

Linguagem de Programação Variáveis, Condição e Operadores Lógicos

Jose.wellington@uniceub.br

Calendário

◀ agosto de 2013 ▶

D	S	T	Q	Q	S	S
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

◀ setembro de 2013 ▶

D	S	T	Q	Q	S	S
25	26	27	28	29	30	31
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	1	2	3	4	5

◀ outubro de 2013 ▶

D	S	T	Q	Q	S	S
29	30	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2

◀ novembro de 2013 ▶

D	S	T	Q	Q	S	S
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

◀ dezembro de 2013 ▶

D	S	T	Q	Q	S	S
24	25	26	27	28	29	30
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Agenda

- Palavras Reservadas
- Operações Lógicas
- Condição IF e ELSE
- Exercício

Netbeans

IDE - Integrated Development Environment



- O NetBeans é uma apenas uma Interface para auxiliar o desenvolvimento.
- Nessa IDE conseguimos trabalhar com a linguagem Java.

IDE - Integrated Development Environment

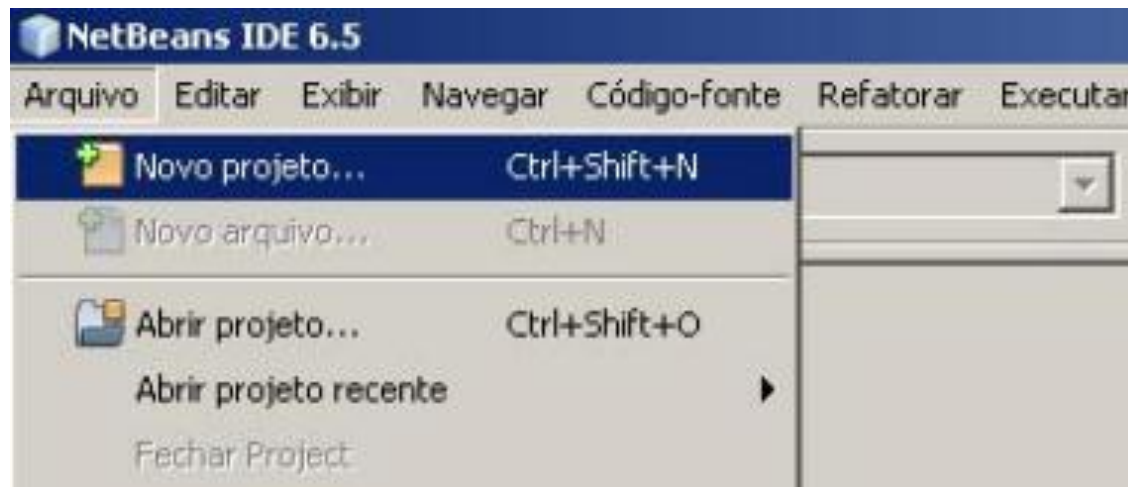


- Editor de código fonte
- Colorização de sintaxe
- Teclas de acesso rápido a recursos
- Edição de telas
- Debug
- Gerenciamento de versões

IDE - Integrated Development Environment

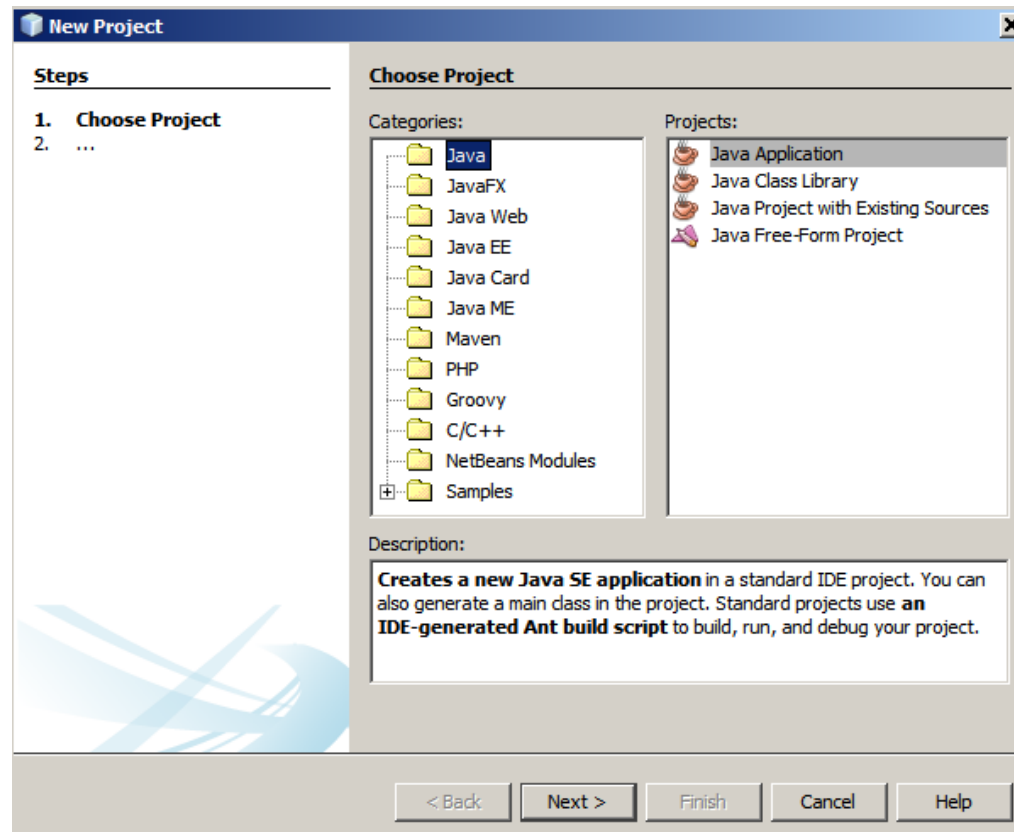
Criando um novo Projeto

Clique em Arquivo-Novo projeto... ou use o atalho Ctrl+Shift+N, ou ainda tem um botão na barra de ferramentas para criar o novo projeto



IDE - Integrated Development Environment

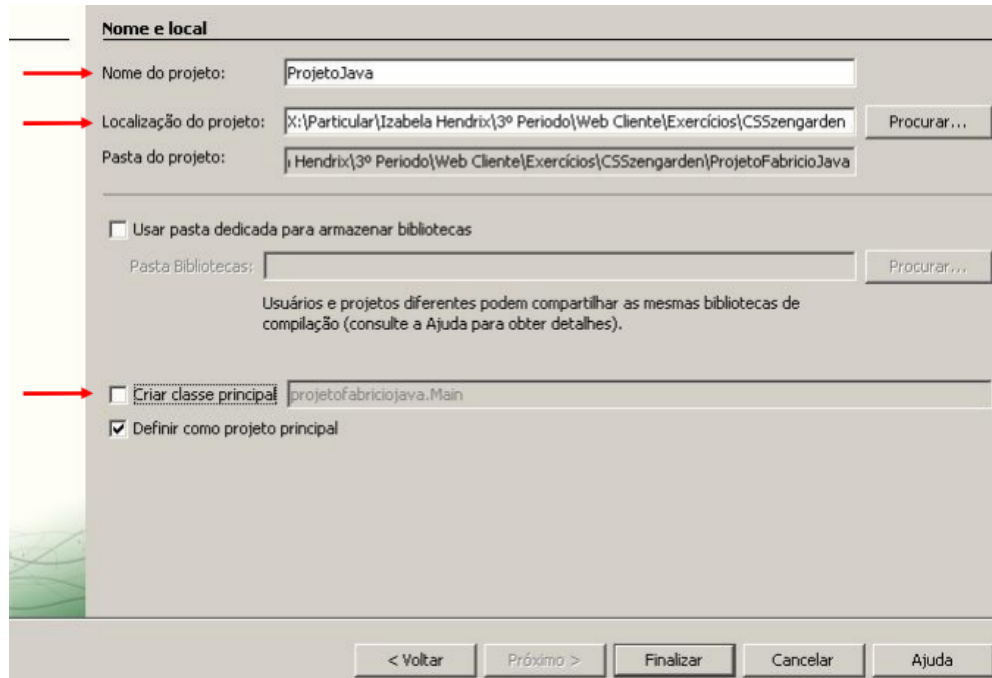
Vai aparecer uma janela como na figura abaixo, selecione no lado esquerdo a Categoria Java.



IDE - Integrated Development Environment

Nome do projeto (lembrando que nome de projetos e classes são sempre escritos em letras minúsculas, mas a primeira letra é maiúscula.

Sempre sem espaços entre as palavras e a primeira letra da segunda palavra maiúscula. Ex.: ProjetoJava).



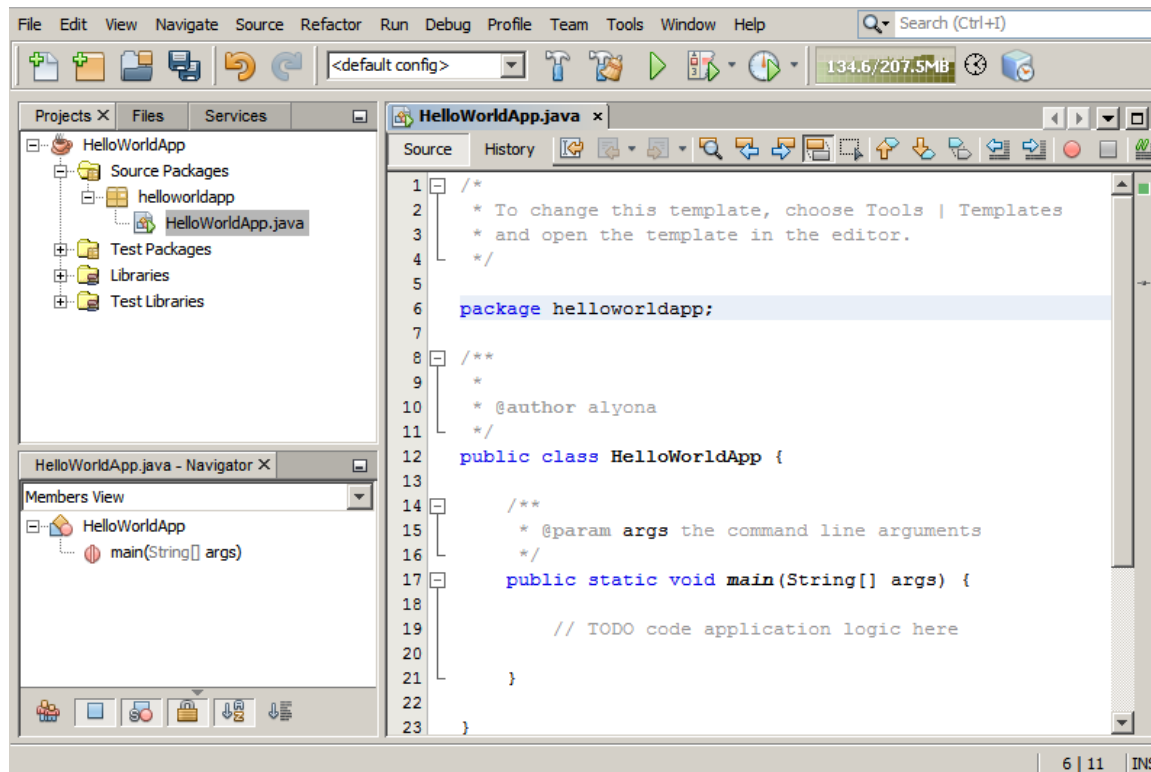
The screenshot shows the 'Nome e local' (Name and location) dialog box in an IDE. It contains the following fields and options:

- Nome do projeto:** ProjetoJava
- Localização do projeto:** X:\Particular\Izabela Hendrix\3º Período\Web Cliente\Exercícios\CSSzengarden (with a 'Procurar...' button)
- Pasta do projeto:** Hendrix\3º Período\Web Cliente\Exercícios\CSSzengarden\ProjetoFabricioJava
- ☐ Usar pasta dedicada para armazenar bibliotecas
- Pasta Bibliotecas:** (with a 'Procurar...' button)
- Usuários e projetos diferentes podem compartilhar as mesmas bibliotecas de compilação (consulte a Ajuda para obter detalhes).
- ☐ Criar classe principal: projetofabriciojava.Main
- ☒ Definir como projeto principal

At the bottom, there are buttons: < Voltar, Próximo >, Finalizar, Cancelar, and Ajuda. Red arrows point to the 'Nome do projeto', 'Localização do projeto', and 'Criar classe principal' fields.

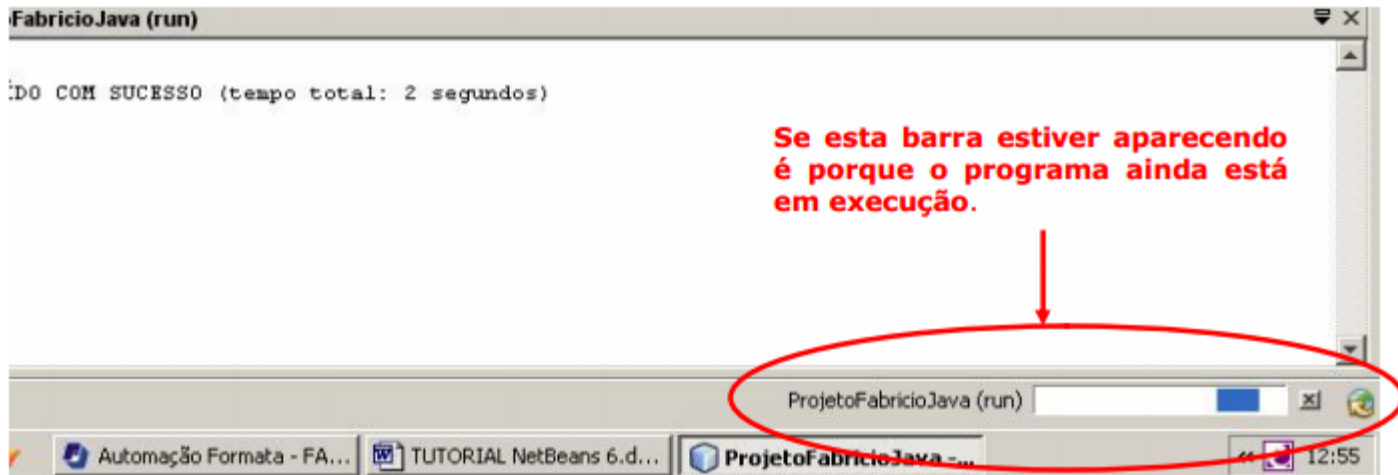
IDE - Integrated Development Environment

A janela Projeto, que contém uma view em árvore dos componentes do projeto, incluindo arquivos de código-fonte, bibliotecas de que seu código depende, e assim por diante.



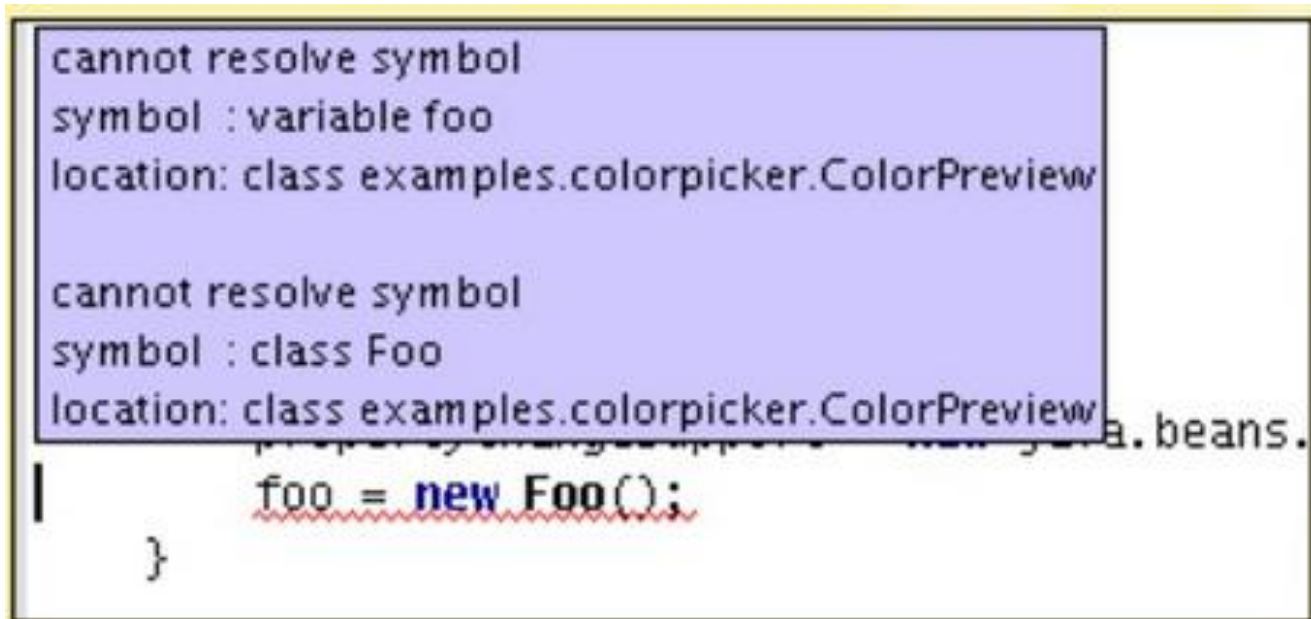
IDE - Integrated Development Environment

Para executar o projeto você pode utilizar o atalho Shift+F6 ou clique em Executar arquivo.



IDE - Integrated Development Environment

Correção em tempo real



The screenshot shows a code editor with a yellow border. A light blue box highlights two error messages:

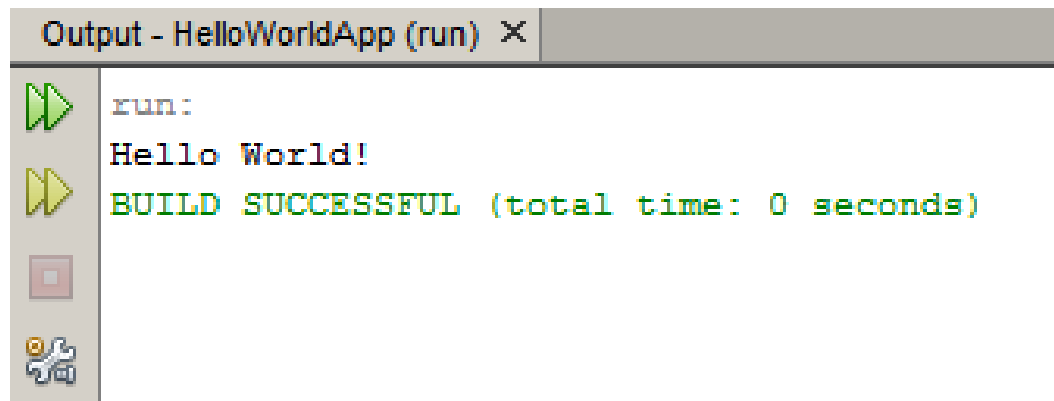
```
cannot resolve symbol  
symbol : variable foo  
location: class examples.colorpicker.ColorPreview  
  
cannot resolve symbol  
symbol : class Foo  
location: class examples.colorpicker.ColorPreview
```

Below the errors, the following code is visible:

```
... java.beans.  
|   foo = new Foo();  
    }
```

IDE - Integrated Development Environment

Parabéns! Seu programa funciona!



```
run:
Hello World!
BUILD SUCCESSFUL (total time: 0 seconds)
```

Variáveis

Tipos Primitivos

Tipo	Descrição	Tamanho	Faixa de valores
<code>char</code>	Armazena um único caracter	1byte	Todos os caracteres disponíveis na tabela ASCII.
<code>short int</code> (<code>short</code>)	Números inteiros "pequenos".	2bytes	com sinal: -32.768 a 32.767 unsigned: 0 a 65.535
<code>int</code>	Números Inteiros.	4bytes	com sinal: -2.147.483.648 a 2.147.483.647 unsigned: 0 a 4.294.967.295
<code>long int</code> (<code>long</code>)	Números inteiros "longos" (grandes).	4bytes	com sinal: -2.147.483.648 a 2.147.483.647 unsigned: 0 a 4.294.967.295
<code>bool</code>	Valores booleanos. Podem ser: true (verdadeiro) ou false (falso).	1byte	true ou false
<code>float</code>	Números reais.	4bytes	3.4e +/- 38 (7 dígitos)
<code>double</code>	Números reais de dupla precisão.	8bytes	1.7e +/- 308 (15 dígitos)
<code>Long double</code>	Números reais longos de dupla precisão.	8bytes	1.7e +/- 308 (15 dígitos)

Tipos Primitivos

```
int idade = 31;  
double preco = 12.45;  
char sexo = 'M';  
boolean casado = true;
```


Tipos Primitivos

O Java é uma linguagem fortemente tipada, ou seja, para usarmos **os tipos de informações, temos que declará-los.**

Vamos declarar um inteiro:

```
int idade;
```

Tipos Primitivos

Inicializando uma variável

Poderíamos atribuir o valor a uma variável de duas maneiras, uma na declaração:

```
int idade=21;
```

Outro meio é depois da declaração:

```
int idade;  
idade=21;
```

Tipos Primitivos

O tipo char: armazenando e representando caracteres

Declaração

```
char nome_do_char = 'a';
```

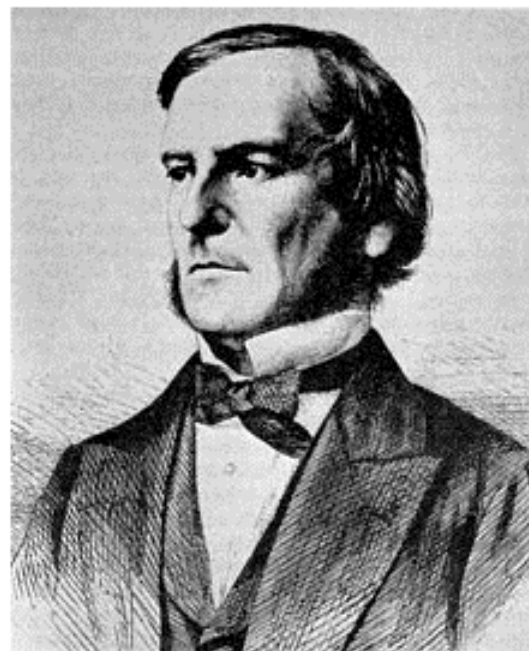
Onde poderíamos substituir 'a' por qualquer caractere alfanumérico, caso tenhamos declarado assim.

Tipos Primitivos

O tipo boolean: a base da lógica Booleano é um tipo de dado que permite apenas dois valores, true (verdadeiro) ou false (falso).

Declaração

```
boolean nome_bool = true;  
boolean nome_bool2 = false;
```



George Boole

Palavras Reservadas

Palavras reservadas

abstract	do	implements	private	throw
boolean	double	import	protected	throws
break	else	instanceof	public	transient
byte	extends	int	return	true
case	false	interface	short	try
catch	final	long	static	void
char	finally	native	super	volatile
class	float	new	switch	while
continue	for	null	synchronized	
default	if	package	this	

⇒ Nota: atualmente as seguintes palavras reservadas não são utilizadas: cast, const, future, generic, goto, inner, operator, outer, rest, var.

Operações Lógicas

Fazendo comparações

Maior que: >

$a > b$ -> retorna 'true' caso 'a' seja maior que 'b', e 'false' caso seja menor

Menor que: <

$a < b$ -> retorna 'true' caso 'a' seja menor que 'b', e 'false' caso seja maior

Fazendo comparações

Maior ou igual a: \geq

$a \geq b \rightarrow$ retorna 'true' caso 'a' seja maior ou igual à 'b', e 'false' caso seja menor

Menor ou igual a: \leq

$a \leq b \rightarrow$ retorna 'true' caso 'a' seja menor ou igual à 'b', e 'false' caso seja maior

Fazendo comparações

Comparação de igualdade: ==

`a == b` -> retorna 'true' caso 'a' seja igual a b, e 'false' caso contrário

Comparação de negação: !=

`a != b` -> retorna 'true' caso 'a' seja diferente de b, e 'false' caso contrário

Operadores lógicos e de negação

Operadores lógicos

O primeiro exemplo é representado por: (condicao_A **&&** condicao_B)

(condicao_A AND condicao_B)

Fazendo comparações

Exemplo é representado por: (condicao_A || condicao_B)

(condicao_A OR condicao_B)

Negando declarações: !

true = !false

false = !true

IF e ELSE

IF

- Em inglês quer dizer 'se'.

```
if ( condição ){  
    caso a condição seja verdadeira  
    esse bloco de código será executado  
}
```


IF

```
public class Ifelse {  
    public static void main(String[] args) {  
        if (1 == 2){  
            System.out.println("Você nunca lerá essa mensagem");  
        }  
  
        if (1 == 1){  
            System.out.println("1 é igual a 1 ");  
        }  
    }  
}
```

ELSE

Do inglês: senão

```
if ( condição ){  
    caso a condição seja verdadeira  
    esse bloco de código será executado  
} else {  
    caso a condição seja falsa  
    esse bloco de código que será executado  
}
```



ELSE

```
if( nota >= 7.0 ){  
    System.out.println("Parabéns, você passou direto");}  
else {  
    System.out.println("Não passou direto");}
```

Exercício

Exercício 06

- Faça um programa que crie três variáveis com valores e diga qual deles é o maior?

Agenda

- Palavras Reservadas
- Operações Lógicas
- Condição IF e ELSE
- Exercício