

INSTITUTO MAUÁ DE TECNOLOGIA



# Linguagens I

Introdução ao Java

Prof. Tiago Sanches da Silva

Prof. Murilo Zanini de Carvalho

# O que é Java?



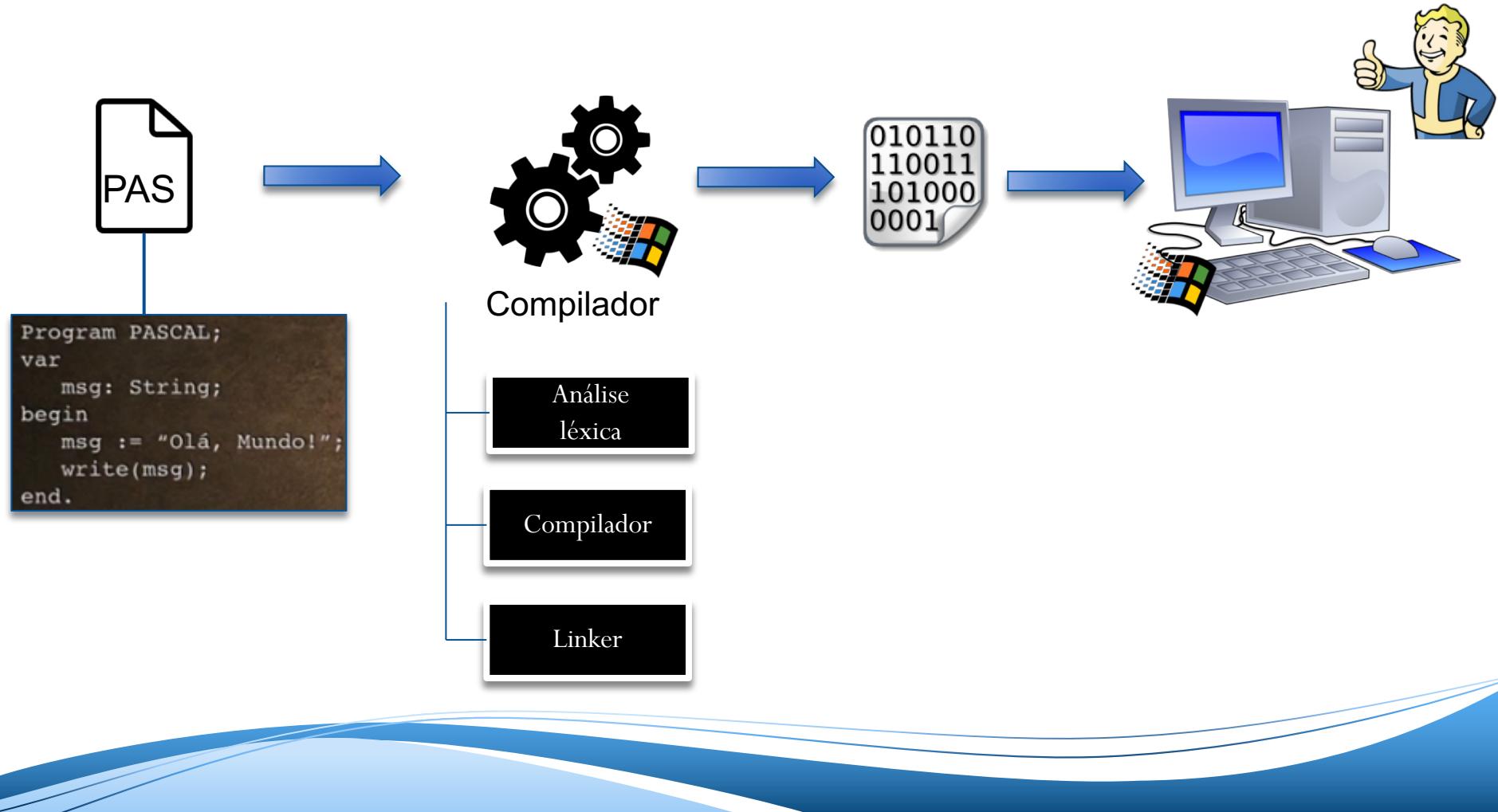
# O que é Java?

Originalmente desenvolvida por uma equipe de desenvolvedores liderada por **James Gosling** na Sun Microsystems (atualmente de propriedade da Oracle) e lançada em 1995, o Java é uma linguagem de programação orientada a objetos que atualmente faz parte do núcleo da Plataforma Java.



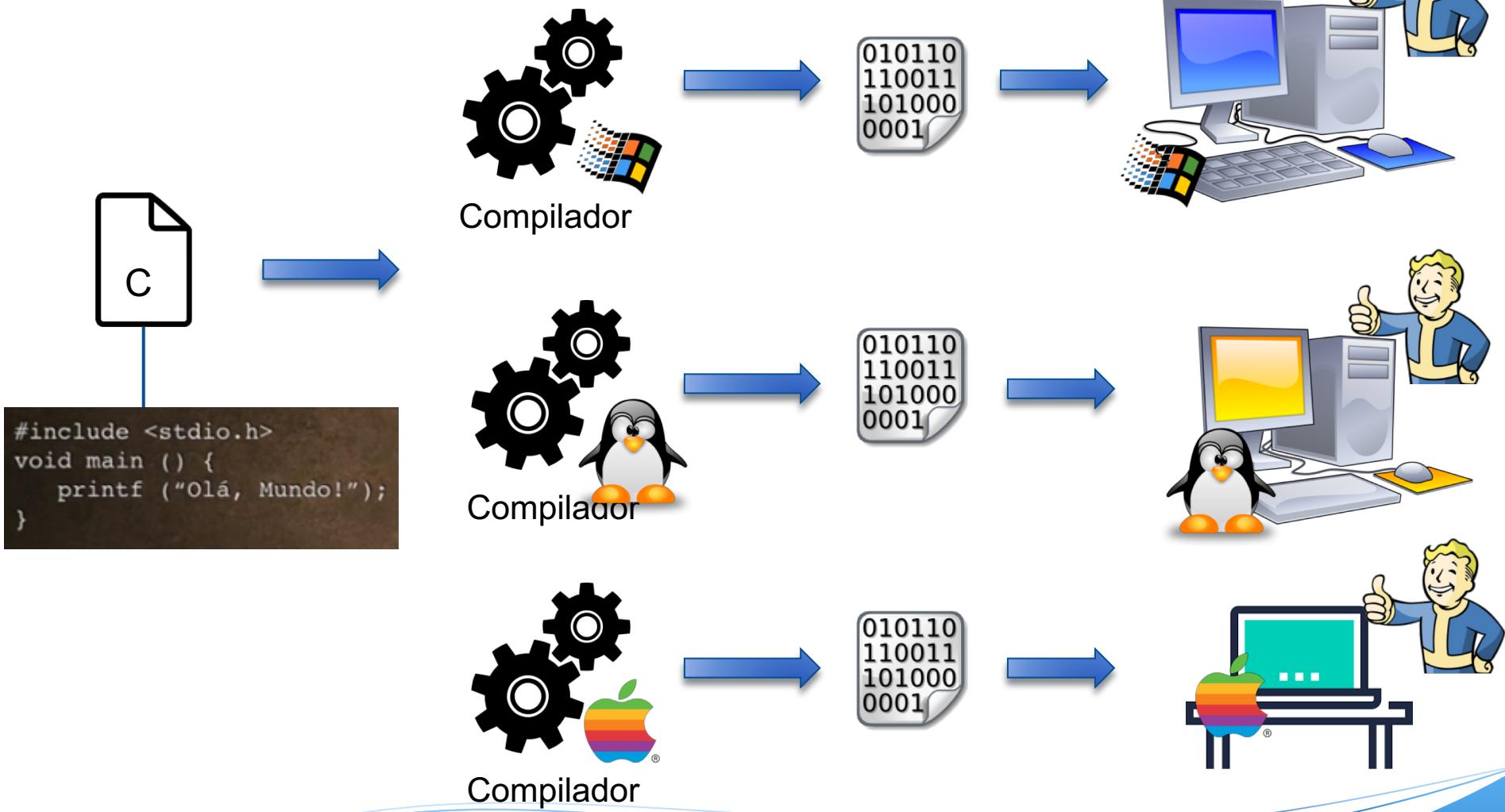
# O que é Java?

Apenas uma plataforma

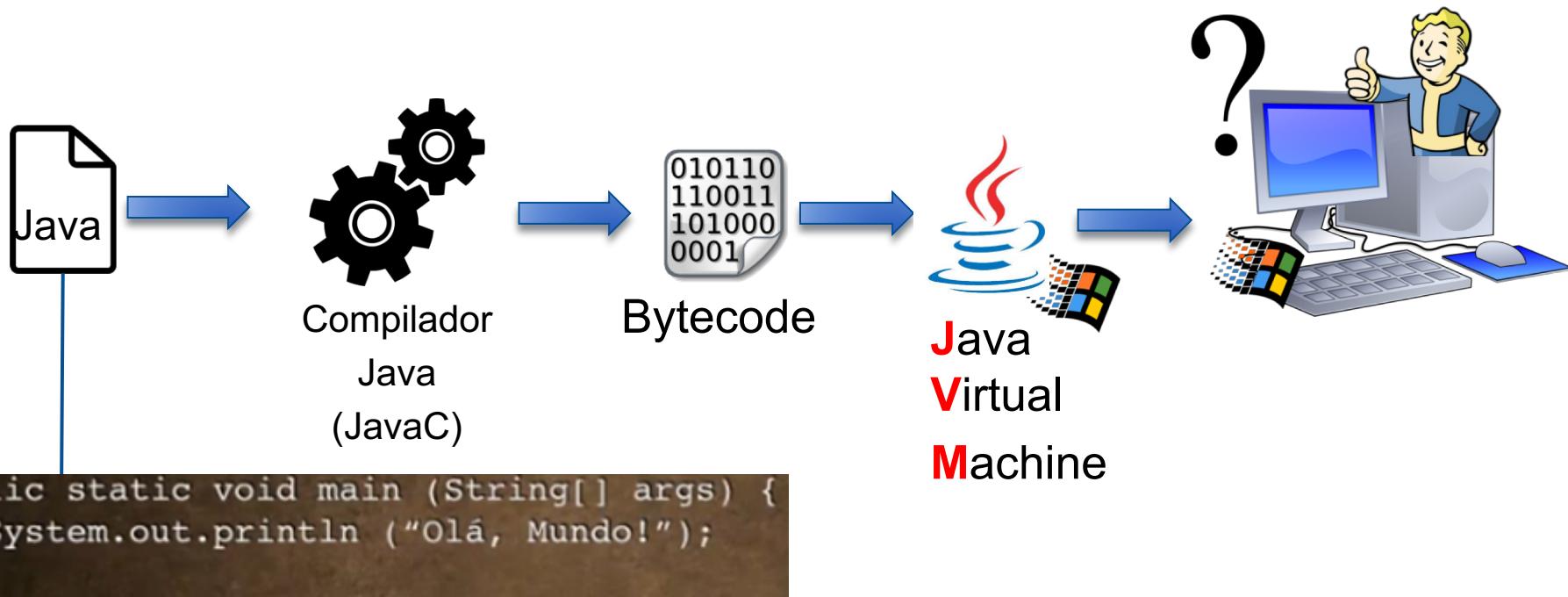


# O que é Java?

## Compilador para várias plataformas

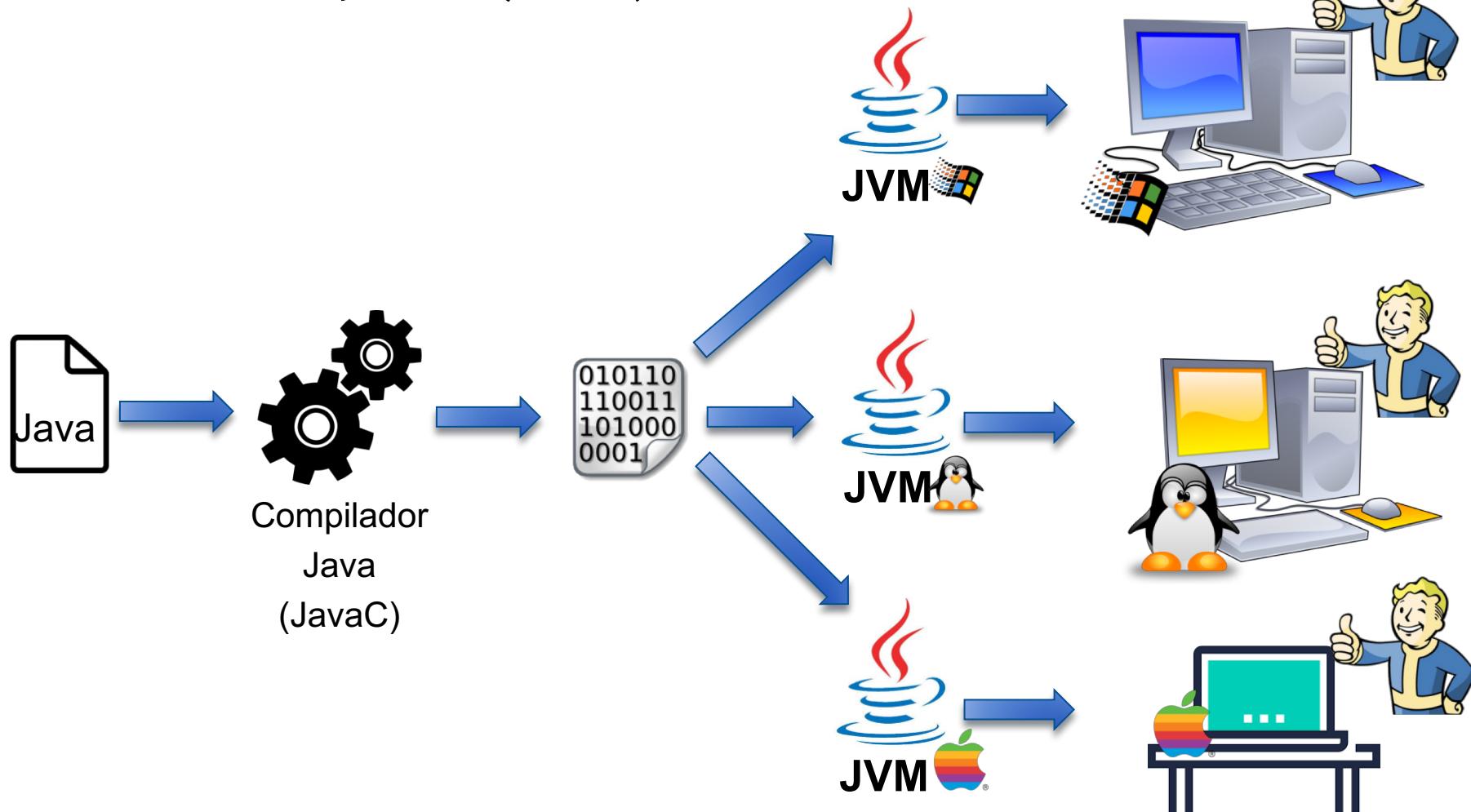


# O que é Java?



# O que é Java?

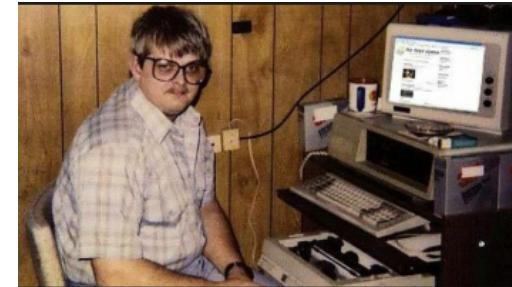
“Write once, run anywhere. (WORA)”



# Plataforma Java



## JRE vs. JDK



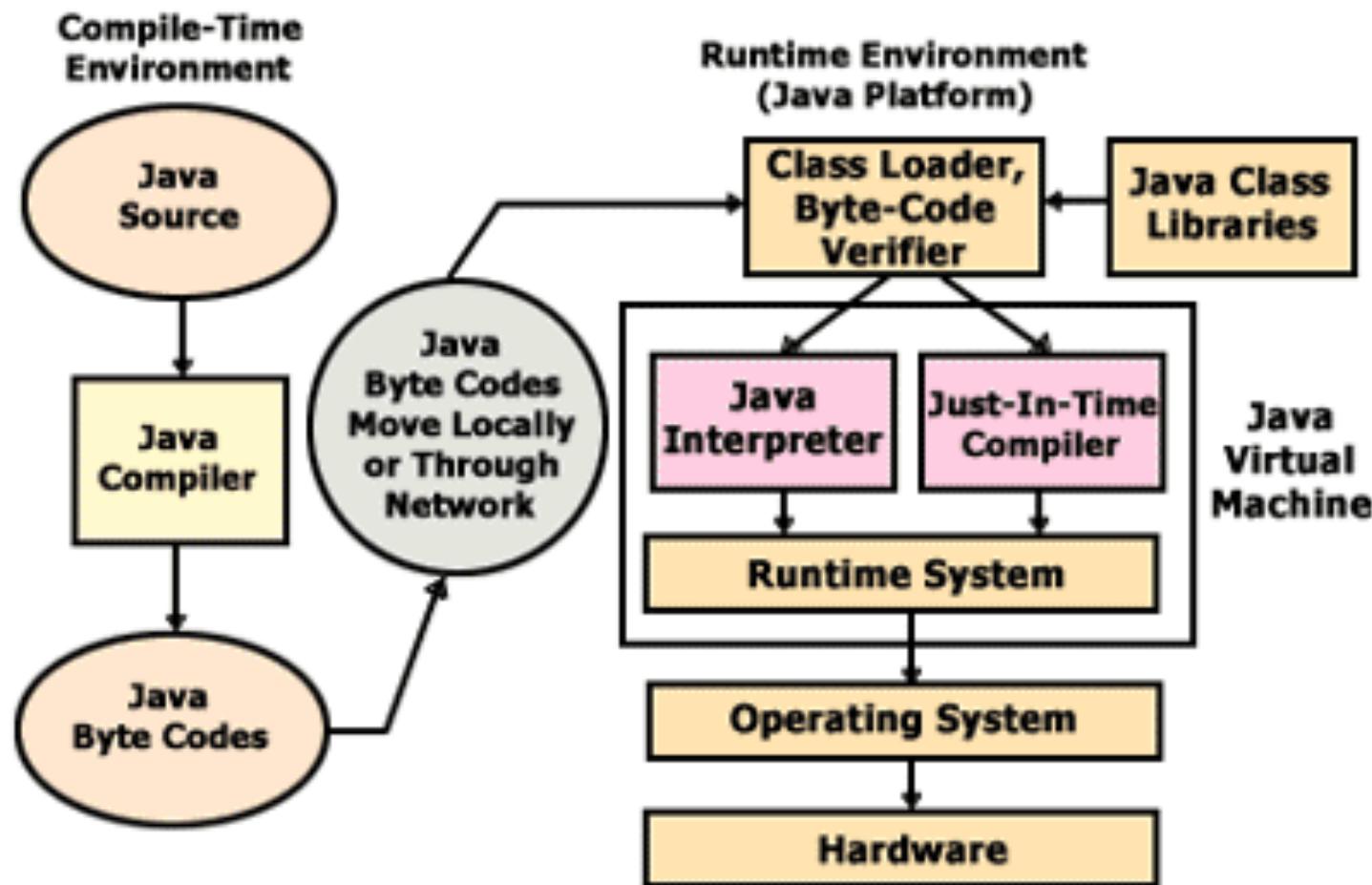
Java Runtime Environment (JRE)



Java Development Kit (JDK)



# Plataforma Java



# **Configure seu ambiente**

# Configuração do ambiente de trabalho

Crie um repositório no seu GitHub: “Linguagens1\_Projetos”;

Configure o Gitbash (user.name/user.email);

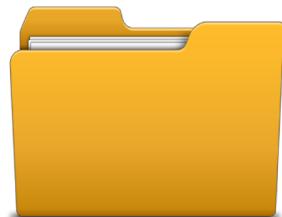
Entre na sua pasta local no computador e clone o repositório criado;

Dentro do repositório **local** crie uma nova pasta chamada “pratica1”;  
Essa será a pasta de trabalho para esse primeiro dia de pratica;

Dentro da pasta “pratica1” serão criadas novas pastas segundo necessidade, uma pra cada exercício. **Por exemplo**, se no dia de hoje tiverem 3 exercícios diferentes, dentro da pasta “pratica1” então deve conter 3 pastas nomeadas da seguinte forma: “exercicio1”, “exercicio2” e “exercicio3”.

Nova semana, nova pasta pratica!

# Configuração do ambiente de trabalho



RA\_Nome

[Pasta do aluno no computador local]



Linguagens1\_Projetos



pratica1



exercicio1



exercicio2



exercicio3

# Exercício 1

Abra um arquivo e responda as seguintes perguntas:

1. Em qual pasta o jdk do java está instalado? (não incluir a pasta \bin nesta etapa)
2. E qual o caminho completo até os executáveis javac e java, que foram instalados do computador?
3. Como compilamos um programa por linha de comando utilizando o javac?
4. E como executamos este programa?
5. O que são variáveis do ambiente? Para que elas servem?
6. Qualquer um pode criar uma variável de ambiente?
7. Como eu crio um variável de ambiente no Windows/Linux (escolha apenas 1)?
8. O que é JAVA\_HOME? Por que preciso dele?

# Conan Mode



# Exercício 2

Abra o bloco de notas e escreva seu primeiro código em Java.

PS1: “Sim no bloco de notas.”

```
class MeuPrograma {  
    public static void main(String[] args) {  
  
        // Mensagem de saída do sistema  
        System.out.println("Minha primeira aplicação Java!!");  
  
    }  
}
```

PS2: É uma imagem para você não copiar!

Utilizando o console, tente compilar o código.

1. Responda em um arquivo quais arquivos foram gerado(s)?
2. Como podemos executa-lo(s)?

Problemas nesse passo? Chame o professor!

# Exercício 3

Modificar o exercício anterior para que exiba como saída as linhas:

“Olá mundo!”

“Estou programando no modo Conan. :)”

Compile e execute o programa.

Leitura auxiliar: <http://www.devmedia.com.br/system-out-objeto-de-saida-em-java/25240>

PS: Crie a pasta “exercicio3”, mesmo que seja um exercício baseado no anterior.

# Ex. Dirigido 1 – Método main

Para um programa Java executar, é necessário definir um método especial para ser o ponto de entrada do programa, ou seja, para ser o primeiro método a ser chamado quando o programa for executado. O método main precisa ser **public**, **static**, **void** e receber um array de strings como argumento.

```
class MeuPrograma {  
    public static void main String[] args) {  
        // Mensagem de saída do sistema  
        System.out.println("Minha primeira aplicação Java!!");  
    }  
}
```

Método de entrada da aplicação.

Classe de entrada da aplicação. (nome da aplicação)

# Ex. Dirigido 1 – args?

```
class ExemploArgs {  
    public static void main(String[] args) {  
  
        int i;  
        for( i = 0; i < args.length; i++) {  
            System.out.println( args[i] );  
        }  
    }  
}
```

Execução da virtual machine

```
\exemplo_args>java ExemploArgs Aqui vão os argumentos, não esqueça que é um array
```

Argumentos de entrada (args)

Nome da aplicação que quero rodar

arg[0] arg[1] arg[2] ... arg[n-1]

# Ex. Dirigido 1 – args?

```
class ExemploArgs {  
    public static void main(String[] args) {  
  
        int i;  
        for( i = 0; i < args.length; i++) {  
            System.out.println( args[i] );  
        }  
    }  
}
```

```
\exemplo_args>java ExemploArgs Aqui vão os argumentos, não esqueça que é um array
```

Como será a saída?

# Ex. Dirigido 1 – args?

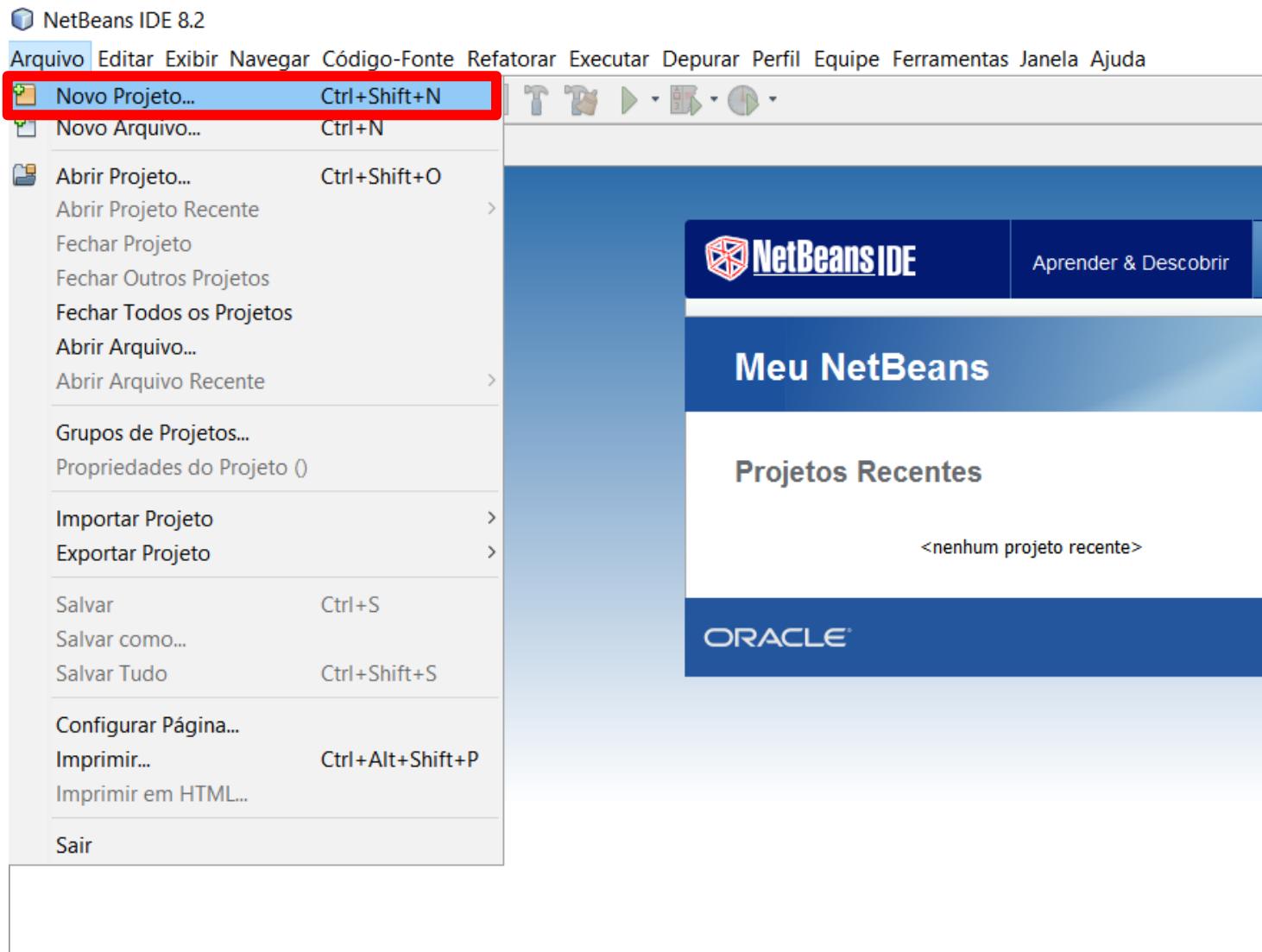
```
class ExemploArgs {  
    public static void main(String[] args) {  
  
        int i;  
        for( i = 0; i < args.length; i++) {  
            System.out.println( args[i] );  
        }  
    }  
}
```

```
\exemplo_args>java ExemploArgs Aqui vão os argumentos, não esqueça que é um array
```

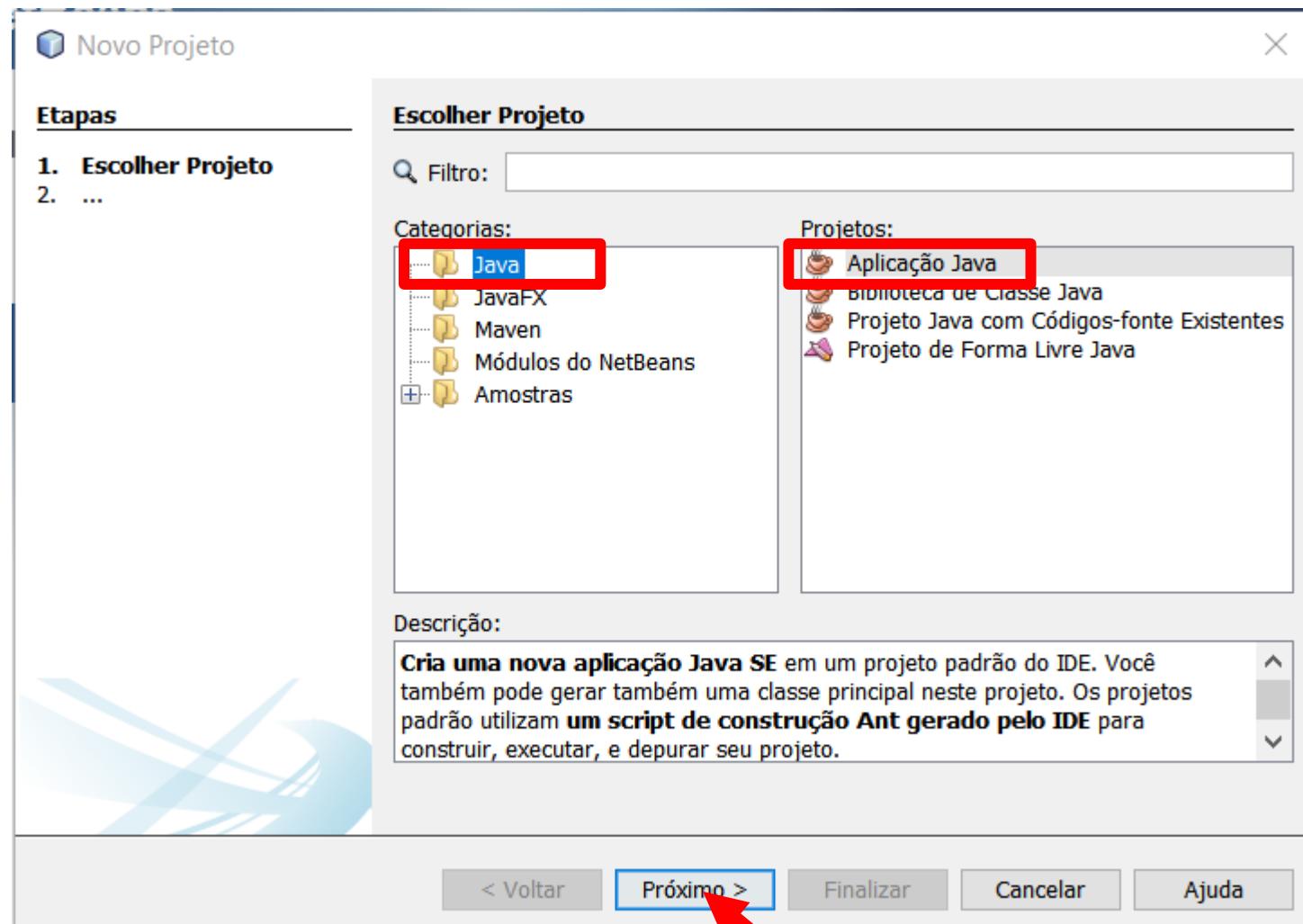
# NetBeans



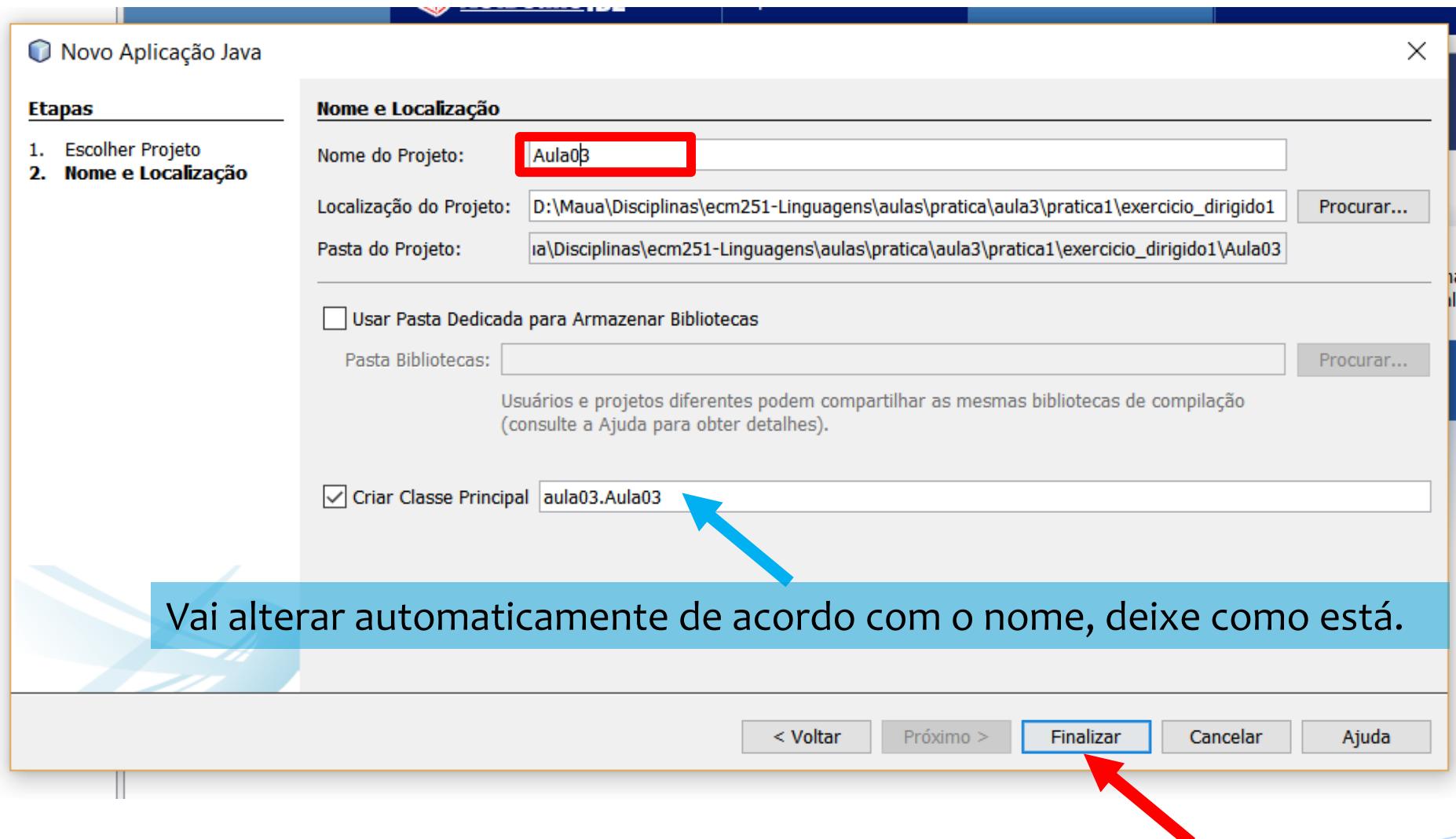
# Ex. Dirigido – Criando uma aplicação Java



# Ex. Dirigido 1 – Criando uma aplicação Java



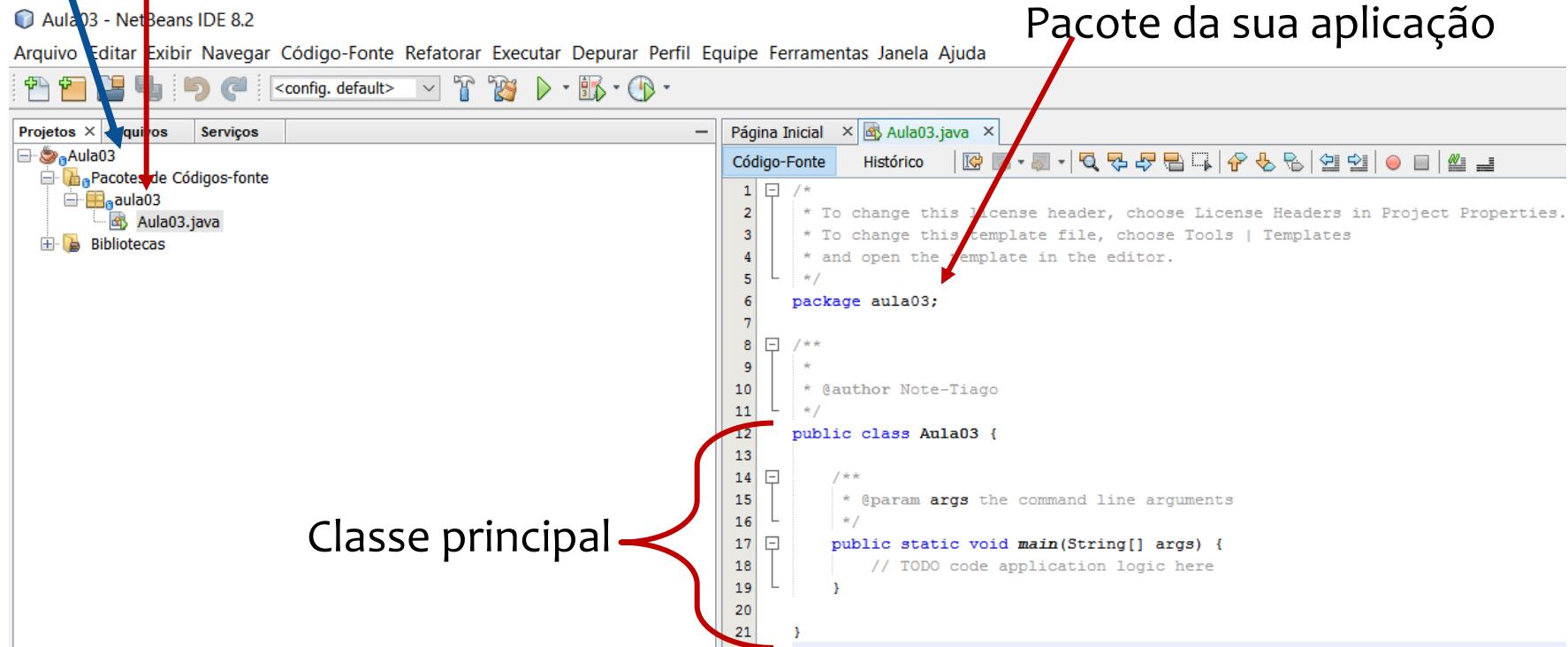
# Ex. Dirigido – Criando uma aplicação Java



# Ex. Dirigido – Criando uma aplicação Java

Seu projeto (nome da aplicação)

Nome do seu pacote, conterá todas classes da sua aplicação



# IntelliJ

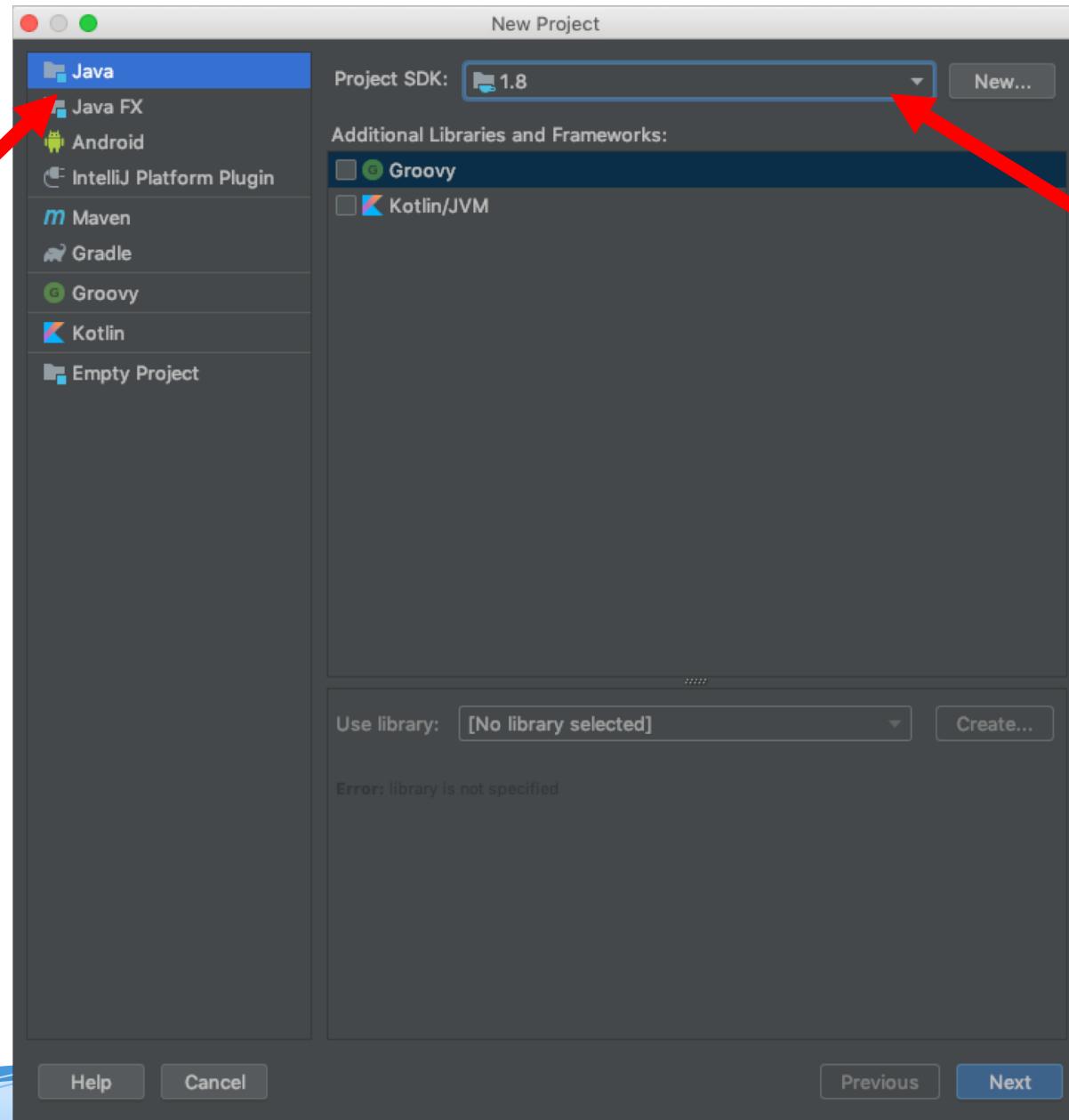


# Ex. Dirigido – Criando uma aplicação Java



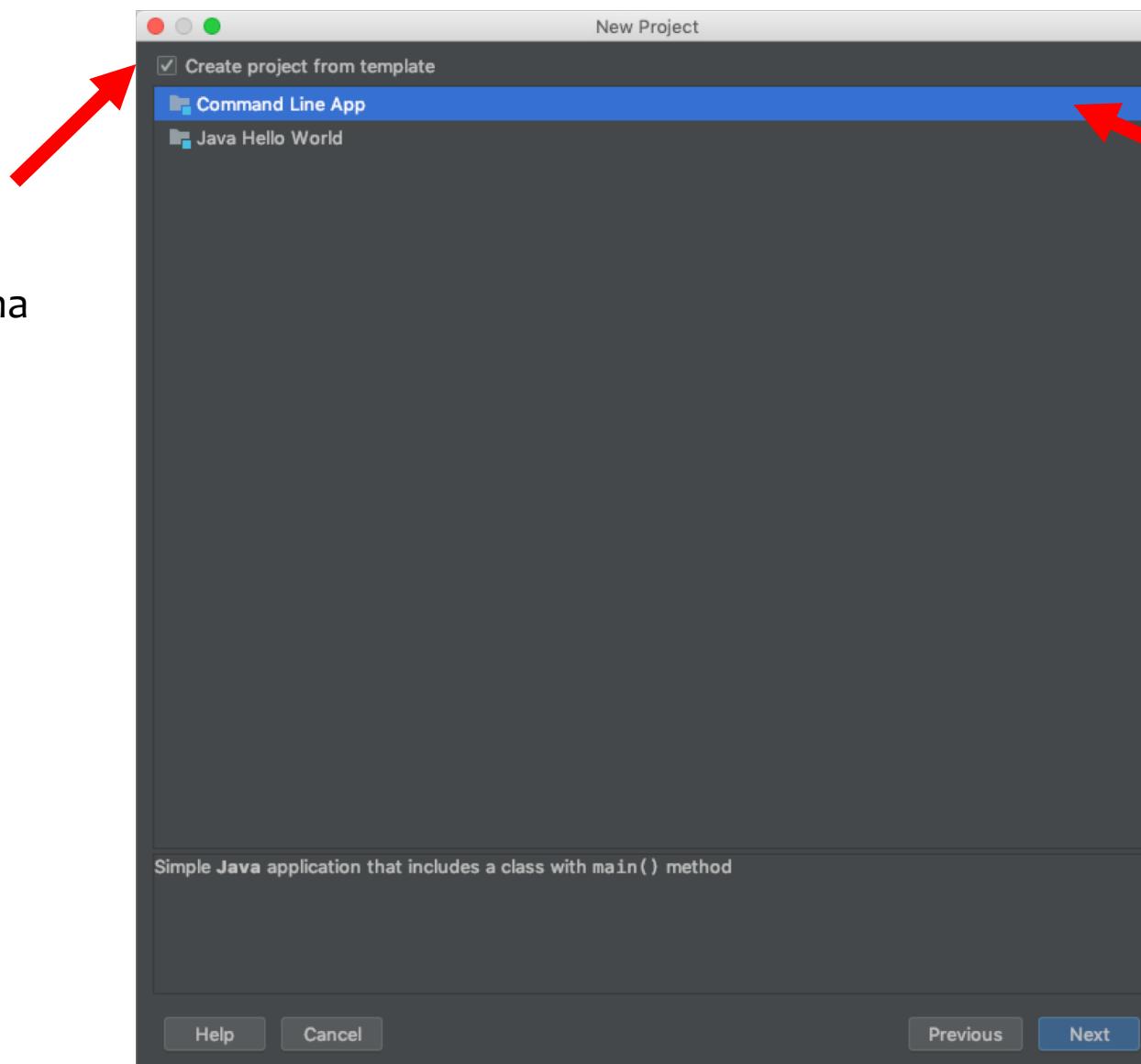
# Ex. Dirigido – Criando uma aplicação Java

Escolher  
plataforma  
Java



Escolher a  
versão do  
SDK que  
será  
utilizada no  
projeto.

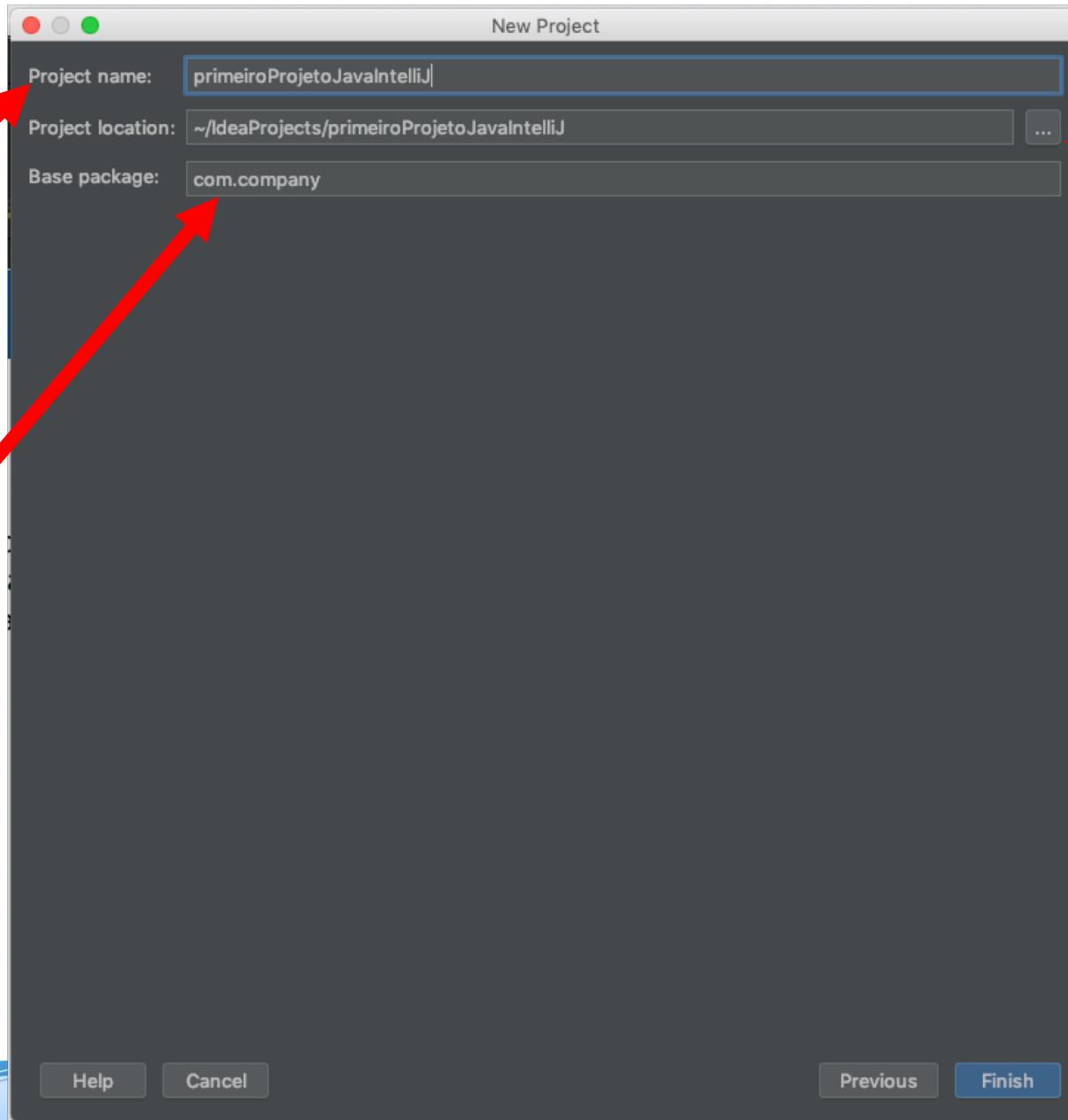
# Ex. Dirigido – Criando uma aplicação Java



Escolher  
plataforma  
Java

Escolher o  
template de  
linha de  
comando.

# Ex. Dirigido – Criando uma aplicação Java

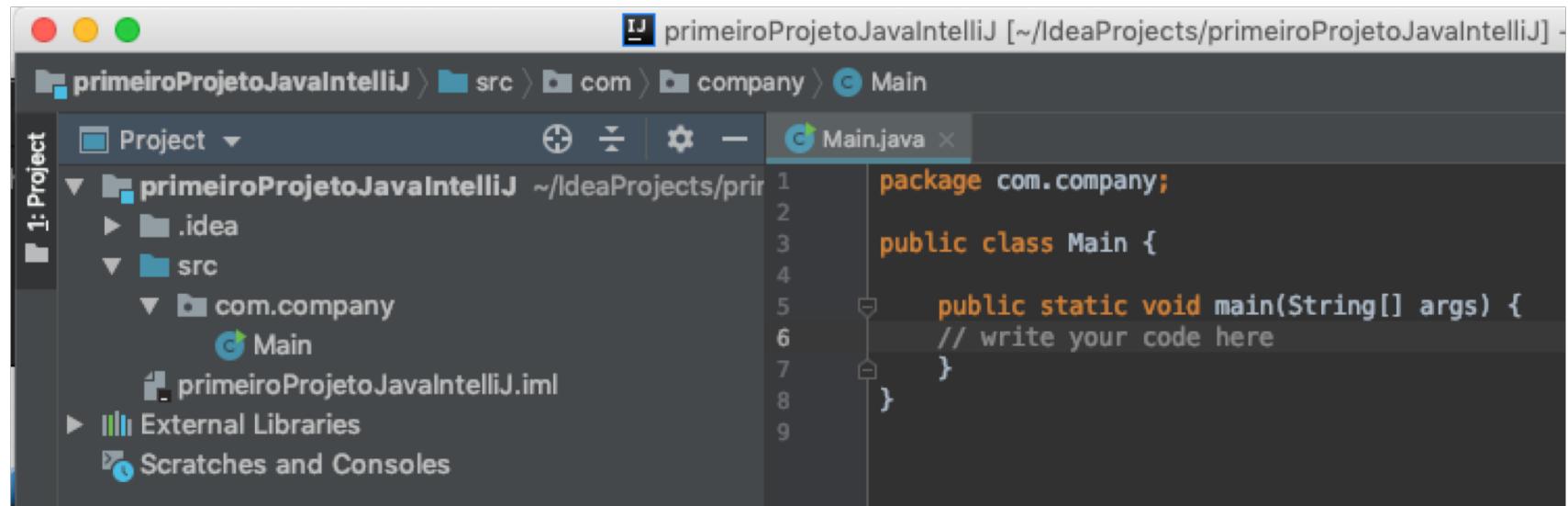


Escolher um nome para o projeto

Apenas por enquanto, vamos deixar isso dessa forma.

Por padrão, os projetos são criados em Usuário/Idea Projects, esse caminho pode ser alterado.

# Ex. Dirigido – Criando uma aplicação Java



# Perguntas?