



# Ambiente Java

Para desenvolver e executar qualquer programa Java, você precisa instalar o JDK no seu sistema. Você pode fazer o download da versão mais recente do Java aqui. <https://www.oracle.com/java/technologies/javase-downloads.html>). Depois de abrir o link, você poderá ver algo como a figura abaixo. Clique na caixa vermelha para fazer o download do JDK.



Após selecionar a plataforma Windows e clique no link Download (o nome do arquivo depende da versão do JDK).

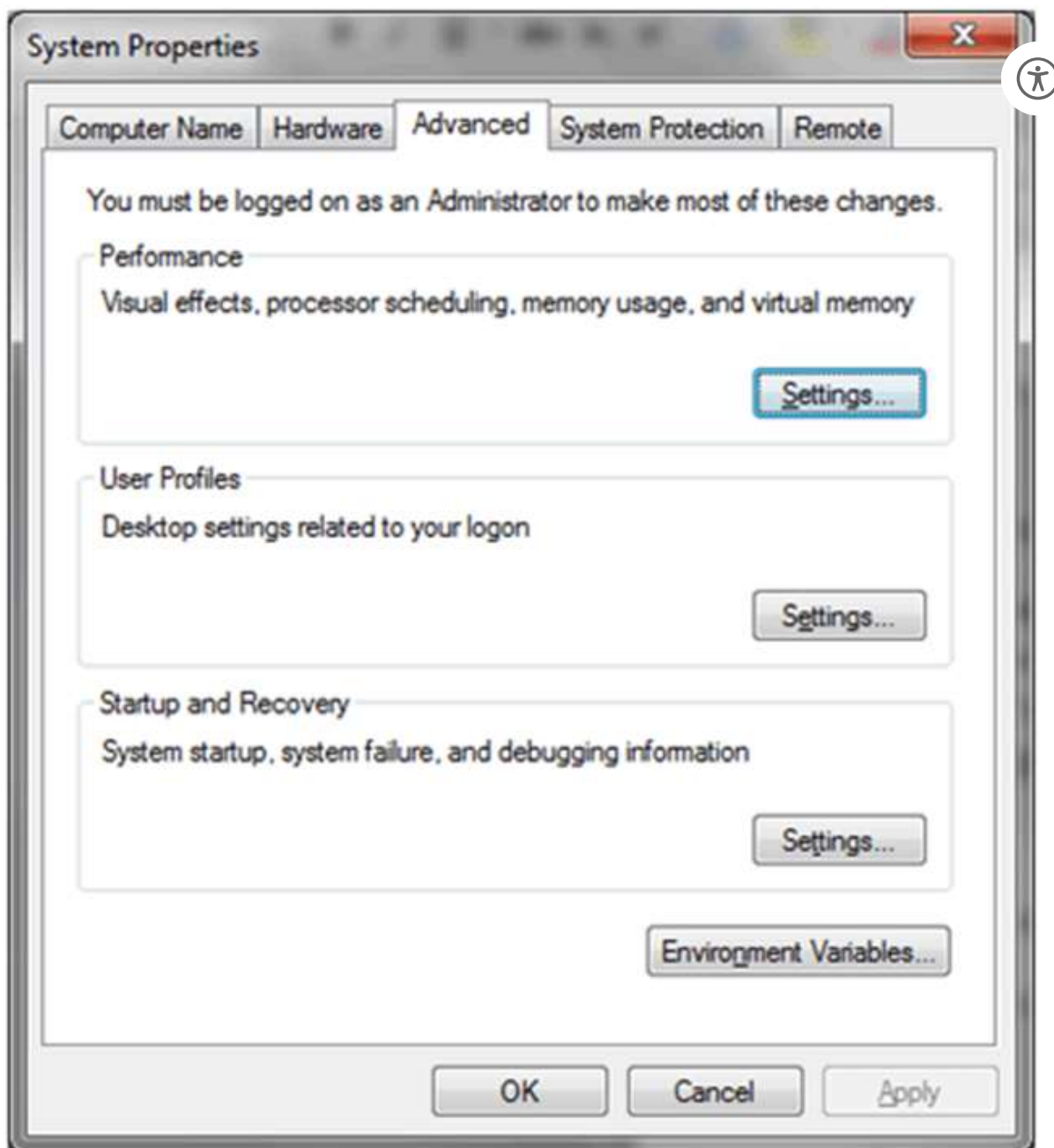
Depois que o download for concluído, você poderá iniciar a instalação. É muito semelhante à instalação de outro software no Windows. Após alguns minutos do processo de instalação, você verá uma janela pop-up perguntando onde instalar o JRE. Assumirei que você aceitou o diretório padrão, mas você pode selecionar outro. Em breve, você verá uma mensagem informando que a instalação foi bem-sucedida.

Você pode verificar novamente a instalação usando o prompt de comando. Use o comando “java -version” para verificar a versão do JRE instalada.



Se você não conseguir ver a versão Java como acima no prompt de comando, precisará definir o caminho da classe na variável do sistema. Para definir o PATH permanentemente:

Escolha Iniciar, Configurações, Painel de controle e clique duas vezes no Sistema, selecione a guia Avançado e clique em Variáveis de ambiente. Procure por “Caminho” nas Variáveis de Usuário e Variáveis de Sistema. Se você não tiver certeza de onde adicionar o caminho, adicione-o à extremidade direita do “Caminho” nas Variáveis do usuário. O valor deve ser o diretório de instalação.



O novo caminho entra em vigor em cada nova janela do Prompt de Comando que você abre após definir a variável PATH.

O programa Java pode ser escrito usando um editor de texto simples e compilado usando a janela de comando. Esta não é uma maneira produtiva de programação. Programadores profissionais usam um dos IDEs (Integrated Development Environments), que inclui um editor, um compilador, ajuda de digitação antecipada, um depurador e muito mais. Existem vários IDEs Java populares.

- Eclipse: IDE grátis com longa lista de plugins e ótimo suporte da comunidade

- NetBeans (<http://netbeans.org/downloads/>) : IDE gratuito e leve




- IntelliJ IDEA (<http://www.jetbrains.com/idea/download/>) : Ele vem em dois sabores. Versão gratuita e paga. A versão paga é realmente poderosa, com ótimos plugins.
- RAD (<http://www.ibm.com/developerworks/downloads/r/rad/>) : Grátis para avaliação.
- Site de download do Eclipse <https://www.eclipse.org/downloads/packages/>

O Eclipse não precisa de nenhuma instalação. Basta baixar o arquivo zip e descompactá-lo em uma unidade de disco de sua escolha. Você encontrará o arquivo eclipse.exe na pasta Eclipse - basta executar este programa. Você verá imediatamente uma janela pop-up solicitando que você selecione uma área de trabalho, que é um diretório no disco rígido em que um ou mais de seus projetos serão armazenados.


OBS.: Mas se você quiser, pode baixar o Eclipse executável, dessa forma é só dar duplo clique, que a instalação será padrão (Next/Next).

## Eclipse IDE 2020-03 R Packages




**Eclipse IDE for Java Developers**  
196 MB 513,890 DOWNLOADS  
The essential tools for any Java developer, including a Java IDE, a Git client, XML Editor, Mylyn, Maven and Gradle integration

Windows 64-bit  
Mac Cocoa 64-bit  
Linux 64-bit



**Eclipse IDE for Enterprise Java Developers (includes Incubating components)**  
400 MB 467,099 DOWNLOADS  
Tools for developers creating Java Enterprise and Web applications, including a Java IDE, tools for Web Services, JPA and Data Tools, JSF, Mylyn, Maven and Gradle, Git, and more.  
[Click here to file a bug against Eclipse Web Tools Platform.](#)  
[Click here to file a bug against Eclipse Platform.](#)  
[Click here to file a bug against Maven integration for web projects.](#)  
[Click here to report an issue against Eclipse Wild Web Developer \(incubating\).](#)

Windows 64-bit  
Mac Cocoa 64-bit  
Linux 64-bit



**Eclipse IDE for C/C++ Developers (includes Incubating components)**  
235 MB 133,752 DOWNLOADS  
An IDE for C/C++ developers with Mylyn integration.

Windows 64-bit  
Mac Cocoa 64-bit  
Linux 64-bit

## Executando o programa Java no Windows

Vamos tentar executar o primeiro programa Java. Para começar a escrever um programa Java, você pode usar qualquer editor de texto sem formatação. O arquivo que contém o código Java hello world deve ser a extensão .java. Crie um diretório dentro da unidade C como “c: / javadevelopment /”. Agora salve o código abaixo como HelloWorld.java. (Se você estiver usando o Bloco de notas, selecione Todos os arquivos no menu suspenso Salvar como tipo para evitar o anexo automático do sufixo .txt.

### Etapas 1: Criar HelloWorld.java

Digite o seguinte código no editor de texto.

```
package br.com.java.aula;
class HelloWorld {
    /**
     * comentários
     */
    public static void main(String[] args) {
        // comentário: imprimindo resultado no console
        System.out.println("Hello World! Oi mundo");
    }
}
```



## **Etapas 2: Compilar HelloWorld.java**

javac HelloWorld.java (pressione Enter)

## **Etapas 3:**

Após compilar o arquivo java, você pode ver o arquivo .class criado (HelloWorld.class). Este é o código do nosso programa java que pode ser executado em qualquer plataforma. Podemos tentar rodar em nosso JRE usando o comando java – java HelloWorld (pressione enter)

Lembre-se de que Java é uma linguagem que diferencia maiúsculas de minúsculas, o que significa que, se você nomeou o programa HelloWorld com H maiúsculo e W maiúsculo, não deve tentar iniciar o programa helloworld.

## **Executando o programa Java no Linux**

### **Etapas 1: Crie um arquivo de origem:**

Você pode usar qualquer editor de texto para criar e editar arquivos de origem. Crie o diretório usando os seguintes comandos.

cd / tmp



mkdir javadevelopment

cd javadevelopment

Crie o arquivo de origem Java usando o editor vi e digite as seguintes linhas ou copie as seguintes linhas no arquivo.

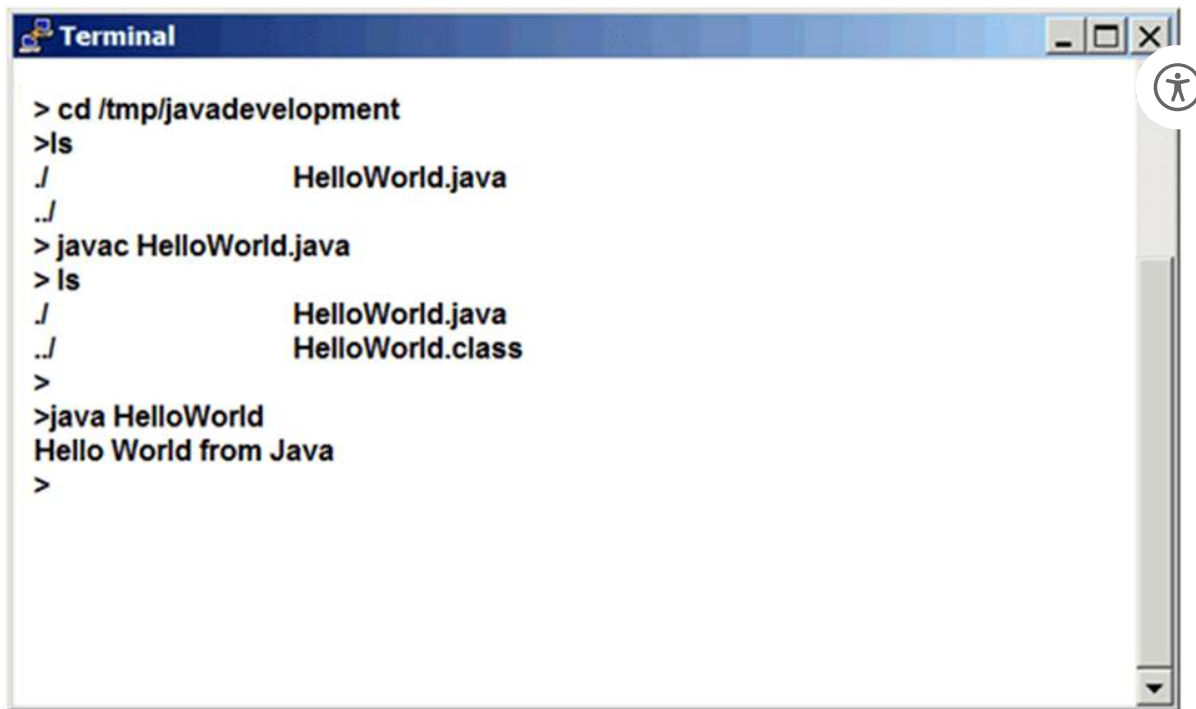
```
public class HelloWorld {  
    public static void main(String[] args) {  
        System.out.println("Hello World! Ol mundo");  
    }  
}
```

## **Etapas 2: Compilar o arquivo Java:**

Para compilar seu arquivo de origem, altere seu diretório atual para o diretório em que seu arquivo está salvo. Digite o comando (javac HelloWorld.java) e pressione enter. Após o término do processo de compilação, você poderá ver o arquivo HelloWorld.class criado na mesma pasta.

## **Etapas 3: Execute o programa Java:**

Digite o comando (java HelloWorld) e pressione enter para ver a saída na tela.

A screenshot of a Windows Terminal window titled "Terminal". The window has a blue title bar with standard minimize, maximize, and close buttons. On the right side of the title bar, there is a small circular icon with a person inside. The terminal content shows a series of commands and their outputs:

```
> cd /tmp/javadevelopment
> ls
./          HelloWorld.java
../
> javac HelloWorld.java
> ls
./          HelloWorld.java
../          HelloWorld.class
>
> java HelloWorld
Hello World from Java
>
```

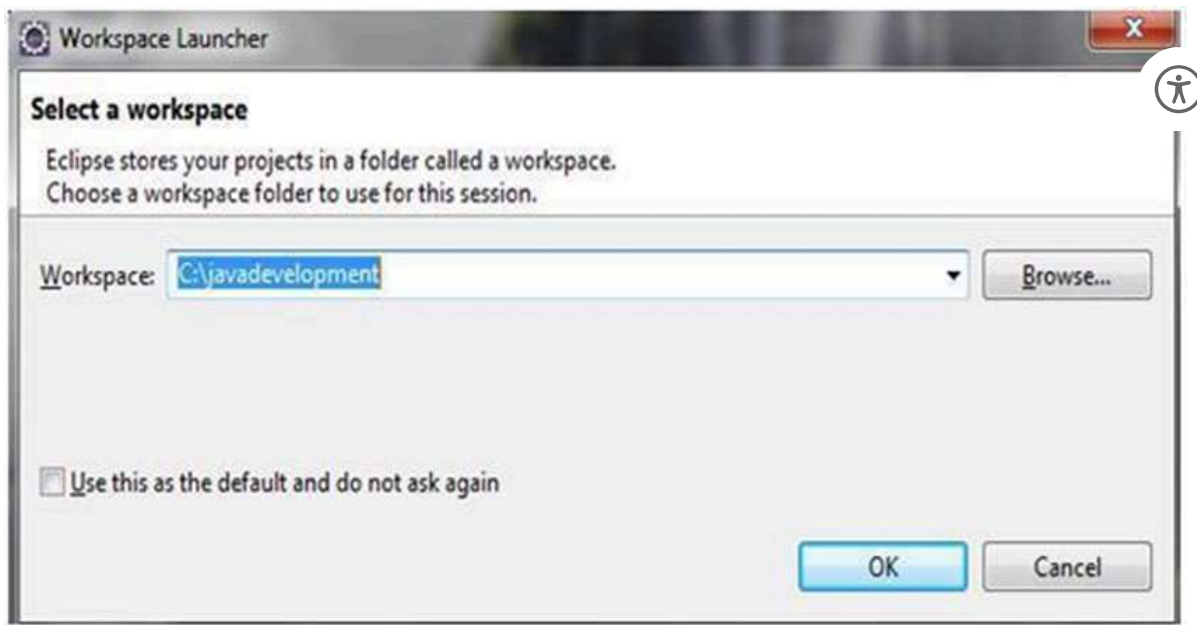
## Compilando, Executando e Depurando Programas Java

Vamos criar o primeiro programa usando o Eclipse IDE, que é um produto de código aberto que foi originalmente criado com uma doação substancial de código pela IBM para a comunidade Java e, a partir desse momento, o Eclipse era um produto 100% orientado à comunidade. Ele começou como um IDE para o desenvolvimento de programas Java, mas agora possui diferentes tipos de linguagens de programação, como PHP, Spring, Scala etc.

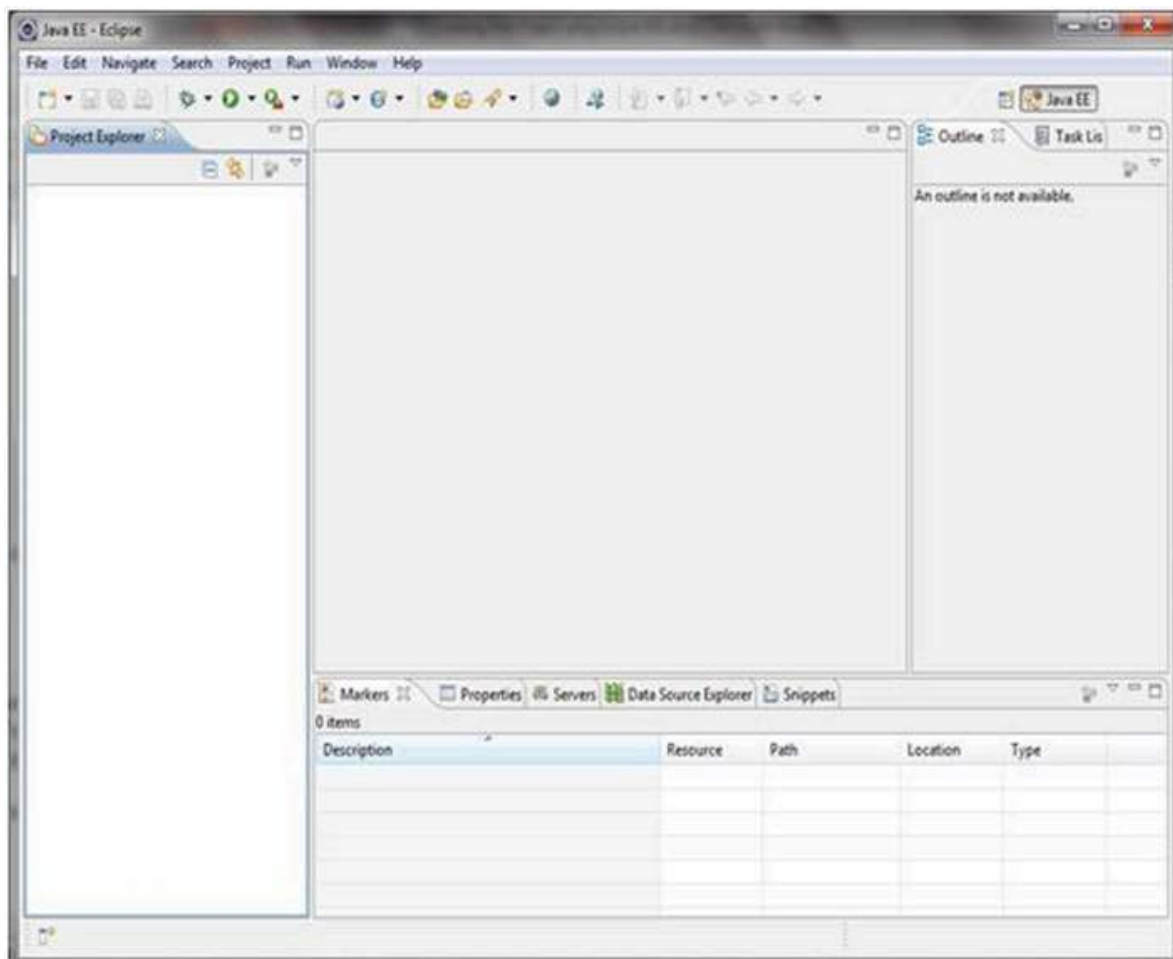
Anteriormente, vimos como fazer o download do eclipse. Após descompactar o eclipse, você pode ver dentro da pasta o arquivo exe. Clique duas vezes em eclipse.exe (você também pode criar um atalho na área de trabalho).

A nova caixa de diálogo na figura abaixo será aberta para criar ou abrir a área de trabalho (Área de Trabalho é a pasta / Diretório em que todos os projetos java residirão). Usaremos o diretório "C: \ javadevelopment" para a área de trabalho. Após selecionar a área de trabalho, clique em ok e você poderá ver a página de boas-vindas do eclipse, que pode ser fechada clicando em "X".





A Figura abaixo é uma captura instantânea do IDE recém-instalado com um espaço de trabalho vazio.

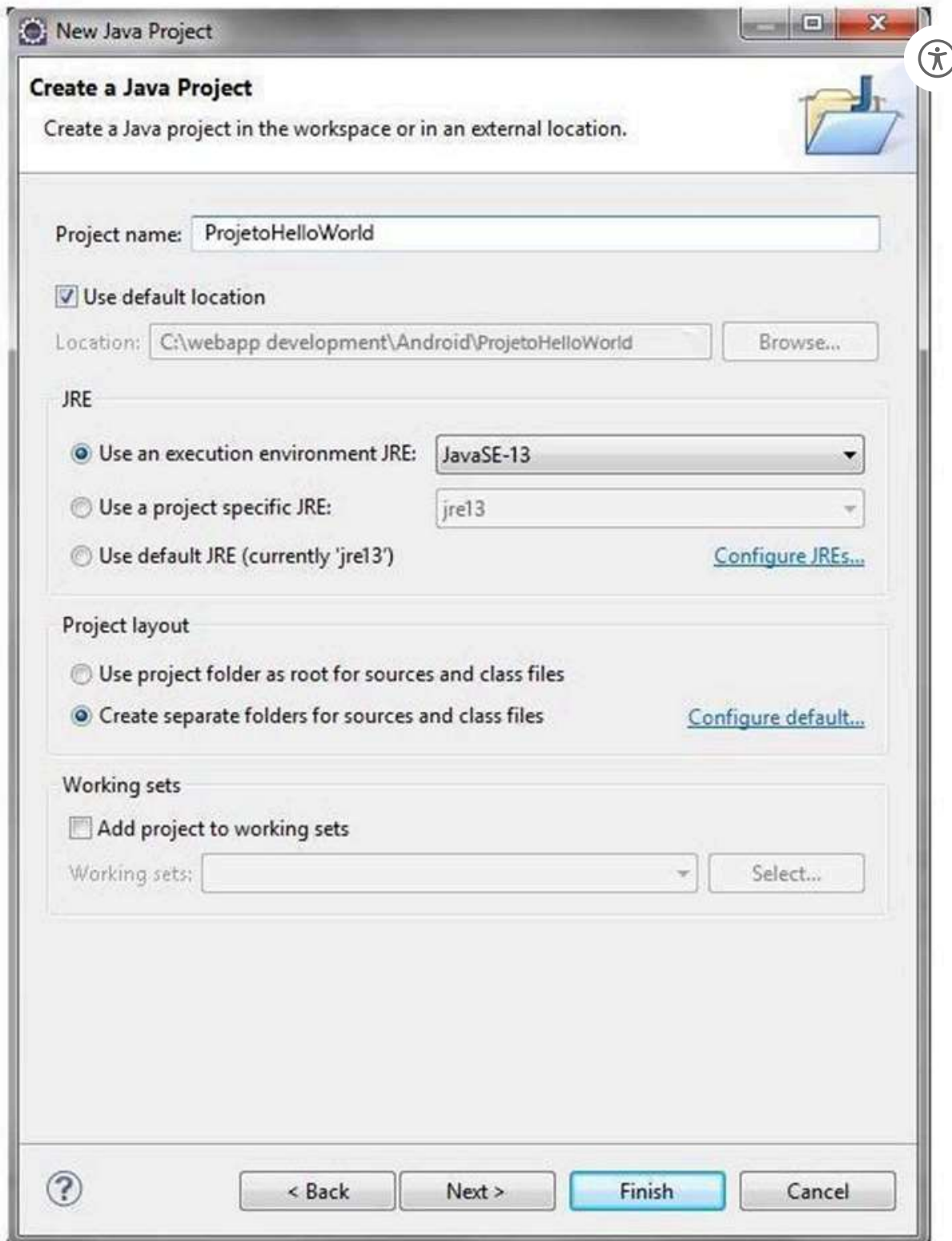


Para ser mais preciso, vemos a perspectiva do Java EE (observe a guia Java EE na parte superior), que é uma coleção de visualizações. À esquerda, você vê a visualização do Project Explorer. A área no meio é reservada para a visualização do editor de código - você começa a usá-la assim que cria sua primeira classe Java. A visualização Estrutura de tópicos está à direita - você verá os nomes de suas classes, métodos e variáveis quando estiverem disponíveis.

Existem muitas outras visualizações que você pode abrir e fechar sozinho selecionando Janela ⇌ Mostrar Visualização. Isso inclui console, pesquisa, servidores e outros.

Vamos começar criando o primeiro projeto java chamado ProjetoHelloWorld

**Etapas 1:** Selecione Arquivo -> Novo -> Projeto Java e digite ProjetoHelloWorld como o nome do projeto na janela pop-up, conforme mostrado na Figura abaixo.

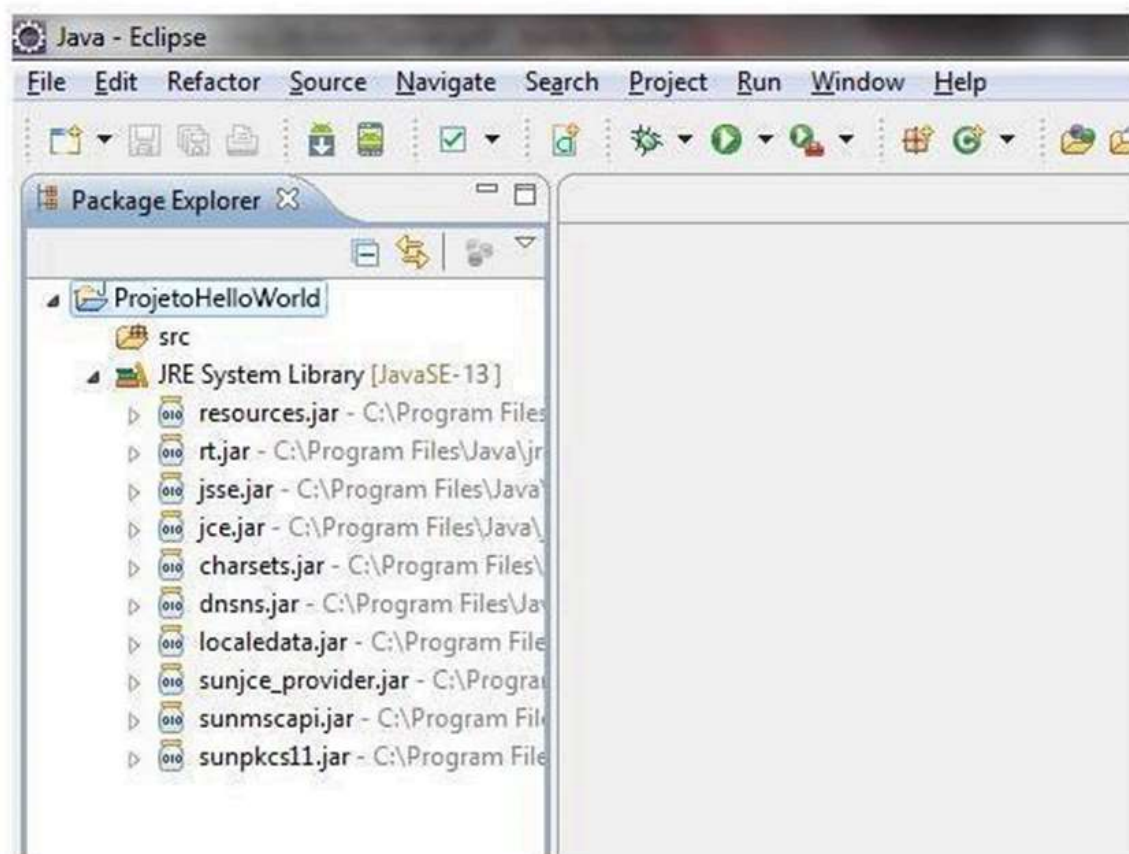


Você pode selecionar a versão do JRE com a qual deseja trabalhar. O Eclipse pode compilar o código com qualquer versão do JRE instalada selecionada por você.

Clique em Avançar> Você será solicitado a escolher uma pasta para manter as classes compiladas (referidas na Figura abaixo). Continuaremos com a seleção padrão e clique em Concluir.

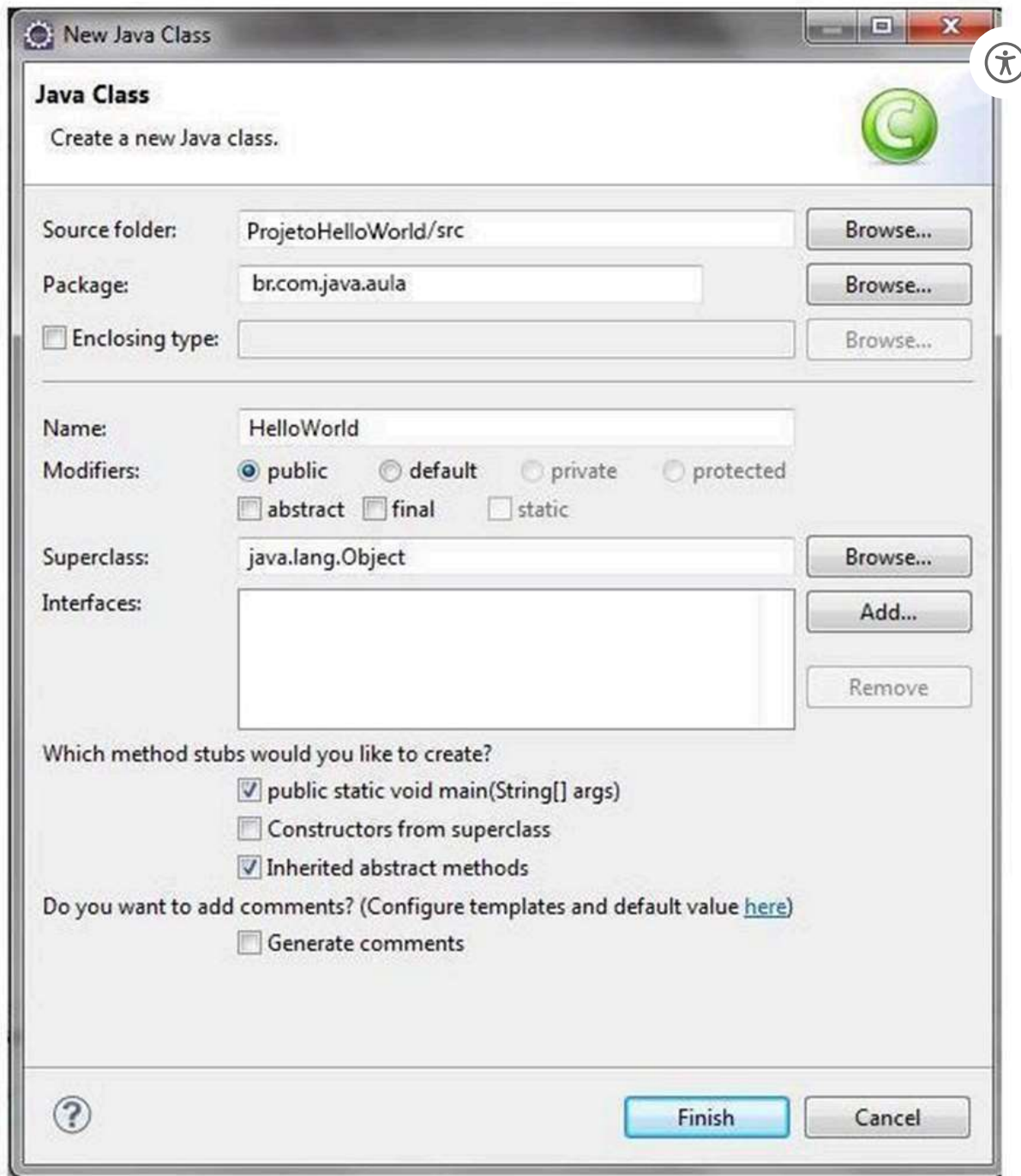


Este projeto possui uma pasta vazia, src - você pode salvar o código fonte do HelloWorld.java quando estiver pronto. A pasta JRE contém todas as bibliotecas necessárias que suportam a JVM na qual o HelloWorld será executado.



**Etapa 2:** Crie o programa Java “HelloWorld” dentro do pacote `br.com.java.aula` (você pode decidir o nome do pacote ou o nome da classe conforme sua escolha).

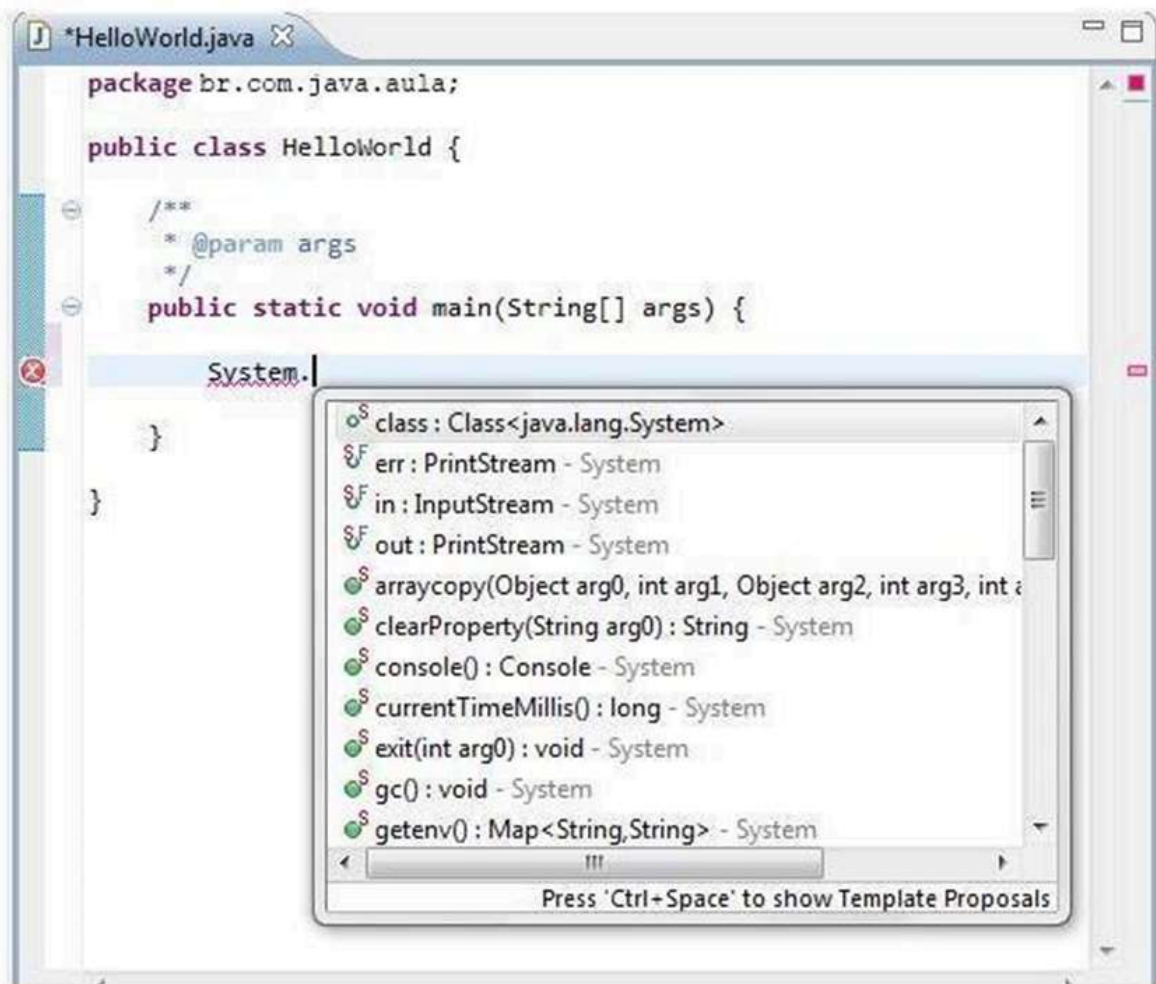
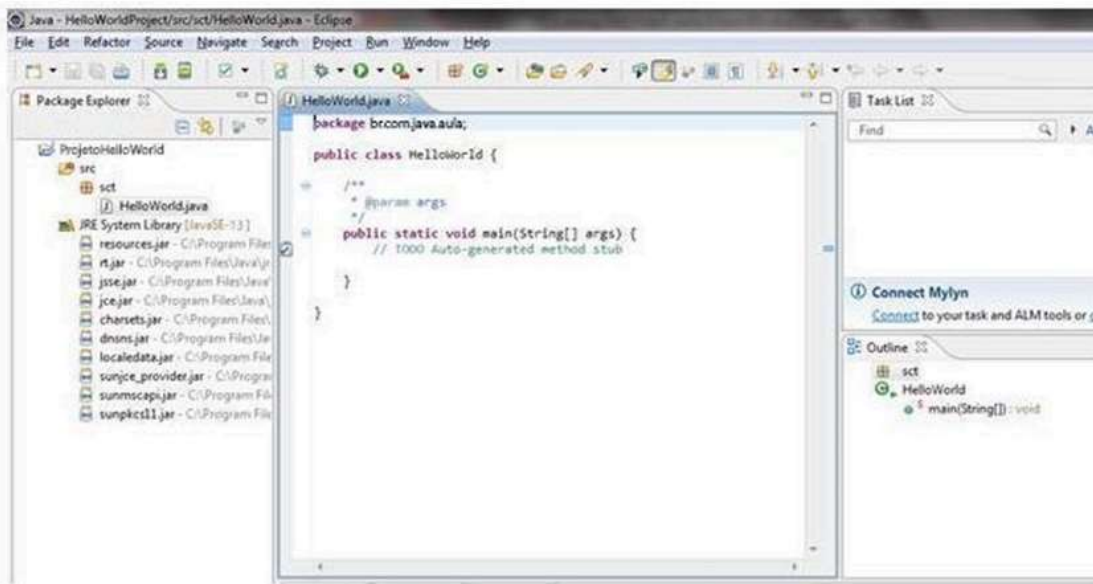
Selecione **Arquivo > Novo > Classe** e digite HelloWorld no campo Nome e `br.com.java.aula` no campo Pacote, conforme mostrado na Figura referida abaixo. Marque também a caixa de seleção do método principal, que criará o método principal automaticamente após clicar no botão Concluir.



Agora você poderá ver a tela onde você pode copiar e colar nosso conteúdo da classe HelloWorld ou apenas uma linha no método main (). Se você tentar escrever a declaração, o IDE o ajudará com as opções disponíveis, conforme mostrado apenas na Figura abaixo. Dessa forma, podemos reduzir o erro de sintaxe de programação.

```
System.out.println ("Hello World! Oi mundo");
```





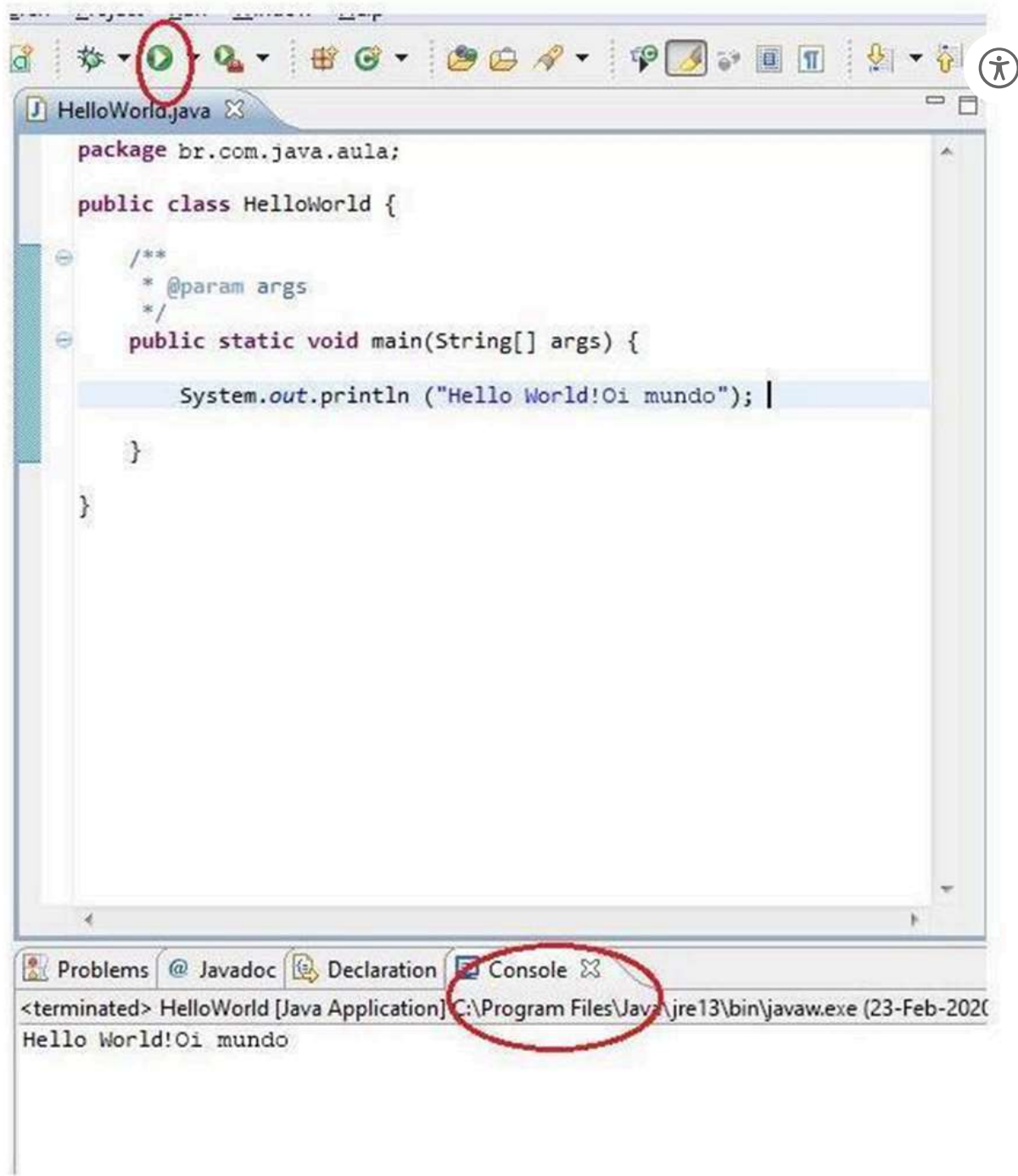
Quando terminamos de escrever nosso código, podemos salvar o arquivo pressionando a pequena imagem de disquete na barra de ferramentas ou usando a combinação de teclas Ctrl + S. Por padrão, salvar o código resulta em uma

chamada do compilador. Se você não cometeu nenhum erro de sintaxe, o Eclipse criará HelloWorld.class no diretório bin do ProjetoHelloWorld. Se houver algum erro de compilação será mostrado como sublinhado vermelho e marca de cruz vermelha, como mostrado na Figura mencionada acima.

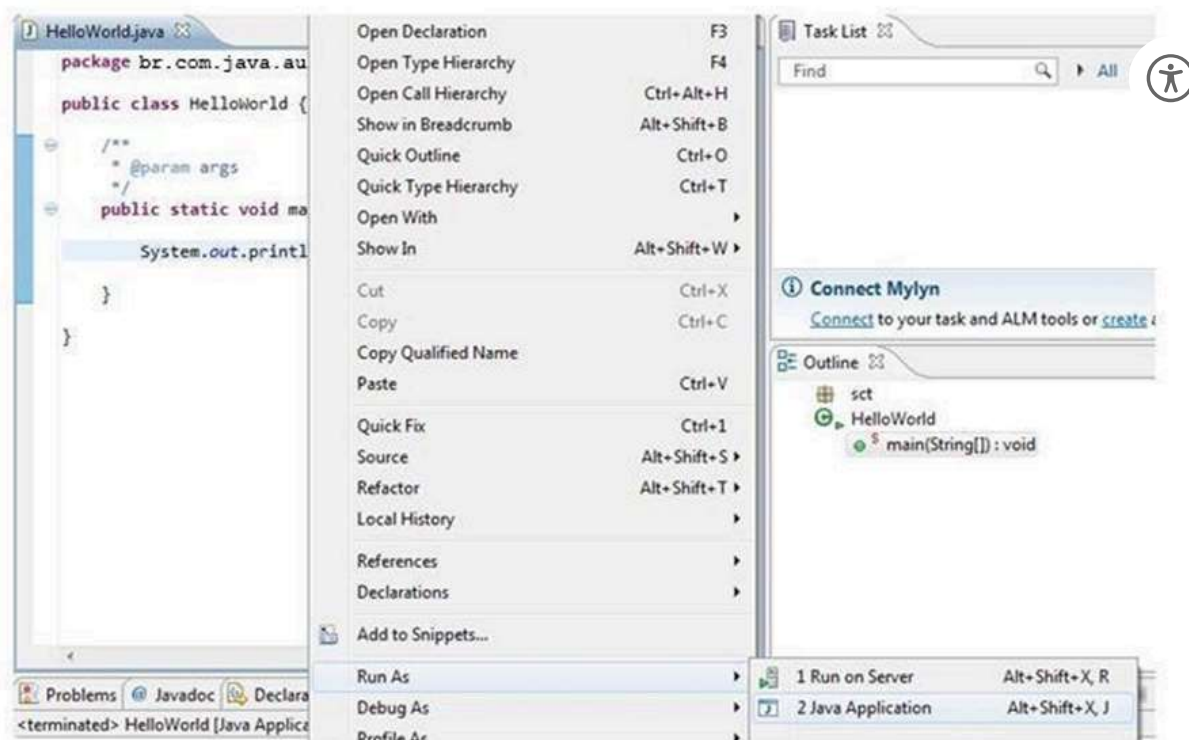
### **Etapas 3:** Executando o programa Java:

Agora você pode executar o programa pressionando o botão verde redondo na barra de ferramentas. A saída do programa será mostrada no painel de visualização do console na parte inferior do Eclipse IDE.





Outra maneira de executar o programa java: clique com o botão direito do mouse no espaço em branco do editor e o menu será aberto, com a opção de executar o programa, conforme mostrado na Figura abaixo.



## Atividade extra

Vídeo: “Ferramentas Free de desenvolvimento”

Link: <https://www.youtube.com/watch?v=RDMhOjA8hNo>

## Referências Bibliográficas

BARNES, D. J. KOLLING, M. **Programação orientada a objetos com java: uma introdução prática usando o bluej**. 4.ed. Pearson: 2009.

FELIX, R. (Org.). **Programação orientada a objetos**. Pearson: 2017.



MEDEIROS, L. F. de. **Banco de dados: princípios e prática**. Intersaberes: 2013;

ORACLE. Java Documentation, 2021. **Documentação oficial da plataforma Java**. Disponível em: < <https://docs.oracle.com/en/java/> >.

**Ir para exercício**