (numero > 0)  
7         ? "O número $numero é positivo!!!"  
8         : "O número $numero é negativo!!!";  
9 }  
10
```

Switch

Essa estrutura de decisão é muito parecida com a estrutura *IF*, se destacando pelo fato de possuir uma estrutura voltada para validação de diversos cenários distintos. Podemos dizer que a estrutura *Switch* permite uma bateria de testes de forma organizada e simplificada, uma ótima alternativa para momentos em que temos diversos cenários para validar.

O *Switch* faz uso de uma estrutura de casos, onde cada caso representa um cenário possível. Sendo que existe também um caso padrão para a possibilidade do valor testado não se enquadrar em nenhum dos dados previstos na estrutura do *Switch*.

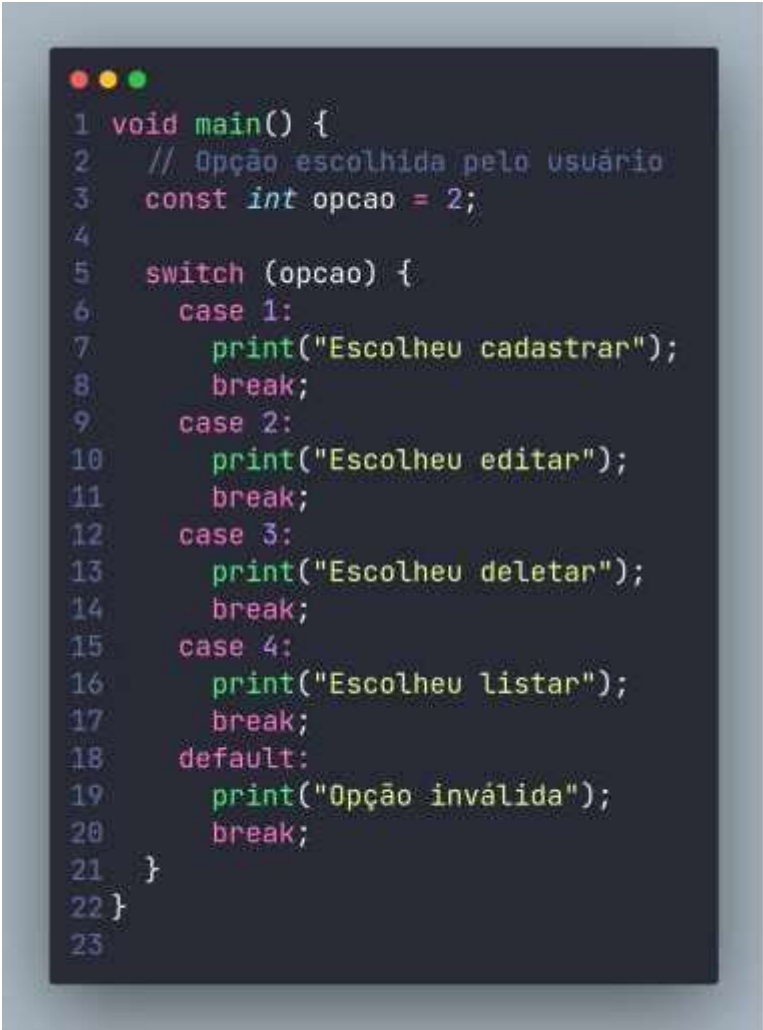
Outro aspecto importante sobre essa estrutura é o fato de não possuir um conjunto de delimitadores para seus cenários, ou seja, não utilizamos chaves “{}” para delimitar o encerramento de um cenário, assim como faríamos no *IF*. Mas sim utilizamos uma

instrução *break*, para que seja possível parar a execução da estrutura sempre que um cenário for atendido.

Também é válido ressaltar que o bloco de cenário padrão (*default*) não é obrigatório, mas recomendado para informar que nenhum cenário previsto é válido para o valor informado.

No primeiro exemplo encontramos uma validação de *strings* representando uma estrutura de menu, onde o usuário poderá escolher entre uma gama limitada de opções previamente definidas, onde cada opção representa uma operação dentro do sistema.

Vejamos um exemplo do uso dessa estrutura. Nesse exemplo é simulada a escolha do usuário dentro uma gama limitada de opções. Logo que a escolha é feita, a estrutura Switch entra em ação para verificar a operação escolhida pelo usuário.



```
1 void main() {
2     // Opção escolhida pelo usuário
3     const int opcao = 2;
4
5     switch (opcao) {
6         case 1:
7             print("Escolheu cadastrar");
8             break;
9         case 2:
10            print("Escolheu editar");
11            break;
12        case 3:
13            print("Escolheu deletar");
14            break;
15        case 4:
16            print("Escolheu listar");
17            break;
18        default:
19            print("Opção inválida");
20            break;
21    }
22 }
23
```