

Plataforma de desenvolvimento Eclipse

Prof. Felipe Scheidt

2015

O que é?

- Eclipse é uma IDE
 - *integrated development environment*
 - ou Ambiente de desenvolvimento integrado
- É um software que possui um conjunto de funcionalidades embutidas, cuja finalidade é prover um modo mais fácil e produtivo de desenvolver programas.

Plugins

- Possui diversos plugins que adicionam funcionalidades ao Eclipse, dentre elas:
 - Php eclipse
 - JavaScript
 - Java
 - C/C++
 - Web
 - Modelagem



The screenshot displays a list of Eclipse IDE plugins available in the marketplace. Each entry includes an icon, the plugin name, its size, and the number of times it has been downloaded. A 'Details' link is provided for each plugin.

Plugin Name	Size	Downloaded Times
Eclipse IDE for Java EE Developers	212 MB	397,535
Eclipse Classic 3.7.2	174 MB	297,679
Eclipse IDE for Java Developers	128 MB	116,411
Eclipse IDE for C/C++ Developers (includes Incubating components)	108 MB	61,067
Eclipse IDE for Java and Report Developers	243 MB	19,878
Eclipse IDE for JavaScript Web Developers	110 MB	18,662
Eclipse Modeling Tools	272 MB	9,712

Ferramentas

- Possui diversas funcionalidades, por exemplo:
 - Um editor de texto;
 - Um compilador;
 - Um editor gráfico;
 - Um *debugger*
 - Um gerenciador de plugins

Instalação

- Acessar o site:
- www.eclipse.org
- “downloads”, selecionar a versão:
- Eclipse IDE for Java EE Developers, ~254 MB

Versão:



Instalação

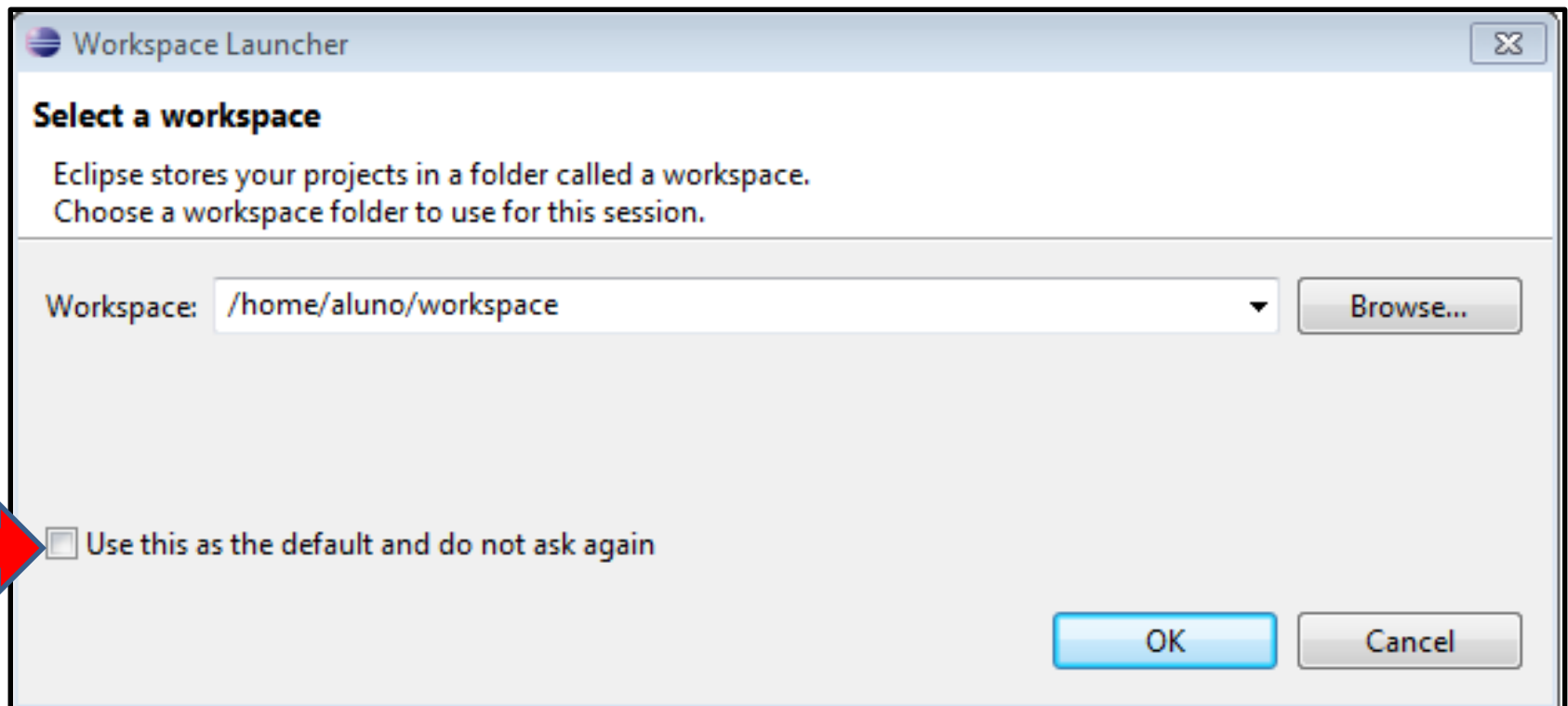
- Não é necessário instalar o eclipse, basta descompactar em uma pasta o arquivo baixado do site.
- Dentro da pasta eclipse, deve ser executado o seguinte arquivo para iniciar o eclipse:
 - eclipse.exe (windows)
 - eclipse (linux)

Workspace

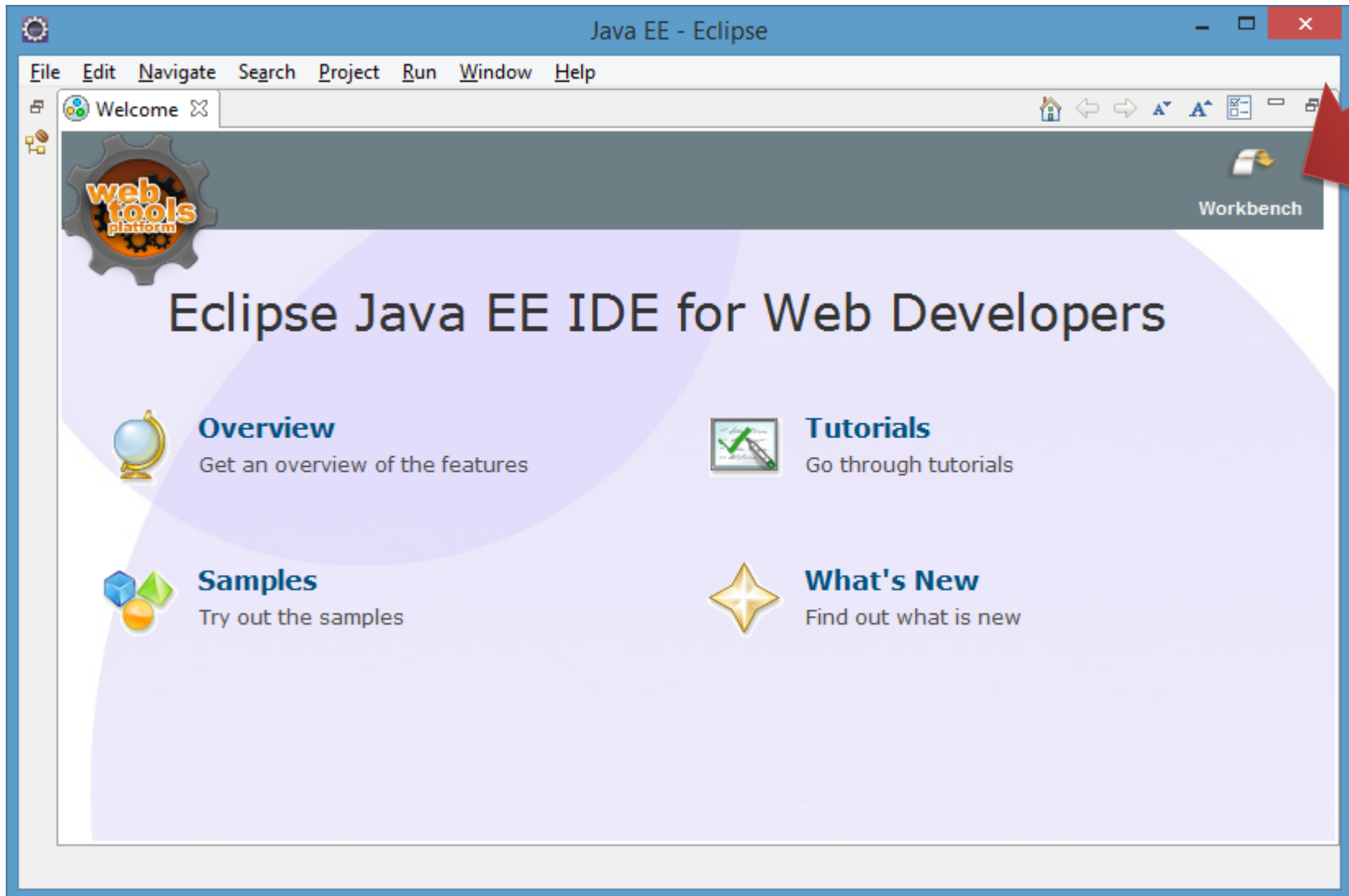
- É um diretório onde estão armazenados os arquivos utilizados em um projeto, isso inclui:
 - código java
 - imagens
 - html
 - javascript
 - css
 - outros

Criando o workspace

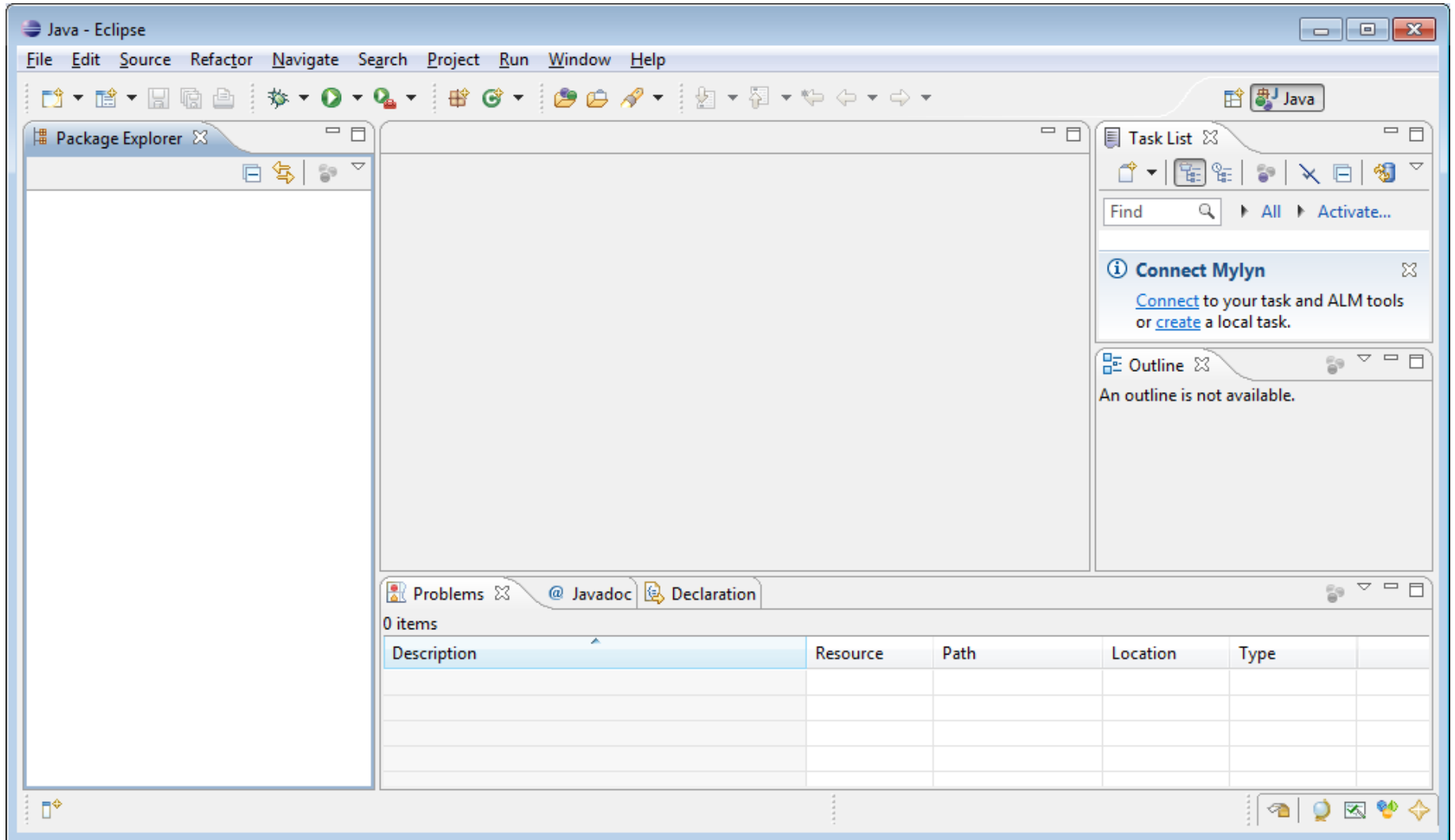
- Configure para o diretório:
`/home/aluno/workspace`




Tela inicial



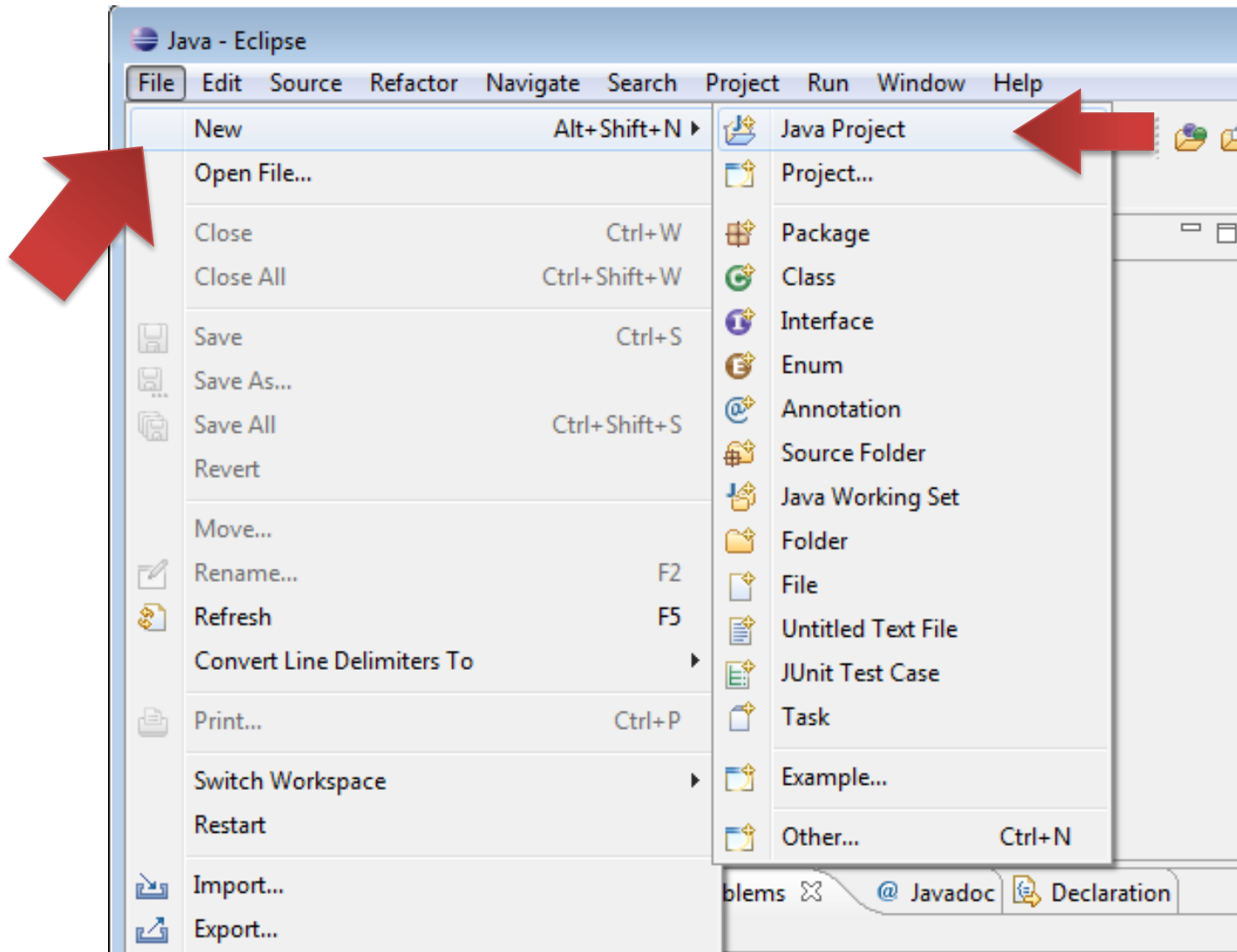
Visão do eclipse



Criando um novo projeto

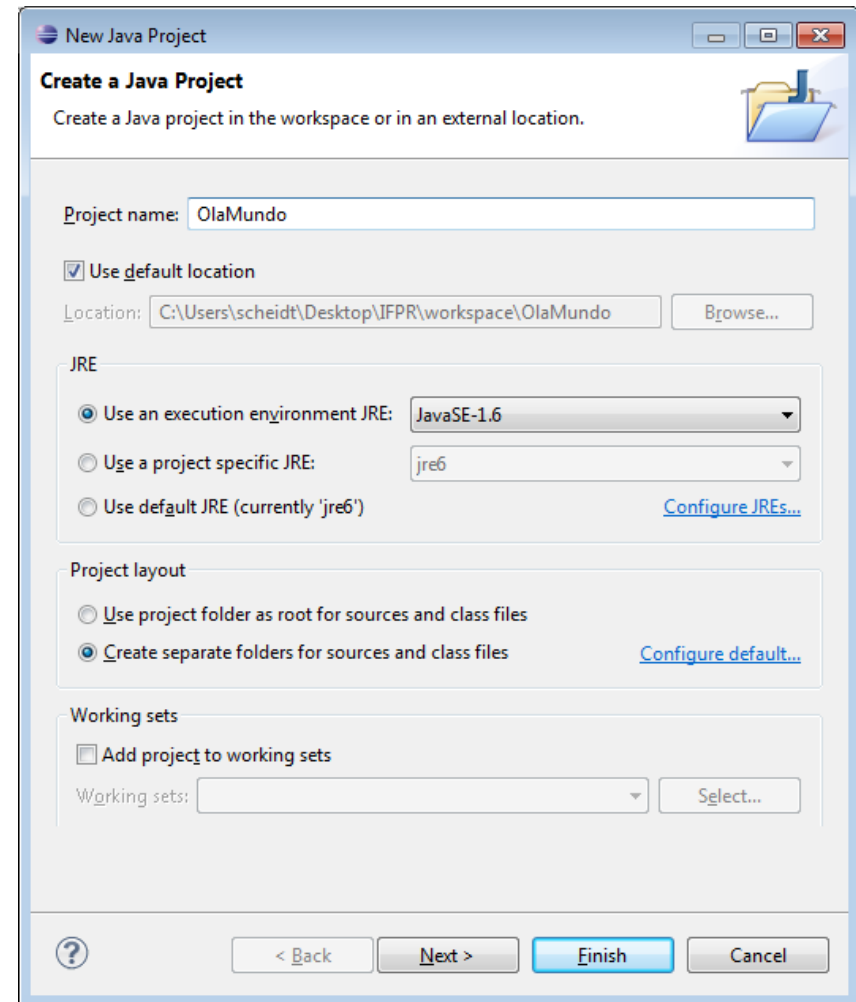
- Primeiramente clique em
 - File -> New -> Java Project 
- Um *wizard* de criação de projetos aparecerá

Criando um novo projeto



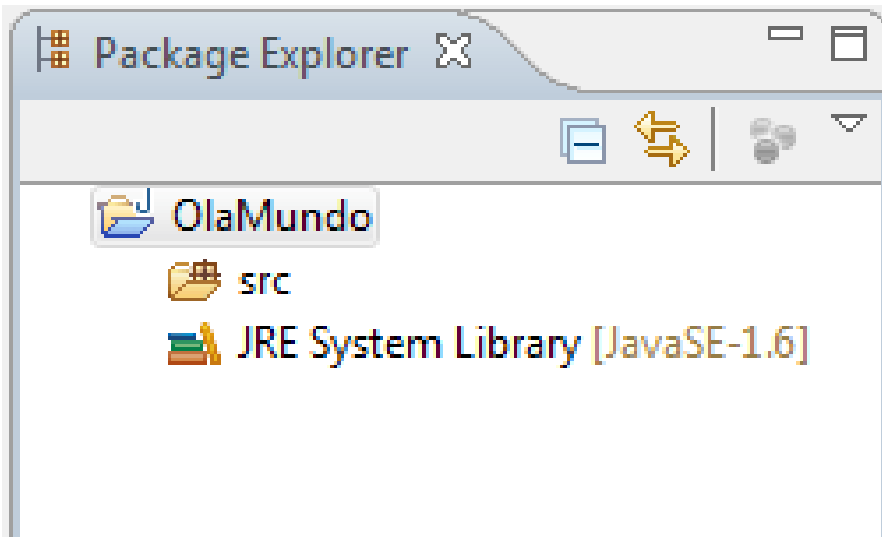
Definindo o projeto

- Coloque um nome para o projeto:
OlaMundo
- Demais configurações, deixar padrão.
- Clique em **Finish**



Estrutura do projeto

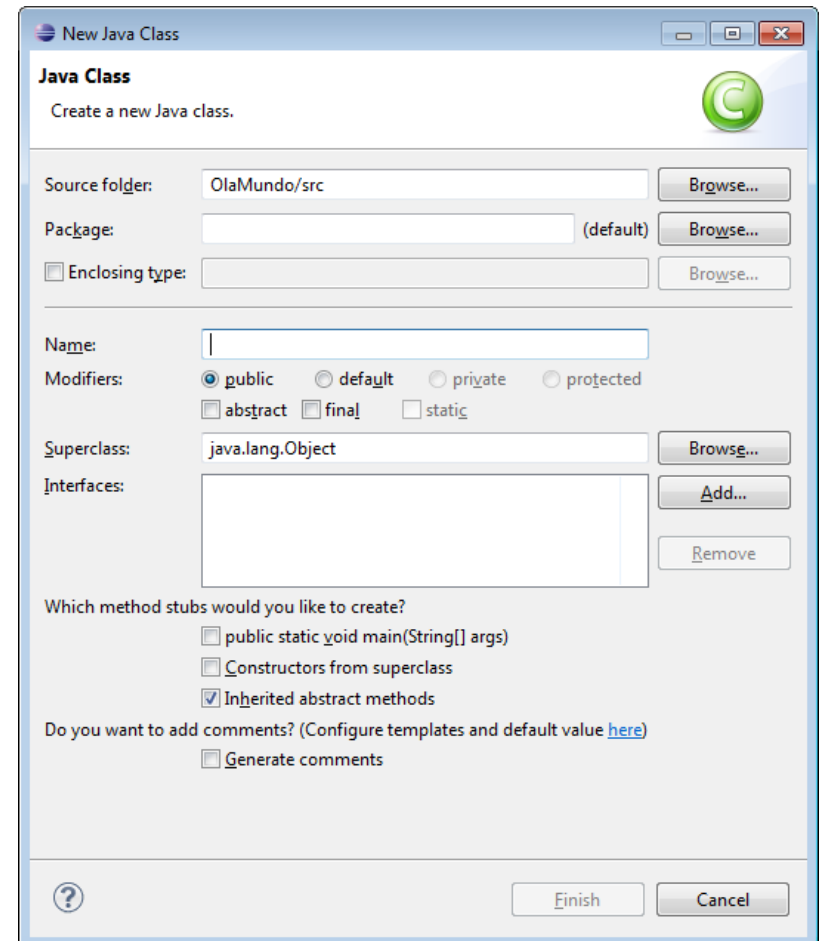
- No package Explorer vai aparecer o projeto recém criado.



O Package Explorer permite visualizar toda a estrutura de diretórios e arquivos contidos no projeto

Criando uma classe

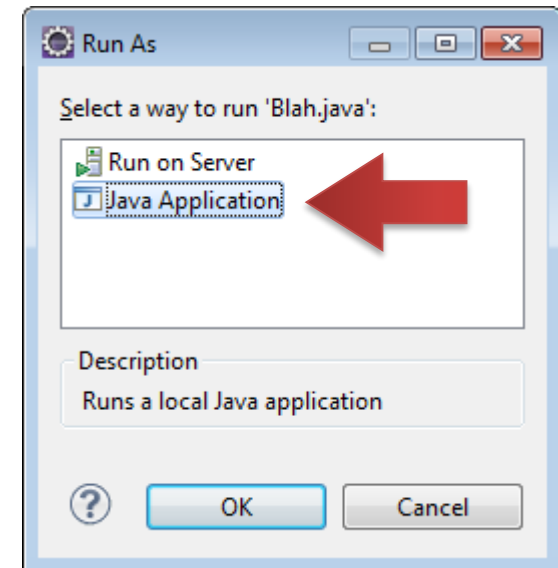
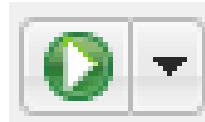
- Clicar botão direito na pasta src:
 - New -> Class
- No campo Name digite:
Ola
- No campo Package:
br.ifpr.oo2



Executando o código

```
public class Exemplo {  
  
    public static void main(String[] args) {  
        System.out.println(">>>...");  
    }  
  
}
```

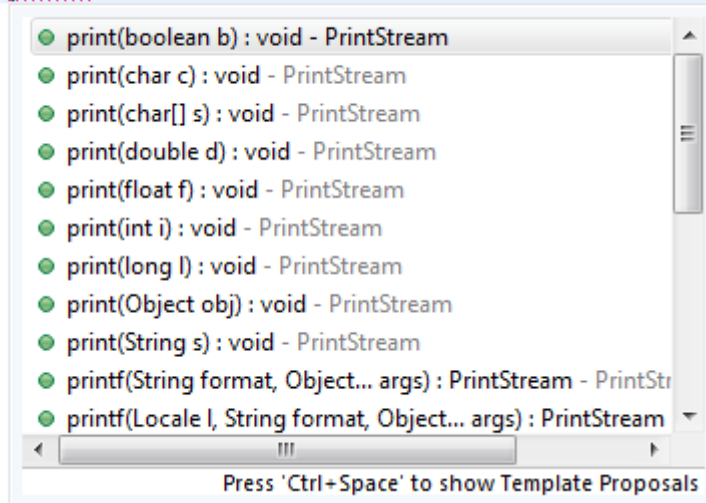
- Usar o atalho **CTRL + F11**
- ou clique no botão:



Principais funcionalidades

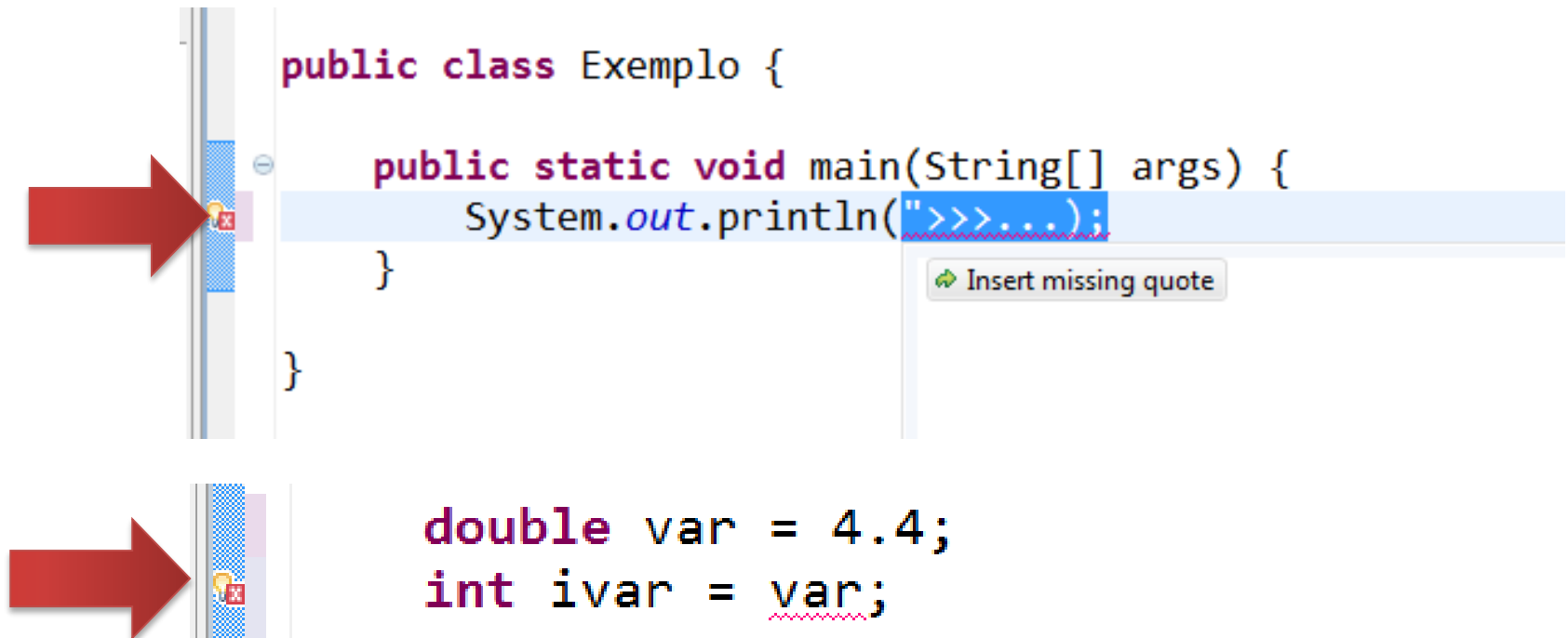
- Detecção de erro
- Função autocompletar.

System.out.prin



Sugestão de correção

- Possui um assistente que sugere correções a serem feitas no código, quando encontrado um erro:



The image shows two examples of code correction suggestions in an IDE. In the first example, a red arrow points to a lightbulb icon next to a code error. The code is a Java class named 'Exemplo' with a 'main' method. The string literal in the 'println' statement is missing a closing quote. A tooltip suggests 'Insert missing quote'. In the second example, a red arrow points to a lightbulb icon next to a code error. The code declares a 'double' variable and then assigns it to an 'int' variable. A tooltip suggests a correction.

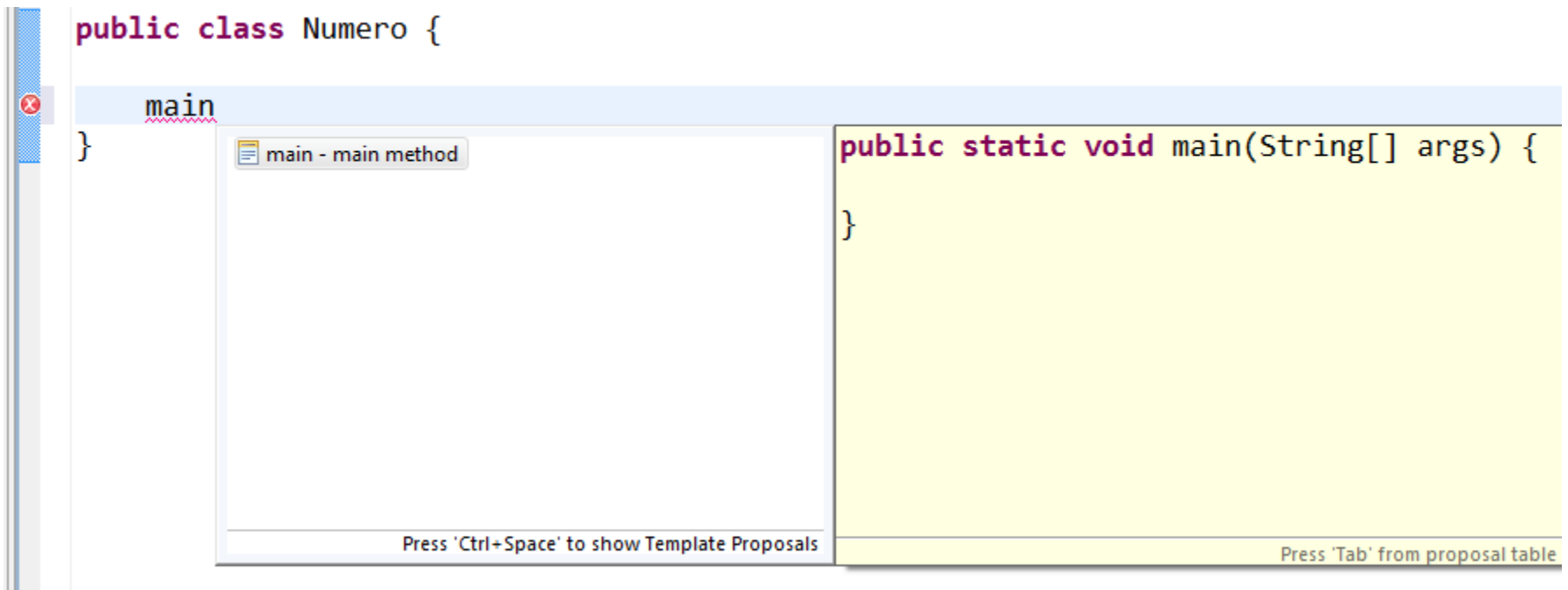
```
public class Exemplo {  
    public static void main(String[] args) {  
        System.out.println(">>>...");  
    }  
}
```

Insert missing quote

```
double var = 4.4;  
int ivar = var;
```

Função ctrl+space

- Digite main e logo em seguida: **CTRL + SPACE**
- O eclipse completa a palavra com a definição do método main.
Teste para sysout



The screenshot shows the Eclipse IDE interface. On the left, a snippet of Java code is visible: `public class Numero {` followed by a line where the word `main` is partially typed and underlined with a red dashed line. Below this, a small window titled "main - main method" is open, showing the full definition of the `main` method: `public static void main(String[] args) {` followed by a closing brace `}`. At the bottom of the IDE, there are two instructional messages: "Press 'Ctrl+Space' to show Template Proposals" on the left and "Press 'Tab' from proposal table" on the right.

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

Press 'Ctrl+Space' to show Template Proposals

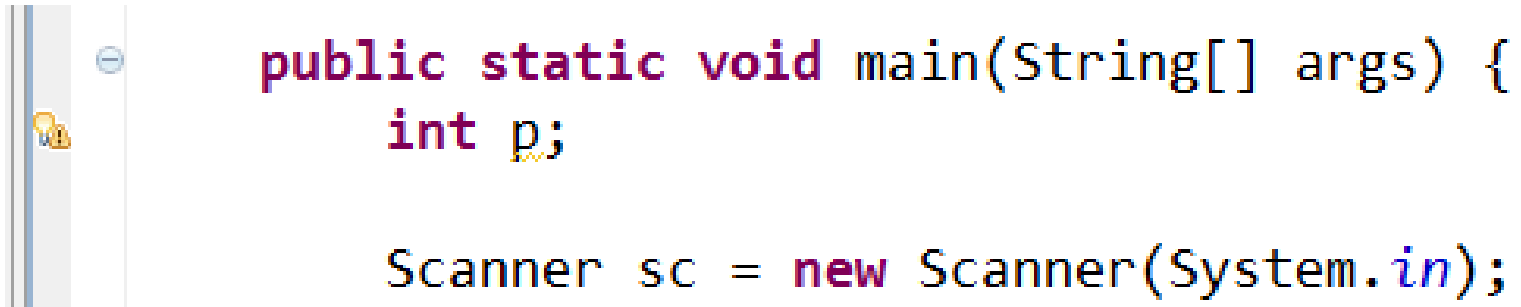
Press 'Tab' from proposal table

Função `ctrl+space`

- É uma funcionalidade do eclipse que permite injetar código a partir de palavras chaves.
- Também usado para importar um pacote.

Avisos (warning)

- Eclipse sublinha em amarelo, quando algo deve ser verificado no código, entretanto, não impede a compilação:

A screenshot of the Eclipse IDE's code editor. On the left side, there is a vertical toolbar with a yellow warning icon (a lightbulb with a lightning bolt) and a minus sign. The code in the editor is as follows:

```
public static void main(String[] args) {  
    int p;  
  
    Scanner sc = new Scanner(System.in);
```

- Nesse exemplo, a variável “p” foi declarada, mas não é utilizada.

Teclas de atalho

CTRL + /

Comenta uma linha

CTRL + D

Deleta uma linha

CTRL + A

Seleciona tudo

CTRL + SHIF + F

Indentação do código

CTRL + SHIF + O

Importa pacotes necessários

CTRL + M

Maximiza a janela de trabalho

Teclas de atalho

ALT + UP/DOW

Move uma linha para cima/baixo

CTRL + T

Encontra rapidamente uma classe

CTRL + SHIFT + R

Encontra rapidamente qualquer arquivo

CTRL + 1

Quick fix

Atividade

- 1) Escreva um programa no Eclipse que lê 2 números inteiros do teclado e imprime o **maior** deles;
- 2) Altere este programa para que ele imprima também a **soma** e a **média** aritmética dos 2 números. Crie métodos para melhor organizar o seu código.
- 3) Faça um programa que possui uma classe chamada **Idade**. Essa classe possui um método – **getMeses(int idade)** – que calcula a idade em meses. Teste o funcionamento da mesma.

Atividade

- 4) Implemente uma classe Computador, que possui: (1) uma placa mãe. Essa placa mãe possui: (a) uma placa de vídeo, (b) um HD e (c) um driver de DVD. Instancie as classes e relacione as mesmas com a classe Computador. Crie ao final um método start que inicializa do computador.