

Laboratório de Java

Ambiente Eclipse (Lab V)

Prof. Guido Araújo



ESPECIALIZAÇÃO EM
ENGENHARIA
DE SOFTWARE



UNICAMP



Eclipse

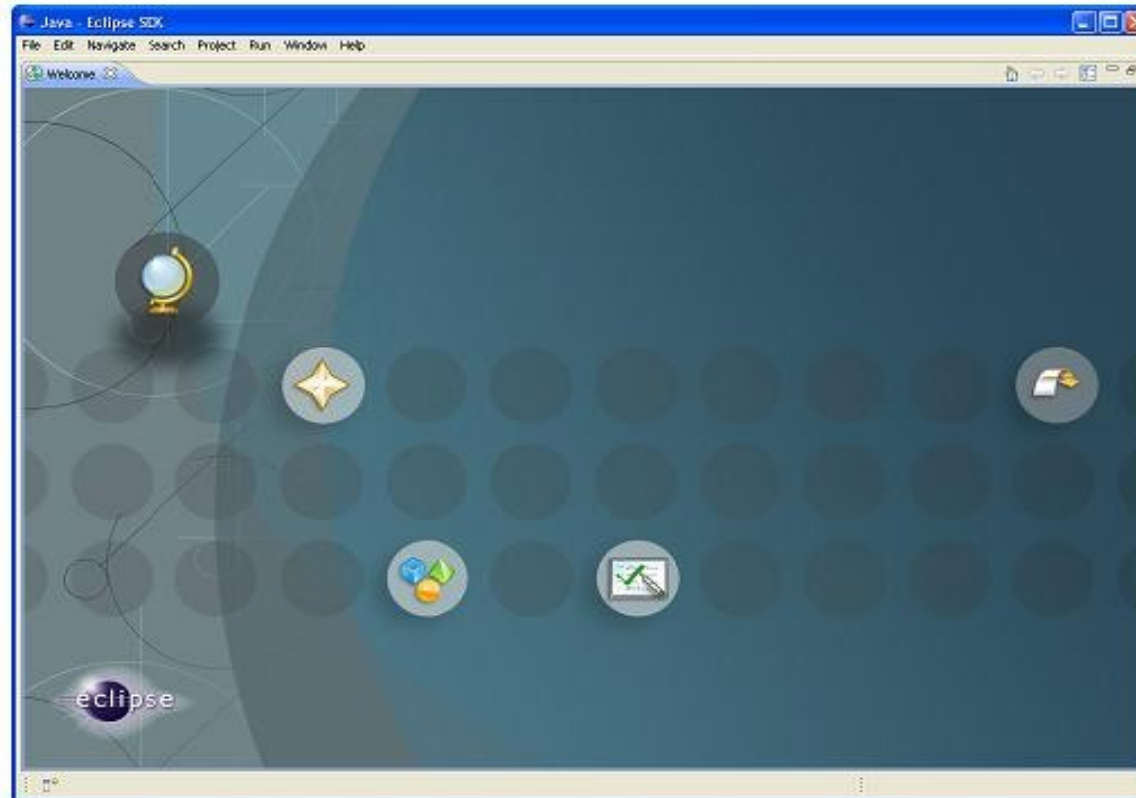


ESPECIALIZAÇÃO EM
ENGENHARIA
DE SOFTWARE



UNICAMP

Tela de Boas Vindas



A tela de boas-vindas será apresentada ao se abrir o eclipse pela primeira vez



ESPECIALIZAÇÃO EM
ENGENHARIA
DE SOFTWARE



UNICAMP

Ícones da Tela de Boas-Vindas



OverView – Informações básicas sobre o Eclipse.



What's New – Novidades sobre o Eclipse



Samples – Exemplos de código.



Tutorials – Tutorias sobre o Eclipse.



WorkBench – Ponto de partida para se começar a trabalhar no Eclipse.

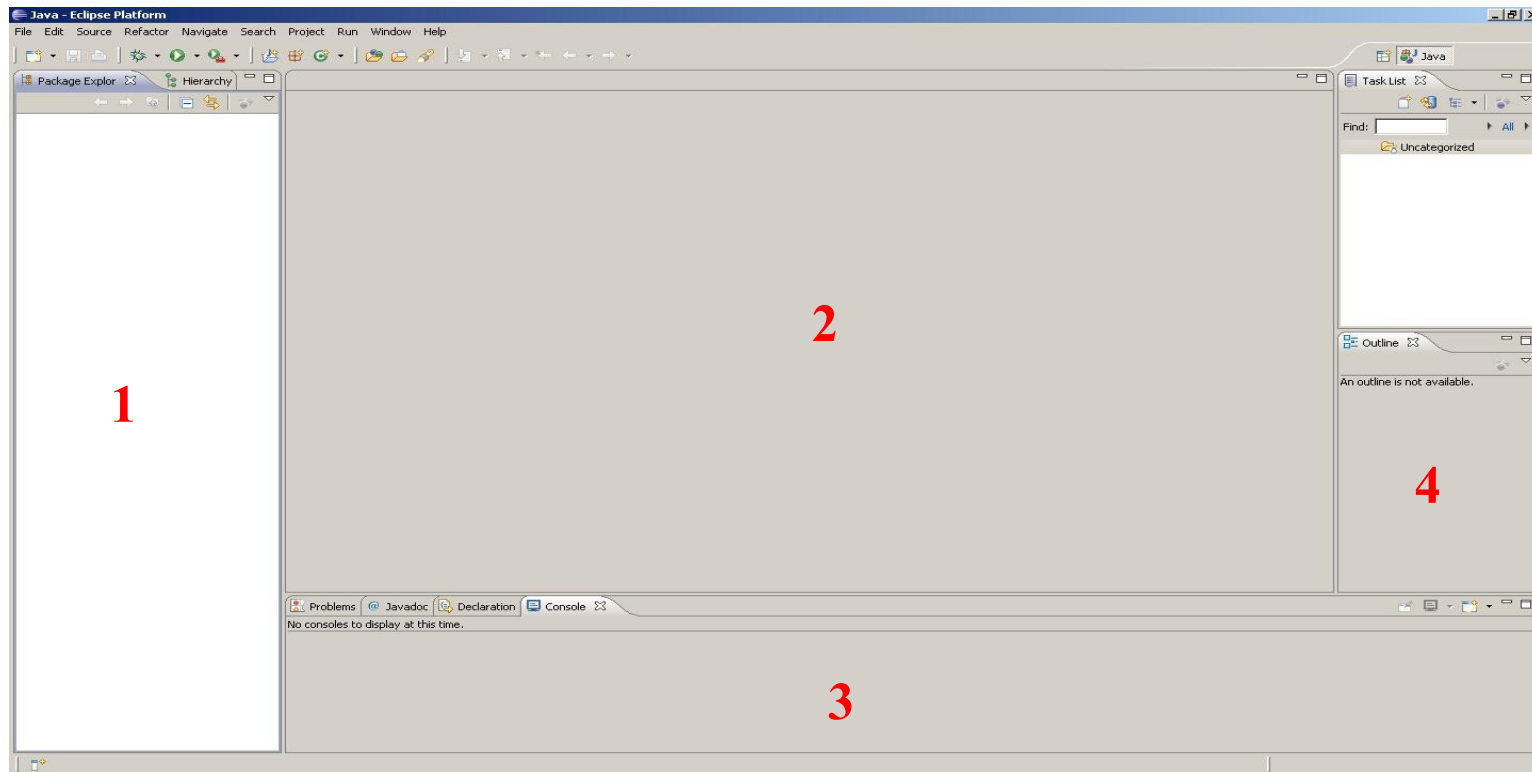


ESPECIALIZAÇÃO EM
ENGENHARIA
DE SOFTWARE



UNICAMP

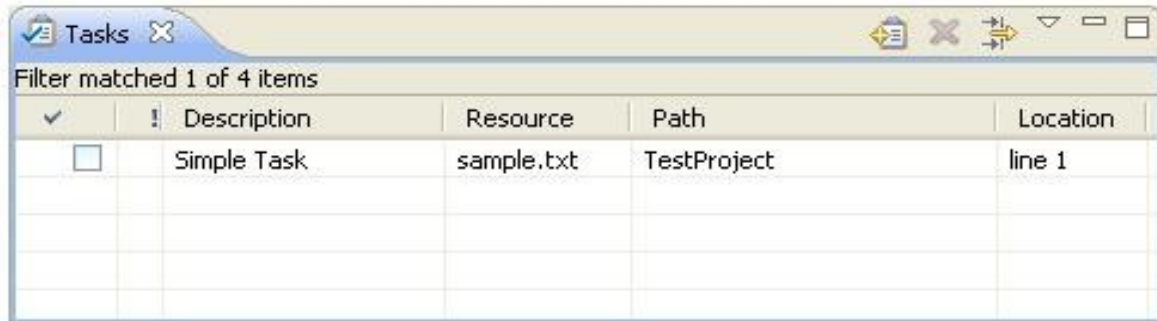
Abas do WorkBench



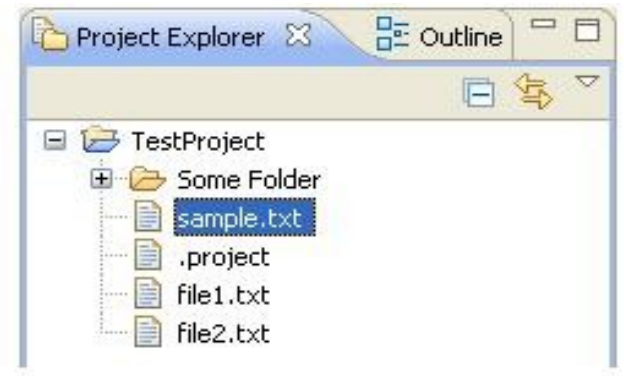
- 1 Package Explorer: Os projetos ficam localizados aqui.
- 2 Java View: Janela onde se visualiza o código java dos programas.
- 3 Console: A saída padrão das aplicações java.
- 4 OutLine: Todas as funções e includes do código.



Abas do WorkBench



✓	!	Description	Resource	Path	Location
<input checked="" type="checkbox"/>		Simple Task	sample.txt	TestProject	line 1
<input type="checkbox"/>					
<input type="checkbox"/>					
<input type="checkbox"/>					
<input type="checkbox"/>					

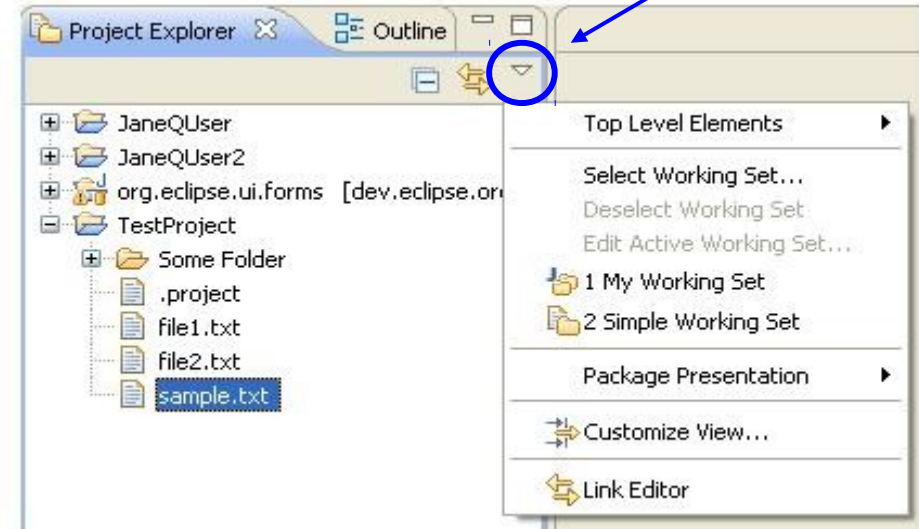


Cada aba pode aparecer “solta” ou “presa” ao ambiente do workbench.

Para seleccionar uma aba, basta clicar sobre a mesma.



Abas do WorkBench

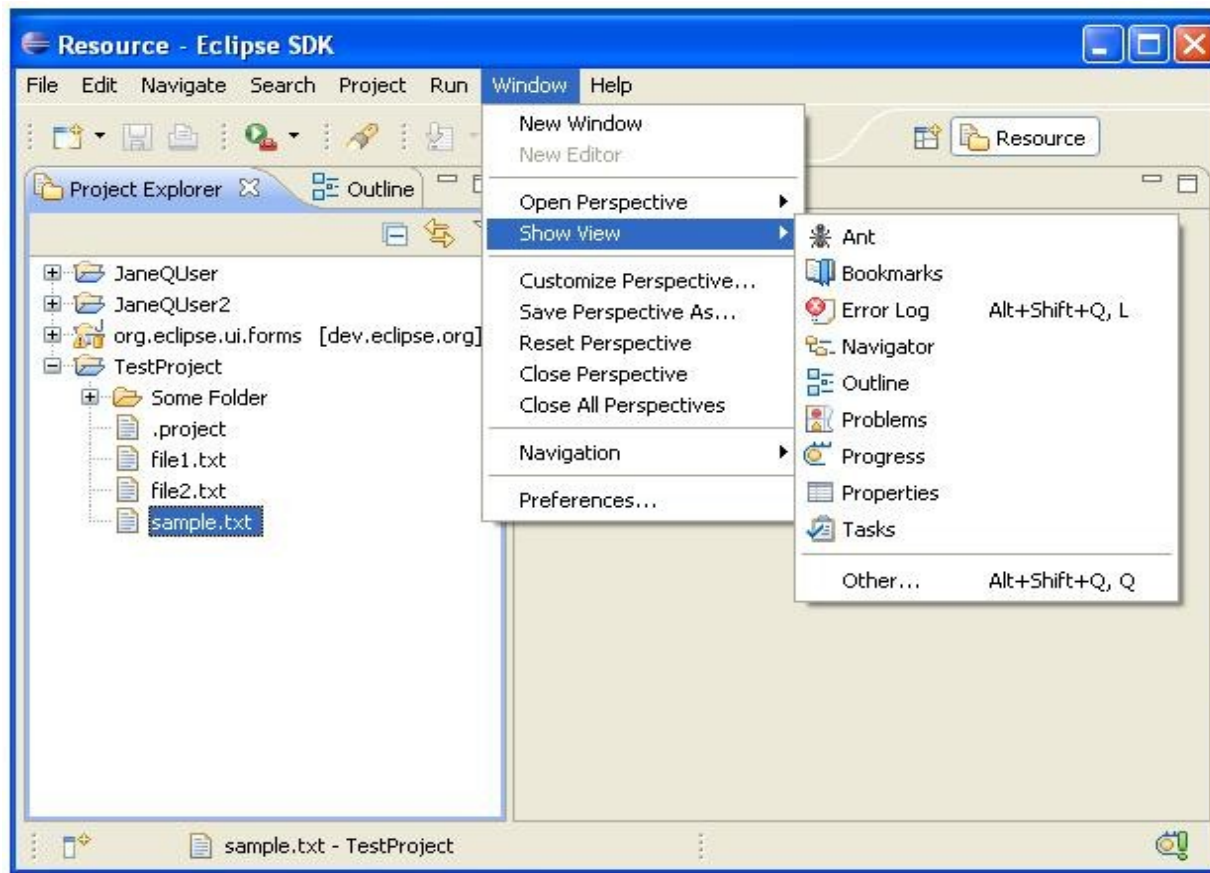


Cada aba possui dois menus.

- O primeiro é acessado através de um “right-click” no título da aba, permitindo que a aba seja manipulada da mesma forma que no menu associado na janela do workbench.
- O segundo menu é acessado clicando na flecha que aponta para baixo. Tal menu é chamado de “view pull-down menu” e contém ações que estão relacionadas com o conteúdo da aba.



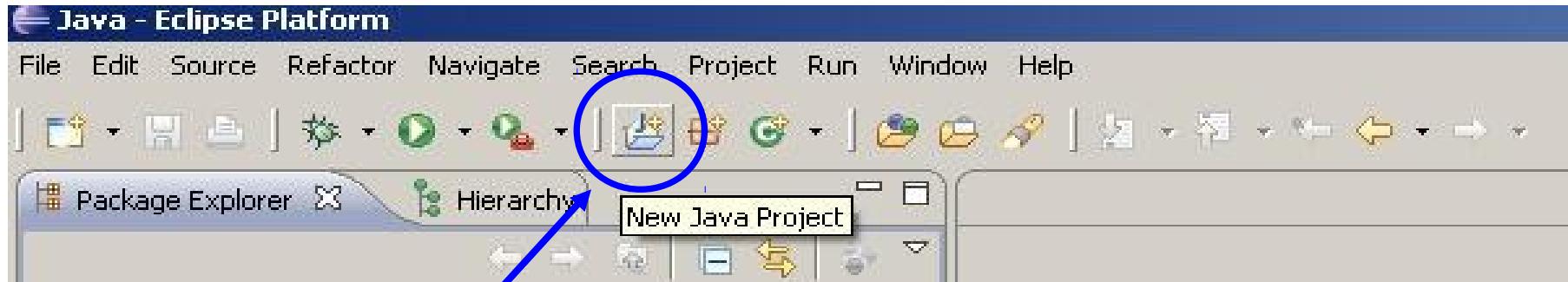
Abas do WorkBench



As abas também podem ser visualizadas através do menu **Window > Show View**. Durante a utilização do workbench, você pode redimensionar, mover e fechar abas. Para voltar a mostrar a perspectiva padrão, utilize o menu **Window > Reset Perspective**.



Criando um Projeto



Clique em New Java Project para criar o seu primeiro projeto.



Criando um Projeto

New Java Project

Create a Java project in the workspace or in an external location.

Project name:

Contents

- ☒ Create new project in workspace
- ☐ Create project from existing source

Directory: [Browse...](#)

JRE

- ☐ Use default JRE (Currently 'jre1.6.0_04') [Configure default...](#)
- ☐ Use a project specific JRE:
- ☒ Use an execution environment JRE:

Project layout

- ☐ Use project folder as root for sources and class files
- ☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

- ☐ Add project to working sets

Working sets: [Select...](#)

[?](#) [< Back](#) [Next >](#) [Finish](#) [Cancel](#)

Campo para inserir o nome do Projeto

WorkSpace onde todos os projetos são salvos



Criando um Projeto

New Java Project

Create a Java project in the workspace or in an external location.

Project name:

Contents

- ☒ Create new project in workspace
- ☐ Create project from existing source

Directory: [Browse...](#)

JRE

- ☐ Use default JRE (Currently 'jre1.6.0_04') [Configure default...](#)
- ☐ Use a project specific JRE:
- ☒ Use an execution environment JRE:

Project layout

- ☐ Use project folder as root for sources and class files
- ☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

- ☐ Add project to working sets

Working sets: [Select...](#)

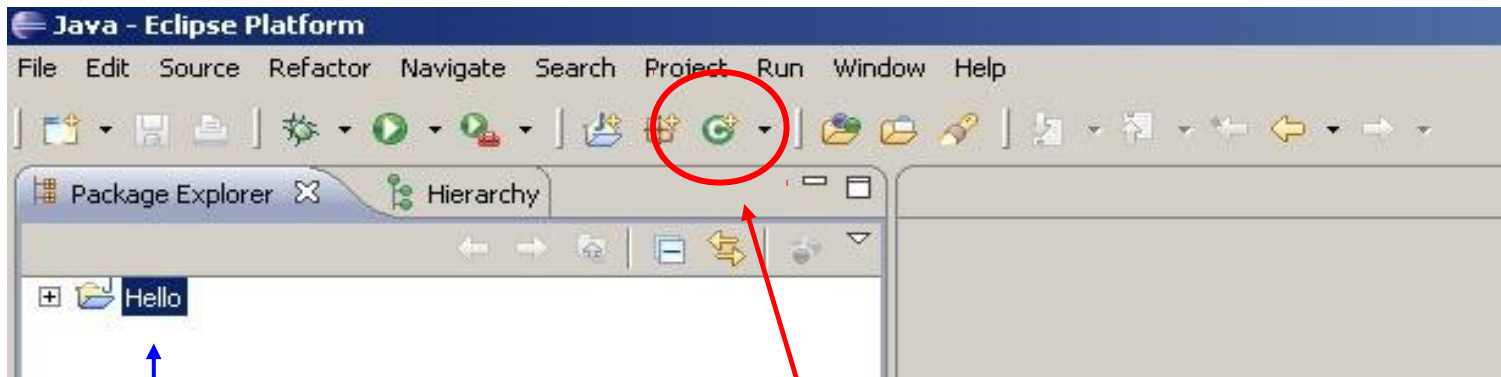
[? < Back](#) [Next >](#) **Finish** [Cancel](#)

Entre com o nome do Projeto.
Clique em Finish.



Criando um Projeto

O projeto já foi criado e já aparece na aba Package Explorer. O próximo passo é criar uma classe para poder inserir código.



Nosso primeiro Projeto.

Clique no botão New Java Class para inserir a primeira classe do Projeto.



ESPECIALIZAÇÃO EM
ENGENHARIA
DE SOFTWARE



UNICAMP

Criando um Projeto

New Java Class

Java Class
Create a new Java class.

Source folder: Hello/src Browse...

Package: (default) Browse...

☐ Enclosing type: Browse...

Name:

Modifiers: ☒ public ☐ default ☐ 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 as configured in the [properties](#) of the current project?
☐ Generate comments

Finish Cancel

Campo para inserir o nome da Classe



Criando um Projeto

New Java Class

Java Class

The use of the default package is discouraged.

Source folder:

Package:

☐ Enclosing type:

Name:

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

Superclass:

Interfaces:

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 as configured in the [properties](#) of the current project?

☒ Generate comments

Vamos chamar a classe de HelloWorld_Main

Se marcarmos esta opção o Eclipse vai gerar comentários automaticamente contendo informações sobre o Projeto e o autor do mesmo.



Clique em Finish para criar a classe.



Criando uma Classe para o Projeto



Podemos observar que o Eclipse já inseriu algumas porções de comentários e criou um método vazio.

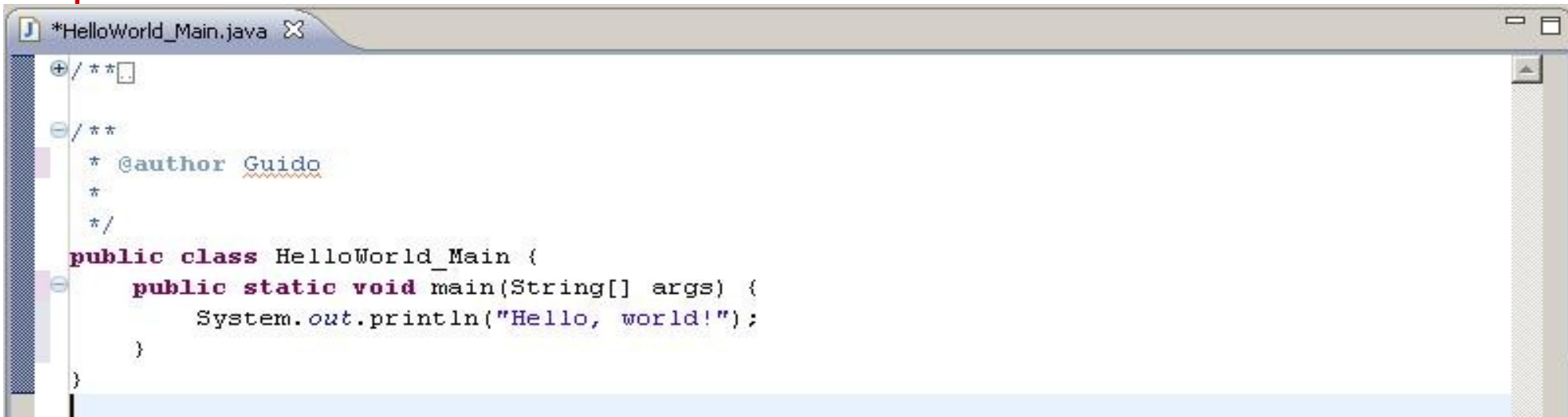
Os sinais de  e  que aparecem na lateral podem ser clicados para expandir ou contrair porções do código.



Inserindo Código e Executando o Projeto

Agora já podemos inserir código no nosso projeto. Para começar, vamos criar o programa Hello World padrão:

```
public class HelloWorld_Main {  
    public static void main(String[] args) {  
        System.out.println("Hello, world!");  
    }  
}
```

A screenshot of a Java IDE window titled '*HelloWorld_Main.java'. The code editor shows the following content: a multi-line comment with '@author Guido', followed by the Java code for a 'Hello World' program. The code is:

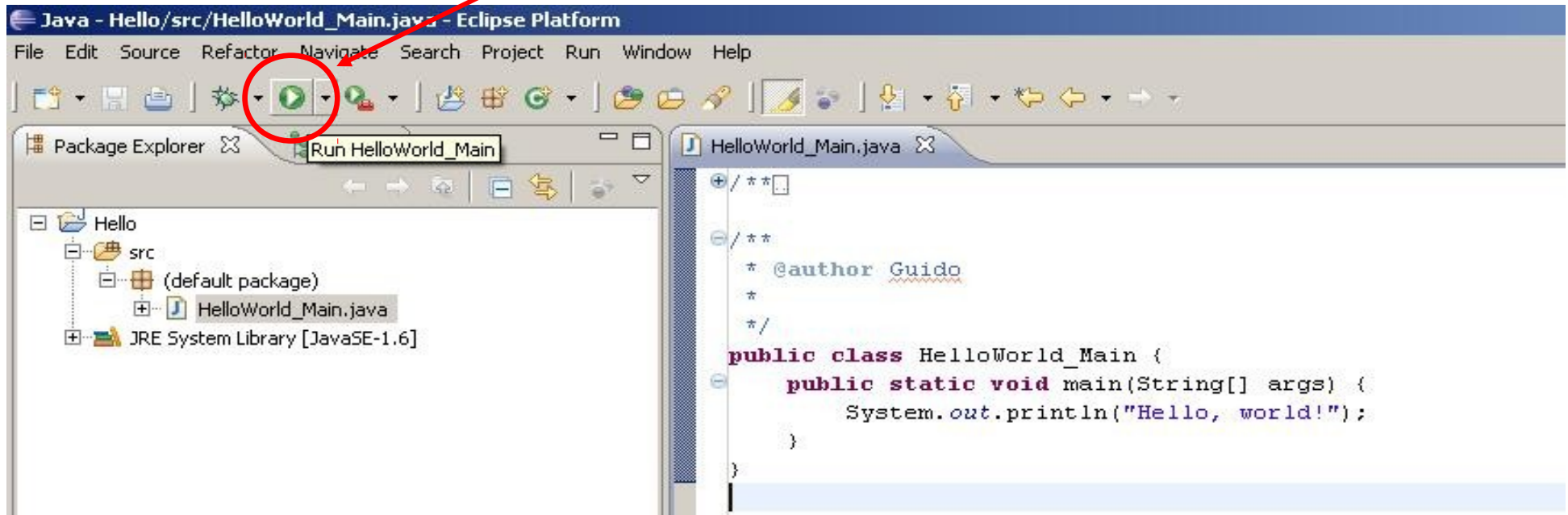
```
public class HelloWorld_Main {  
    public static void main(String[] args) {  
        System.out.println("Hello, world!");  
    }  
}
```



Inserindo Código e Executando o Projeto

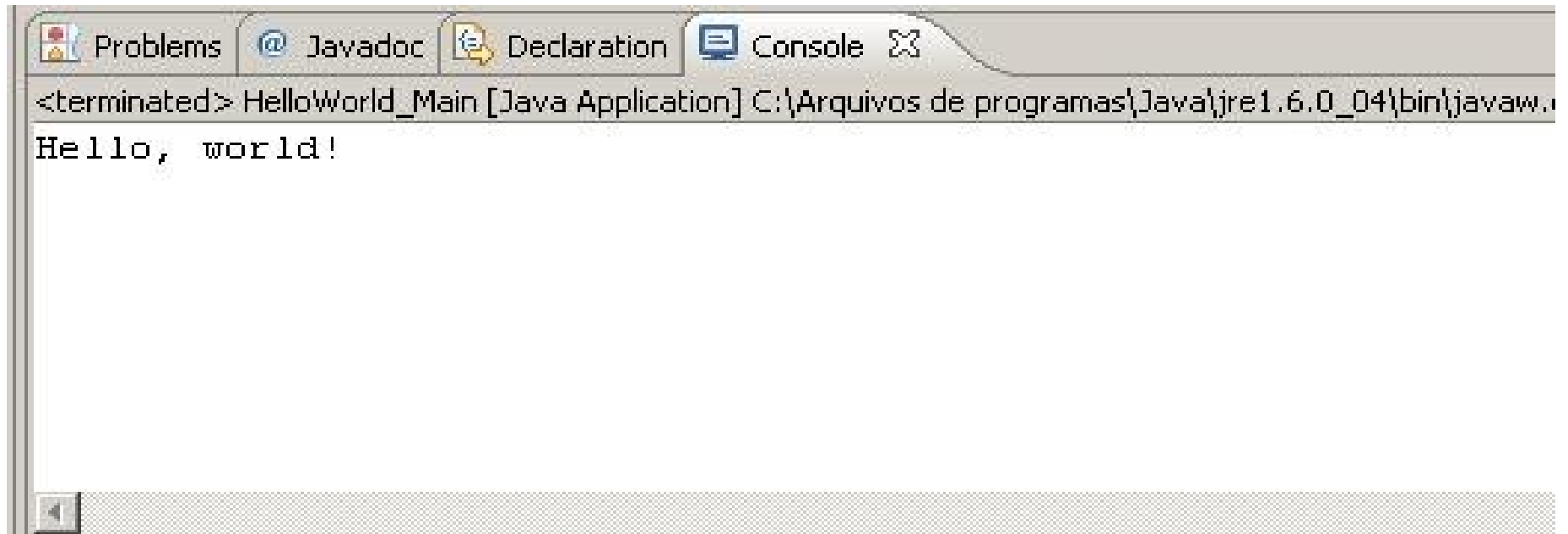
O próximo passo é executar o código escrito, isto é feito pressionando o botão run:

Botão RUN

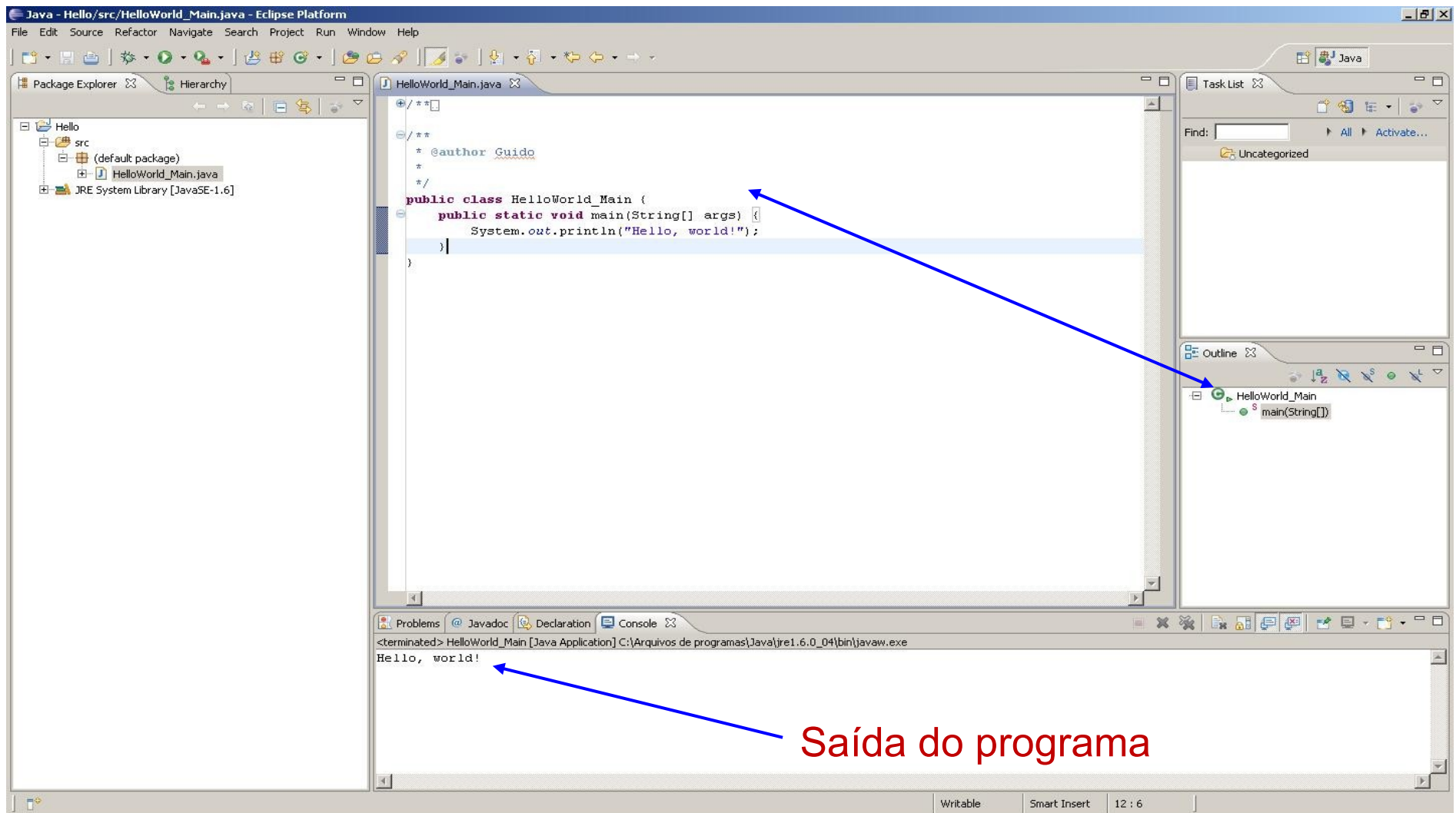


Inserindo Código e Executando o Projeto

A saída do programa será mostrada na aba Console:



Visão Geral do Projeto



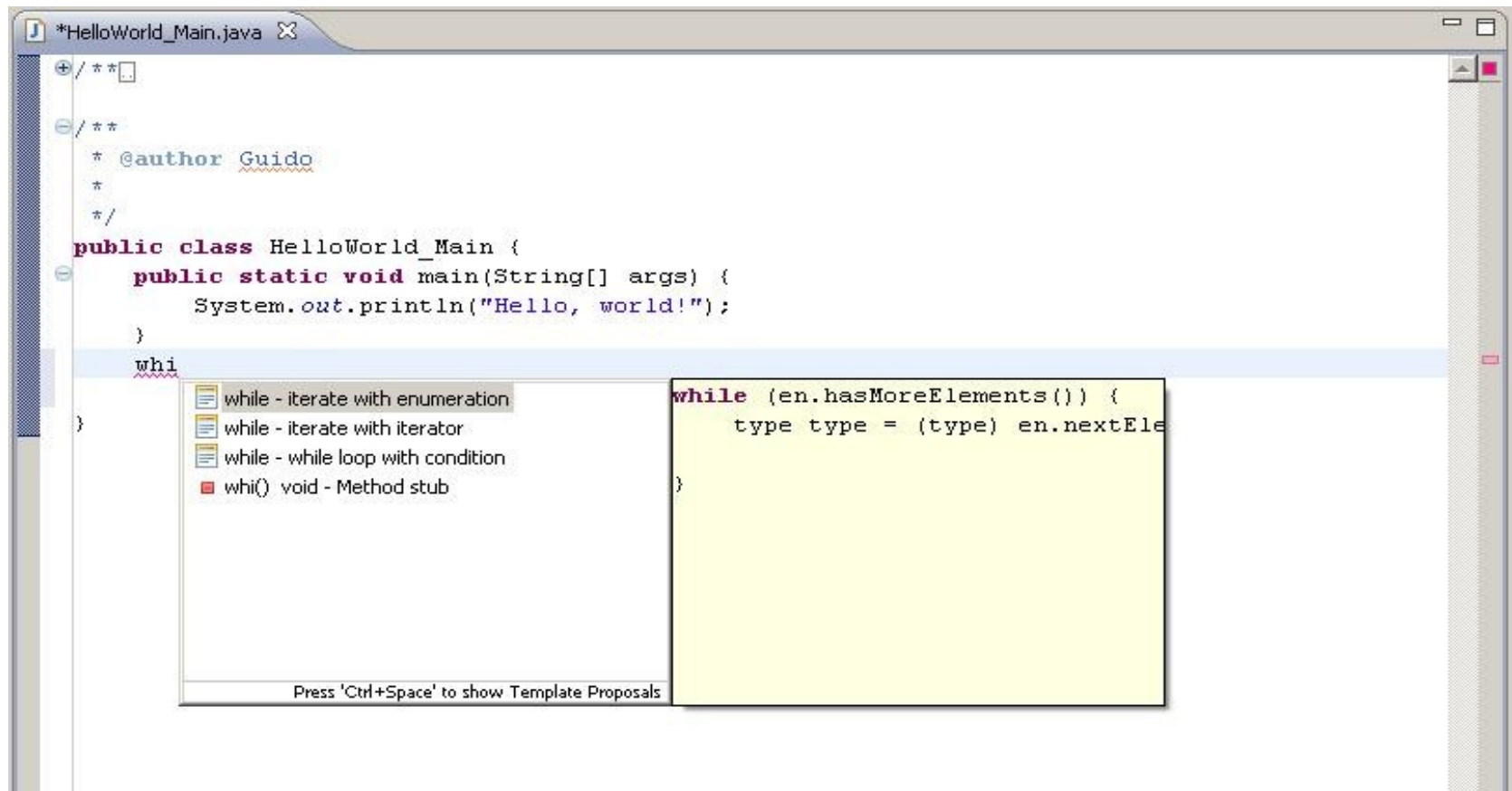
ESPECIALIZAÇÃO EM
ENGENHARIA
DE SOFTWARE



UNICAMP

Dica

Quando estiver escrevendo código, você pode pressionar as teclas Ctrl+Barra de Espaço. Esta sequência de teclas faz com que o Eclipse complete o código sendo digitado ou sugira opções a serem usadas como no exemplo abaixo:



Um exemplo com leitura de dados

```
import java.io.*;
public class HelloWorld_Main {
    public static void main(String[] args) {
        // Pergunta ao usuario o seu nome
        System.out.print("Entre o seu nome: ");
        // Abre a entrada padrao
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
        // Cria uma variavel para armazenar o nome
        String userName = null;
        // Uso da estrutura try/catch com o método readLine()
        try {
            userName = br.readLine();
        }
        catch (IOException ioe) {
            System.out.println("Erro ao tentar ler o seu nome!");
            System.exit(1);
        }

        // Pergunta o RA do usuario
        System.out.print("Entre o seu RA: ");
        // Cria uma variavel para armazenar o RA
        String ra = null;
        // Uso da estrutura try/catch com o método readLine();
        try{
            ra = br.readLine();
        }
        catch(IOException ioe){
            System.out.println("Erro ao tentar ler o RA!");
            System.exit(1);
        }

        // Imprime o resultado final
        System.out.println(userName + ", seu RA eh... " + ra);
    }
}
```



Dicas e Truques

Busca Incremental: Pressione Ctrl+J para iniciar a busca incremental. Basta começar a digitar a palavra e o Eclipse irá marcar todas as ocorrências que ele encontrar. Abaixo o que acontece ao se realizar a busca incremental ao se procurar por **tr**:

```
public class HelloWorld Main {  
    public static void main(String[] args) {  
        // Pergunta ao usuário o seu nome  
        System.out.print("Entre o seu nome: ");  
        // Abre a entrada padrão  
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));  
        // Cria uma variável para armazenar o nome  
        String userName = null;  
        // Uso da estrutura try/catch com o método readLine()  
        try {  
            userName = br.readLine();  
        }  
        catch (IOException ioe) {  
            System.out.println("Erro ao tentar ler o seu nome!");  
            System.exit(1);  
        }  
        System.out.print("Entre o seu RA: ");  
        String ra = null;  
        try{  
            ra = br.readLine();  
        }  
        catch(IOException ioe)  
        {  

```

Para finalizar a busca, basta pressionar ENTER ou ESC.



Dicas e Truques

Tornando todas as letras em **MAIÚSCULO**: Selecione a palavra(s) e pressione Ctrl+Shift+X

```
System.out.print("Entre o seu RA: ");  
String ra = null;  
try{  
    ra = br.readLine();  
}
```



```
System.out.print("Entre o seu RA: ");  
String ra = null;  
TRY{  
    ra = br.readLine();  
}
```

Tornando todas as letras em **MINÚSCULO**: Selecione a palavra(s) e pressione Ctrl+Shift+Y

```
System.out.print("Entre o seu RA: ");  
String ra = null;  
TRY{  
    ra = br.readLine();  
}
```

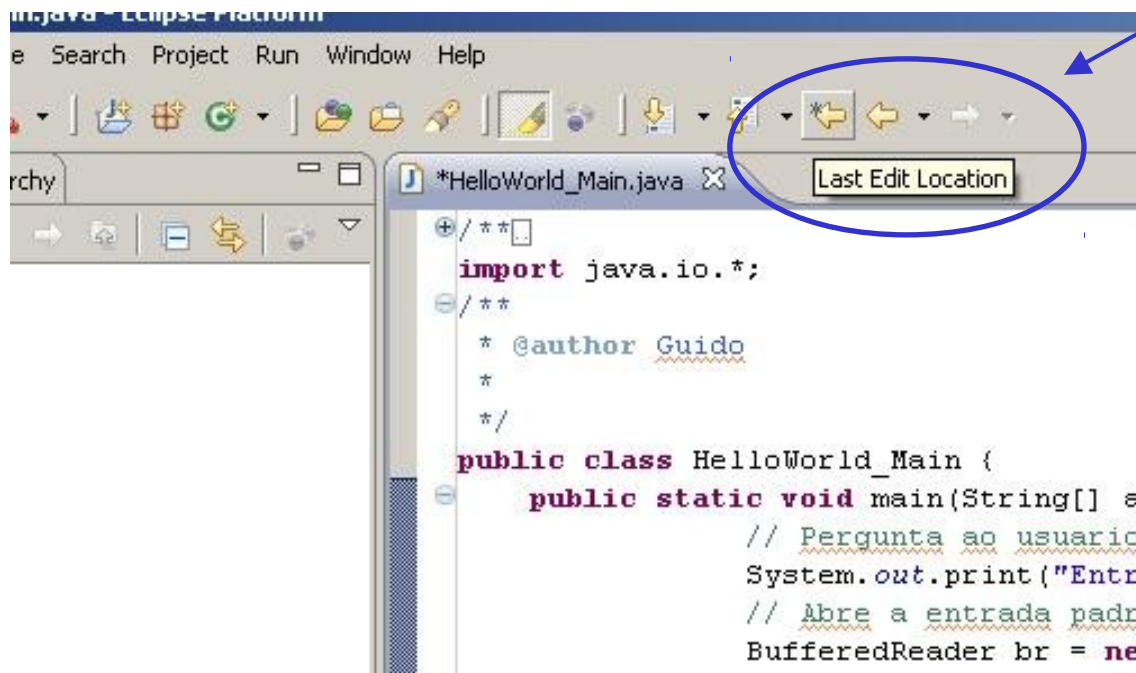


```
System.out.print("Entre o seu RA: ");  
String ra = null;  
try{  
    ra = br.readLine();  
}
```



Dicas e Truques

Ir para a última alteração: Ao se pressionar Ctrl+Q o Eclipse posicionará o cursor na última alteração realizada no código. Você também pode utilizar um botão da barra de ferramentas.



Dicas e Truques

Verificação Automática: Ao digitar código, o Eclipse procura automaticamente por erros em seu programa, caso ele encontre algum, a linha será marcada com um ícone de erro.

