



Domain Driven Design

PROF. Rafael Martins Ronqui
profrafael.ronqui@fiap.com.br

1. Sintaxe do Java

AGENDA

- O que é Java?
- IDE para programar em Java que iremos utilizar
- Inserir comentários na linguagem Java
- Primeiro Olá mundo em Java
- Primeiro projeto (Passo a passo, para criar um projeto no eclipse com 13 steps)
- Explicação da sintaxe básica:
 - class
 - metodo
 - public, static e void
 - main e args[]
 - public e System.out.println
- Regras para nomeação de classes
- Variáveis
- Constantes



- É uma linguagem de programação que foi criada pela Sun Microsystems no ano de 1995.
- Exemplo de utilização dessa linguagem são os serviços bancários online, os vários aplicativos da Receita Federal brasileira como o programa de declaração de imposto de renda (desktop e não web).

- IDE que iremos utilizar: Eclipse
 - Mas o que é IDE?
 - I → Interface/Intergrated; / Interface
 - D → Development; / Desenvolvimento
 - E → Environment / Ambiente
 - IDEs para Java:
 - **Eclipse** → Foco no Java, porém também é possível desenvolvimento para outras linguagens.
 - NetBeans → Foco no Java, porém também é possível desenvolvimento para outras linguagens.
 - Genexus → Ferramenta para desenvolvimento com a metodologia de base de conhecimento, é possível o desenvolvimento em Java e até mesmo banco de dados.
 - IntelliJ IDEA → IDE da JetBrains, desenvolvimento em diversas linguagens, com foco no Java.
 - Android Studio → IDE do google para desenvolvimento na plataforma Android.

Inserir comentários na linguagem Java

Mais utilizado:

// Comentário de uma linha

Segundo mais utilizado:

/* Comentário de várias
linhas */

Terceiro mais utilizado, é utilizado para gerar a documentação do código, gerado pelo javadoc, ou podemos acrescentar esse tipo de comentário para ajudar na documentação por exemplo do porquê da criação de determinada classe:

/*Comentário
*de várias linhas
*que geralmente é utilizado
*no início do código para explicar qual
*é o objetivo do código */

- Exemplo:

- Exemplo:

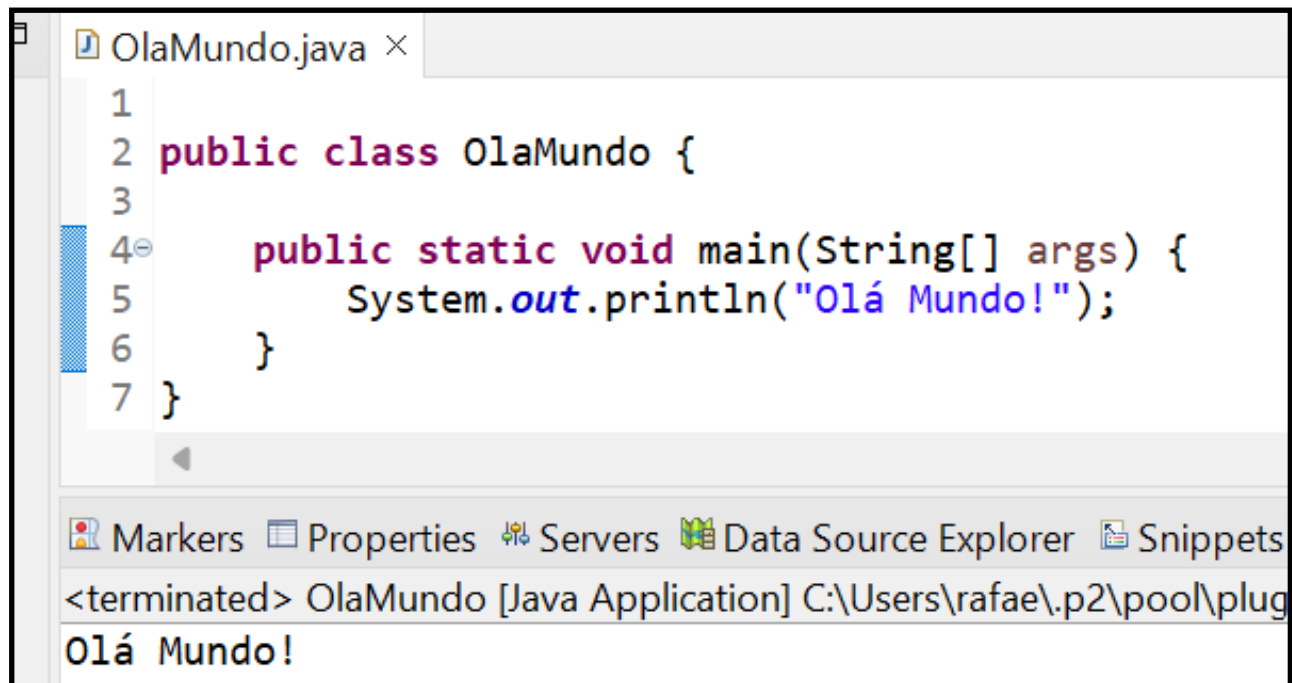
```
1  /**
2   * Classe utilizada para demonstrar o uso de comentários.
3   *
4   * Depois que uma classe Java é compilada, os comentários não vão
5   * para os códigos compilados .class, dessa forma os comentários
6   * não interferem no tamanho final do projeto.
7   */
8  public class ExemploComentario {
9      public static void main(String[] args) {
10         //Imprime uma mensagem para o usuário.
11         System.out.println("Demonstrando o uso dos comentários.");
12
13         /* Podemos utilizar esta forma quando queremos escrever um
14            comentário mais longo para exemplificar o código. */
15     }
16 }
```

Primeiro Olá mundo em Java:

- Exemplo:

Iremos explicar
cada linha,
nos próximos slides

```
public class OlaMundo {  
    public static void main(String[] args) {  
        System.out.println("Olá Mundo!");  
    }  
}
```



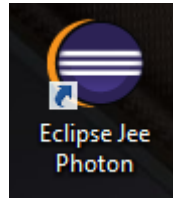
The screenshot shows an IDE window titled 'OlaMundo.java'. The code is as follows:

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

Below the code editor, the 'Markers' tab is active, showing the output of the program:

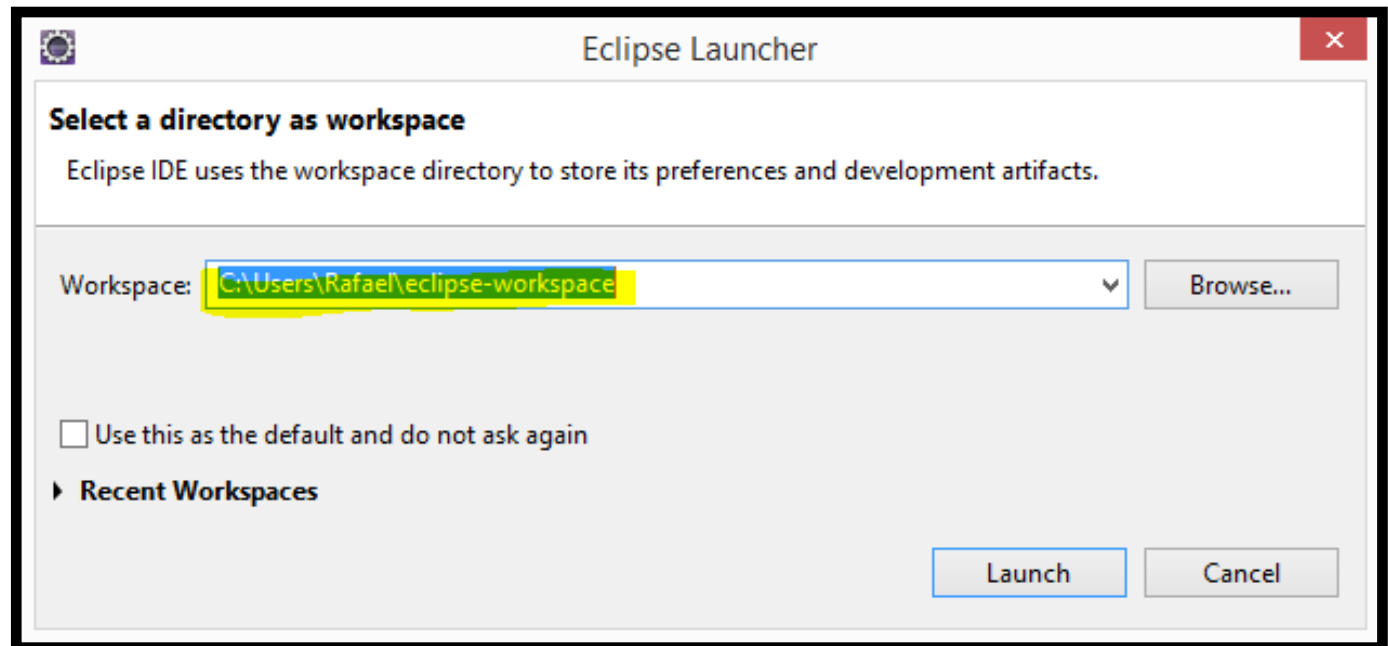
```
<terminated> OlaMundo [Java Application] C:\Users\rafae\.p2\pool\plug  
Olá Mundo!
```


Primeiro projeto



Step 1: Abrir o eclipse:

Step 1: Obs.: Quando o eclipse abre, conforme tela abaixo, ele indica aonde serão armazenados os projetos.

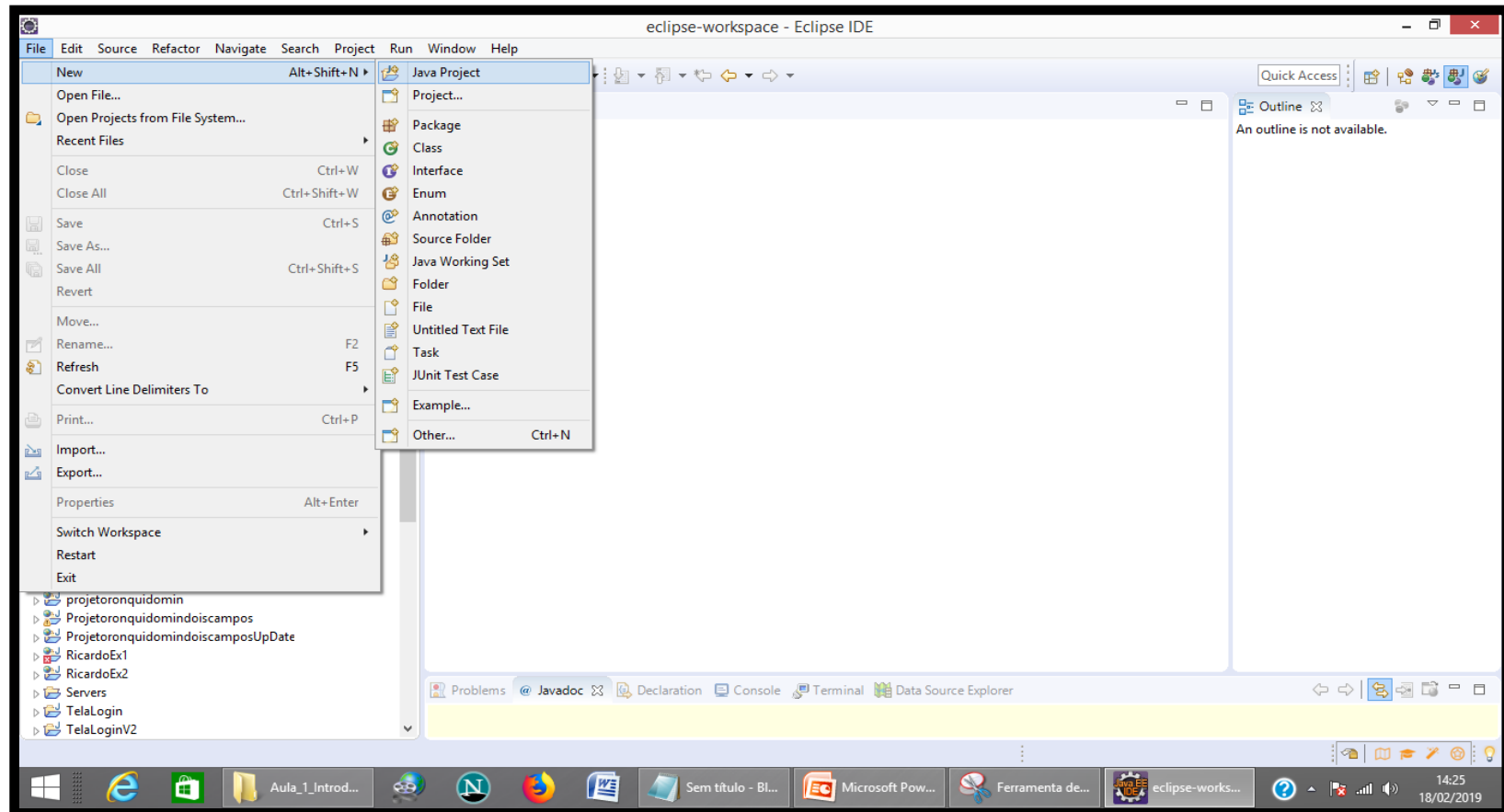


Primeiro projeto

Step 2: Clicar em File;

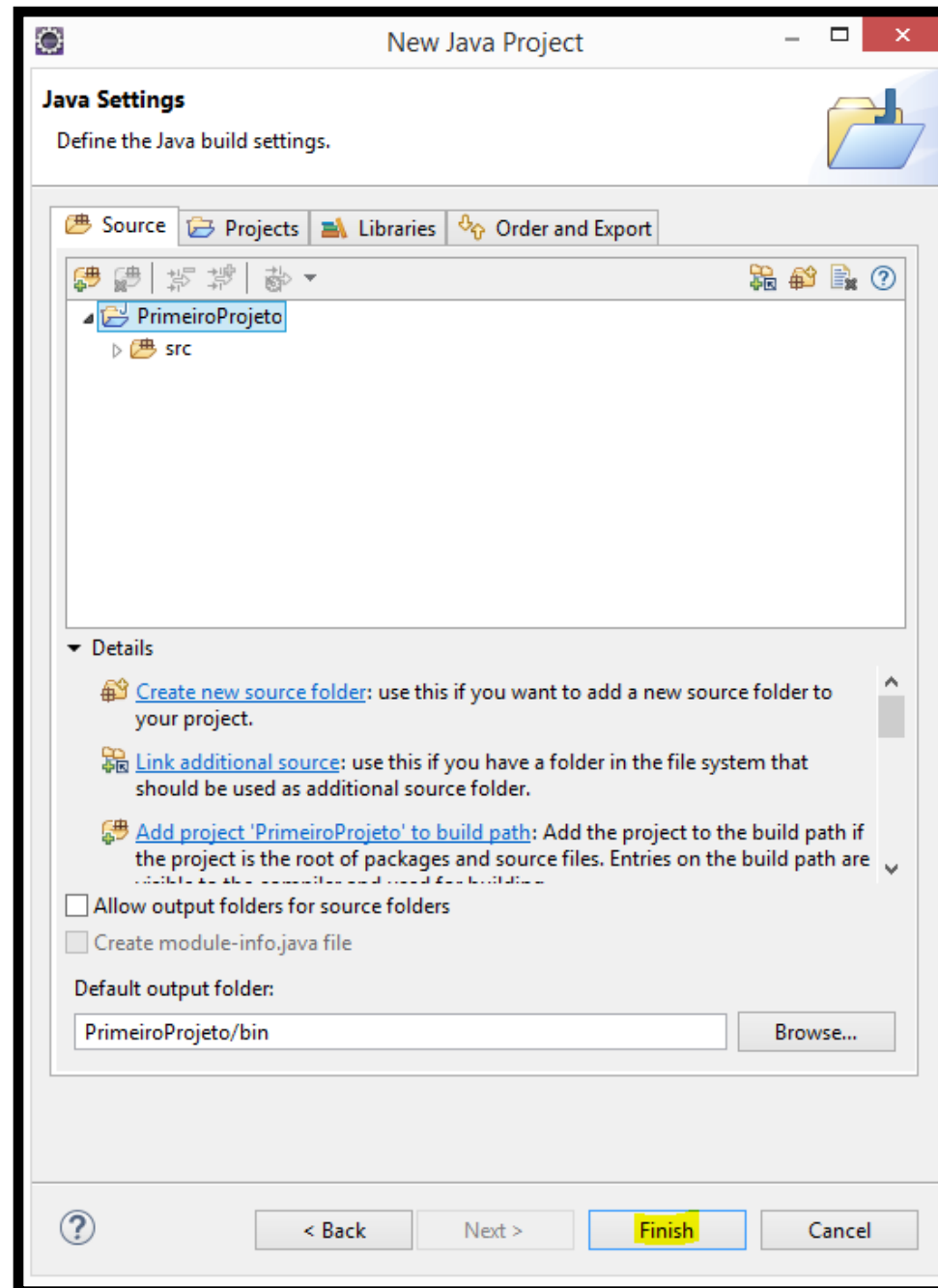
Step 3: New project:

Caso não apareça a opção Java Project clicar em Other e após escrever Java Project



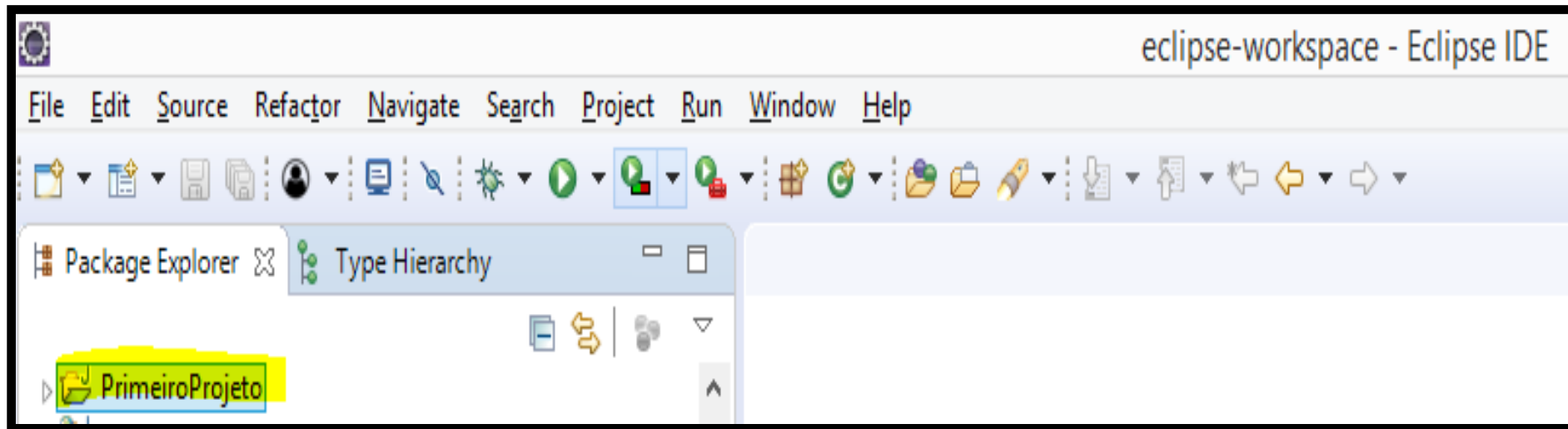
Primeiro projeto

Step 4: Clicar em Finish:



Primeiro projeto

Conforme abaixo, o projeto foi criado.

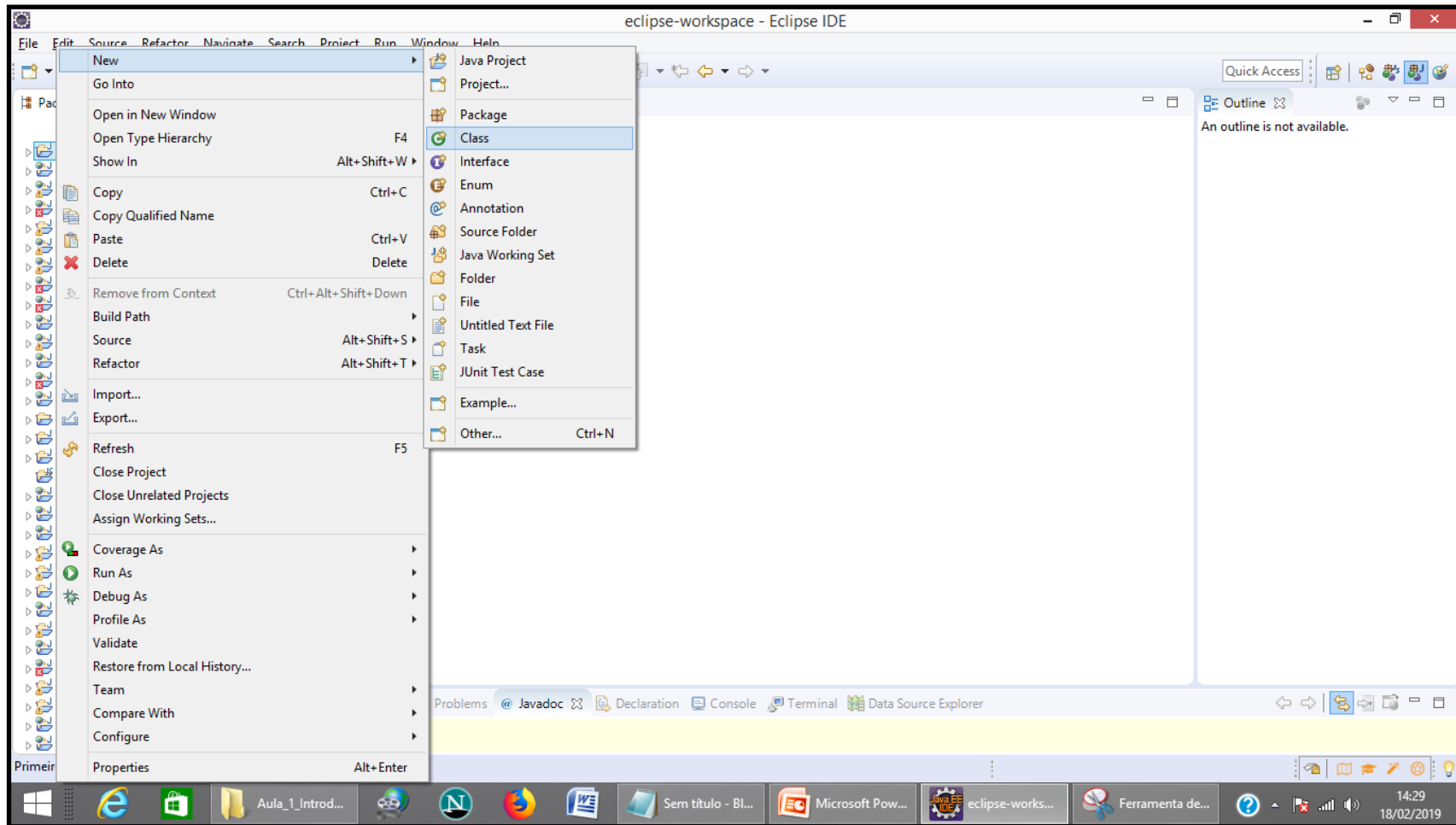


Primeiro projeto

Step 5: Criando uma classe, clicar com o botão direito no projeto;

Step 6: New;

Step 7: Class:



Primeiro projeto

Step 8: Colocar o nome da classe no campo “Name”;

Step 9: Flegar o campo “public static void (String[] args);

Step 10: Clicar em Finish:

New Java Class

Java Class

⚠ The use of the default package is discouraged.

Source folder: PrimeiroProjeto/src Browse...

Package: (default) Browse...

☐ Enclosing type: Browse...

Name: PrimeiroProjeto

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

Superclass: java.lang.Object Browse...

Interfaces: Add...
Remove

Which method stubs would you like to create?

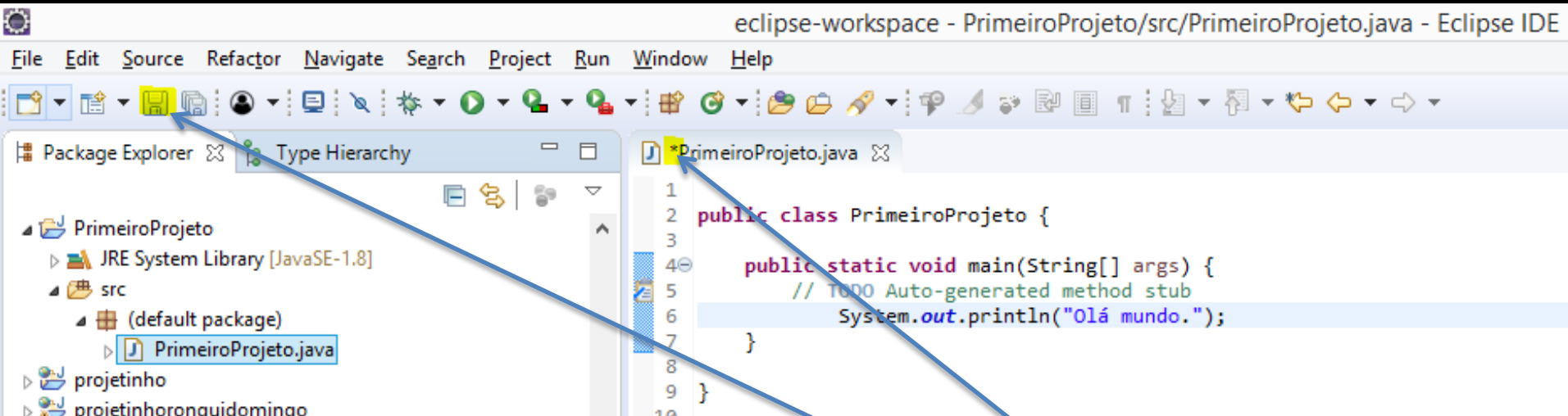
☒ public static void main(String[] args)
☐ Constructors from superclass
☒ Inherited abstract methods

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

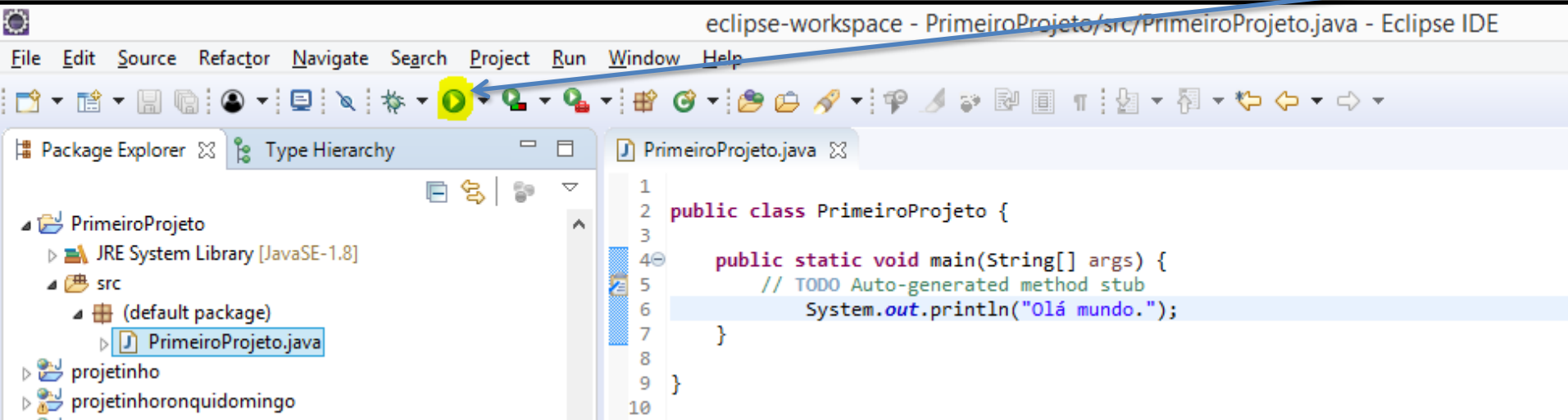
? Finish Cancel

Primeiro projeto

Step 11: Digitar o código abaixo, conforme o lado direito da imagem:

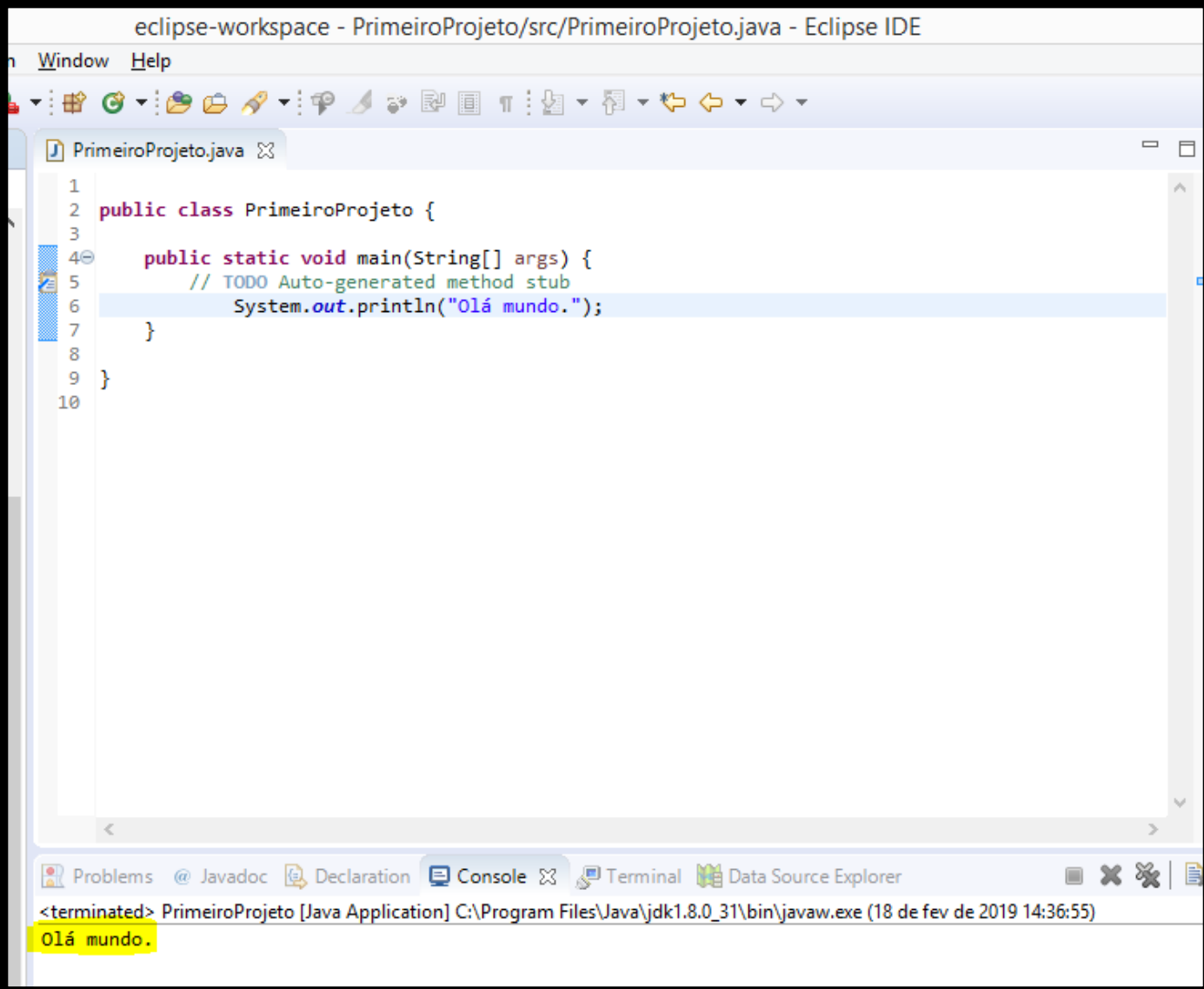


Step 12: Após digitar o código, veja na imagem acima do lado direito, como tem o sinal *, significa que o projeto ainda não foi salvo, então clicar para salvar, após, clicar no “play”.



Primeiro projeto

Step 13: Após clicar no “Play”, irá exibir a frase “Olá mundo”, que está abaixo em amarelo.



The screenshot shows the Eclipse IDE interface. The main editor window displays the file `PrimeiroProjeto.java` with the following code:

```
1 public class PrimeiroProjeto {  
2  
3  
4     public static void main(String[] args) {  
5         // TODO Auto-generated method stub  
6         System.out.println("Olá mundo.");  
7     }  
8  
9 }  
10
```

The line `System.out.println("Olá mundo.");` is highlighted in blue. The bottom of the IDE shows the `Console` tab with the output:

```
<terminated> PrimeiroProjeto [Java Application] C:\Program Files\Java\jdk1.8.0_31\bin\javaw.exe (18 de fev de 2019 14:36:55)  
Olá mundo.
```

The output `Olá mundo.` is highlighted in yellow.

Dúvidas...



FIM