

# ILP-010 – Linguagem de Programação

---

ADS – Análise e Desenvolvimento de Sistemas  
Prof. Vagner Macedo

Aula 07  
Estruturas de controle condicional (“if-else” / “switch-case”)

# Controle de Fluxo

- Podemos classificar os comandos aceitos pela linguagem Java em basicamente quatro categorias:
  - Tomada de Decisões
    - If-Else
    - Switch-Case
  - Laços e Repetições
    - For
    - While
    - Do-While

# Controle de Fluxo

- Podemos classificar os comandos aceitos pela linguagem Java em basicamente quatro categorias:
  - Tratamento de Exceções
    - Try-Catch-Finally
    - Throw
  - Desvios
    - Break
    - Continue
    - Return

# If-Else

- A forma mais simples de controle de fluxo é o comando *if*; e a segunda forma mais simples é a *if-else*.
- Esta segunda define um **ou** outro caminho a ser seguido de acordo com a avaliação de uma **expressão** booleana, ou seja, sempre retornará **TRUE** ou **FALSE**.

# If-Else

- Sintaxe:

```
if ( [condição] )  
    [comando 1] // se true  
else  
    [comando 2] // se false
```

# If-Else

- Operadores de comparação:

== igual

!= diferente

> maior

>= maior ou igual

< menor

<= menor ou igual

&& e

|| ou

# If-Else

- Exemplo 1:

```
int value1 = 1, value2 = 2;
```

```
if(value1 == value2) System.out.println("value1 == value2");
```

```
if(value1 != value2) System.out.println("value1 != value2");
```

```
if(value1 > value2) System.out.println("value1 > value2");
```

```
if(value1 < value2) System.out.println("value1 < value2");
```

```
if(value1 <= value2) System.out.println("value1 <= value2");
```

- Saída:

```
value1 != value2
```

```
value1 < value2
```

```
value1 <= value2
```

# If-Else

- Exemplo 2:

```
int value1 = 1;
```

```
int value2 = 2;
```

```
if((value1 == 1) && (value2 == 2))
```

```
    System.out.println("value1 is 1 AND value2 is 2");
```

```
if((value1 == 1) || (value2 == 1))
```

```
    System.out.println("value1 is 1 OR value2 is 1");
```

- Saída:

```
value1 is 1 AND value2 is 2
```

```
value1 is 1 OR value2 is 1
```



# If-Else

- Outro operador de condição é o:  
? (interrogação)
- Este operador também é conhecido por **operador ternário**, pois usa 3 operandos.
- Sintaxe:

[var] = [condição] ? [valor1] : [valor2]

# If-Else

- Exemplo:

```
int x = 1;  
int y = 2;  
int z = (x==y) ? 4 : 8;  
System.out.println(z);
```

- Saída:

- 8

# If-Else

- Exemplo:

```
public static void main(String[] args) {  
    int i = 8;  
    // if que dirá se 8 é par ou não.  
    if (i % 2 == 0)  
        System.out.println("Oito é divisível por dois");  
    else  
        System.out.println("Oito não é divisível por dois");  
  
    // Verifica se o nome é Fatec (sempre utilizar o “equals” para comparar Strings)  
    String nome = "Fatec ";  
    if (nome.equals("Fatec "))  
        System.out.println("Nome é Fatec");  
    else  
        System.out.println("Nome não é Fatec");  
}
```

# Switch-Case

- Há situações em que se sabe de antemão quais condições assumem o valor true.
- Nesses casos, a linguagem Java provê um comando de controle de fluxo chamado ***switch***.

# Switch-Case

- Sintaxe:

```
switch([expressão]) {  
  case [constante 1]:  
    [comando 1]  
    break;  
  case [constante 2]:  
    [comando 2]  
    break;  
  .  
  .  
  .  
  case [constante n]:  
    [de comando n]  
    break;  
  default:  
    [comando]  
}
```

---

# Switch-Case

- A única limitação do switch é que, para versões antigas do Java (< 8), ele deve retornar um valor numérico inteiro.
- O switch não funciona com String, float, char e boolean, por exemplo.
- Nas versões de Java que não aceitam Switch para Strings, é possível utilizar o método hashCode (), que retorna um código inteiro único para cada String diferente.

**Obrigado!**

