

A decorative wavy line in yellow and white on the left side of the slide.

ESTRUTURA CONDICIONAL- SWITCH-CASE

MICHELLE NERY NASCIMENTO
PUC MINAS

ALGORITMOS E TÉCNICAS DE PROGRAMAÇÃO

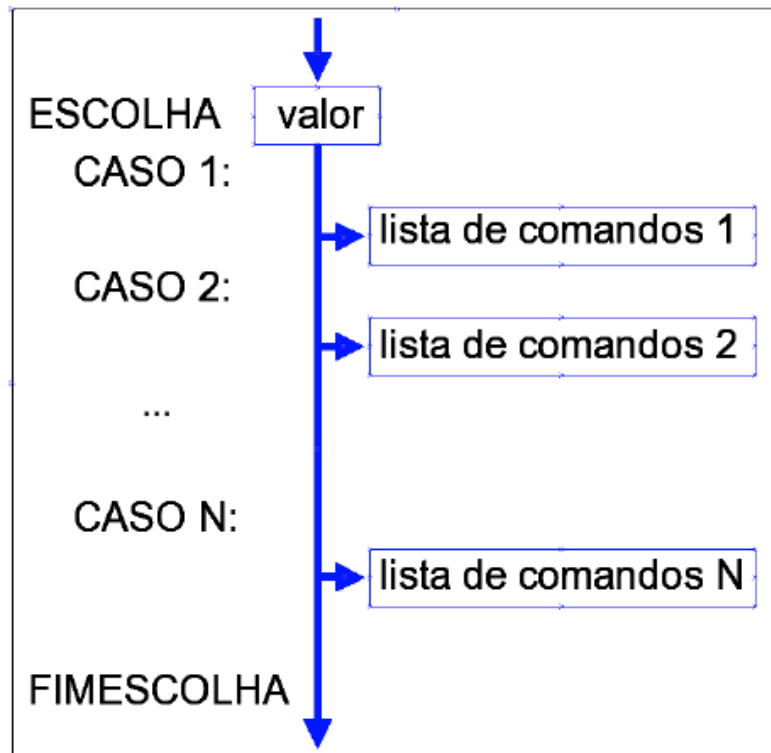
ESTRUTURA CONDICIONAL

- Permite a escolha do grupo de ações e estruturas a ser executado quando determinada condição **é ou não satisfeita**.
- A condição é representada por **expressões lógicas** e deve ser do tipo byte, short, int, char. A partir do JDK 7, a expressão também pode ser uma String.

COMANDO SWITCH-CASE

- Permite mais de uma comparação

Fluxo de controle:



```
switch (expressão) {  
    case valorI:  
        comandosI;  
        break;  
  
    case valorK:  
        comandosK;  
        break;  
  
    default:  
        comandos;  
}
```

COMANDO SWITCH-CASE

- A expressão é avaliada e o seu valor é comparado com os valores associados às cláusulas **case** em sequência.
- Quando o valor associado a uma cláusula é igual ao da expressão, os comandos da cláusula são executados até encontrar um **break**.

COMANDO SWITCH-CASE

- **Se não existir um break** na cláusula selecionada, os comandos das cláusulas seguintes são executados em ordem até encontrar um **break** ou **o fim do switch**.
- Se nenhuma das cláusulas contém o valor de seleção, a cláusula **default** (se existir) é executada.

```
import java.util.*;
public class Switch Exemplo{
    Scanner sc = new Scanner (System.in);
    public static void main(String [] args){
        int i;
        System.out.print("Entre com um número inteiro positivo");
        i = sc.nextInt();
        switch (i) {
            case 0 :
                System.out.println("zero");
                break;
            case 1 :
                System.out.println("um");
                break;
            case 2 :
                System.out.println("dois");
                break;
            case 3 :
                System.out.println("três");
                break;
            default:
                System.out.println("i é quatro ou mais");
        }
    }
}
```

- É possível ter cases vazios:

```
switch (i) {  
    case 1 :  
    case 2 :  
    case 3 :  
        System.out.println("i é 1, 2 ou 3");  
        break;  
    case 4 :  
        System.out.println("quatro");  
        break;  
}
```

- Se i tiver 1,2 ou 3, a primeira instrução **println()** será executada.
- Isso é comum quando vários **cases** compartilham o mesmo código.

- **É possível ter instruções switch aninhadas:**

```
switch (ch1) {  
    case 'A' :  
        System.out.println("A do switch mais externo");  
        switch (ch2) {  
            case 'A' :  
                System.out.println("A do switch mais interno");  
                break;  
            case 'B' :  
                System.out.println("B do switch mais interno");  
                break;  
            // ...  
        } //fim do switch mais interno  
        break;  
    case 'B' :// ...  
  
}
```


Exemplo com String:

```
String comando = "cancelar";  
switch (comando) {  
    case "conectar" :  
        System.out.println("Conectando...");  
        break;  
    case "cancelar" :  
        System.out.println("Cancelando...");  
        break;  
    case "desconectar" :  
        System.out.println("Desconetando...");  
        break;  
    default:  
        System.out.println("Comando errado!");  
        break;  
}
```

- Sob que condições devo usar **if-else-if** em vez de **switch**?
- Em geral, use **if-else-if** quando:
 - as condições que controlam o processo de seleção não dependem de um único valor, ou seja, de uma única variável;
 - ou quando precisa testar valores de ponto flutuante ou outros tipos que não sejam válidos em uma expressão **switch**.

EXERCÍCIOS

1) Faça um programa que receba o código de origem de um animal e mostre seu nome, conforme tabela a seguir. Caso o usuário digite outro código, deve aparecer na tela “Código Inválido”.

Código	Produto
1	Cavalo
2	Leão
3	Onça
4	Tigre

2) Faça uma calculadora com as quatro operações básicas. Caso a operação não exista, digite “operação inválida”. Apresente um menu para o usuário.

3) Uma loja fornece 5% de desconto para funcionários (F) e 10% de desconto para clientes especiais (E). Faça um algoritmo que calcule o total a ser pago por uma pessoa. o algoritmo deverá ter o valor total da compra efetuada e um código que identifique se o comprado é um cliente comum (C) ou um dos dois tipos de cliente que recebem desconto.