

DESENVOLVIMENTO A DESKTOP(DAW)

Ementa

Introdução.

Ambiente de Desenvolvimento.

Sintaxe Básica da Linguagem.

Identificadores e tipos primitivos.

Operadores.

Conversão de tipos.

Estrutura de controle.

Manipulação de Strings.

Manipulação de Array.





Horário das aulas

| Horário / dias | Segunda-feira | Terça-feira | Quarta-feira | Quinta-feira | Sexta-feira |
|----------------|---------------|-------------|--------------|--------------|-------------|
| 18:40 / 19:20 | | | | DAW | |
| 19:20 / 20:00 | | | | DAW | |
| 20:00 / 20:40 | | | | | DAW |
| 20:40 / 21:20 | | | | | DAW |



Datas e Informações

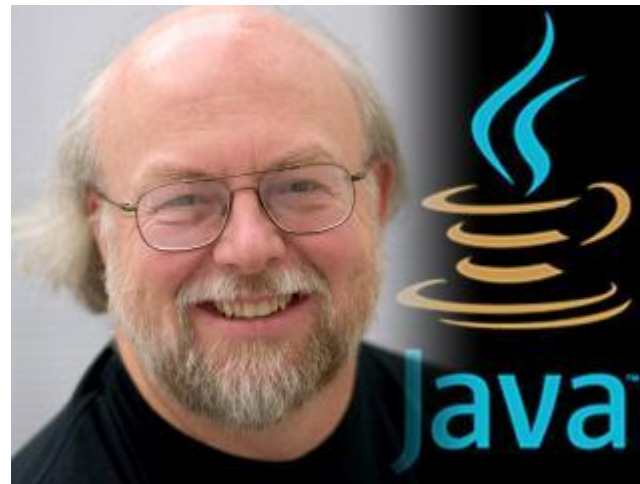
- Nota 1: AT1 + AT2 (5 pontos + 5 pontos)
 - data:
- Nota 2: Prova (10 pontos)
 - data:
- Recuperação: Prova (10 pontos)
 - data:



Um pouca de história...

Origem de Java

Java surgiu do **projeto Green**, em **1991**, na Sun Microsystems, que tinha como objetivo **prever qual seria a próxima onda em computação**.



James Gosling
Pai do Java



Um pouca de história...

A tecnologia Java surgiu em um pequeno e secreto projeto da Sun Microsystems chamado “The Green Project” em 1991.

O time era **composto por 13 pessoas e liderado por James Gosling.**

Com a ideia que envolviam problemas com dispositivos eletrônicos da década de 90: Tvs, geladeiras, videocassetes entre outros dispositivos. Pois cada um deles tinha um código diferente necessitando de uma linguagem própria para funcionar ou reescrevendo quando tivesse que trocar algum chip e isso não fazia o maior sentido.

Sua equipe tiveram a ideia de escrever apenas um código e funcionasse para todas as máquinas. Usando um intermediário para tradutor do código e cada aparelho pudesse entender.



Um pouca de história...

Adoção do Java

Algumas empresas que investem em Java



A Tecnologia Java

A tecnologia Java é composta por dois componentes fundamentais:

- A linguagem de programação Java.
- A Plataforma Java
 - Que por sua vez é composta por: Java Virtual Machine - JVM (Máquina Virtual Java)
 - Java Application Programming Interface (API Java)

Algumas empresas que investem em Java



A Tecnologia Java

Linguagem Java Código fonte da
Aplicação

Plataforma Java

Dependente do Hardware
Normalmente um SO

MeuPrograma.java

API Java

JVM (Máquina Virtual Java)

Plataforma Nativa



A Linguagem Java

O objetivo principal da linguagem Java é ser **Simples, poderosa e eficiente**. **Gosling definiu algumas características que mostram que java cumpre esses requisitos.**

- Simples e Orientada a Objetos
- Robusta e Segura
- Neutra em relação à arquitetura e Portável.



A Linguagem Java

Simple e Orientada a Objetos.

Java foi **desenvolvida para ser uma linguagem com uma sintaxe simples, resumida e de fácil aprendizado**. Os principais conceitos da linguagem são absorvidos rapidamente pelo desenvolvedor o que em pouco tempo o deixa apto a produção.



A Linguagem Java

Neutra em Relação à Arquitetura e Portável.

Conceito neutra em relação a arquitetura queremos dizer que o código compilado Java pode rodar em diversas plataformas de hardware sem que seja preciso modificar ou recompilar o código.



A Linguagem Java

- Java ME - Java Micro Edition - dispositivos embarcados e móveis - IoT
 - <http://www.oracle.com/technetwork/java/javame>
- **Java SE - Java Standard Edition - core - desktop e servidores**
 - <http://www.oracle.com/technetwork/java/javase>
- Java EE - Java Enterprise Edition - aplicações corporativas
 - <http://www.oracle.com/technetwork/java/javaee>



A Linguagem Java

- **Java Virtual Machine (JVM)**

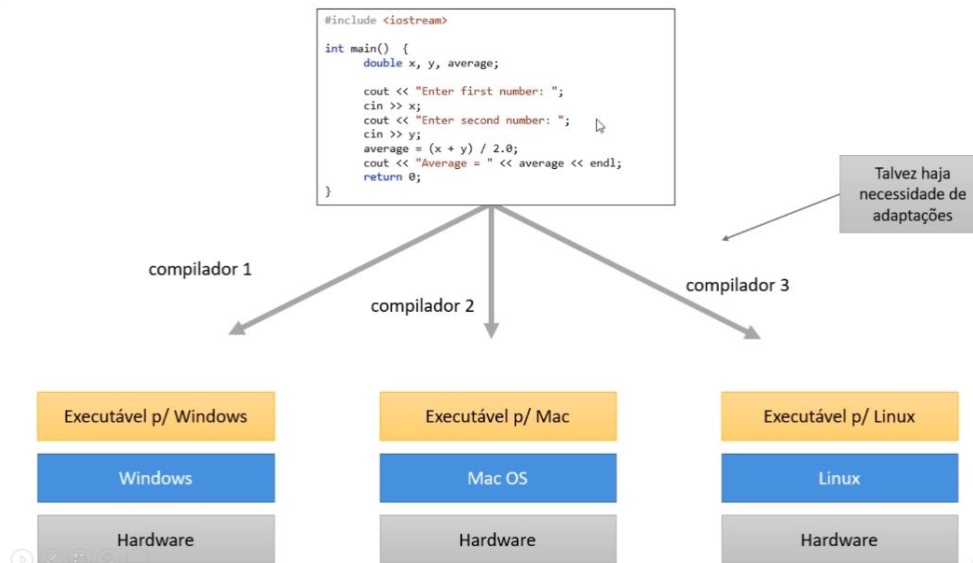
- Componente da plataforma Java que assegura a independência das aplicações entre diferentes plataformas.
- É uma espécie de tradutor existente nos dispositivos para traduzir comandos da linguagem Java para a linguagem da máquina nativa.



Compiladas mais rápidas C++

Código fonte passar por um compilador e vai gerar o arquivo executável exclusivo para aquela plataforma que estamos usando.

Gerando problema de portabilidade, eventualmente para que rode em todos os sistemas operacionais talvez temos que fazer mudanças no código.



Interpretadas mais lenta, porém mais flexível PHP

O php distribui um interpretador para cada sistema operacional.

Permite que o mesmo código sem nenhuma alteração rodem em qualquer sistema operacional mas isso deixa o programa um pouco mais lento.

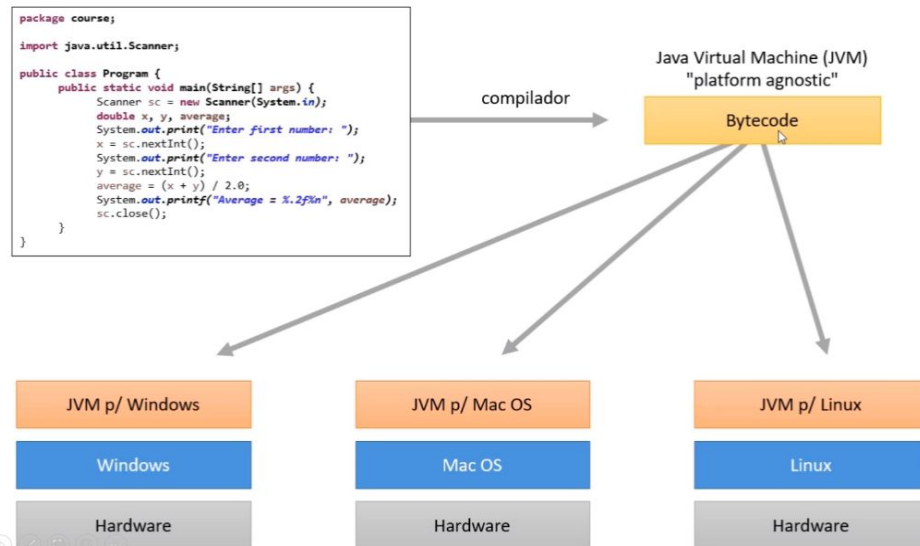
```
<?php
print "Enter first number: ";
$x = trim(fgets(STDIN));
print "Enter second number: ";
$y = trim(fgets(STDIN));
$average = ($x + $y) / 2;
print "Average = $average";
?>
```



Híbrida Java

Ele é compilado e interpretado.

Já compilado verifica se está correto no aspecto sintático.



Palavras Reservadas

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



Palavras Reservadas

Modificadores de acesso

- **private:** acesso apenas dentro da classe
- **protected:** acesso por classes no mesmo pacote e subclasses
- **public:** acesso de qualquer classe



Palavras Reservadas

Modificadores de classes, variáveis ou métodos

- **abstract:** classe que não pode ser instanciada ou método que precisa ser implementado por uma subclasse não abstrata
- **class:** especifica uma classe
- **extends:** indica a superclasse que a subclasse está estendendo
- **final:** impossibilita que uma classe seja estendida, que um método seja sobrescrito ou que uma variável seja reinicializada



Palavras Reservadas

Modificadores de classes, variáveis ou métodos

- **implements:** indica as interfaces que uma classe irá implementar
- **interface:** especifica uma interface
- **native:** indica que um método está escrito em uma linguagem dependente de plataforma, como o C.
- **new:** instancia um novo objeto, chamando seu construtor



Palavras Reservadas

Modificadores de classes, variáveis ou métodos

- **static:** faz um método ou variável pertencer à classe ao invés de às instâncias
- **strictfp:** usado em frente a um método ou classe para indicar que os números de ponto flutuante seguirão as regras de ponto flutuante em todas as expressões
- **synchronized:** indica que um método só pode ser acessado por uma thread de cada vez
- **transient:** impede a serialização de campos
- **volatile:** indica que uma variável pode ser alterada durante o uso de threads



Palavras Reservadas

Controle de fluxo dentro de um bloco de código

- **break:** sai do bloco de código em que ele está
- **case:** executa um bloco de código dependendo do teste do switch
- **continue:** pula a execução do código que viria após essa linha e vai para a próxima passagem do loop
- **default:** executa este bloco de código caso nenhum dos teste de switch-case seja verdadeiro
- **do:** executa um bloco de código uma vez, e então realiza um teste em conjunto com o while para determinar se o bloco deverá ser executado novamente.



Palavras Reservadas

Controle de fluxo dentro de um bloco de código

- **else:** executa um bloco de código alternativo caso o teste if seja falso.
- **for:** usado para realizar um loop condicional de um bloco de código
- **if:** usado para realizar um teste lógico de verdadeiro o falso
- **instanceof:** determina se um objeto é uma instância de determinada classe, superclasse ou interface
- **return:** retorna de um método sem executar qualquer código que venha depois desta linha (também pode retornar uma variável)
- **switch:** indica a variável a ser comparada nas expressões case
- **while:** executa um bloco de código repetidamente enquanto a condição for verdadeira



Palavras Reservadas

Tratamento de erros

- **assert:** testa uma expressão condicional para verificar uma suposição do programador
- **catch:** declara o bloco de código usado para tratar uma exceção
- **finally:** bloco de código, após um try-catch, que é executado independentemente do fluxo de programa seguido ao lidar com uma exceção
- **throw:** usado para passar uma exceção para o método que o chamou
- **throws:** indica que um método pode passar uma exceção para o método que o chamou
- **try:** bloco de código que tentará ser executado, mas que pode causar uma exceção



Palavras Reservadas

Controle de pacotes

- **import**: importa pacotes ou classes para dentro do código.
- **package**: especifica a que pacote todas as classes de um arquivo pertencem.



Palavras Reservadas

Primitivos

- **boolean**: um valor indicando verdadeiro ou falso
- **byte**: um inteiro de 8 bits
- **char**: um caracter unicode
- **double**: um número de ponto flutuante de 64 bits
- **float**: um número de ponto flutuante de 32 bits
- **int**: um inteiro de 32 bits
- **long**: um inteiro de 64 bits
- **short**: um inteiro de 32 bits



Palavras Reservadas

Variáveis de referência

- **super:** refere-se a superclasse imediata
- **this:** refere-se a instância atual do objeto



Palavras Reservadas

Retorno de um método

- **void:** indica que o método não tem retorno



Instalando

Baixar o JRE java

<https://www.java.com/pt-BR/>

Baixar o JRE java serve apenas para executar aplicações em java.

<https://www.java.com/pt-BR/>



The screenshot shows the Java website interface. At the top, there is a red header bar with the Java logo on the left, a search bar with the text "Pesquisar" on the right, and links for "Fazer Download" and "Ajuda" in the center. Below the header, the main text reads "JAVA E VOCÊ, FAÇA DOWNLOAD HOJE" in large, bold, grey letters. Underneath this text is a red button that says "Download Gratuito do Java". Below the button are two links: "» O que é o Java?" and "» Precisa de Ajuda?". Further down, there is a section titled "Sobre o Java (Site em Inglês)" which contains a row of six icons: "go.java", "Java + Alice", "Java + Greenfoot", "Java Developer Conference", "Oracle Academy", and "Java Magazine". At the bottom of the page, there is a footer with links: "Selecionar Idioma", "Sobre o Java", "Suporte", and "Desenvolvedores", followed by the "ORACLE" logo on the right.

Instalando

Baixar o JRE do java, que vem com a virtual machine e as bibliotecas.

Recursos da Ajuda

- » [O que é o Java?](#)
- » [Remover Versões Mais Antigas](#)
- » [Desativar o Java](#)
- » [Mensagens de Erro](#)
- » [Solucionar Problemas do Java](#)
- » [Outro Tipo de Ajuda](#)

Java para Windows de 64 bits

Recomendado Version 8 Update 291 (tamanho do arquivo: 80.7 MB)

Data da release - 20 de abril de 2021



Importante Atualização de Licença do Oracle Java

A Licença do Oracle Java mudou para as releases com datas a partir de 16 de abril de 2019.

O novo [Contrato de Licença do Oracle Technology Network para Oracle Java SE](#) é substancialmente diferente das licenças anteriores do Oracle Java. A nova licença permite determinados usos, como uso pessoal e para desenvolvimento, sem custo -- entretanto, outros usos autorizados de acordo com licenças anteriores do Oracle Java talvez não estejam mais disponíveis. Analise os termos com atenção antes de fazer download deste produto e utilizá-lo. Uma seção de Perguntas Mais Frequentes está disponível [aqui](#).

A licença para uso comercial e o suporte estão disponíveis com uma [Assinatura do Java SE](#) de baixo custo.

A Oracle também oferece a mais recente release do OpenJDK de acordo com a [Licença GPL](#) de código-fonte aberto em [jdk.java.net](#).











Detectamos que você está usando o Google Chrome e talvez não consiga usar o plug-in do Java neste browser. A partir da Versão 42 (lançada em abril de 2015), o Chrome desativou a forma padrão pela qual os browsers suportam plug-ins. [Mais informações](#)

**Concordar e Iniciar
Download Gratuito**

Instalando

Baixar o JDK do java
kit para desenvolver aplicações Java

| | | |
|-------------------------------------|-----------|--|
| Linux x64 Compressed Archive | 138.22 MB |  jdk-8u291-linux-x64.tar.gz |
| macOS x64 | 207.42 MB |  jdk-8u291-macosx-x64.dmg |
| Solaris SPARC 64-bit (SVR4 package) | 133.69 MB |  jdk-8u291-solaris-sparcv9.tar.Z |
| Solaris SPARC 64-bit | 94.74 MB |  jdk-8u291-solaris-sparcv9.tar.gz |
| Solaris x64 (SVR4 package) | 134.48 MB |  jdk-8u291-solaris-x64.tar.Z |
| Solaris x64 | 92.56 MB |  jdk-8u291-solaris-x64.tar.gz |
| Windows x86 | 155.67 MB |  jdk-8u291-windows-i586.exe |
| Windows x64 | 168.67 MB |  jdk-8u291-windows-x64.exe |



Instalando

Antes do download tem que ser criada uma conta no site da oracle.



Início de sessão na conta Oracle

Nome de utilizador i

Senha i

Iniciar sessão

[Precisa de ajuda?](#)

Não tem uma Conta Oracle?

Criar Conta

© Oracle | [Condições de Utilização](#) | [Política de Privacidade](#)

Instalando o Eclipse IDE para desenvolvimento Java

Fazer o download

<https://www.eclipse.org/downloads/>



The Eclipse Installer 2021-03 R now includes a JRE for macOS, Windows and Linux.



Get **Eclipse IDE 2021-03**
Install your favorite desktop IDE packages.

[Download x86_64](#)

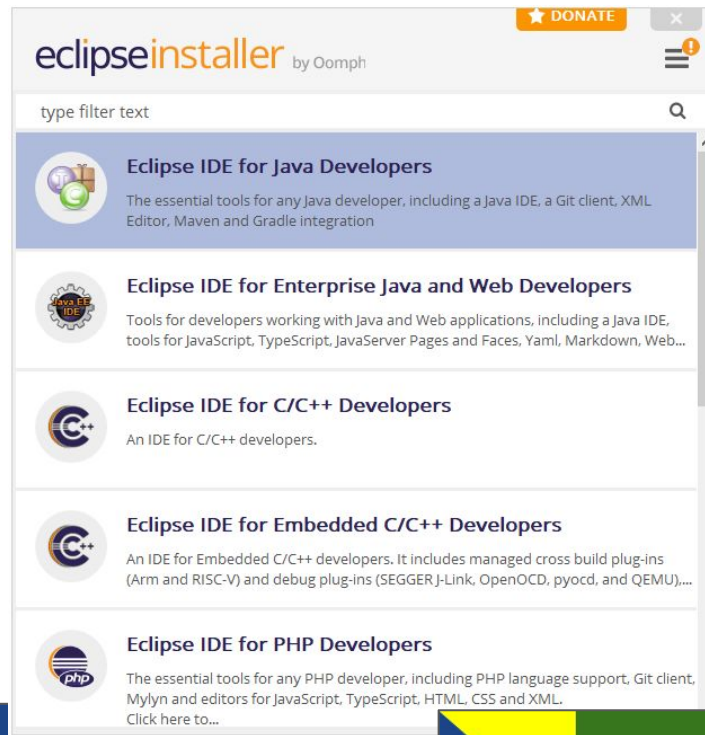
[Download Packages](#) | [Need Help?](#)



Instalando o Eclipse IDE para desenvolvimento Java

Selecione a primeira opção para instalar.

•



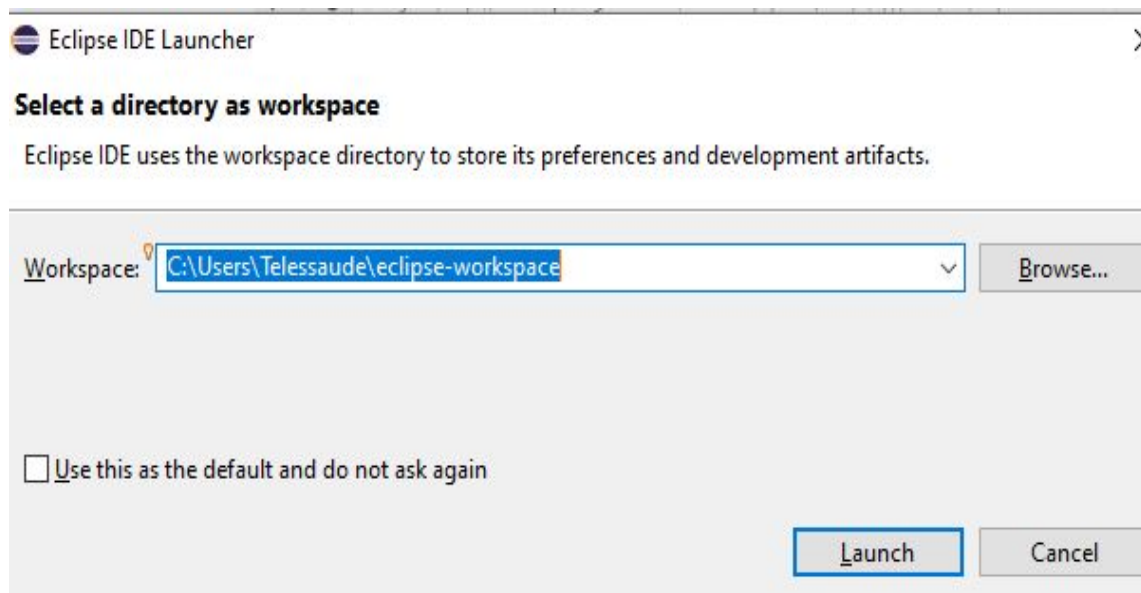
Instalando o Eclipse IDE para desenvolvimento Java

Selecione a primeira opção para instalar.



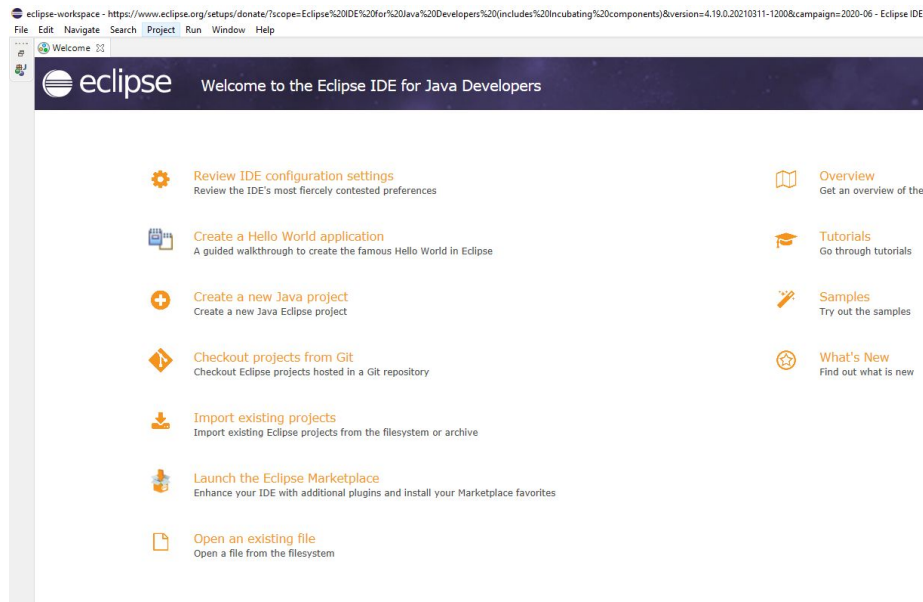
Instalando o Eclipse IDE para desenvolvimento Java

Informar o local onde vai ficar os projetos em java.



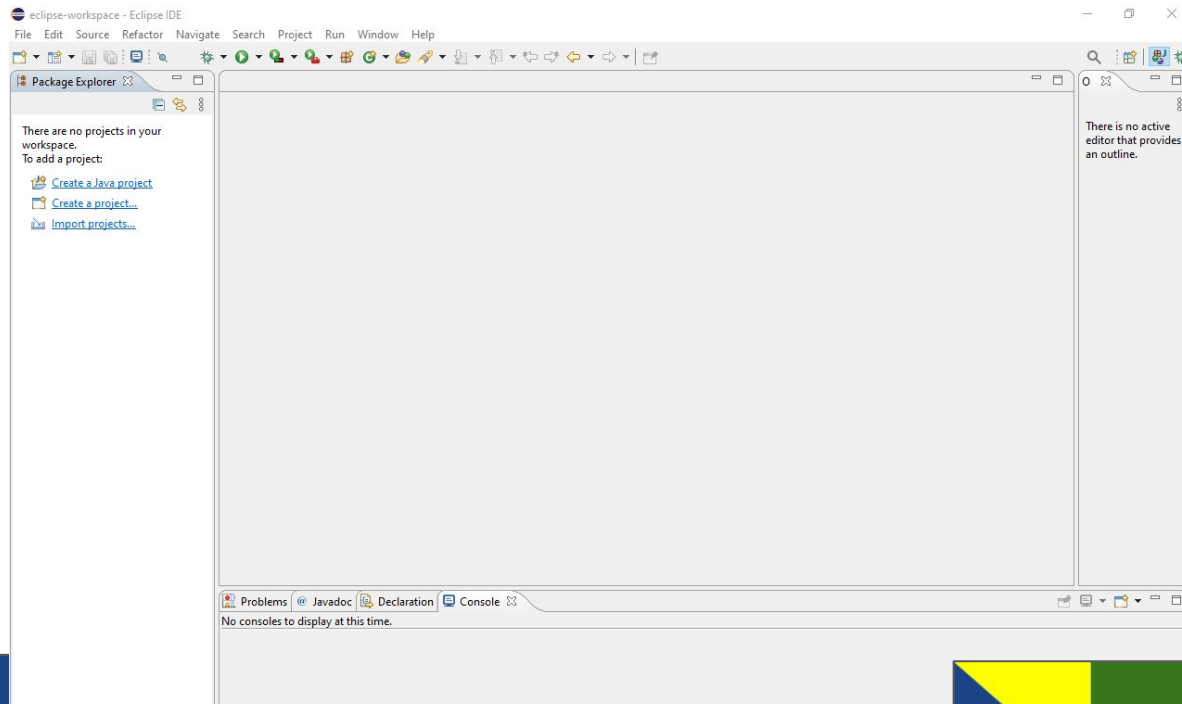
Instalando o Eclipse IDE para desenvolvimento Java

Parte visual do Eclipse



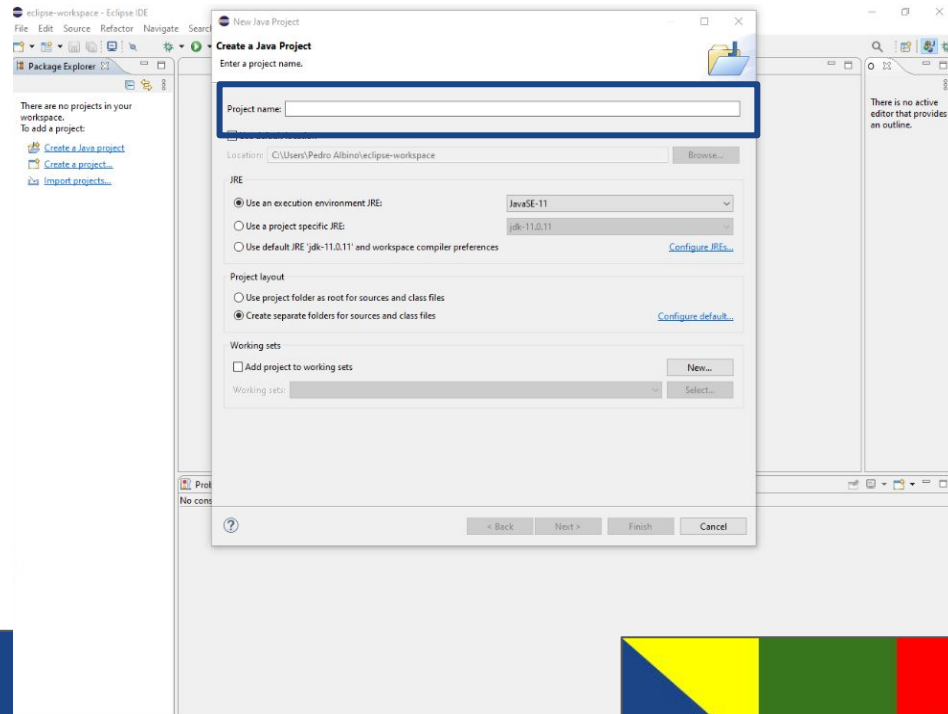
Instalando o Eclipse IDE para desenvolvimento Java

Área do Eclipse



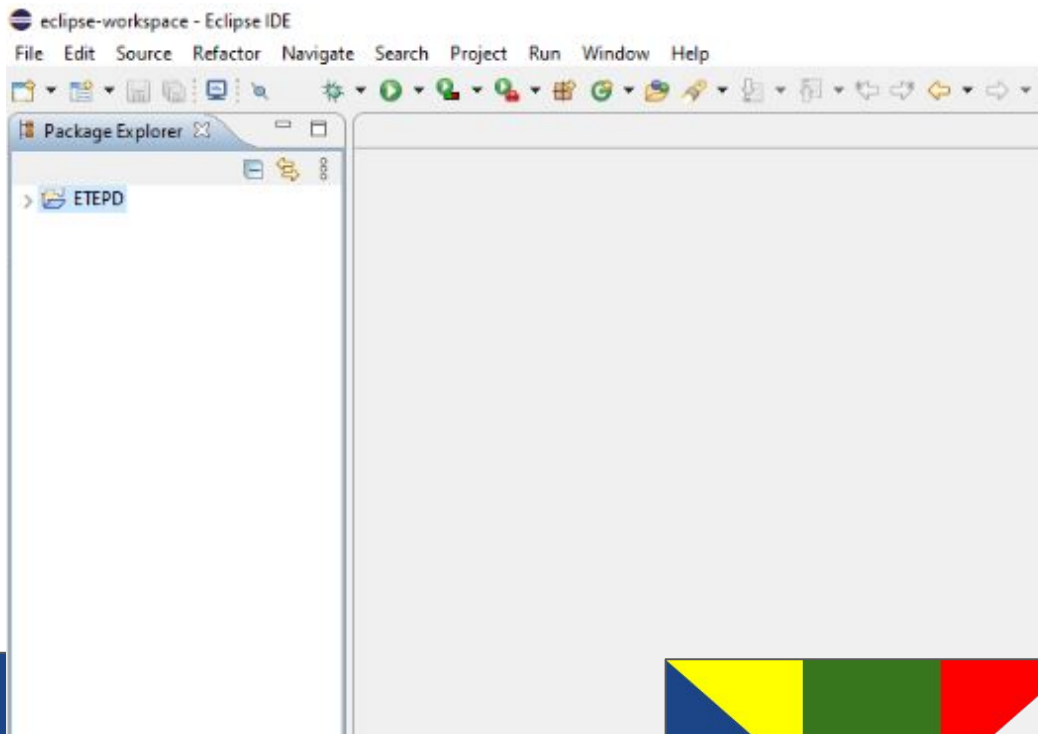
Instalando o Eclipse IDE para desenvolvimento Java

Colocar o nome do projeto depois próximo e finalizar.



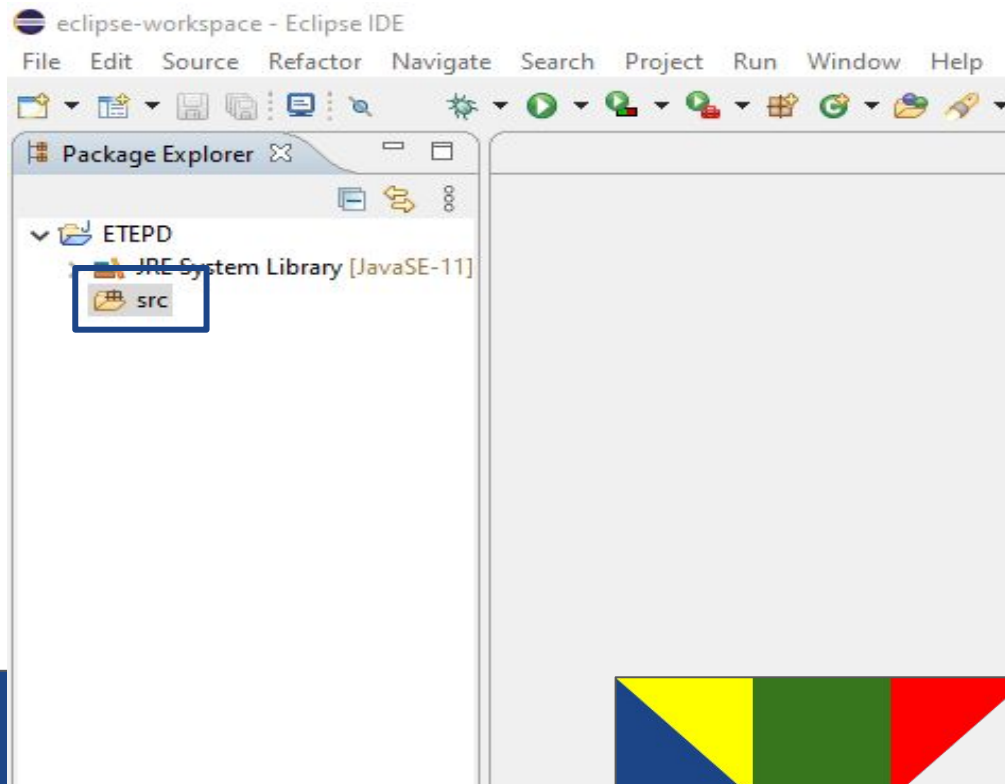
Instalando o Eclipse IDE para desenvolvimento Java

O projeto depois que foi criado vai mostrar assim.



Instalando o Eclipse IDE para desenvolvimento Java

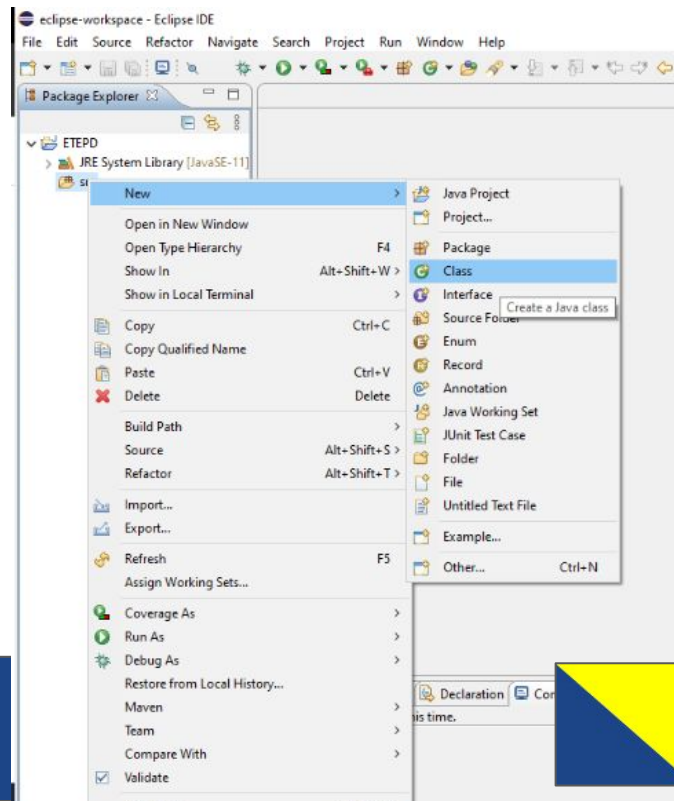
Pasta src onde fica toda as classes do projeto.



Instalando o Eclipse IDE para desenvolvimento Java

Criando nossa primeira class

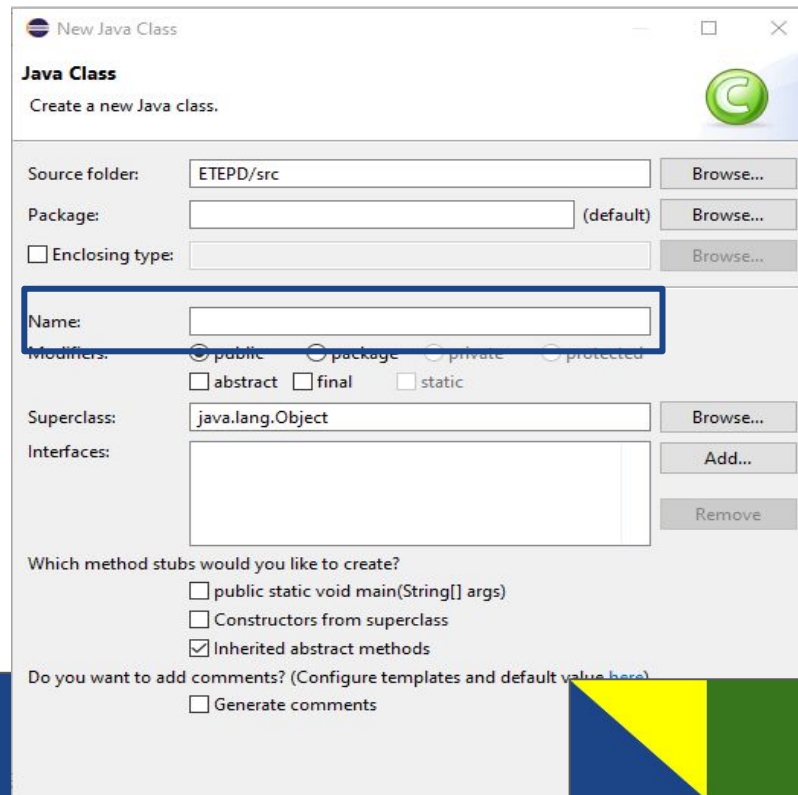
A pasta src selecionado, clique com o botão direito do mouse, New -> Class



Instalando o Eclipse IDE para desenvolvimento Java

Para criar uma classe temos que escrever no padrão chamado CamelCase.

Ex:MinhaClasse



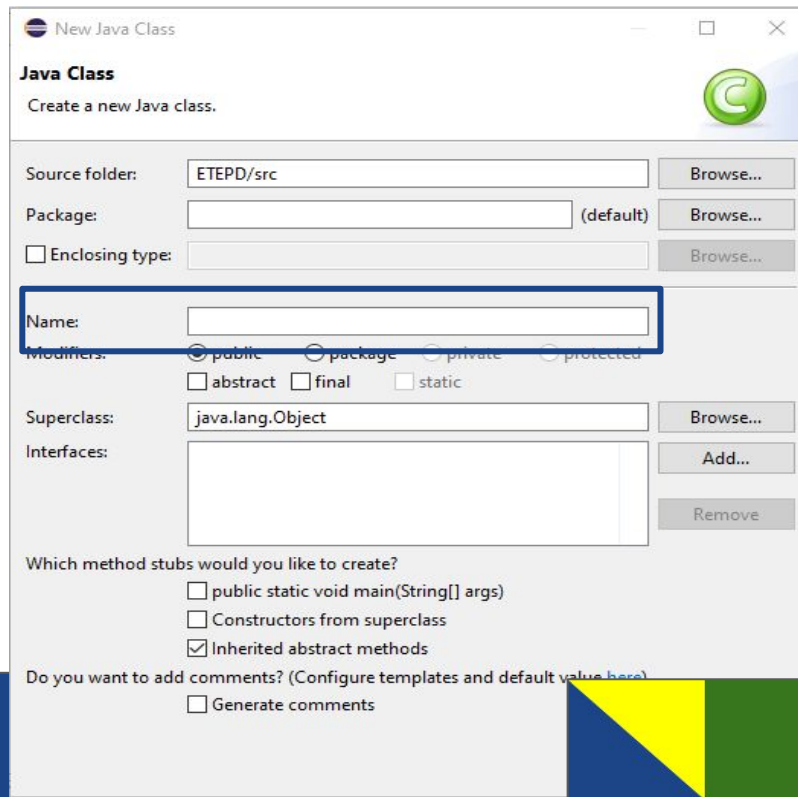
The screenshot shows the 'New Java Class' dialog box in the Eclipse IDE. The dialog is titled 'New Java Class' and has a subtitle 'Create a new Java class.' with a green 'C' icon. The fields are as follows:

- Source folder:** ETEPD/src (with a 'Browse...' button)
- Package:** (empty) (default) (with a 'Browse...' button)
- Enclosing type:** (empty) (with a 'Browse...' button)
- Name:** (empty) (highlighted with a blue box)
- Modifiers:** ☒ public, ☐ package, ☐ private, ☐ protected, ☐ abstract, ☐ final, ☐ static
- Superclass:** java.lang.Object (with a 'Browse...' button)
- Interfaces:** (empty list) (with 'Add...' and 'Remove' buttons)
- Which method stubs would you like to create?**
 - ☐ public static void main(String[] args)
 - ☐ Constructors from superclass
 - ☒ Inherited abstract methods
- Do you want to add comments? (Configure templates and default value [here](#))**
 - ☐ Generate comments

Primeira classe no Java

Para criar uma classe temos que escrever no padrão chamado CamelCase.

Ex:MinhaClasse



New Java Class

Java Class
Create a new Java class.

Source folder: ETEPD/src Browse...

Package: (default) Browse...

☐ Enclosing type: Browse...

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static

Superclass: java.lang.Object Browse...

Interfaces: Add... Remove

Which method stubs would you like to create?
☐ public static void main(String[] args)
☐ Constructors from superclass
☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))
☐ Generate comments

Primeira classe no Java

eclipse-workspace - ETEPD/src/OlaMundo.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- ETEPD
 - JRE System Library [JavaSE-11]
 - src
 - (default package)
 - OlaMundo.java

```
1  
2 public class OlaMundo {  
3  
4     //Aqui  
5     //vem  
6     //o código  
7     //em  
8     //Java  
9 }  
10
```



Criando um método main()

O método main() é onde o programa começará a ser executado.

```
public static void main(String[] args) {  
  
}
```



Criando um método main()

A classe que foi criada ficará assim

```
public class OlaMundo {  
  
    public static void main(String[] args) {  
  
    }  
  
}
```



Criando um método main()

Executando o primeiro projeto

undo.java - Eclipse IDE

te Search Project Run Window Help

*OlaMundo.java

```
1 public class OlaMundo {  
2  
3     public static void main(String[] args) {  
4  
5         System.out.println("Olá Mundo");  
6  
7     }  
8  
9 }
```

Problems Javadoc Declaration Console

<terminated> OlaMundo (1) [Java Application] C:\Program Files\Java\jdk-11.0.11\bin\javaw.exe (30 de set de 2021 23:09:45 - 23:09:47)

Olá Mundo

Sintaxe Básica

Primeiro a classe criada.
Segundo o método main dentro da classe.

```
public class SintaxeBasica {  
    public static void main(String[] args) {  
  
    }  
}
```



// comentário de uma linha.

/*

* comentário de bloco

*

* */

```
public class SintaxeBasica {  
    public static void main(String[] args) {  
  
    }  
}
```



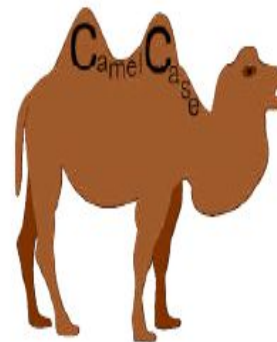
Tipos de Variáveis

| Descrição | Tipo | Tamanho | Valores | Valor padrão |
|-------------------------------------|---------|---------|--|--------------|
| tipos numéricos inteiros | byte | 8 bits | -128 a 127 | 0 |
| | short | 16 bits | -32768 a 32767 | 0 |
| | int | 32 bits | -2147483648 a 2147483647 | 0 |
| | long | 64 bits | -9223372036854770000 a 9223372036854770000 | 0L |
| tipos numéricos com ponto flutuante | float | 32 bits | -1,4024E-37 a 3,4028E+38 | 0.0f |
| | double | 64 bits | -4,94E-307 a 1,79E+308 | 0.0 |
| um caractere Unicode | char | 16 bits | '\u0000' a '\uFFFF' | '\u0000' |
| valor verdade | boolean | 1 bit | {false, true} | false |

Tipos de Variáveis

Nomes de variáveis

- Não pode começar com dígito: use uma letra ou _
- Não pode ter espaço em branco
- Não usar acentos ou til
- Sugestão: use o padrão "camel case"



Errado:

```
int 5minutos;  
int salário;  
int salário do funcionario;
```

Correto:

```
int _5minutos;  
int salario;  
int salarioDoFuncionario;
```



Tipos de Variáveis

Sintaxe:

`<tipo> <nome> = <valor inicial>;`
└──────────┘
(opcional)

Exemplos:

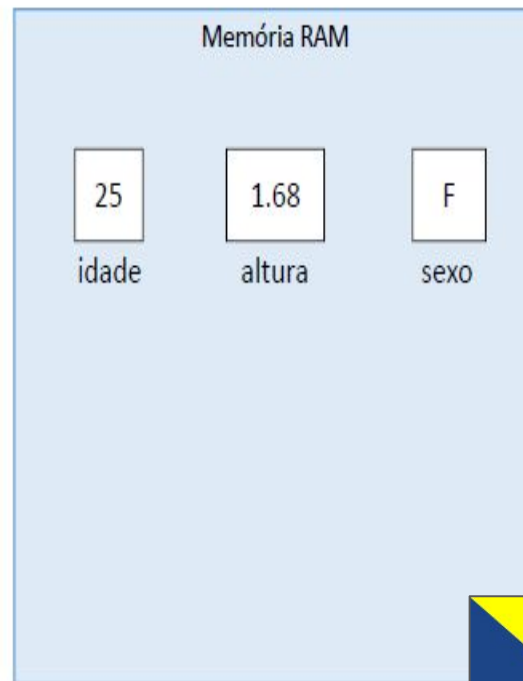
```
int idade = 25;
```

```
double altura = 1.68;
```

```
char sexo = 'F';
```

Uma variável possui:

- Nome (ou identificador)
- Tipo
- Valor
- Endereço



Tipos de Variáveis

Tipos de Variáveis - Exercício

- Qual o tipo de dados para os exemplos abaixo ?
 - Peso
 - Altura
 - Idade
 - Nome
 - CPF
 - Telefone
 - Lâmpada Acesa (verdadeiro ou falso)

E vamos colocar essas informações no compilador do JAVA.

Tipos de Variáveis

Tipos de Variáveis - Exercício
Resposta

```
double peso = 90;  
double altura = 1.73;  
int idade = 33;  
String nome = "Pedro";  
long cpf = 990988988001;  
int telefone = 999999999;  
boolean lampada = true;
```


Operadores Aritméticos

| Operador | Significado |
|----------|--------------------------|
| + | adição |
| - | subtração |
| * | multiplicação |
| / | divisão |
| % | resto da divisão ("mod") |



Operadores Aritméticos

```
1
2 public class Adicao {
3
4     public static void main(String[] args) {
5         int valor1 = 10;
6         int valor2 = 5;
7         int total;
8
9         total = valor1 + valor2;
10
11        System.out.println(total);
12
13    }
14 }
15
```

Problems Javadoc Declaration Console

<terminated> Adicao [Java Application] C:\Program Files\Java\jdk-11.0.11\bin\javaw.exe (3 de out de 2021 12:54:43 – 12:54:43)



Operadores Aritméticos

Operadores Aritméticos MOD

Resto da Divisão

```
public class Mod {  
  
    public static void main(String[] args) {  
        int valor1 = 5;  
        int valor2 = 2;  
        int mod;  
  
        mod = valor1 % valor2;  
  
        System.out.println(mod);  
    }  
}
```



Operadores Aritméticos Exercícios

Faça um programa que receba dois valores inteiros, e depois mostrar na tela a soma desses números.

Valores de Entrada:

Valor1 = 10

Valor2 = 30



Operadores Aritméticos Exercícios

Faça um programa que receba dois valores inteiros, e depois mostrar na tela a soma desses números.

Valores de Entrada:

Valor1 = 10

Valor2 = 30



Operadores Aritméticos Exercícios

Faça um programa que receba dois valores inteiros, e depois mostrar na tela a soma desses números.

Valores de Entrada:

Valor1 = -30

Valor2 = 10



Operadores Aritméticos Exercícios

Faça um programa que receba dois valores inteiros, e depois mostrar na tela o resto da divisão.

Valores de Entrada:

Valor1 = 7

Valor2 = 2



Exercícios

Construa um algoritmo que tenha 4 variáveis como notas e mostre a média.



Exercícios

Faça um algoritmo que receba dois números com variáveis e faça a multiplicação dos números lidos e mostre o resultado.



Exercícios

Faça um algoritmo que receba dois números como variáveis e faça a divisão dos números lidos e mostre o resultado.



Exercícios

Faça um algoritmo que receba dois números e ao final mostre a soma, subtração, multiplicação e a divisão dos números lidos.



Exercícios

Faça um algoritmo que calcule o Índice de Massa Corpórea (IMC), onde o usuário informe seu peso e sua altura. Fórmula para calcular o IMC = $\text{peso}/(\text{altura}^* \text{altura})$.



Operadores Comparativos

| Operador | Significado |
|----------|----------------|
| > | maior |
| < | menor |
| >= | maior ou igual |
| <= | menor ou igual |
| == | igual |
| != | diferente |



Operadores Comparativos

Exemplo

(suponha x igual a 5)

$X > 0$

Resultado: V

$X == 3$

Resultado: F

$10 \leq 30$

Resultado: V

$X \neq 2$

Resultado: V



Operadores Lógicos

| Operador | Significado |
|----------|-------------|
| && | E |
| | OU |
| ! | NÃO |



Operadores Lógicos

Operador Lógico E

Você pode obter uma habilitação de motorista se:

- For aprovado no exame psicotécnico,
E
- For aprovado no exame de legislação,
E
- For aprovado no exame de direção

**Todas condições
devem ser
verdadeiras!**



Operadores Lógicos

(suponha x igual a 5)

Operador Lógico E

$X \leq 20 \ \&\& \ X == 10$ Resultado: F

$\underbrace{\hspace{1.5cm}}_V \quad \underbrace{\hspace{1.5cm}}_F$

$X > 0 \ \&\& \ X != 3$ Resultado: V

$\underbrace{\hspace{1.5cm}}_V \quad \underbrace{\hspace{1.5cm}}_V$

$X \leq 20 \ \&\& \ X == 10 \ \&\& \ X != 3$ Resultado: F

$\underbrace{\hspace{1.5cm}}_V \quad \underbrace{\hspace{1.5cm}}_F \quad \underbrace{\hspace{1.5cm}}_V$



Operadores Lógicos

Tabela verdade “E”

| A | B | A && B |
|---|---|--------|
| F | F | F |
| F | V | F |
| V | F | F |
| V | V | V |



Operadores Lógicos

Operador Lógico OU

Você pode estacionar na vaga especial se:

- For idoso(a),

OU

- For uma pessoa com deficiência,

OU

- For uma gestante

**Pelo menos uma
condição deve
ser verdadeira!**



Operadores Lógicos

(suponha x igual a 5)

Operador Lógico OU

$X == 10 \ || \ X \leq 20$ Resultado: V

$\underbrace{\hspace{1.5cm}}_F \quad \underbrace{\hspace{1.5cm}}_V$

$X > 0 \ || \ X \neq 3$ Resultado: V

$\underbrace{\hspace{1.5cm}}_V \quad \underbrace{\hspace{1.5cm}}_V$

$X \leq 0 \ || \ X \neq 3 \ || \ X \neq 5$ Resultado: V

$\underbrace{\hspace{1.5cm}}_F \quad \underbrace{\hspace{1.5cm}}_V \quad \underbrace{\hspace{1.5cm}}_F$



Operadores Lógicos

Tabela Verdade “OU”

| A | B | A B |
|---|---|--------|
| F | F | F |
| F | V | V |
| V | F | V |
| V | V | V |



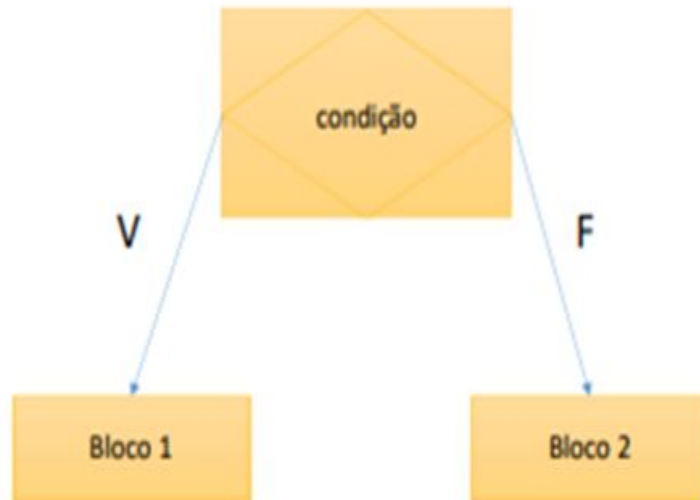
Estrutura Condicional



Estrutura Condicional Simples

Estrutura condicional:

É uma **estrutura de controle** que permite definir que um certo **bloco de comandos** somente será executado dependendo de uma **condição**



Estrutura Condicional Simples

Simples:

REGRA:

V: executa o bloco de comandos.

F: pula o bloco de comandas.

```
if ( <condição> ) {  
    <comando 1>  
    <comando 2>  
}
```



Estrutura Condicional Simples

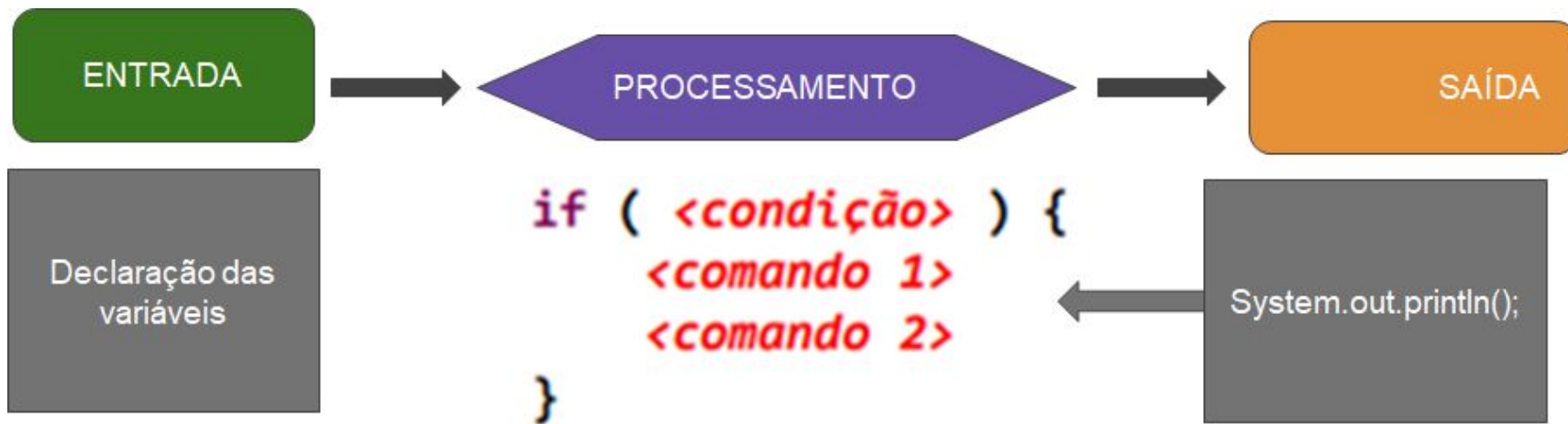
```
public class CondicionalSimples {  
    public static void main(String[] args) {  
        if( condicao ) {  
        }  
    }  
}
```



Estrutura Condicional Simples

```
public class Exercicio {  
  
    public static void main(String[] args) {  
        // TODO Auto-generated method tub  
  
        int x = 5;  
  
        if(x > 3) {  
            //Instruções a serem executadas  
        }  
    }  
}
```

Estrutura Condicional Simples



Estrutura Condicional Simples

Faça um programa que tenha uma variável inteira e informe se esse valor é positivo



Estrutura Condicional Simples

```
public class Exer2 {  
  
    public static void main(String[] args) {  
        int valor1 = 10;  
  
        if(valor1 > 0) {  
            System.out.println("É positivo");  
        }  
    }  
}
```



Estrutura Condicional Simples

Calcule a média entre dois valores e escreva o resultado da média. Se a média for maior ou igual 6 enviar uma mensagem de **“aprovado”**.



Estrutura Condicional Simples

Faça um algoritmo que receba os valores **A**, **B**, **C** e imprima na tela se a soma de **A + B** é menor que **C**.



Estrutura Condicional Simples

Faça um algoritmo que leia dois valores inteiros A e B se os valores forem iguais deverá se somar os dois, caso contrário multiplique A por B.



Estrutura Condicional Simples

Encontrar o dobro de um número caso ele seja positivo e o seu triplo caso seja negativo, imprimindo o resultado.

