

BCM0505

Processamento da Informação



Universidade Federal do ABC

IDEs

Prof. Rogério Neves

rogerio.neves@ufabc.edu.br

Portugol Studio

- ❑ IDE simples para teste de pseudocódigo
- ❑ Análogo a JAVA

```
3 funcao inicio(cadeia parametros[])
4 {
5     real altura, pesoideal
6     caracter sexo
7
8     escreva("Digite o sexo [M ou F] : ")
9     leia (sexo)
10    escreva("Digite a altura :")
11    leia (altura)
12
13    se (sexo == 'M') {
14        pesoideal = (72.7 * altura) - 58
15    }
16    senao {
17        pesoideal = (62.1 * altura) - 44.7
18    }
```



TIPOS DE DADOS

PRINCIPAIS TIPOS DE DADOS EXISTENTES

Descrição	Em PORTUGOL Studio	Em Java
Caractere	caracter	char
Literal	cadeia	String (classe)
Inteiro	inteiro	int
Inteiro longo	Não se aplica	long
Real (ponto flutuante)	real	float
Real “longo”	Não se aplica	double
Booleano	Não se aplica	boolean

Java, Tipos primitivos

Type	Values	Default	Size	Range
byte	signed integers	0	8 bits	-128 to 127
short	signed integers	0	16 bits	-32768 to 32767
int	signed integers	0	32 bits	-2147483648 to 2147483647
long	signed integers	0	64 bits	-9223372036854775808 to 9223372036854775807
float	IEEE 754 floating point	0.0	32 bits	+/-1.4E-45 to +/-3.4028235E+38, +/-infinity, +/-0, NaN
double	IEEE 754 floating point	0.0	64 bits	+/-4.9E-324 to +/-1.7976931348623157E+308, +/-infinity, +/-0, NaN
char	Unicode character	\u0000	16 bits	\u0000 to \uFFFF
boolean	true, false	false	1 bit used in 32 bit integer	NA

Regras para nomear variáveis em JAVA



- Pode conter um ou mais caracteres;
- Deve começar sempre por uma letra;
- Pode ser seguidos por letras e números;
 - casa21
- Não pode ter espaços entre as letras;
 - casa 21 (errado!!) – usar_underline_para_separar
- Não pode conter caracteres especiais;
 - (?, ç, @, #, !, etc....)
- Não pode ser uma palavra reservada;
 - float, int, for , ...
- Ser sucinto e utilizar nomes coerentes

Precedência de Operadores JAVA

Operador	Observação
()	Parênteses para agrupar expressões
x++, y--	Incremento e decremento pós-fixado
--x, --y, !	Incremento e decremento pós-fixado, negação
*, /, %	Operadores aritméticos de multiplicação, divisão e módulo • observação: “%” não respeita a propriedade associativa com multiplicação/divisão $2 * 3 \% 5 \neq 2 * (3 \% 5)$
+, -	Operadores aritméticos de adição e subtração
<, >, <=, >=	Operadores relacionais
==, !=	Operadores de igualdade e diferença
&&	Operador lógico e
	Operador lógico ou
=	Operador de atribuição

Netbeans

- Explorar o IDE NetBeans
- Iniciar a prática da programação em JAVA

Sobre o NetBeans

- O IDE NetBeans versão 5.0 ou superior, que pode ser obtido a partir de <http://java.sun.com/javase/downloads/> ou <http://www.netbeans.org/downloads/index.html>
- Quando se instala o NetBeans, o JDK é instalado automaticamente
- O Java SE Development Kit versão 5 (JDK 5), ou superior, pode ser obtido a partir de <http://java.sun.com/javase/downloads/> (escolher ultima versão),
- **Acessar roteiro das atividades da aula no TIDIA**

Project Explorer (Projeto) showing the 'helloWorld' project structure:

- helloWorld
 - Pacotes de códigos-fonte
 - helloworld
 - Main.java
 - Pacotes de testes
 - Bibliotecas
 - Bibliotecas de testes

Main.java editor showing the following code:

```
package helloworld;

public class Main {

    public static void main(String[] args) {

    }

}
```

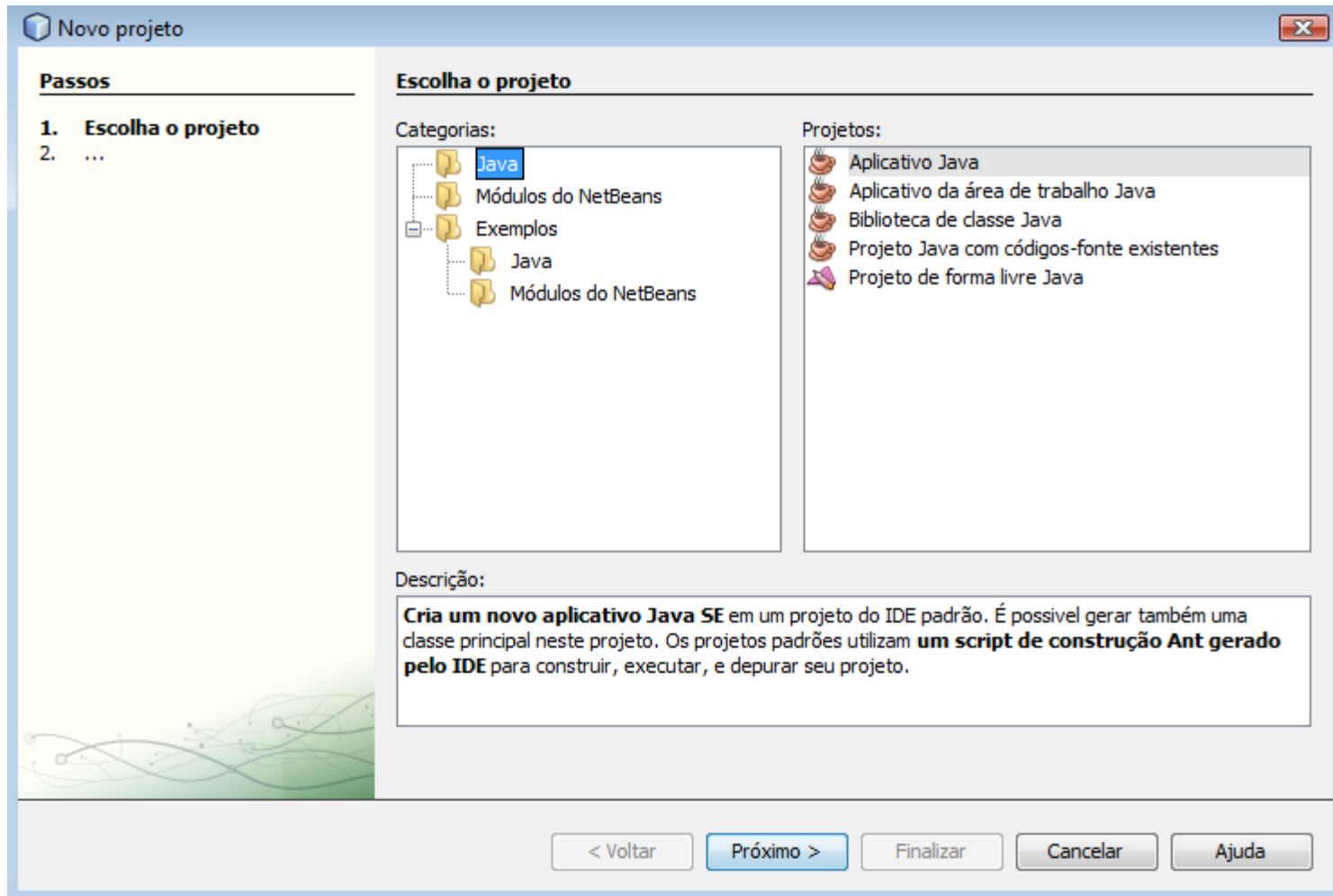
main - Navegador (Navigator) showing the class hierarchy:

- Main
 - main(String[] args)

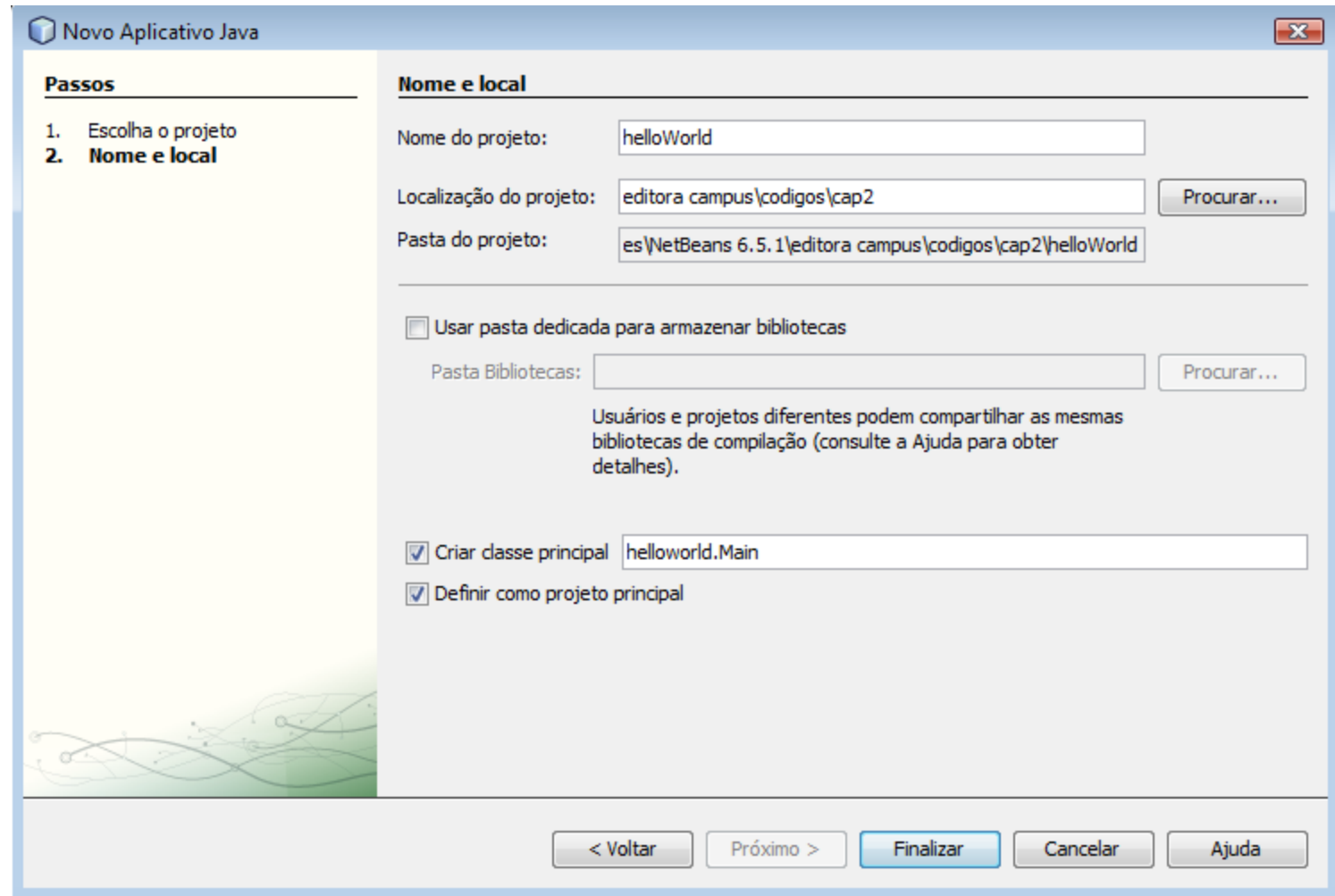
Tasks (Tarefas) window showing the output of the build process:

```
Saída - helloWorld (jar)
Created dir: C:\esteban\profissional\livros\editora campus\codigos\cap2\helloWorld\dist
Building jar: C:\esteban\profissional\livros\editora campus\codigos\cap2\helloWorld\dist\helloWorld.jar
Not copying the libraries.
To run this application from the command line without Ant, try:
java -jar "C:\esteban\profissional\livros\editora campus\codigos\cap2\helloWorld\dist\helloWorld.jar"
CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)
```

Sobre o NetBeans



Sobre o NetBeans



The image shows the 'Novo Aplicativo Java' (New Java Application) dialog box in NetBeans, specifically the 'Nome e local' (Name and Location) step. The dialog is titled 'Novo Aplicativo Java' and has a close button in the top right corner. On the left, a 'Passos' (Steps) pane shows two steps: '1. Escolha o projeto' and '2. Nome e local', with the second step being the active one. The main area is divided into sections for project name, location, and library folder. The 'Nome do projeto' (Project Name) field contains 'helloWorld'. The 'Localização do projeto' (Project Location) field contains 'editora campus\codigos\cap2' and has a 'Procurar...' (Browse...) button. The 'Pasta do projeto' (Project Folder) field contains 'es\NetBeans 6.5.1\editora campus\codigos\cap2\helloWorld'. Below these fields, there is a checkbox 'Usar pasta dedicada para armazenar bibliotecas' (Use dedicated folder for libraries) which is unchecked. The 'Pasta Bibliotecas' (Library Folder) field is empty and has a 'Procurar...' button. A note below this field states: 'Usuários e projetos diferentes podem compartilhar as mesmas bibliotecas de compilação (consulte a Ajuda para obter detalhes)'. At the bottom, there are two checked checkboxes: 'Criar classe principal' (Create main class) with the field 'helloworld.Main' and 'Definir como projeto principal' (Define as main project). At the very bottom of the dialog, there are five buttons: '< Voltar' (Back), 'Próximo >' (Next), 'Finalizar' (Finish), 'Cancelar' (Cancel), and 'Ajuda' (Help).

Novo Aplicativo Java

Passos

1. Escolha o projeto
2. **Nome e local**

Nome e local

Nome do projeto: helloWorld

Localização do projeto: editora campus\codigos\cap2

Pasta do projeto: es\NetBeans 6.5.1\editora campus\codigos\cap2\helloWorld

Usar pasta dedicada para armazenar bibliotecas

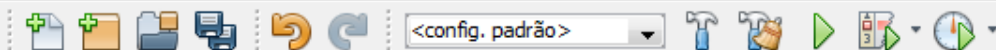
Pasta Bibliotecas:

Usuários e projetos diferentes podem compartilhar as mesmas bibliotecas de compilação (consulte a Ajuda para obter detalhes).

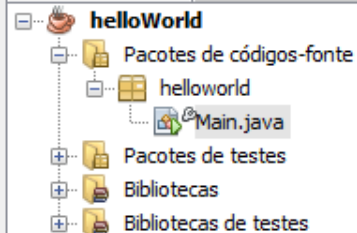
Criar classe principal helloworld.Main

Definir como projeto principal

< Voltar Próximo > Finalizar Cancelar Ajuda

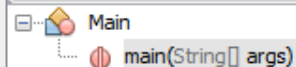


Projetos Arquivos Serviços



main - Navegador

Visualização dos membros



Página Iniciar Main.java *



```
package helloworld;

public class Main {

    public static void main(String[] args) {

    }

}
```

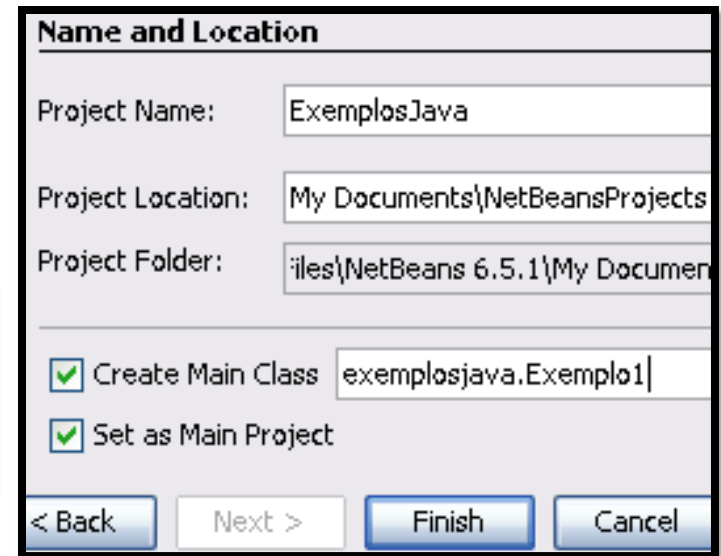
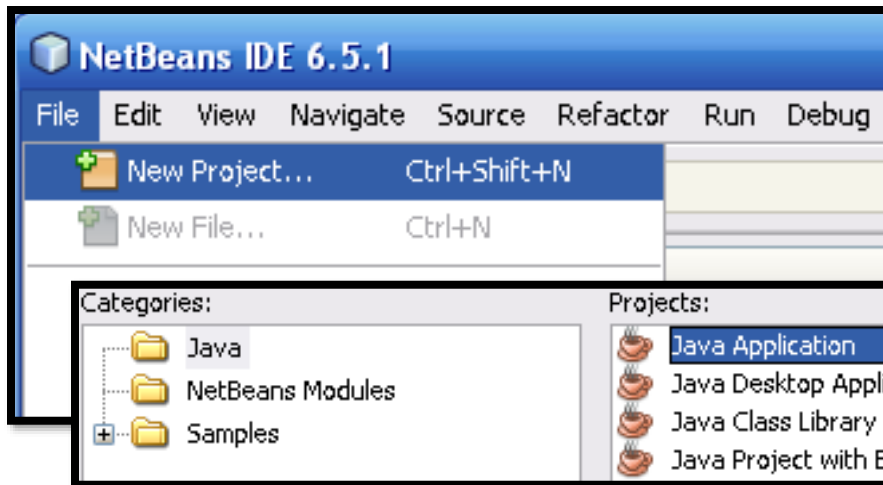
7:1 INS

Tarefas

Saída - helloWorld (jar)

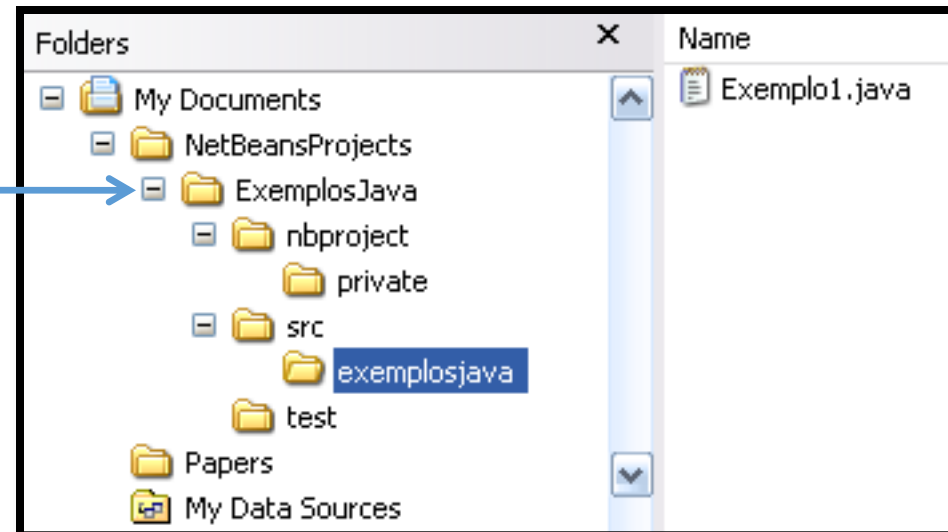
```
Created dir: C:\esteban\profissional\livros\editora campus\codigos\cap2\helloWorld\dist
Building jar: C:\esteban\profissional\livros\editora campus\codigos\cap2\helloWorld\dist\helloWorld.jar
Not copying the libraries.
To run this application from the command line without Ant, try:
java -jar "C:\esteban\profissional\livros\editora campus\codigos\cap2\helloWorld\dist\helloWorld.jar"
CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)
```

Projetos com NetBeans IDE



- Conjunto de classes de um projeto

Códigos fonte
Bytecodes gerados
Arquivos relacionados



Projects | **Files** | **Services**

- ExemplosJava
 - Source Packages
 - exemplosjava
 - Exemplo1.java
 - Test Packages
 - Libraries
 - Test Libraries

```
Start Page x Exemplo1.java x
```

```
/*  
 * To change this template, choose Tools | Templates  
 * and open the template in the editor.  
 */  
  
package exemplosjava;  
  
/**  
 *  
 * @author Spider  
 */  
public class Exemplo1 {  
  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        // TODO code application logic here  
    }  
}
```

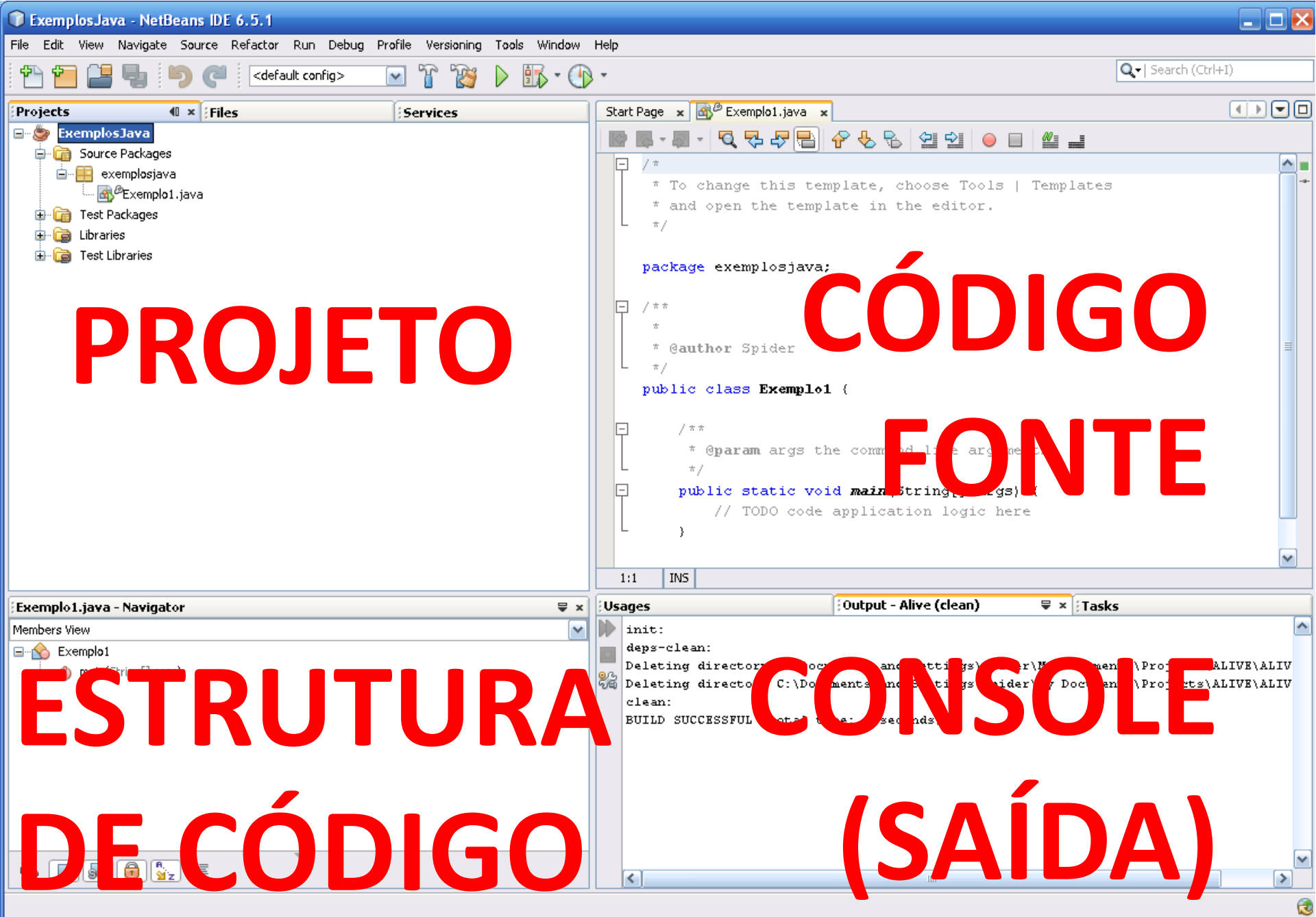
Exemplo1.java - Navigator

Members View

- Exemplo1
 - main(String[] args)

Usages | **Output - Alive (clean)** | **Tasks**

```
init:  
deps-clean:  
Deleting directory C:\Documents and Settings\Spider\My Documents\Projects\ALIVE\ALIV  
Deleting directory C:\Documents and Settings\Spider\My Documents\Projects\ALIVE\ALIV  
clean:  
BUILD SUCCESSFUL (total time: 0 seconds)
```



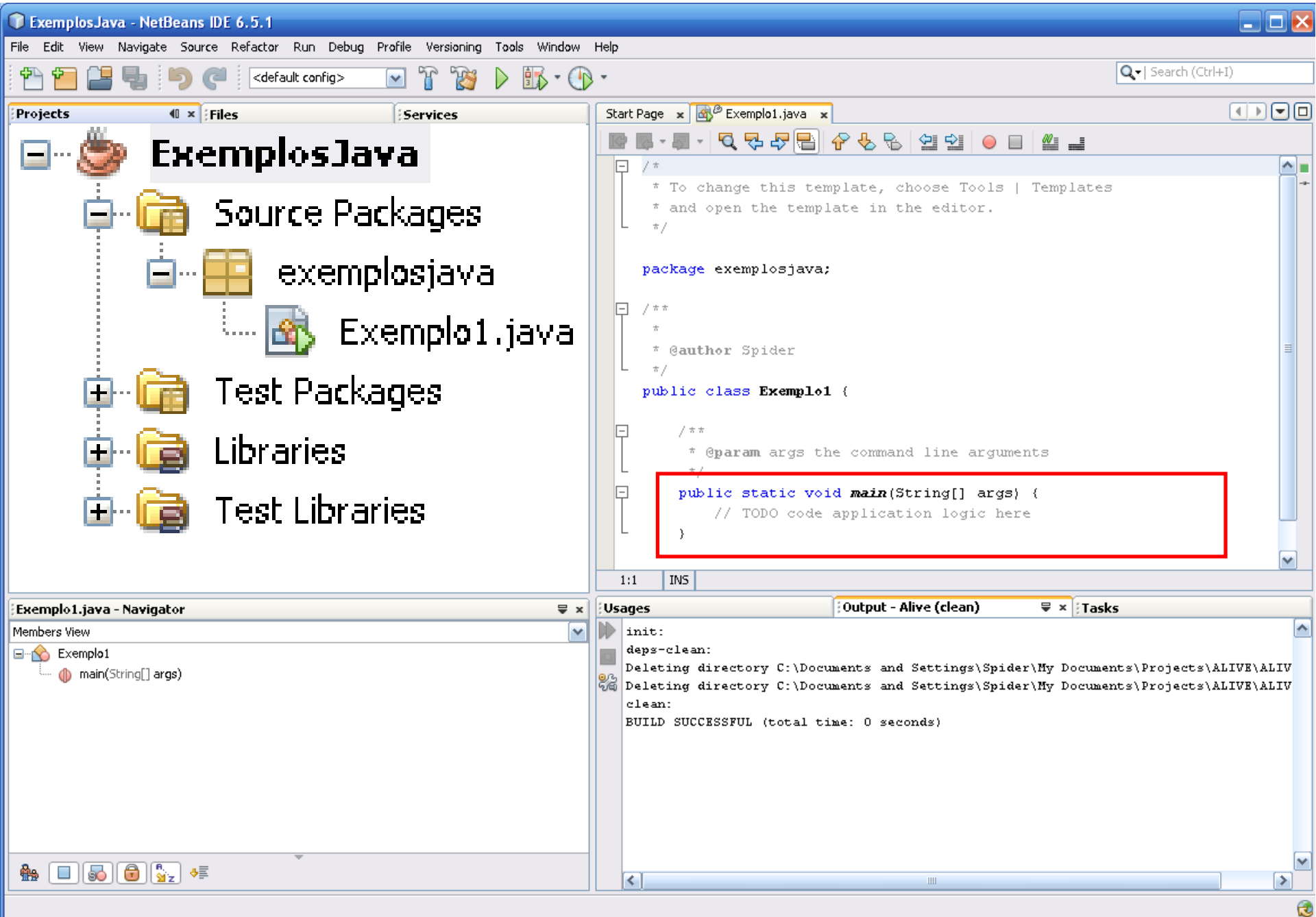
PROJETO

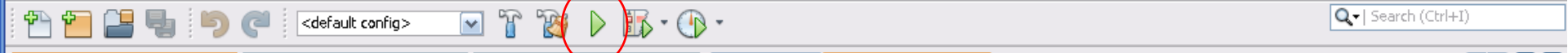
CÓDIGO

FONTE

ESTRUTURA
DE CÓDIGO

CONSOLE
(SAÍDA)





Projects | **Files** | **Services**

- ExemplosJava
 - Source Packages
 - exemplosjava
 - Exemplo1.java**
 - Test Packages
 - Libraries
 - Test Libraries

The file **Exemplo1.java** is circled in red.

```
Start Page x Exemplo1.java x
```

```
/*  
 * To change this template, choose Tools | Templates  
 * and open the template in the editor.  
 */  
  
package exemplosjava;  
  
/**  
 *  
 * @author Spider  
 */  
public class Exemplo1 {  
  
    /**  
     * @param args the command line arguments  
     */  
    public static void main(String[] args) {  
        // TODO code application logic here  
    }  
}
```

Exemplo1.java - Navigator

Members View

- Exemplo1
 - main(String[] args)

Usages | **Output - Alive (clean)** | **Tasks**

```
init:  
deps-clean:  
Deleting directory C:\Documents and Settings\Spider\My Documents\Projects\ALIVE\ALIV  
Deleting directory C:\Documents and Settings\Spider\My Documents\Projects\ALIVE\ALIV  
clean:  
BUILD SUCCESSFUL (total time: 0 seconds)
```

Projects | Files | Services

The Project Explorer shows a tree view of the project 'ExemplosJava'. It contains a 'Source Packages' folder, which is highlighted with a red box and contains a sub-package 'exemplosjava' (also highlighted with a red box). Below 'Source Packages' are 'Test Packages', 'Libraries', and 'Test Libraries' folders. A file 'Exemplo1.java' is shown as a child of the 'exemplosjava' package.

Start Page x Exemplo1.java x

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package exemplosjava;

/**
 *
 * @author Spider
 */
public class Exemplo1 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
    }
}
```

Exemplo1.java - Navigator

Members View

The Navigator shows the 'main(String[] args)' method of the 'Exemplo1' class.

Usages | Output - Alive (clean) | Tasks

```
init:
deps-clean:
Deleting directory C:\Documents and Settings\Spider\My Documents\Projects\ALIVE\ALIV
Deleting directory C:\Documents and Settings\Spider\My Documents\Projects\ALIVE\ALIV
clean:
BUILD SUCCESSFUL (total time: 0 seconds)
```

ExemplosJava - NetBeans IDE 6.5.1

File Edit View Navigate Source Refactor Run Debug Profile Versioning Tools Window Help

<default config> Search (Ctrl+I)

Projects | Files | Services

ExemplosJava

- Source Packages
 - exemplosjava
 - Exemplo1.java
- Test Packages
- Libraries
- Test Libraries

Start Page x Exemplo1.java x

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

package exemplosjava;

/**
 *
 * @author Spider
 */
public class Exemplo1 {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
    }
}
```

1:1 INS

Exemplo1.java - Navigator

Members View

- Exemplo1
 - main(String[] args)

Usages | Output - Alive (clean) | Tasks

```
init:
deps-clean:
Deleting directory C:\Documents and Settings\Spider\My Documents\Projects\ALIVE\ALIV
Deleting directory C:\Documents and Settings\Spider\My Documents\Projects\ALIVE\ALIV
clean:
BUILD SUCCESSFUL (total time: 0 seconds)
```

Código fonte

```

/*
 * Exemplos em Java
 * por Rogerio Neves
 */

package exemplosjava;

public class Exemplo1 {

    /**
     * String[] args são os parâmetros passados por linha de comando
     */
    public static void main(String[] args) {
        // Método principal, lo a ser executado
        String nome = "Pedro";
        int idade = 22;
        System.out.println(nome + " têm " + idade + " anos.\n");
    }
}

```

7:1

INS

ages

Output - ExemplosJava (run)



Tasks

run:

Pedro têm 22 anos.

BUILD SUCCESSFUL (total time: 2 seconds)

Comentários

```
⊟ /*  
  * Exemplos em Java  
  * por Rogerio Neves  
  */
```

```
package exemplosjava;
```

```
public class Exemplo1 {
```

```
⊟ /**  
  * String[] args são os parâmetros passados por linha de comando  
  */
```

```
⊟ public static void main(String[] args) {  
    // Método principal, lo a ser executado  
    String nome = "Pedro";  
    int idade = 22;  
    System.out.println(nome + " têm " + idade + " anos.\n");  
  }  
}
```

```
7:1 | INS
```

```
ages
```

```
Output - ExemplosJava (run)
```

```
Tasks
```

```
run:
```

```
Pedro têm 22 anos.
```

```
BUILD SUCCESSFUL (total time: 2 seconds)
```

Blocos de código

```

/*
 * Exemplos em Java
 * por Rogerio Neves
 */

package exemplosjava;

public class Exemplo1 {

    /**
     * String[] args são os parâmetros passados por linha de comando
     */
    public static void main(String[] args) {
        // Método principal, lo a ser executado
        String nome = "Pedro";
        int idade = 22;
        System.out.println(nome + " têm " + idade + " anos.\n");
    }
}

```

7:1

INS

ages

Output - ExemplosJava (run)



Tasks

run:

Pedro têm 22 anos.

BUILD SUCCESSFUL (total time: 2 seconds)

Blocos de código

```

/*
 * Exemplos em Java
 * por Rogerio Neves
 */

package exemplosjava;

public class Exemplo1 {

    /**
     * String[] args são os parâmetros passados por linha de comando
     */
    public static void main(String[] args) {
        // Método principal, lo a ser executado
        String nome = "Pedro";
        int idade = 22;
        System.out.println(nome + " têm " + idade + " anos.\n");
    }
}

```

7:1

INS

ages

Output - ExemplosJava (run)

☰

×

Tasks

run:

Pedro têm 22 anos.

BUILD SUCCESSFUL (total time: 2 seconds)

Blocos de código

```
/*
 * Exemplos em Java
 * por Rogerio Neves
 */

package exemplosjava;

public class Exemplo1 {

    /**
     * String[] args são os parâmetros passados por linha de comando
     */
    public static void main(String[] args) {
        // Método principal, lo a ser executado
        String nome = "Pedro";
        int idade = 22;
        System.out.println(nome + " têm " + idade + " anos.\n");
    }
}
```

7:1

INS

ages

Output - ExemplosJava (run)

☰

×

Tasks

run:

Pedro têm 22 anos.

BUILD SUCCESSFUL (total time: 2 seconds)

Saída

```

/*
 * Exemplos em Java
 * por Rogerio Neves
 */

package exemplosjava;

public class Exemplo1 {

    /**
     * String[] args são os parâmetros passados por linha de comando
     */
    public static void main(String[] args) {
        // Método principal, lo a ser executado
        String nome = "Pedro";
        int idade = 22;
        System.out.println(nome + " têm " + idade + " anos.\n");
    }
}

```

7:1

INS

ages

Output - ExemplosJava (run)



Tasks

run:

Pedro têm 22 anos.

BUILD SUCCESSFUL (total time: 2 seconds)

Declaração de String

```

/*
 * Exemplos em Java
 * por Rogerio Neves
 */

package exemplosjava;

public class Exemplo1 {

    /**
     * String[] args são os parâmetros passados por linha de comando
     */
    public static void main(String[] args) {
        // Método principal, lo a ser executado
        String nome = "Pedro";
        int idade = 22;
        System.out.println(nome + "têm" + idade + "anos.\n");
    }
}

```

7:1

INS

ages

Output - ExemplosJava (run)



Tasks

run:

Pedro têm 22 anos.

BUILD SUCCESSFUL (total time: 2 seconds)

Fim de linha

```

/*
 * Exemplos em Java
 * por Rogerio Neves
 */

package exemplosjava;

public class Exemplo1 {

    /**
     * String[] args são os parâmetros passados por linha de comando
     */
    public static void main(String[] args) {
        // Método principal, lo a ser executado
        String nome = "Pedro";
        int idade = 22;
        System.out.println(nome + " têm " + idade + " anos.\n");
    }
}

```

7:1

INS

ages

Output - ExemplosJava (run)

☰

x

Tasks

run:

Pedro têm 22 anos.

BUILD SUCCESSFUL (total time: 2 seconds)

Sintaxe JAVA

Exemplos

Meu primeiro programa

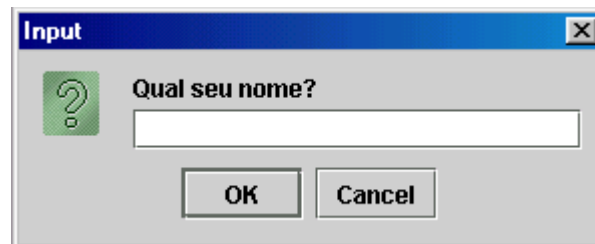
```
public static void main(String[] args) {  
    System.out.println("Alô mundo!");  
}
```

Meu segundo programa

```
public static void main(String[] args) {  
    System.out.println("Meu nome é ...");  
}
```

Entrada com Interface Gráfica

```
class EntradaB {  
    public static void main ( String args [ ] ) {  
  
        String nome =  
            javax.swing.JOptionPane.showInputDialog  
                (null, "Qual seu nome?");  
  
    }  
}
```



Média de 2 números

```
import java.util.*;
public class Nota {
public static void main(String args[])
{
    Scanner entrada = new Scanner(System.in);
    float num1, num2, media;
    System.out.println("Qual a nota 1?");
    num1 = entrada.nextFloat();
    System.out.println("Qual a nota 2");
    num2 = entrada.nextFloat();
    media= (num1+num2)/2f;
    System.out.println("a nota final é: " + media);
}
}
```

Métodos para leitura e escrita

```
Import java.util.*;
public static void main(String args[]) {

    // Seu código aqui

}
public static void escreva(Object texto) {
    System.out.print(texto.toString());
}
public static float leia(String texto) {
    Scanner s = new Scanner(System.in);
    System.out.print(texto);
    return s.nextFloat();
}
```

Métodos para leitura e escrita

```
public class Nota {  
    public static void main(String args[]) {  
        float num1, num2, media;  
        num1 = leia("Qual a nota 1?");  
        num2 = leia("Qual a nota 2?");  
        media= (num1+num2)/2;  
        escreva("a nota final é: " + media);  
    }  
}
```