



INSTITUTO FEDERAL  
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
Bahia

---

# Linguagem de Programação II

Persistência de Dados em Java - Java  
Database Connectivity (JDBC) – Tela de  
Cadastro;



# Roteiro

- Persistência de Dados em Java Java Database Connectivity (JDBC) – Tela de Cadastro;
  - JDBC;
  - Interação entre Aplicações Java e Bancos de Dados;
  - Criação do Banco de Dados e das Tabelas;
  - Criação de um projeto Netbeans com Maven (JDBC);
    - Instalando o Mysql Connector Java;
  - Exemplo – Cadastro de Conta;
    - Classe GerenciadorBD;
    - Tela de Cadastro de Conta.



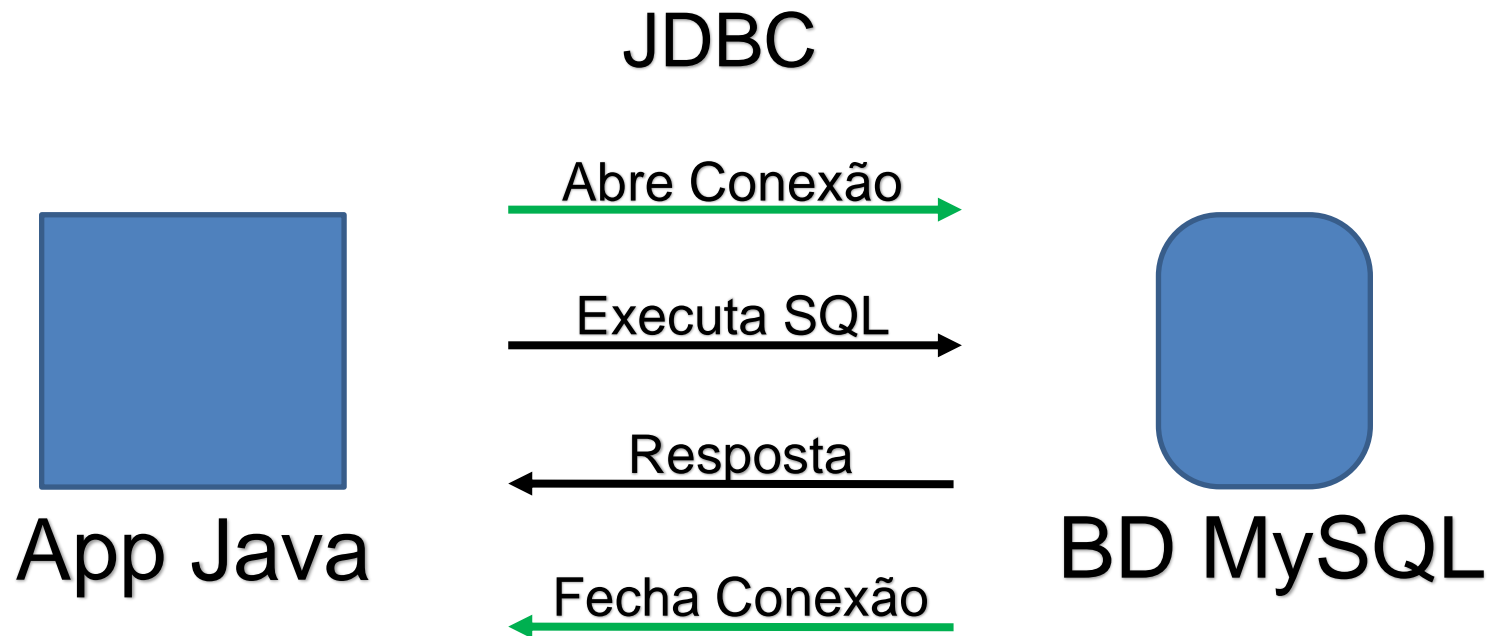
# JDBC

---

- **Persistência de Dados** refere-se à capacidade de **armazenar e recuperar informações de maneira durável** (Ex.: Banco de Dados). **Java Database Connectivity (JDBC)** é uma **biblioteca que facilita a conexão e interação com bancos de dados relacionais** por meio de código **Java**, permitindo operações como **consulta e atualização de dados**.



# Interação - App Java e Bancos de Dados

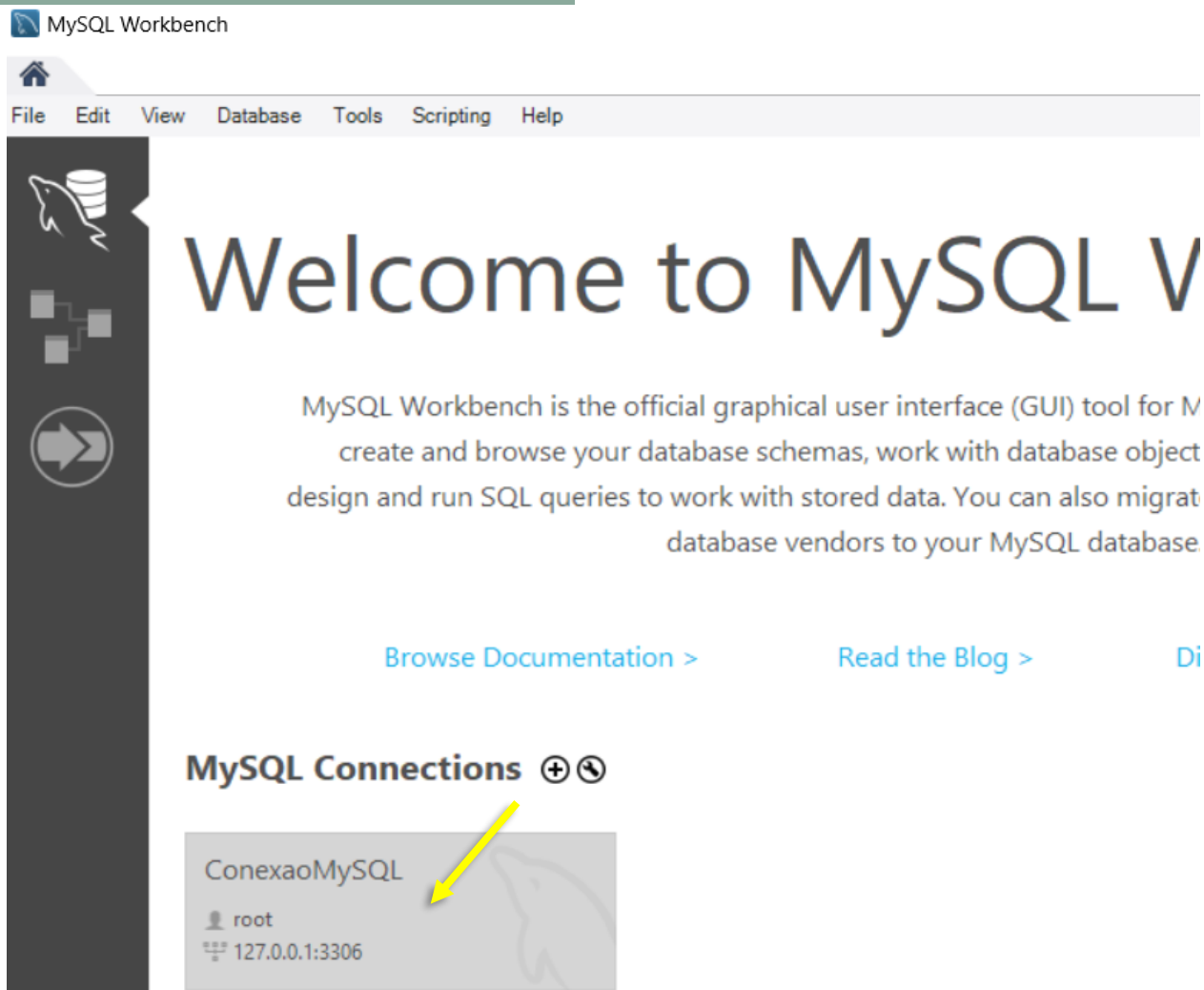


# Criação do BD e das Tabelas - Script

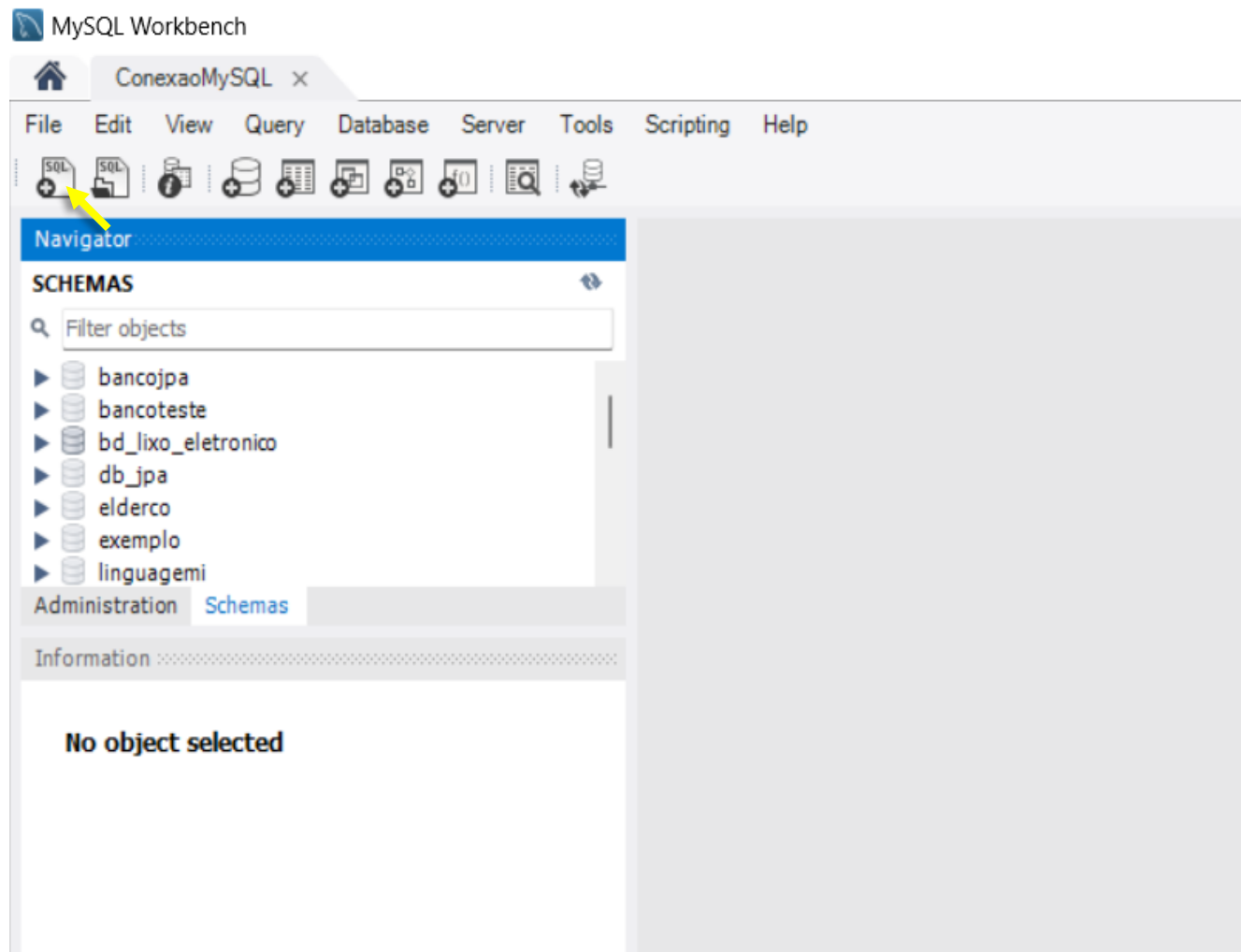
```
1  CREATE SCHEMA IF NOT EXISTS `bancoabc` ;
2
3  USE `bancoabc` ;
4
5  CREATE TABLE IF NOT EXISTS `conta` (
6      `id` int NOT NULL AUTO_INCREMENT,
7      `numero` int NOT NULL,
8      `saldo` double NOT NULL,
9      `limite` double NOT NULL,
10     PRIMARY KEY (`id`)
11 );
```



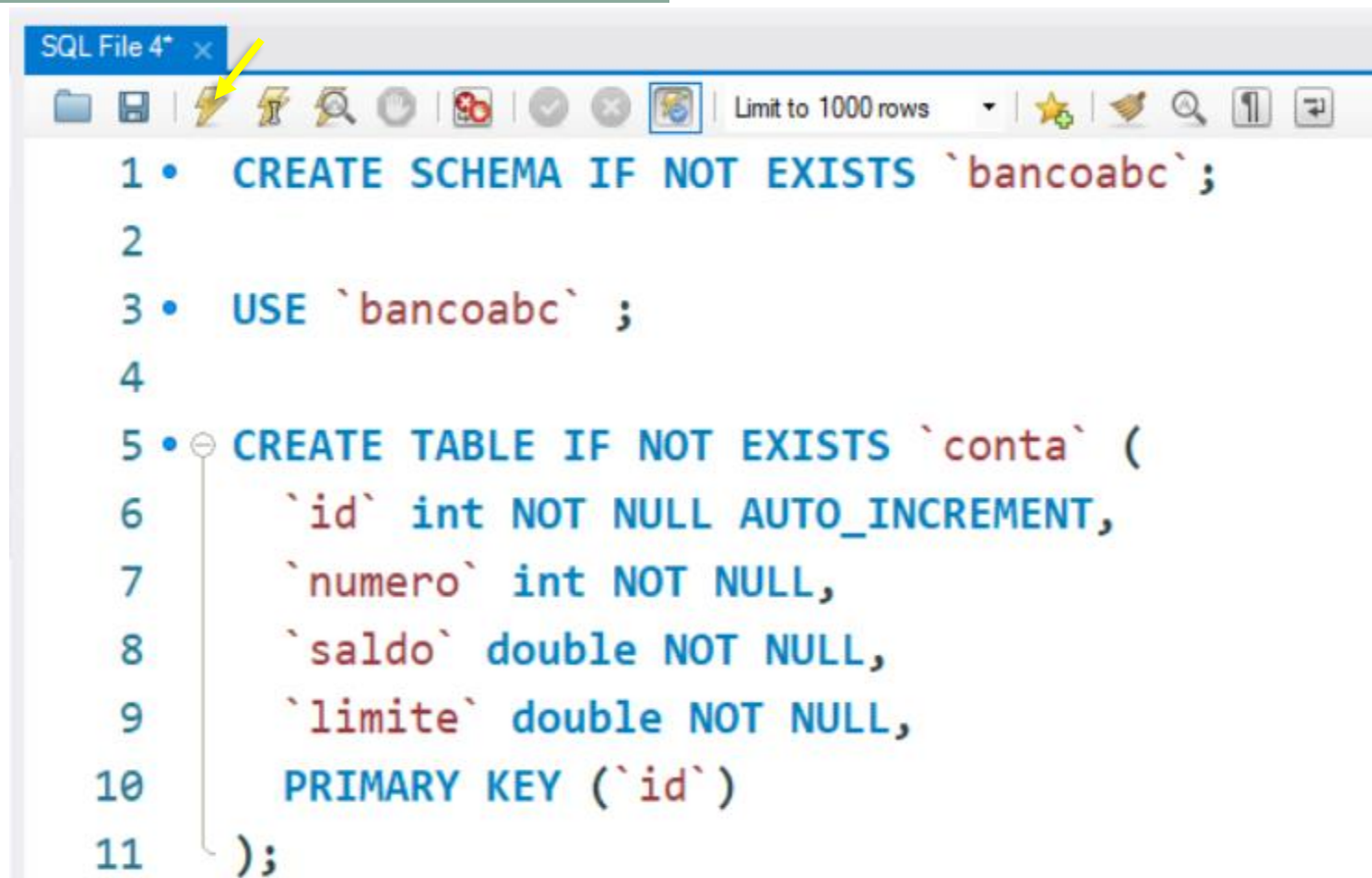
# Criação do BD e das Tabelas - MySQL



# Criação do BD e das Tabelas - MySQL



# Criação do BD e das Tabelas - MySQL



The screenshot shows a MySQL SQL editor window titled "SQL File 4\* x". The toolbar includes icons for file operations, execution, and search. The SQL code is as follows:

```
1 • CREATE SCHEMA IF NOT EXISTS `bancoabc` ;
2
3 • USE `bancoabc` ;
4
5 • CREATE TABLE IF NOT EXISTS `conta` (
6     `id` int NOT NULL AUTO_INCREMENT,
7     `numero` int NOT NULL,
8     `saldo` double NOT NULL,
9     `limite` double NOT NULL,
10    PRIMARY KEY (`id`)
11 );
```

A yellow arrow points to the execution icon (a lightning bolt) in the toolbar. A vertical line with a circle at the top connects the execution icon to the start of the SQL code on line 5.



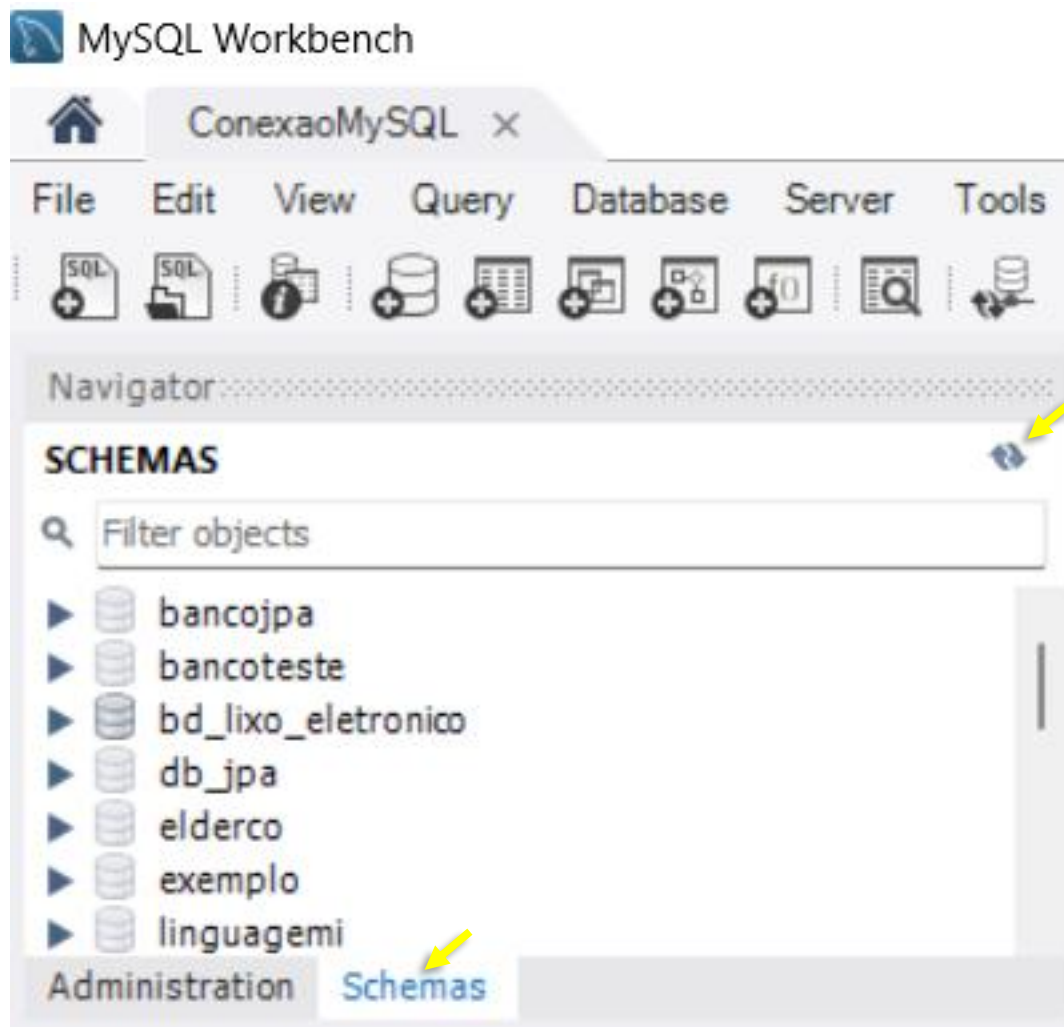


# Criação do BD e das Tabelas - MySQL

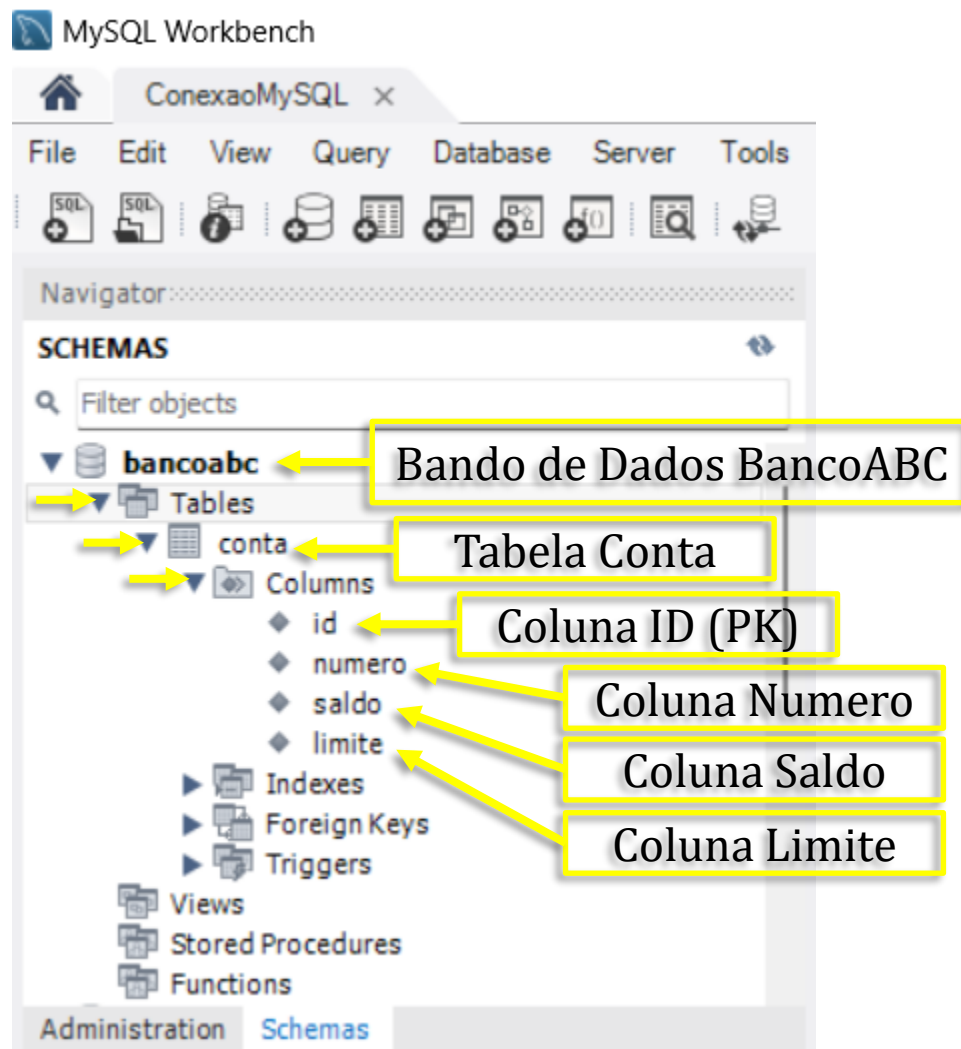
Output				
Action Output				
	#	Time	Action	Message
✓	1	16:51:53	CREATE SCHEMA IF NOT EXISTS 'bancoabc'	1 row(s) affected
✓	2	16:51:53	USE 'bancoabc'	0 row(s) affected
✓	3	16:51:53	CREATE TABLE IF NOT EXISTS 'conta' ( 'id' int NOT NULL AUTO_INCREMENT, 'numero' int N...	0 row(s) affected



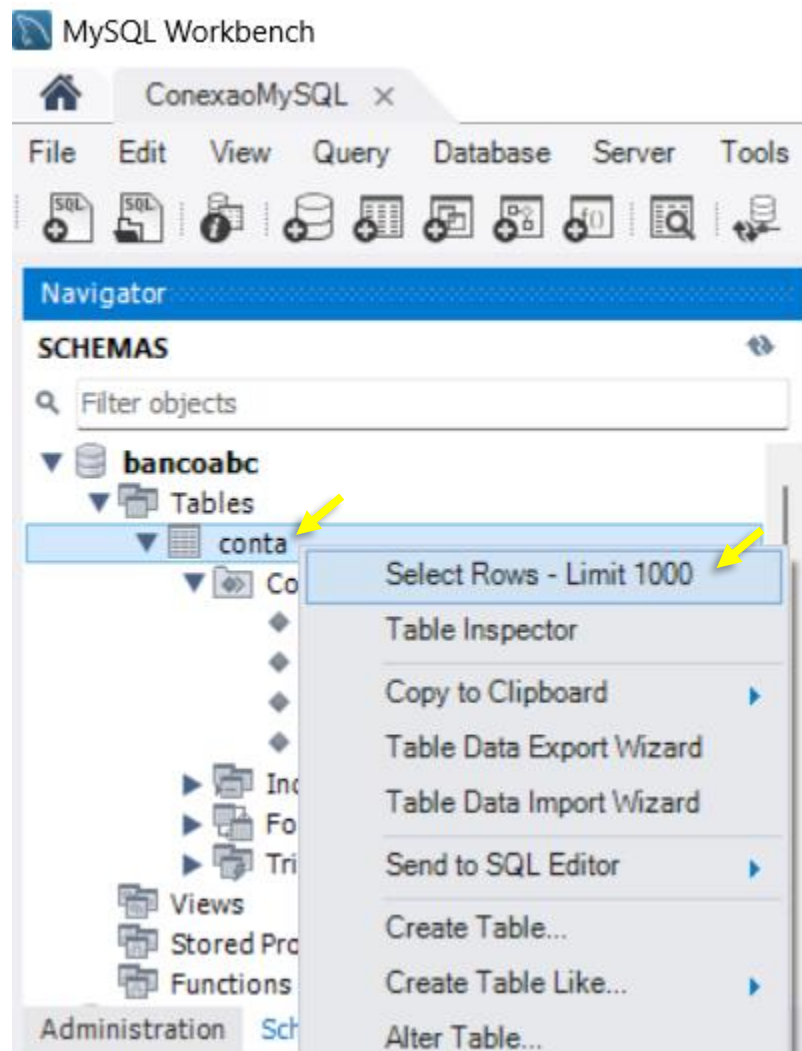
# Criação do BD e das Tabelas - MySQL



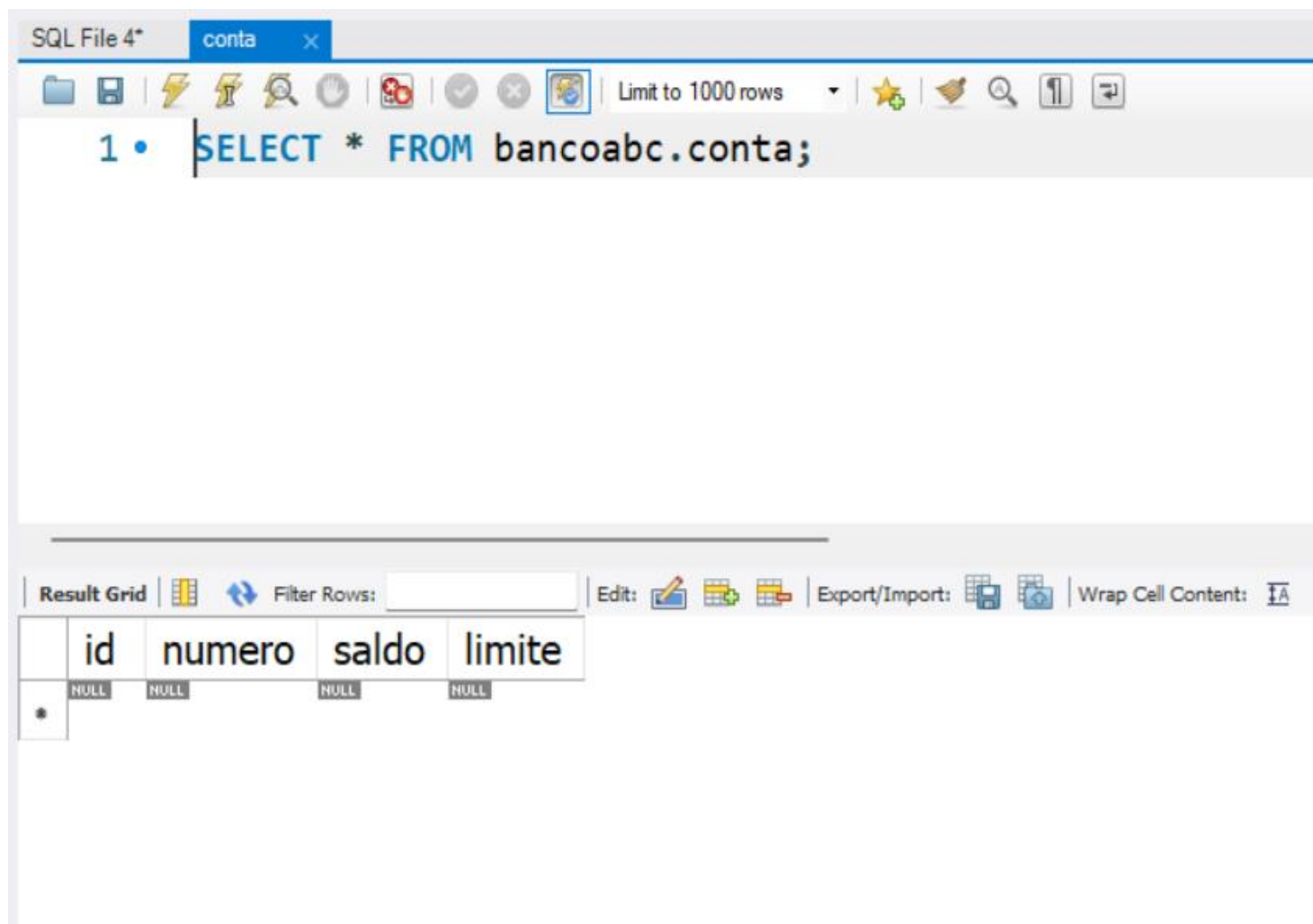
# Criação do BD e das Tabelas - MySQL



# Criação do BD e das Tabelas - MySQL



# Criação do BD e das Tabelas - MySQL



# Criação de Projeto Netbeans (Maven)



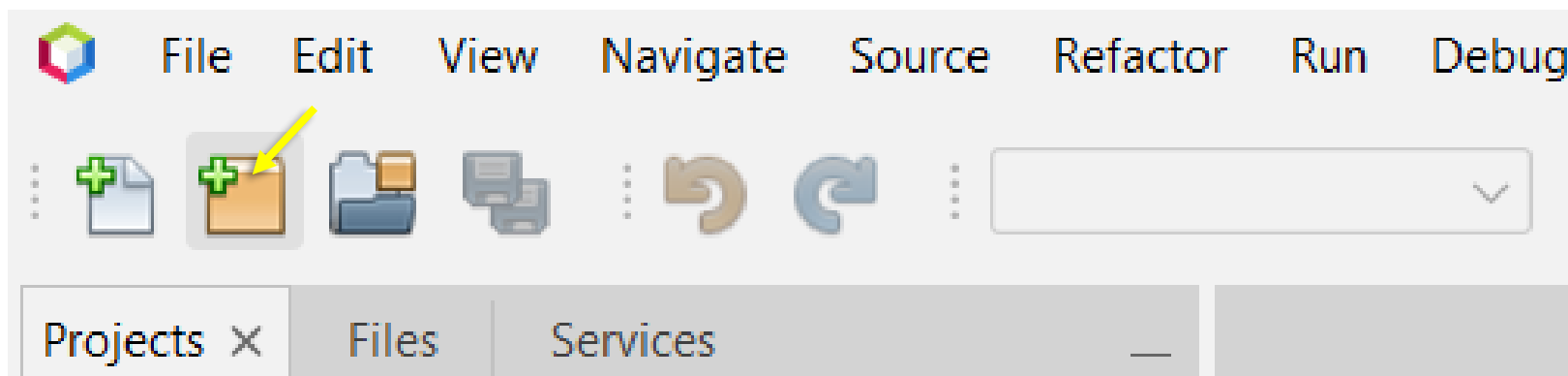
Turning on modules...

Apache NetBeans IDE 16

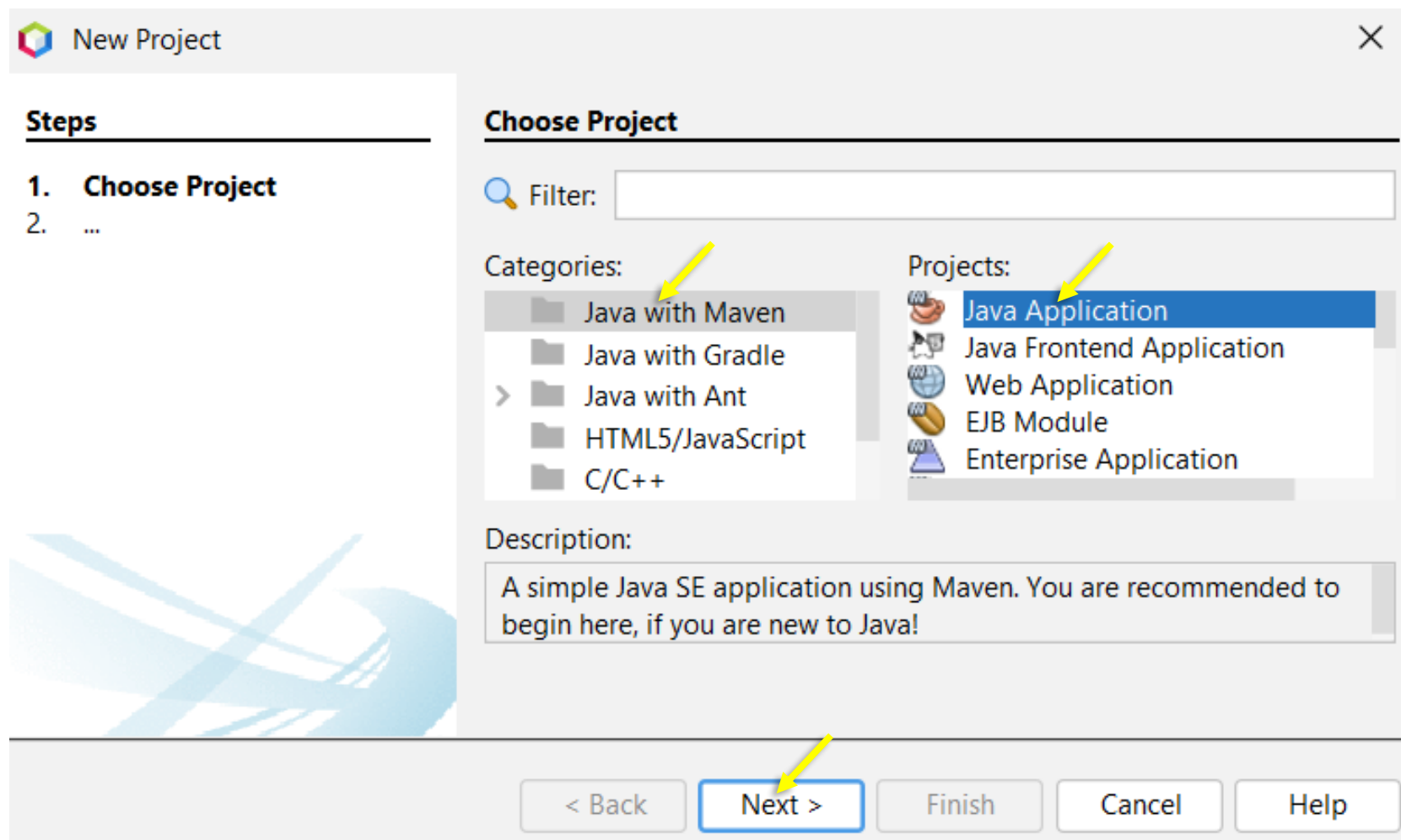


INSTITUTO FEDERAL  
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
Bahia

# Criação de Projeto Netbeans (Maven)



# Criação de Projeto Netbeans (Maven)





# Criação de Projeto Netbeans (Maven)

New Java Application

**Steps**

1. Choose Project
2. **Name and Location**

**Name and Location**

Project Name: ProjetoBancoABC

Project Location: C:\Users\leomi\Desktop\ Browse...

Project Folder: C:\Users\leomi\Desktop\ProjetoBancoABC

Artifact Id: ProjetoBancoABC

Group Id: ifba

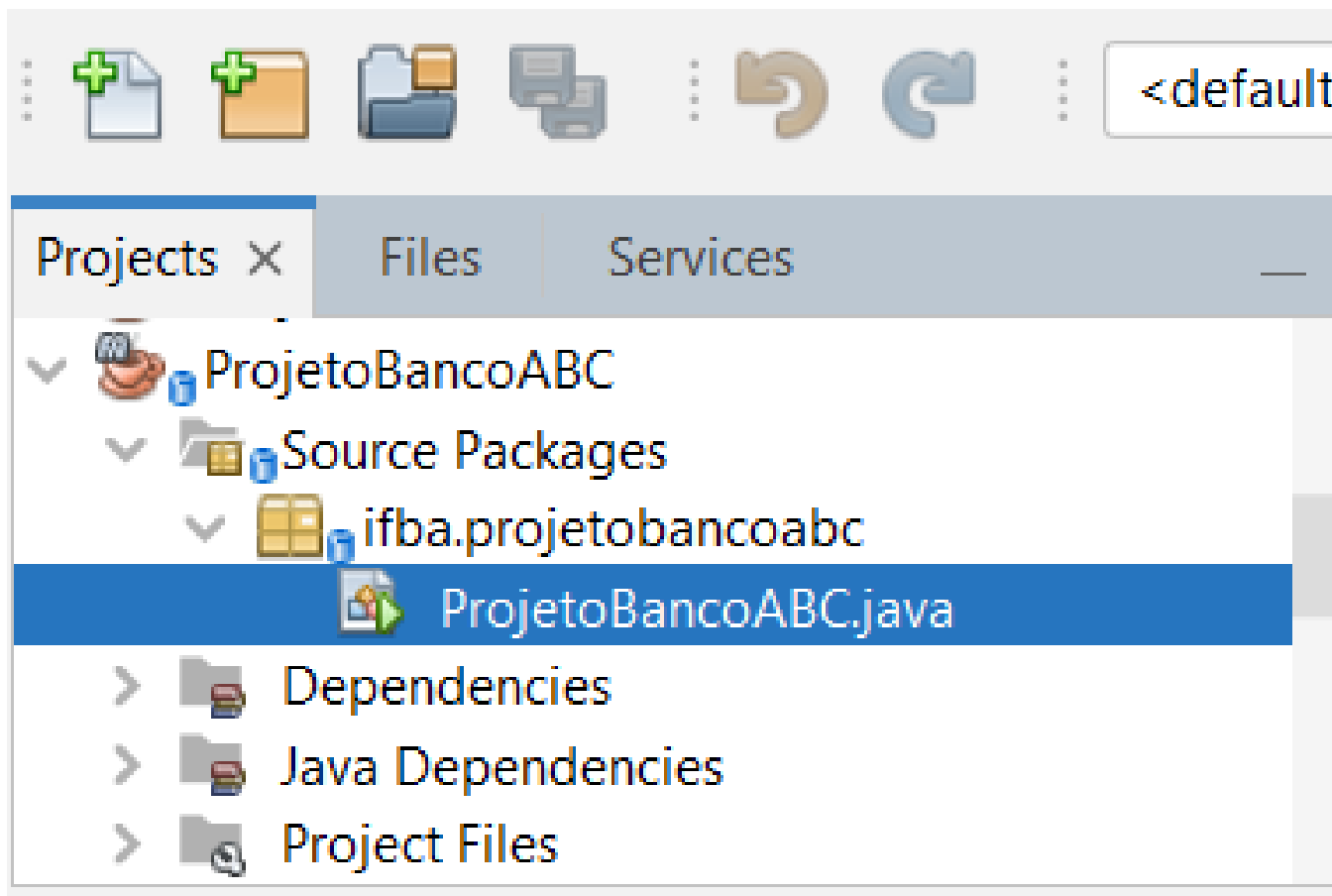
Version: 1.0-SNAPSHOT

Package: ifba.projetoBANCOABC (Optional)

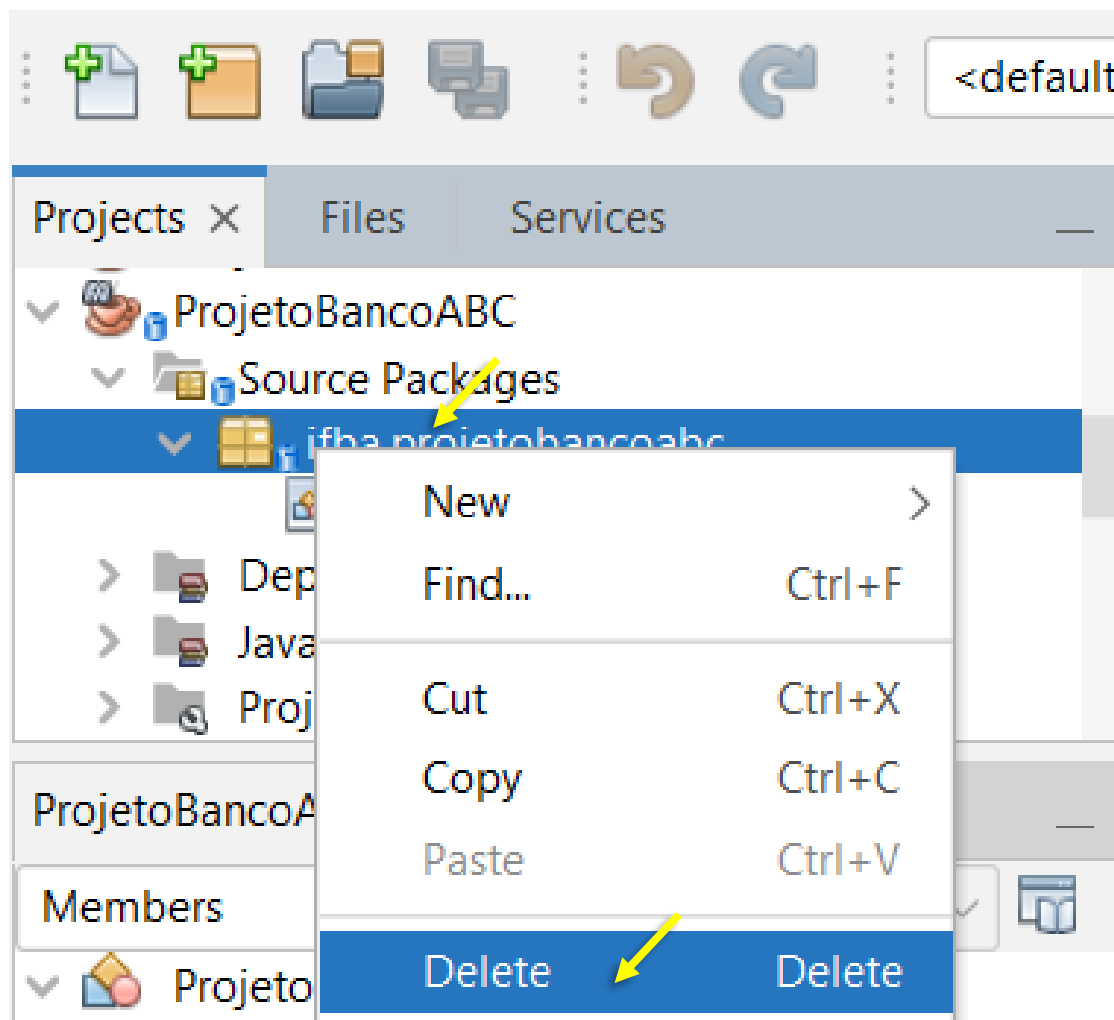
< Back Next > Finish Cancel Help



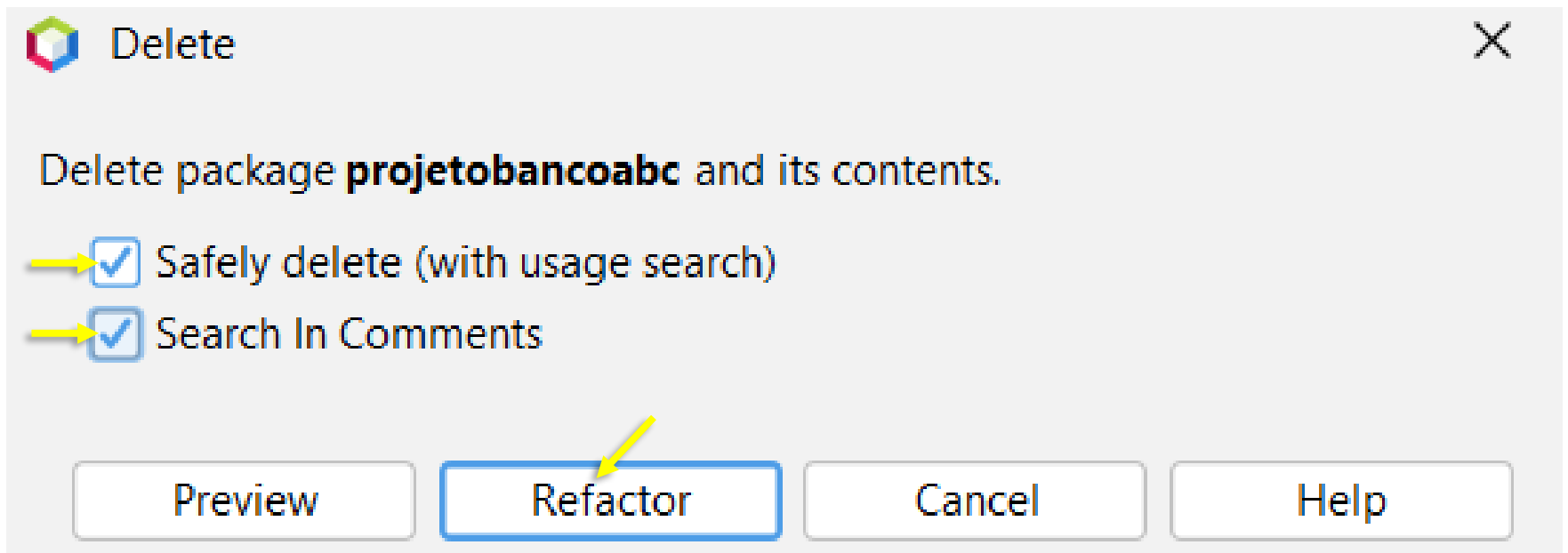
# Criação de Projeto Netbeans (Maven)



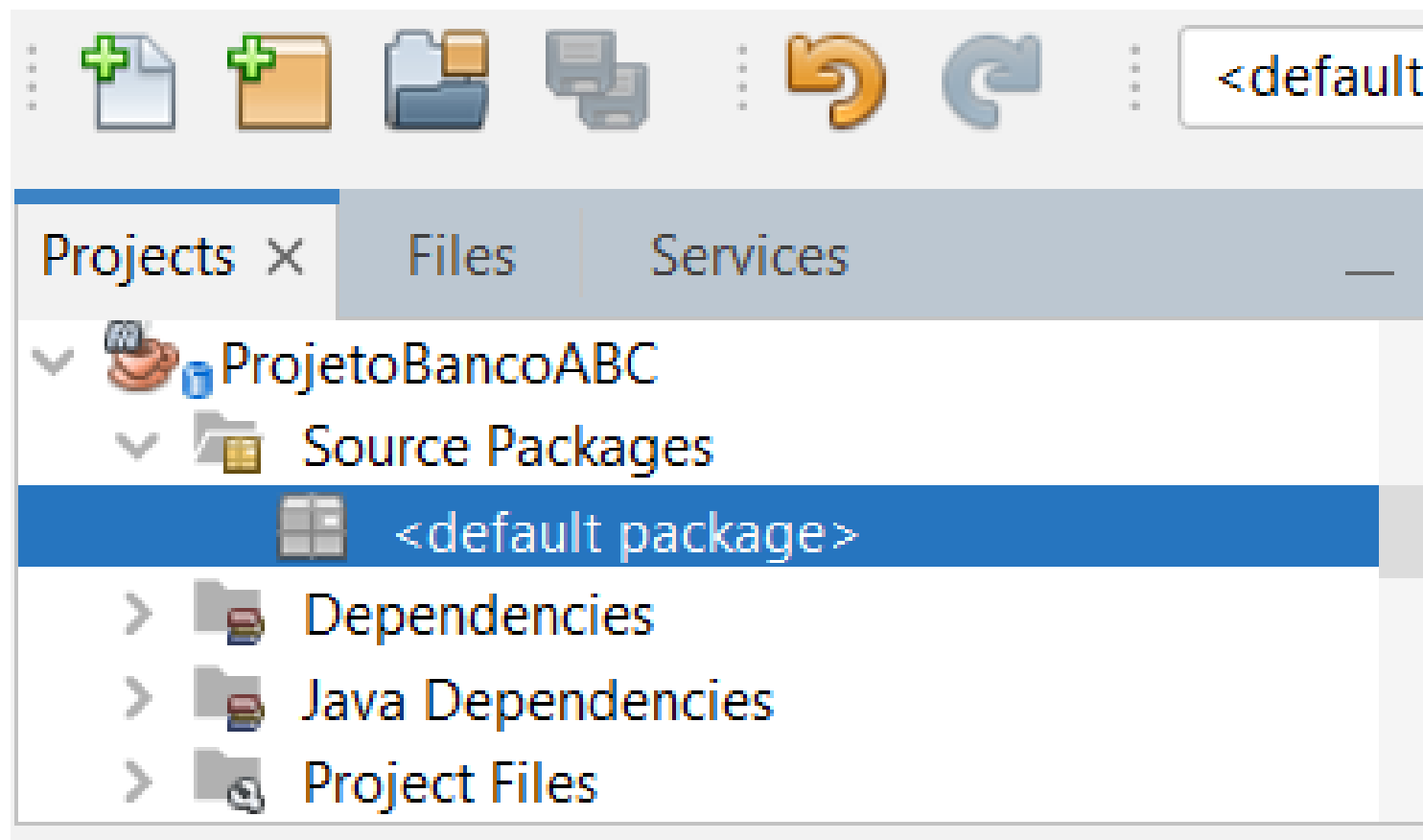
# Criação de Projeto Netbeans (Maven)



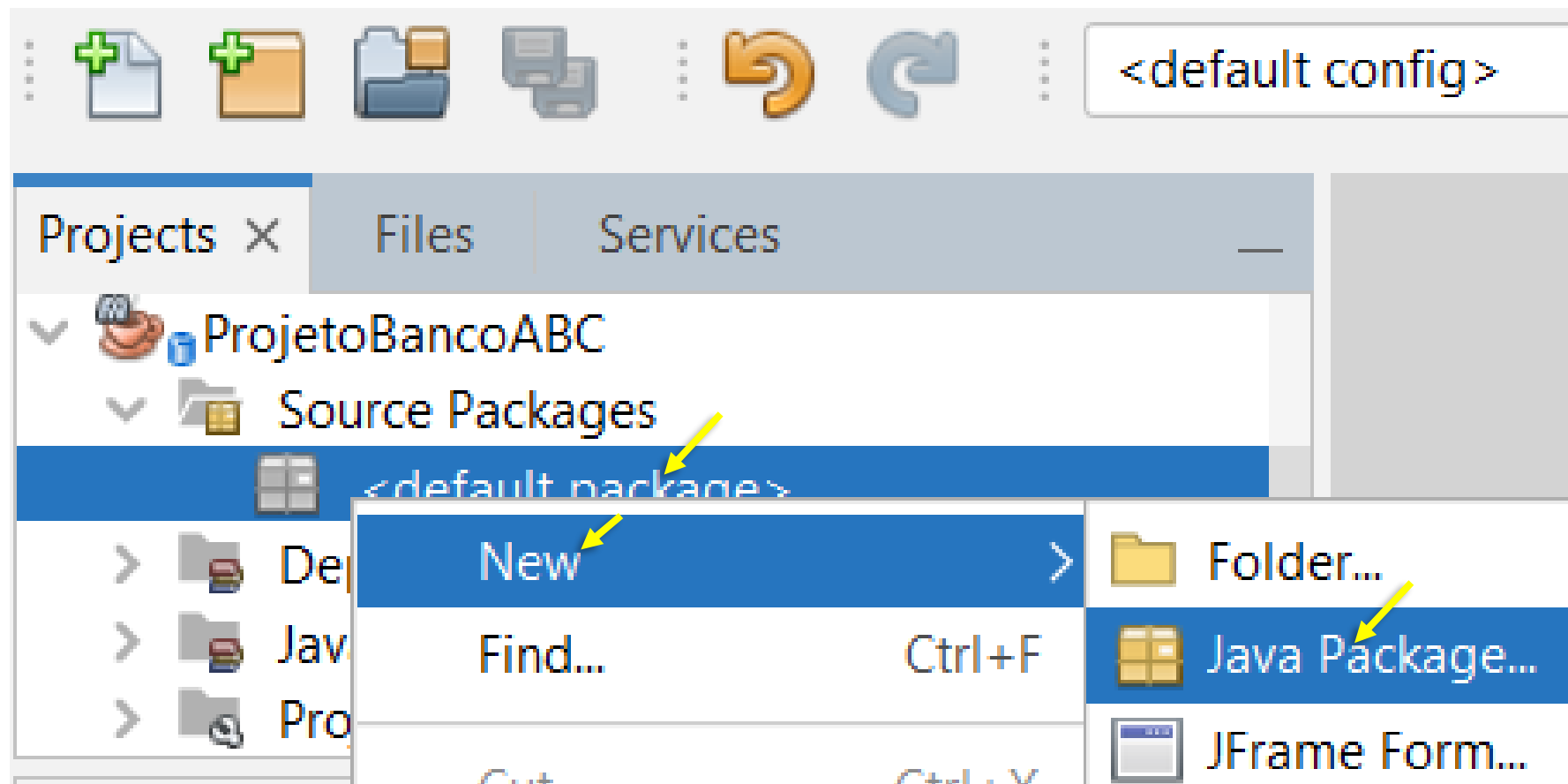
# Criação de Projeto Netbeans (Maven)



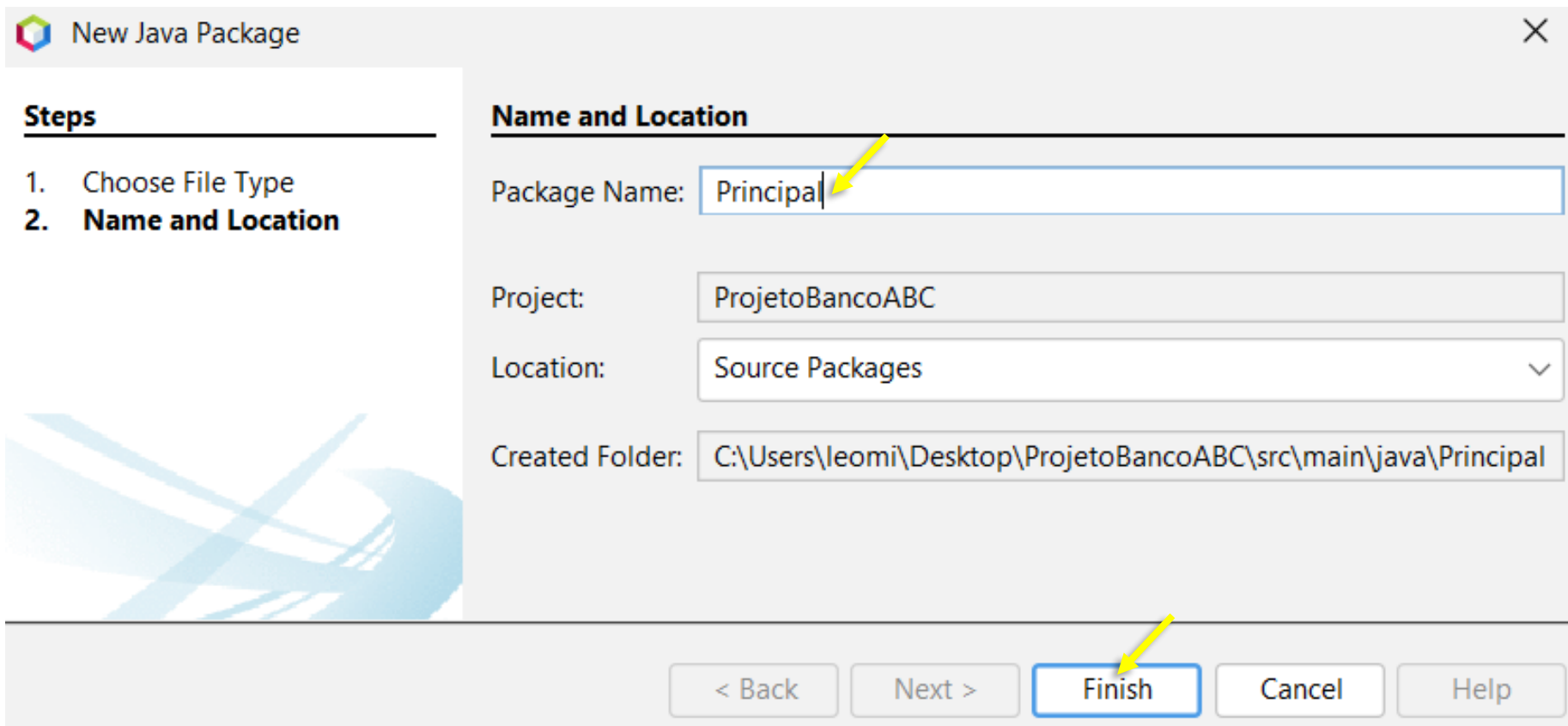
# Criação de Projeto Netbeans (Maven)



# Criação de Projeto Netbeans (Maven)



# Criação de Projeto Netbeans (Maven)



**New Java Package**

**Steps**

1. Choose File Type
2. **Name and Location**

**Name and Location**

Package Name:

Project:

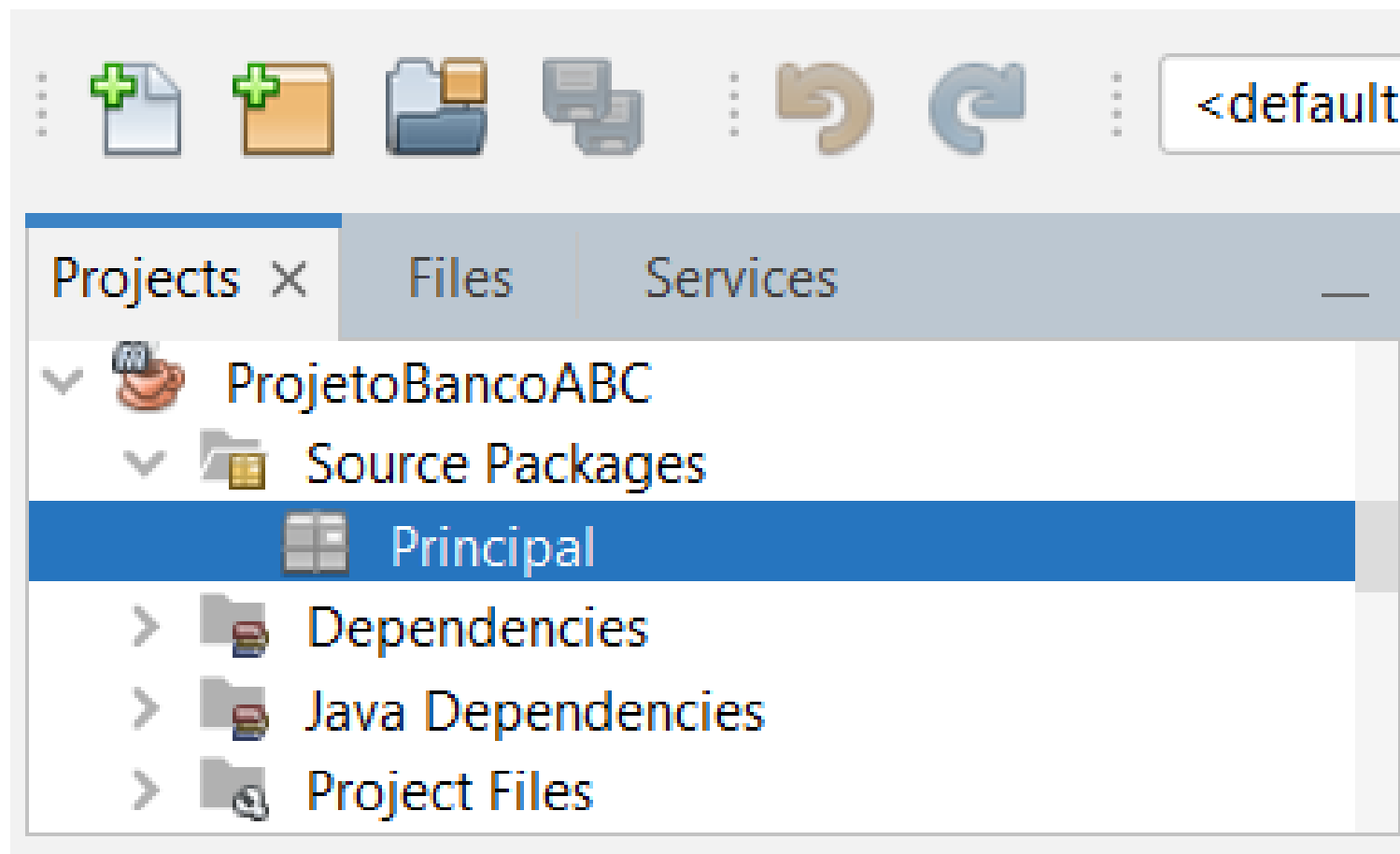
Location:

Created Folder:

< Back   Next >   **Finish**   Cancel   Help

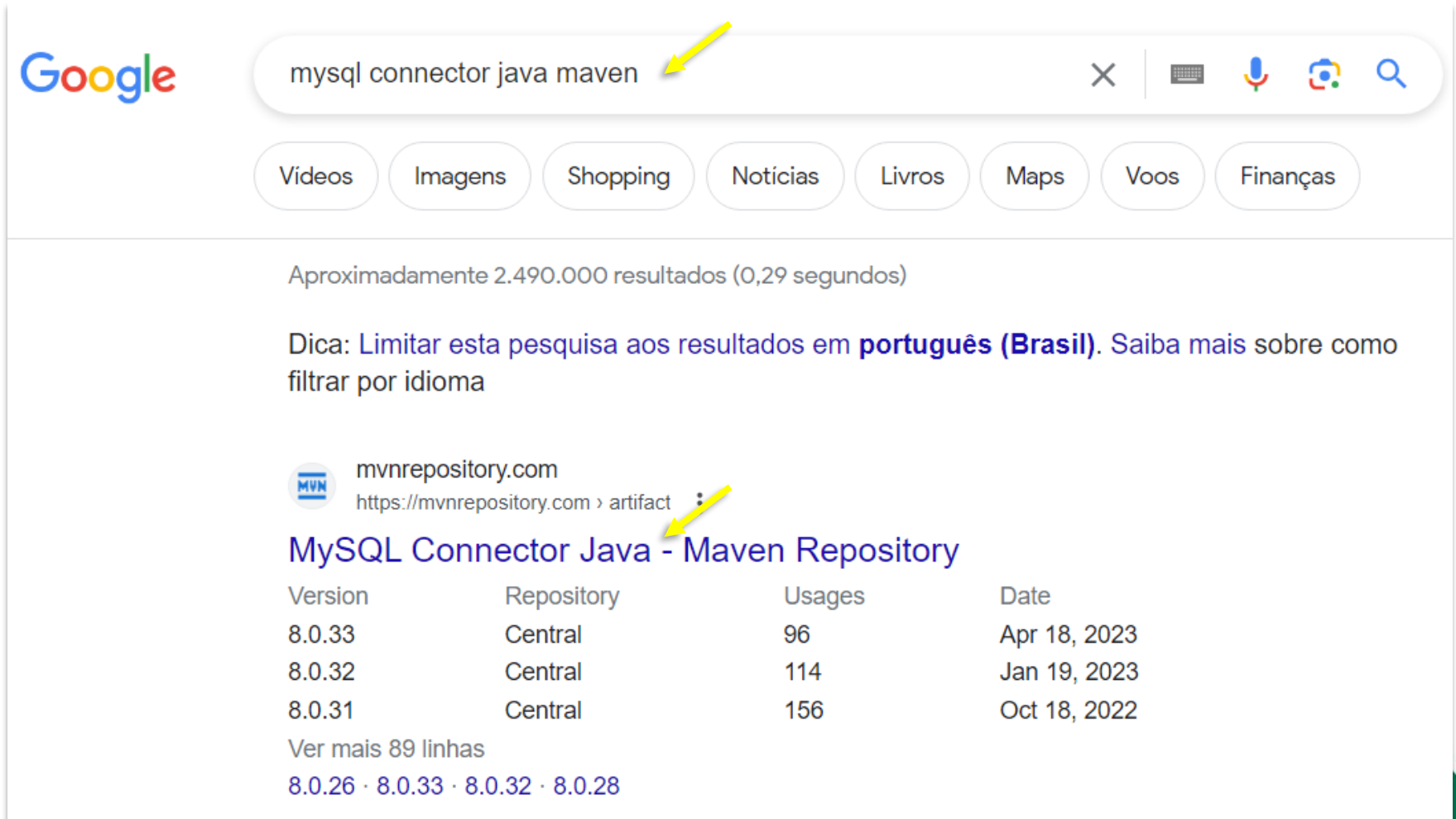


# Criação de Projeto Netbeans (Maven)





# Instalar MySQL Connector Java (Maven)




Google

mysql connector java maven

Vídeos Imagens Shopping Notícias Livros Maps Voos Finanças

Aproximadamente 2.490.000 resultados (0,29 segundos)

Dica: Limitar esta pesquisa aos resultados em **português (Brasil)**. Saiba mais sobre como filtrar por idioma

 mvnrepository.com  
https://mvnrepository.com › artifact

**MySQL Connector Java - Maven Repository**

Version	Repository	Usages	Date
8.0.33	Central	96	Apr 18, 2023
8.0.32	Central	114	Jan 19, 2023
8.0.31	Central	156	Oct 18, 2022

Ver mais 89 linhas

[8.0.26](#) · [8.0.33](#) · [8.0.32](#) · [8.0.28](#)



# Instalar MySQL Connector Java (Maven)

**MVN** REPOSITORY



## MySQL Connector Java

MySQL Connector/J is a JDBC Type 4 driver, which means that it is part of the MySQL client libraries. This driver supports auto-registration with SQLExceptions, support for large update counts, support for local and remote JDBC-4.x XML processing, support for per connection client information.

Categories	<b>JDBC Drivers</b>
Tags	<a href="#">database</a> <a href="#">sql</a> <a href="#">jdbc</a> <a href="#">driver</a> <a href="#">connector</a> <a href="#">mysql</a>
Ranking	<b>#68 in MvnRepository (See Top Artifacts)</b> <b>#1 in JDBC Drivers</b>
Used By	<b>7,240 artifacts</b>

**Note:** This artifact was moved to:

[com.mysql » mysql-connector-j](#)

MySQL Connector/J artifacts moved to reverse-DNS compliant Maven 2+ coordinates

[Central \(92\)](#) [Jahia \(1\)](#) [Redhat GA \(8\)](#) [Redhat EA \(3\)](#) [Imcode \(1\)](#)

Version	Vulnerabilities
<a href="#">8.0.33</a>	



INSTITUTO FEDERAL  
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
Bahia

# Instalar MySQL Connector Java (Maven)

**Note:** This artifact was moved to:

`com.mysql » mysql-connector-j`

MySQL Connector/J artifacts moved to reverse-DNS compliant Maven 2+ coordinates.

Maven

Gradle

Gradle (Short)

Gradle (Kotlin)

SBT

Ivy

Grape

Leiningen

Buildr

```
<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->
```

```
<dependency>
```

```
  <groupId>mysql</groupId>
```

```
  <artifactId>mysql-connector-java</artifactId>
```

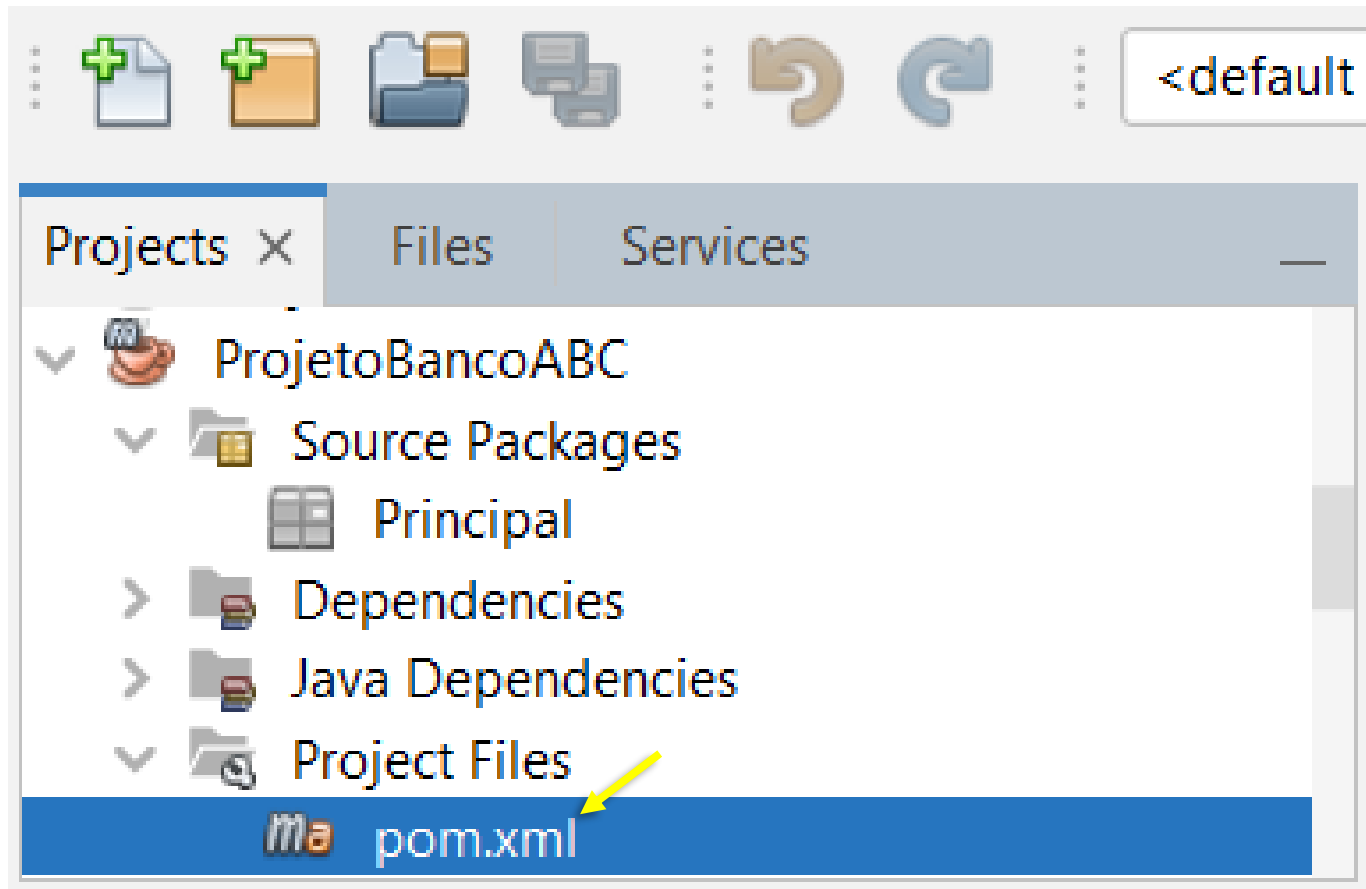
```
  <version>8.0.33</version>
```

```
</dependency>
```

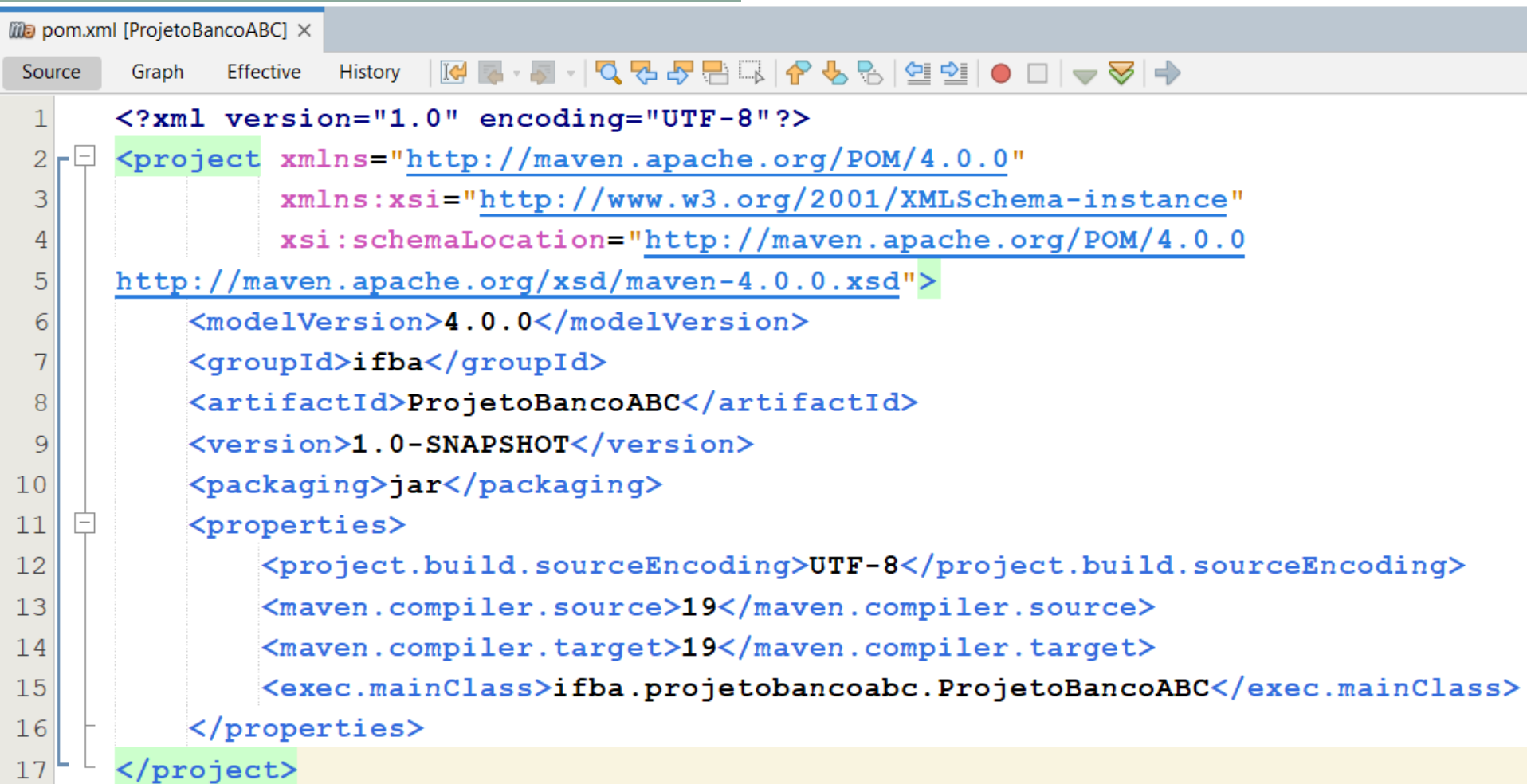
Copia o Texto Selecionado



# Instalar MySQL Connector Java (Maven)



# Instalar MySQL Connector Java (Maven)



The screenshot shows an IDE window titled 'pom.xml [ProjetoBancoABC]'. The interface includes tabs for 'Source', 'Graph', 'Effective', and 'History'. A toolbar with various icons is visible above the code editor. The code editor displays the following XML content:

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0"
3         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
5         http://maven.apache.org/xsd/maven-4.0.0.xsd">
6     <modelVersion>4.0.0</modelVersion>
7     <groupId>ifba</groupId>
8     <artifactId>ProjetoBancoABC</artifactId>
9     <version>1.0-SNAPSHOT</version>
10    <packaging>jar</packaging>
11    <properties>
12        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
13        <maven.compiler.source>19</maven.compiler.source>
14        <maven.compiler.target>19</maven.compiler.target>
15        <exec.mainClass>ifba.projetoBANCOABC.ProjetoBancoABC</exec.mainClass>
16    </properties>
17 </project>
```



# Instalar MySQL Connector Java (Maven)

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <project xmlns="http://maven.apache.org/POM/4.0.0"
3         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
5         http://maven.apache.org/xsd/maven-4.0.0.xsd">
6     <modelVersion>4.0.0</modelVersion>
7     <groupId>ifba</groupId>
8     <artifactId>ProjetoBancoABC</artifactId>
9     <version>1.0-SNAPSHOT</version>
10    <packaging>jar</packaging>
11    <properties>
12        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
13        <maven.compiler.source>19</maven.compiler.source>
14        <maven.compiler.target>19</maven.compiler.target>
15        <exec.mainClass>ifba.projetoBANCOABC.ProjetoBancoABC</exec.mainClass>
16    </properties>
17    <dependencies>
18
19    </dependencies>
20 </project>
```



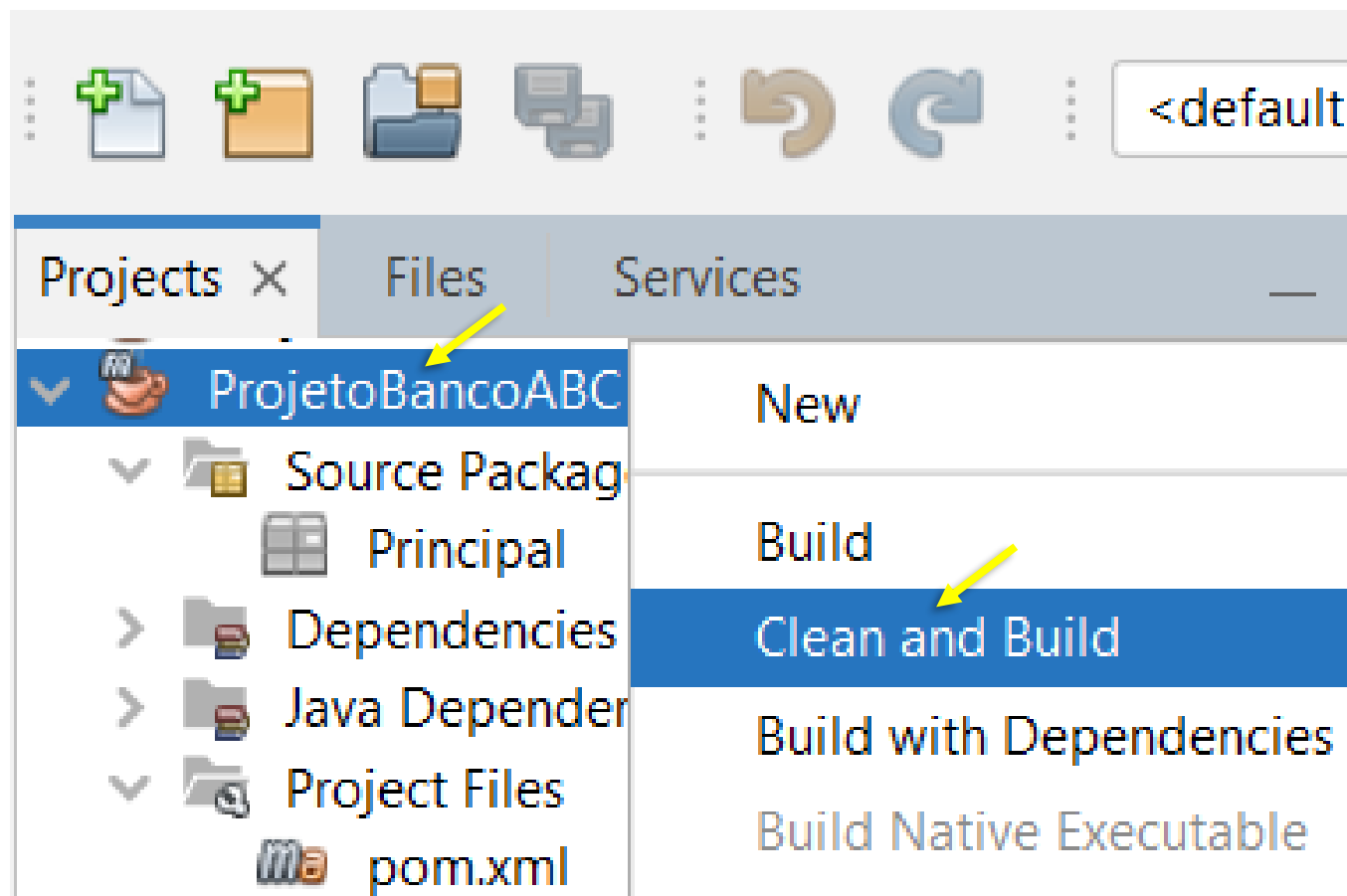
# Instalar MySQL Connector Java (Maven)

Cola o Texto Seleccionado

```
17 <dependencies>
18   <dependency>
19     <groupId>mysql</groupId>
20     <artifactId>mysql-connector-java</artifactId>
21     <version>8.0.33</version>
22   </dependency>
23 </dependencies>
24 </project>
```




# Instalar MySQL Connector Java (Maven)





# Instalar MySQL Connector Java (Maven)

Output - Build (ProjetoBancoABC)

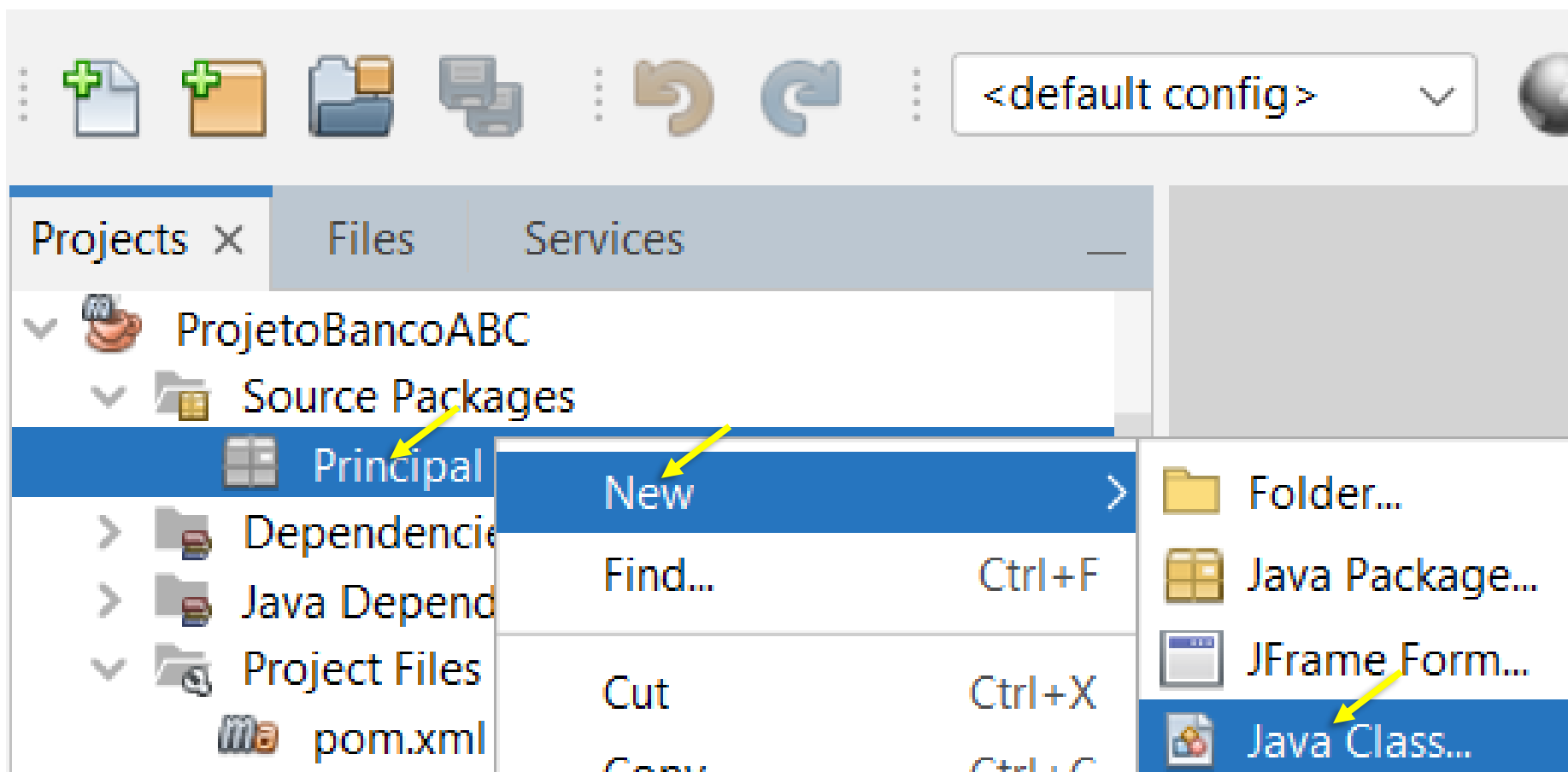


```
--- maven-install-plugin:2.4:install (default-install) @
Installing C:\Users\leomi\Desktop\ProjetoBancoABC\target\
Installing C:\Users\leomi\Desktop\ProjetoBancoABC\pom.xml
-----
BUILD SUCCESS
-----

Total time: 1.982 s
Finished at: 2023-08-27T18:31:47-03:00
-----
```



# Classe GerenciadorBD



# Classe GerenciadorBD

New Java Class

**Steps**

1. Choose File Type
2. **Name and Location**

**Name and Location**

Class Name:

Project:

Location:

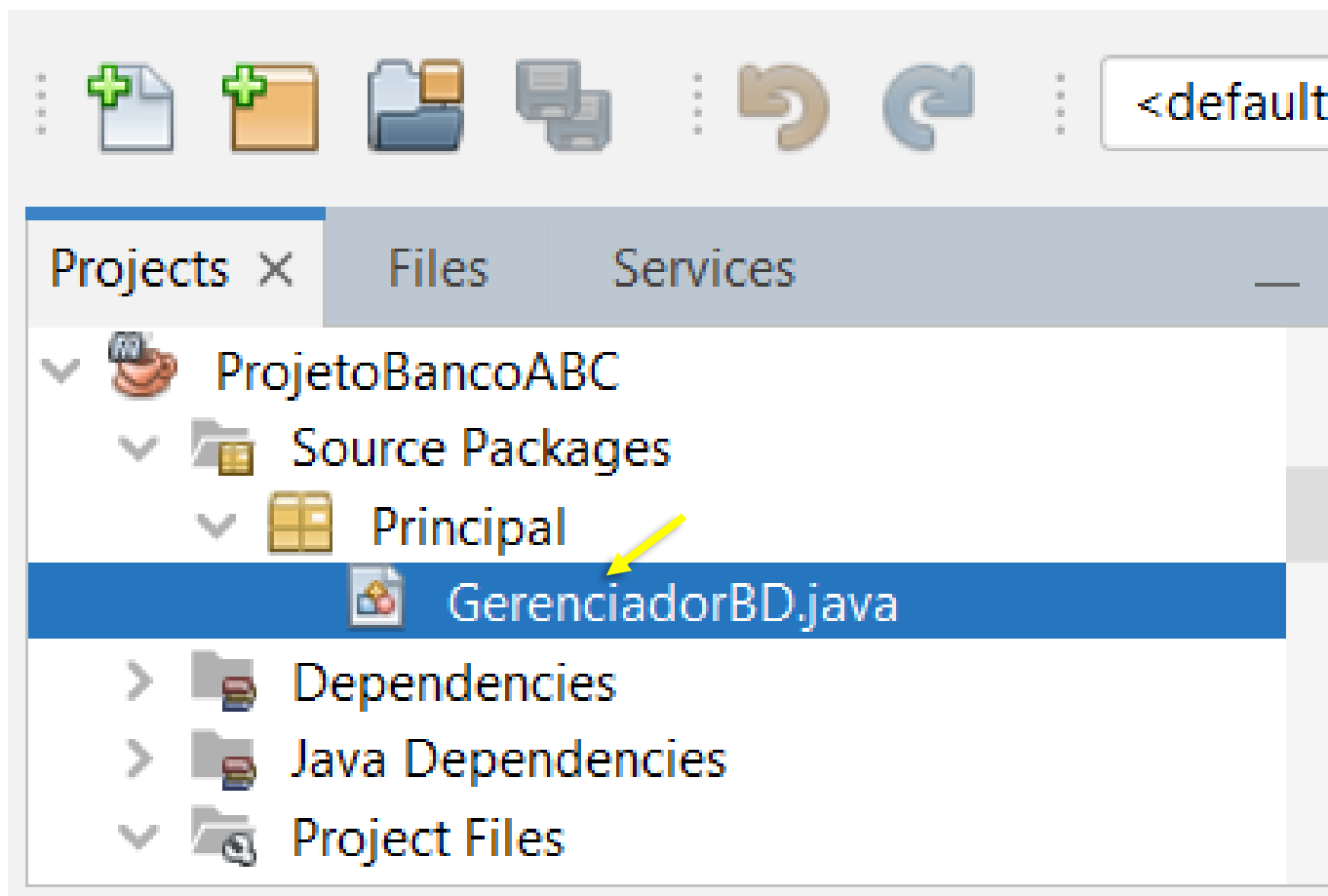
Package:

Created File:

< Back   Next >   **Finish**   Cancel   Help



# Classe GerenciadorBD



Colocar apenas uma classe GerenciadorBD por Projeto

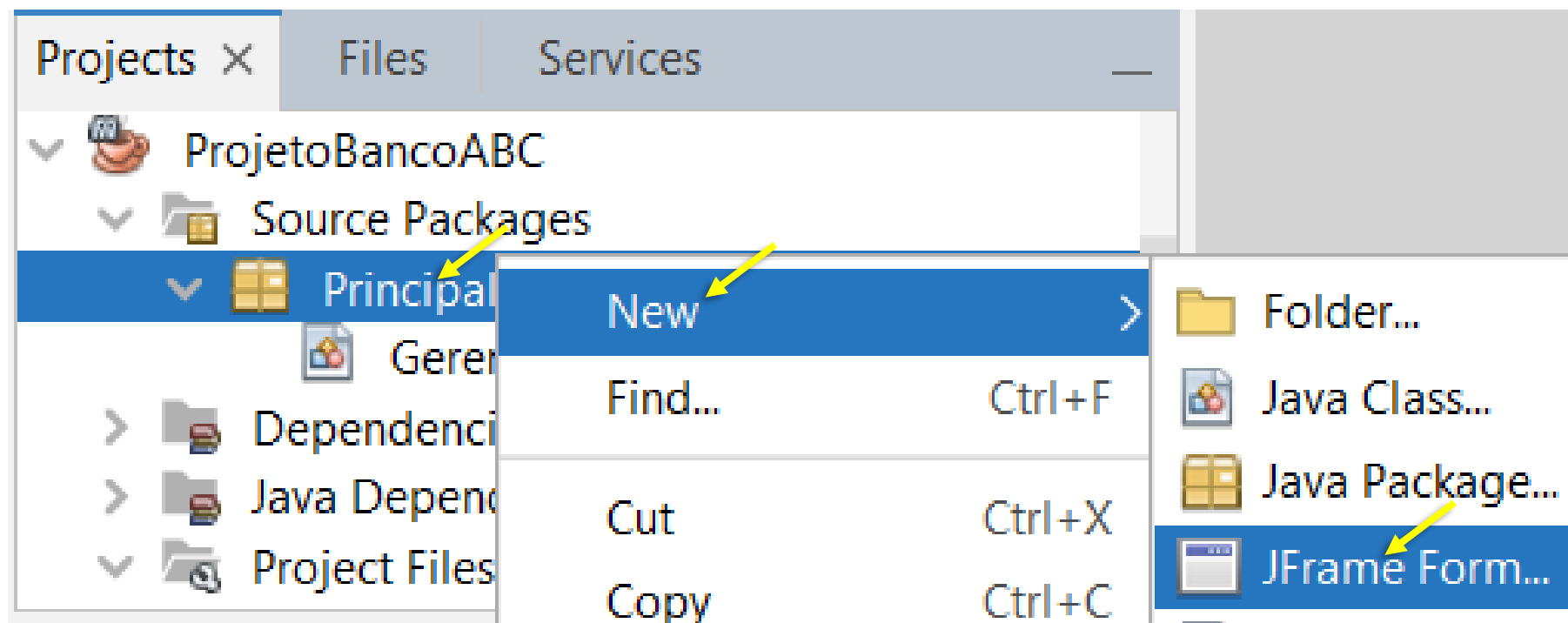


# Classe GerenciadorBD

```
1 package Principal;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5
6 public class GerenciadorBD {
7
8     private final String servidor = "127.0.0.1:3306";
9     private final String banco = "bancoabc";
10    private final String login = "root";
11    private final String senha = "1234";
12
13    public Connection conectar() throws Exception {
14        Class.forName("com.mysql.cj.jdbc.Driver");
15
16        String textoDeConexao = "jdbc:mysql://" + servidor + "/" + banco;
17
18        return DriverManager.getConnection(textoDeConexao, login, senha);
19    }
20 }
```



# Tela de Cadastro de Conta



# Tela de Cadastro de Conta

New JFrame Form

**Steps**

1. Choose File Type
2. **Name and Location**

**Name and Location**

Class Name:

Project:

Location:

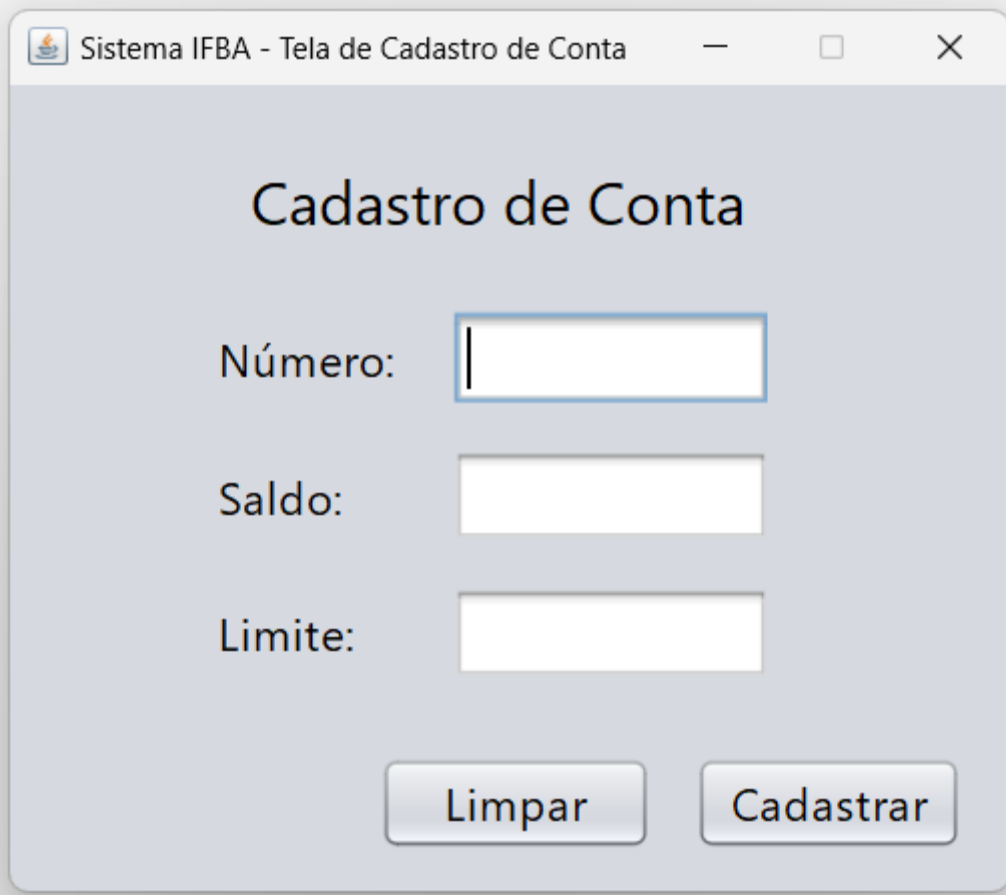
Package:

Created File:

< Back   Next >   **Finish**   Cancel   Help



# Tela de Cadastro de Conta



Sistema IFBA - Tela de Cadastro de Conta

## Cadastro de Conta

Número:

Saldo:

Limite:





# Tela de Cadastro de Conta - Layout

The image shows a Java Swing window titled "Sistema IFBA - Tela de Cadastro de Conta". The window has a title bar with standard OS controls and a close button labeled "resizable: false". The main content area is titled "Cadastro de Conta". It contains three rows of input fields, each preceded by a label. The labels are "Número:", "Saldo:", and "Limite:". The input fields are text boxes. At the bottom, there are two buttons: "Limpar" and "Cadastrar".

Labels and their corresponding UI elements:

- lblTitulo** points to the title "Cadastro de Conta".
- lblNumero** points to the label "Número:".
- txtNumero** points to the text input field for the account number.
- lblSaldo** points to the label "Saldo:".
- txtSaldo** points to the text input field for the account balance.
- lblLimite** points to the label "Limite:".
- txtLimite** points to the text input field for the account limit.
- btnLimpar** points to the "Limpar" button.
- btnCadastrar** points to the "Cadastrar" button.

# Tela de Cadastro de Conta – Import

```
1  package Principal;
2
3  import java.sql.Connection;
4  import java.sql.PreparedStatement;
5  import javax.swing.JOptionPane;
6
7  public class TelaDeCadastroDeConta extends javax.swing.JFrame {
8
9      public TelaDeCadastroDeConta () {
10         initComponents();
11
12         this.setLocationRelativeTo(null);
13     }
```

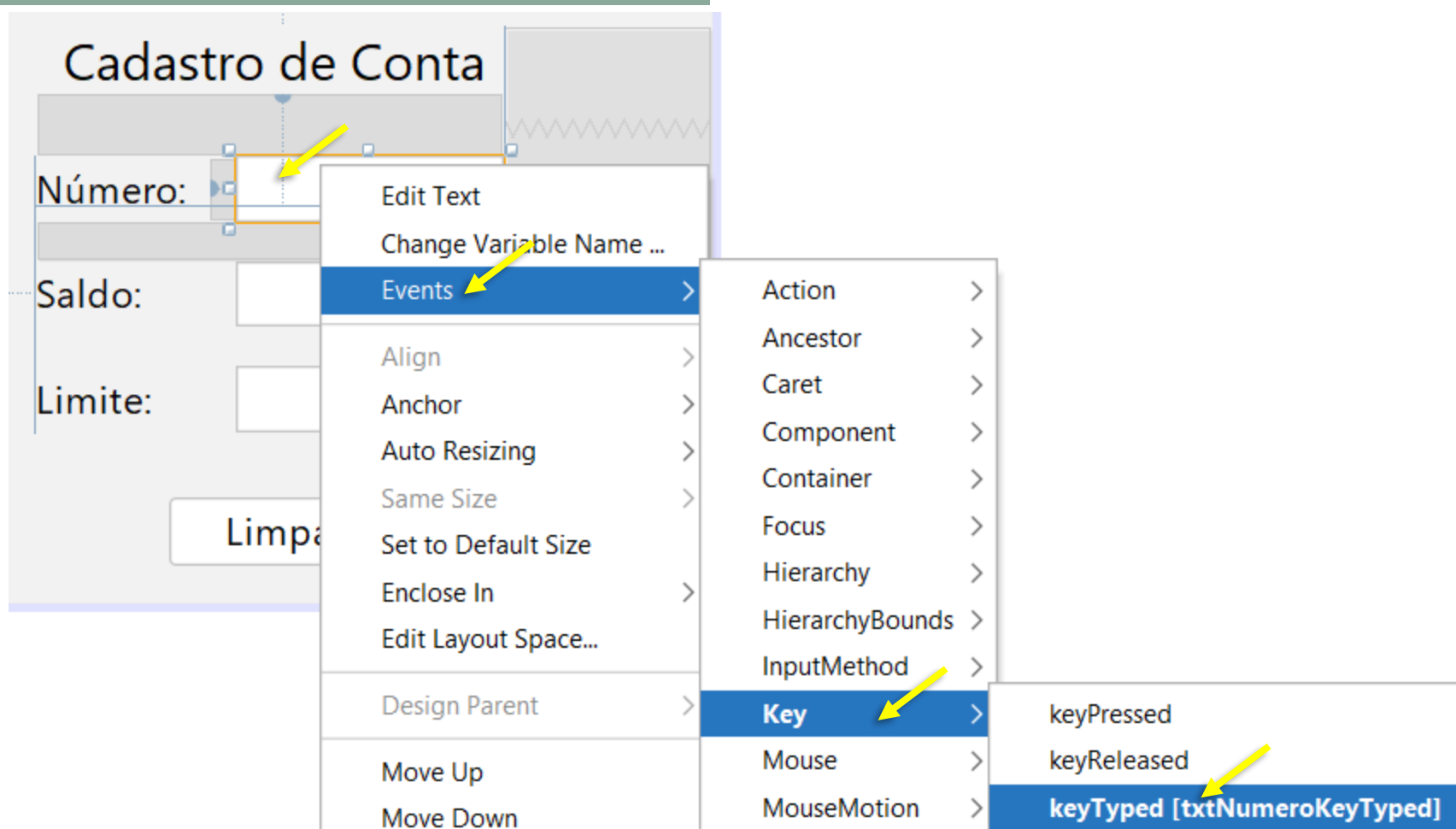


# Tela de Cadastro de Conta – Centralizar

```
1  package Principal;
2
3  import java.sql.Connection;
4  import java.sql.PreparedStatement;
5  import javax.swing.JOptionPane;
6
7  public class TelaDeCadastroDeConta extends javax.swing.JFrame {
8
9      public TelaDeCadastroDeConta () {
10         initComponents();
11
12         this.setLocationRelativeTo(null); ←
13     }
```



# Tela de Cadastro de Conta – Evento 1

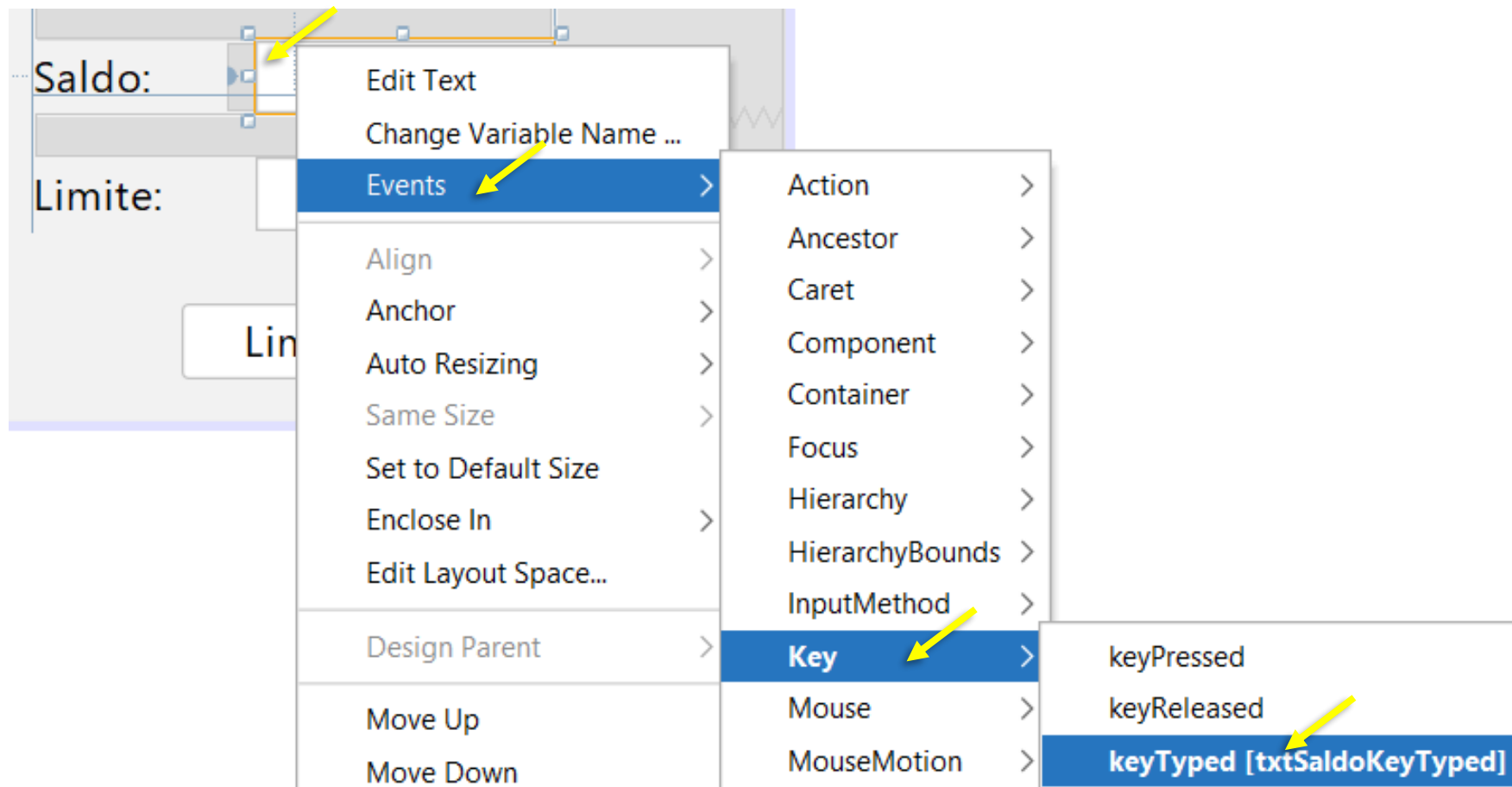


# Tela de Cadastro de Conta – Evento 1

```
private void txtNumeroKeyTyped(java.awt.event.KeyEvent evt) {  
  
    char c = evt.getKeyChar();  
  
    if(!Character.isDigit(c))  
    {  
        evt.consume();  
    }  
  
}
```



# Tela de Cadastro de Conta – Evento 2

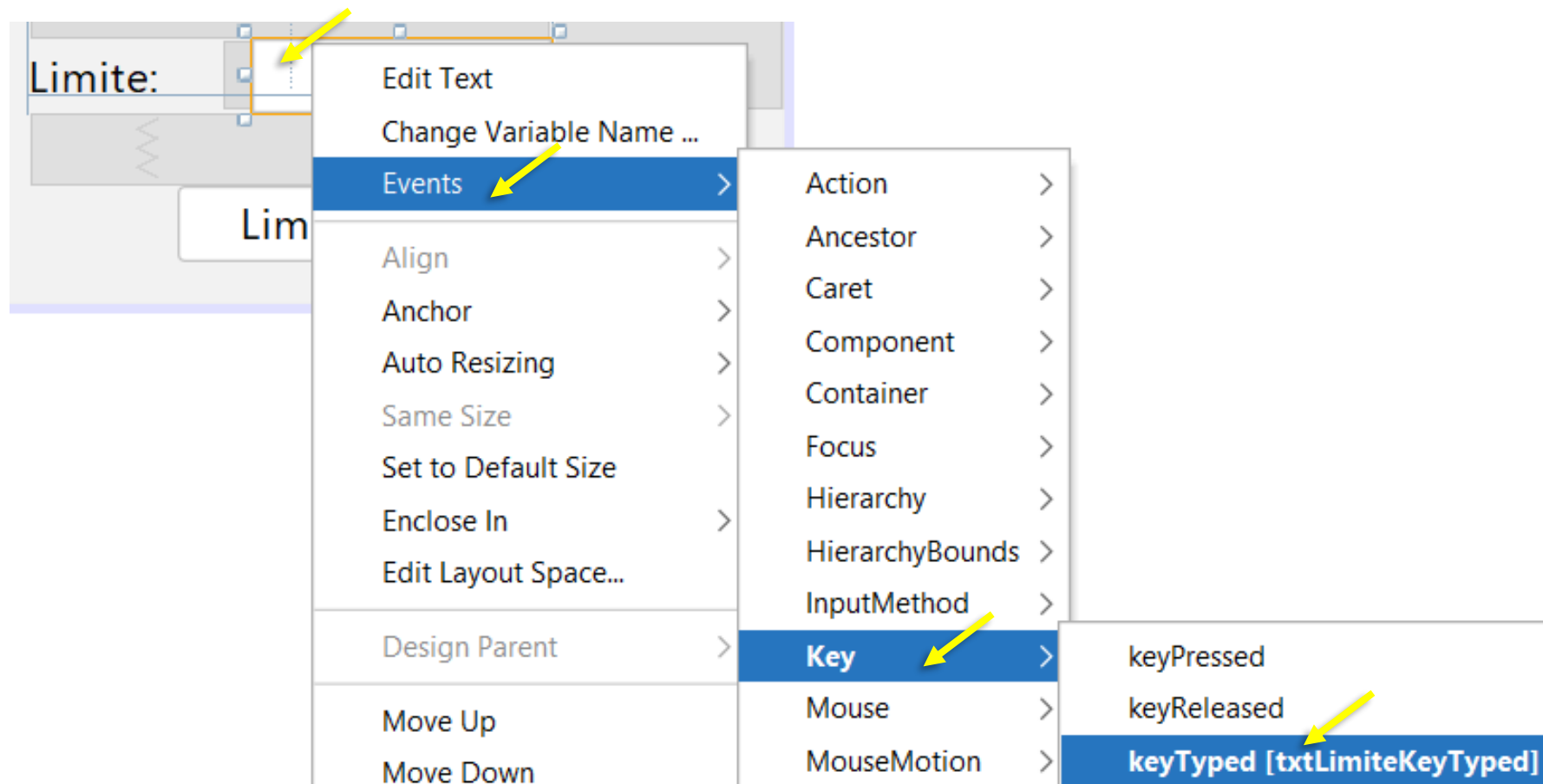


# Tela de Cadastro de Conta – Evento 2

```
private void txtSaldoKeyTyped(java.awt.event.KeyEvent evt) {  
  
    char c = evt.getKeyChar();  
  
    if(!Character.isDigit(c) && c != ',', '.')  
    {  
        evt.consume();  
    }  
  
}
```



# Tela de Cadastro de Conta – Evento 3



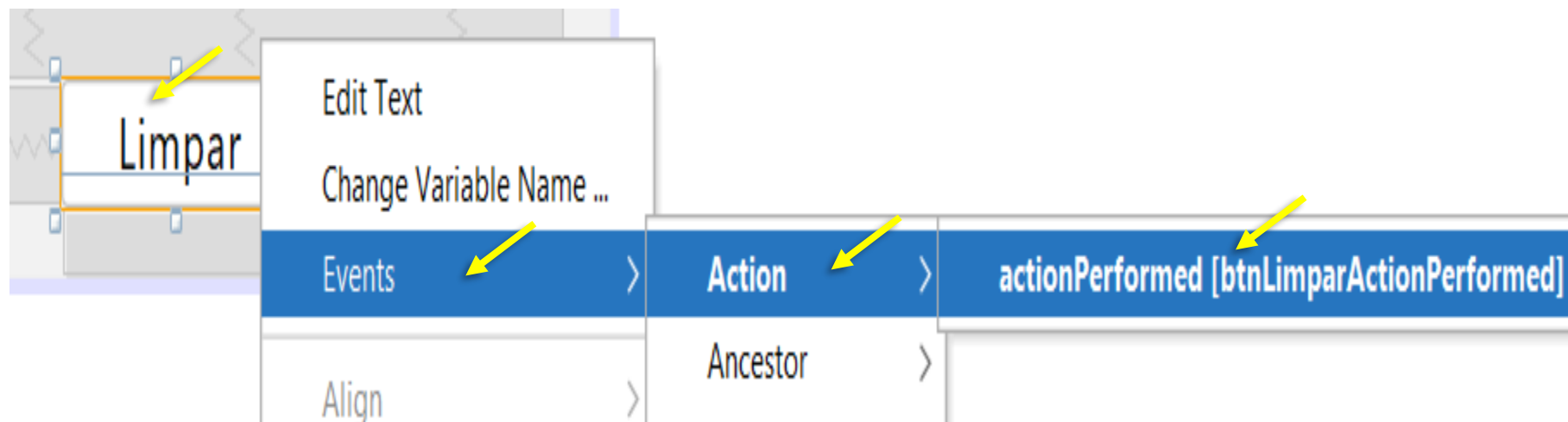


# Tela de Cadastro de Conta – Evento 3

```
private void txtLimiteKeyTyped(java.awt.event.KeyEvent evt) {  
  
    char c = evt.getKeyChar();  
  
    if(!Character.isDigit(c) && c != ',')  
    {  
        evt.consume();  
    }  
  
}
```



# Tela de Cadastro de Conta – Evento 4

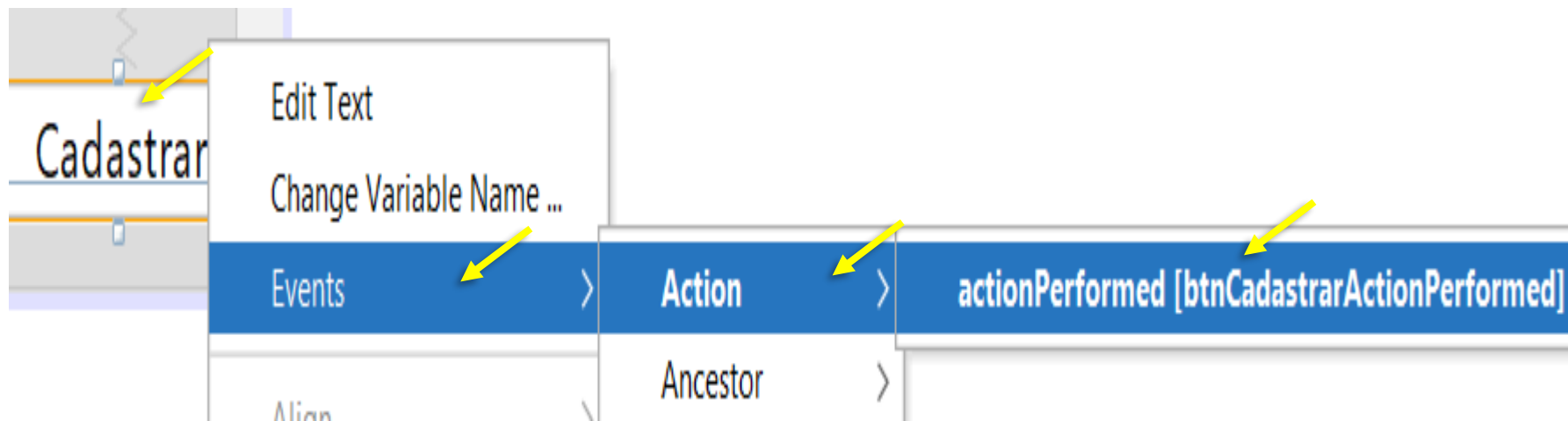


# Tela de Cadastro de Conta – Evento 4

```
private void btnLimparActionPerformed(java.awt.event.ActionEvent evt) {  
    txtNumero.setText("");  
    txtSaldo.setText("");  
    txtLimite.setText("");  
}
```



# Tela de Cadastro de Conta – Evento 5



# Tela de Cadastro de Conta – Evento 5

```
private void btnCadastrarActionPerformed(java.awt.event.ActionEvent evt) {  
    int numero = Integer.parseInt(txtNumero.getText());  
    double saldo = Double.parseDouble(txtSaldo.getText().replace(',', ' '));  
    double limite = Double.parseDouble(txtLimite.getText().replace(',', ' '));  
  
    GerenciadorBD bd = new GerenciadorBD();  
  
    try (Connection conexao = bd.conectar();  
        PreparedStatement comando = conexao.prepareStatement(  
            "INSERT INTO conta (numero, saldo, limite) VALUES (?, ?, ?)"  
        )  
    ) {  
        comando.setInt(1, numero);  
        comando.setDouble(2, saldo);  
        comando.setDouble(3, limite);  
        comando.execute();  
  
        JOptionPane.showMessageDialog(this, "Cadastro realizado com sucesso!");  
  
        txtNumero.setText("");  
        txtSaldo.setText("");  
        txtLimite.setText("");  
    } catch (Exception e) {  
        JOptionPane.showMessageDialog(this, "Erro ao tentar cadastrar uma nova conta!");  
    }  
}
```

# Java Swing – Exercício – Cadastro Conta

Execução

Sistema IFBA - Tela de Cadastro de Conta

## Cadastro de Conta

Número: 1234

Saldo: 1000,99

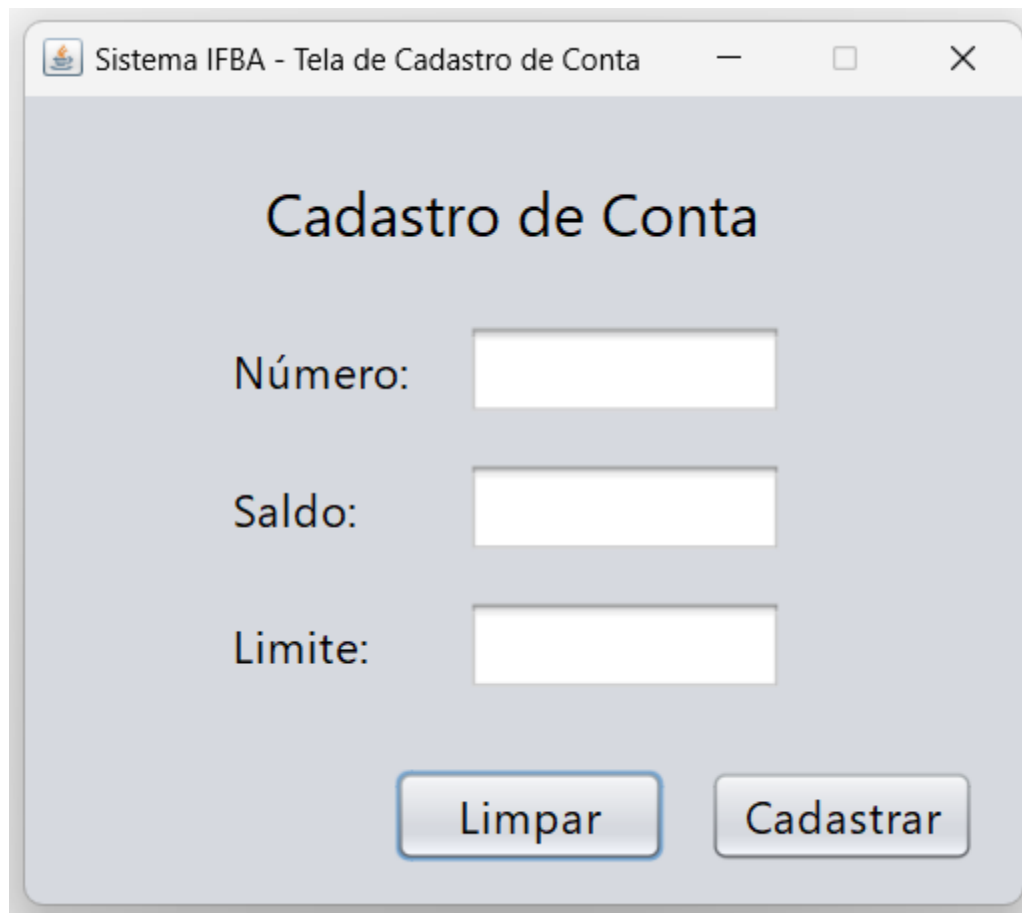
Limite: 200,88

Limpar Cadastrar



# Java Swing – Exercício – Cadastro Conta

Execução



Sistema IFBA - Tela de Cadastro de Conta

## Cadastro de Conta

Número:

Saldo:

Limite:



# Java Swing – Exercício – Cadastro Conta

Execução

Sistema IFBA - Tela de Cadastro de Conta

## Cadastro de Conta

Número: 1234

Saldo: 1000,99

Limite: 200,88

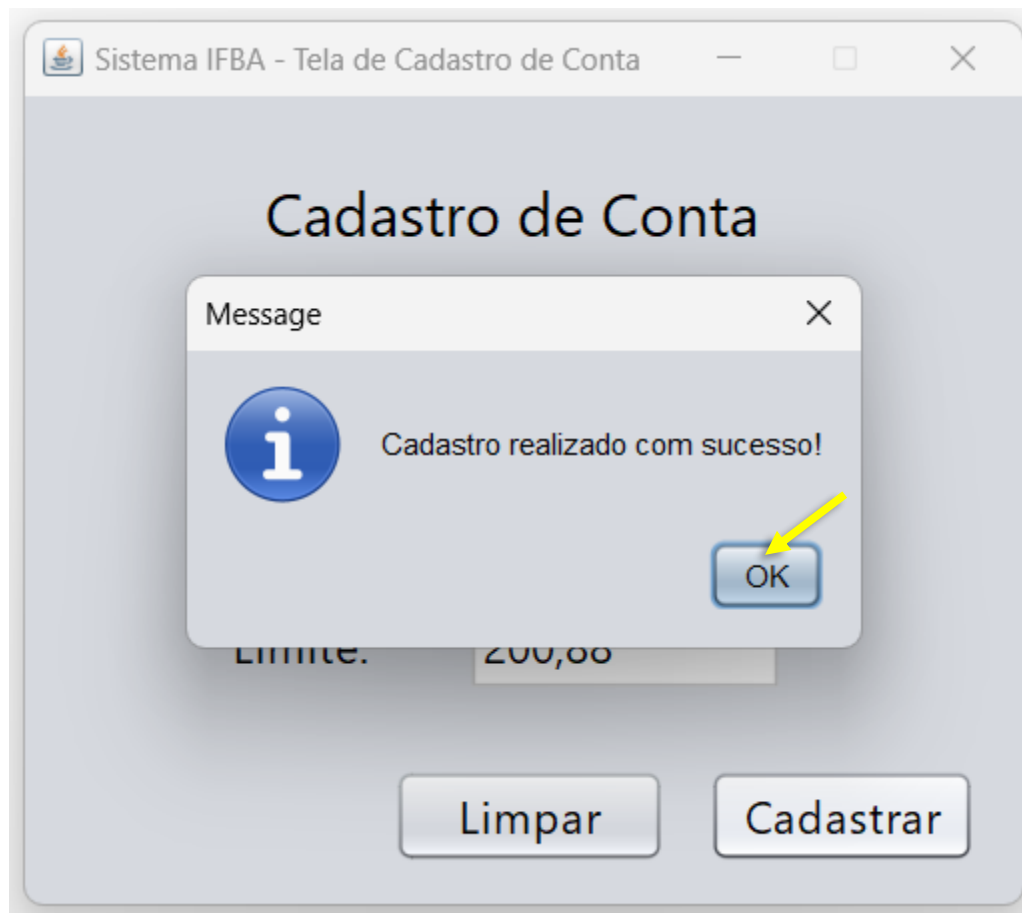
Limpar Cadastrar





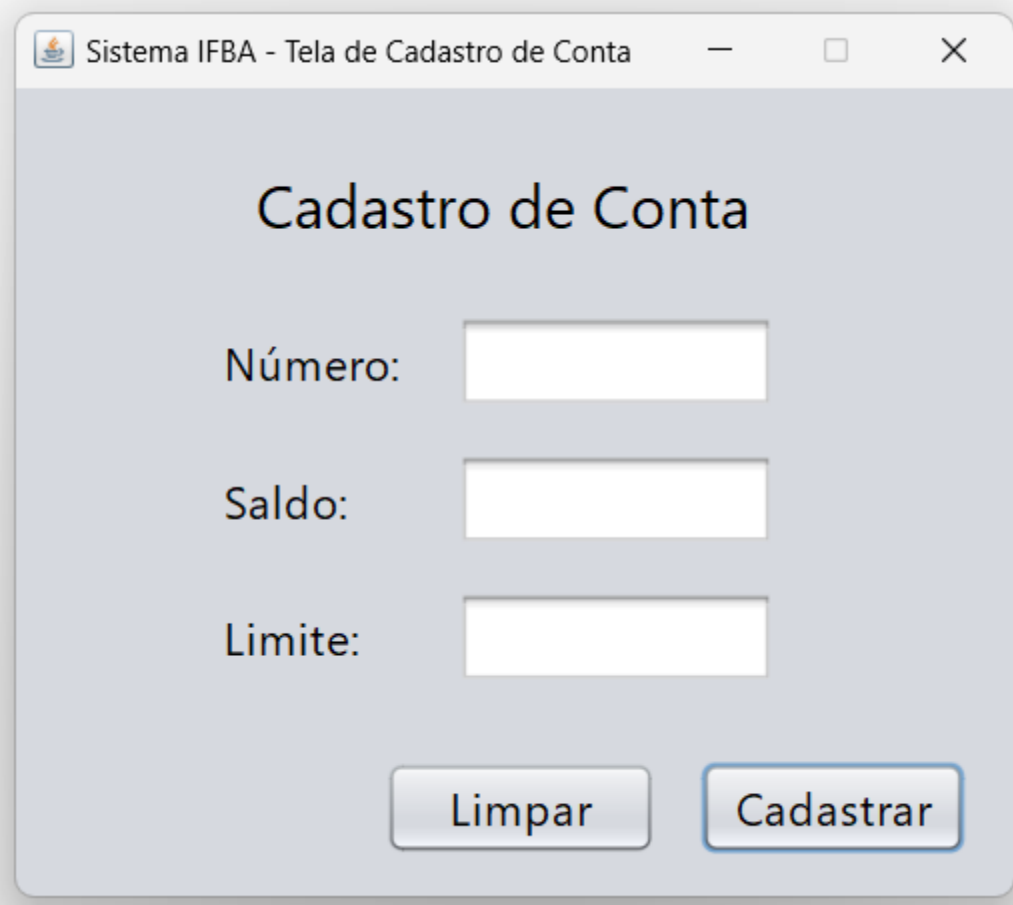
# Java Swing – Exercício – Cadastro Conta

Execução



# Java Swing – Exercício – Cadastro Conta

Execução



Sistema IFBA - Tela de Cadastro de Conta

## Cadastro de Conta

Número:

Saldo:

Limite:



# Java Swing – Exercício – Cadastro Conta

Execução

SQL File 4\*    conta    x

Limit to 1000 rows

```
1 • SELECT * FROM bancoabc.conta;
```

Result Grid    Filter Rows:    Edit:    Export/Import:    Wrap

	id	numero	saldo	limite
▶	1	1234	1000.99	200.88
*	NULL	NULL	NULL	NULL



# Referências

- Junior, Peter Jandl. Java Guia do Programador - 4ª Edição. Novatec Editora.
- SÉRGIO FURGERI. Java Ensino Didático: Desenvolvimento e Implementação de Aplicações. Editora Érica.
- <https://www.youtube.com/playlist?list=PLwQkYMetu0OYFOU71txhtvSYTVUCHjJYD>
- [https://www.youtube.com/playlist?list=PLWd\\_VnthxxLfeEUK08gB7G3316OS5xIT3](https://www.youtube.com/playlist?list=PLWd_VnthxxLfeEUK08gB7G3316OS5xIT3)



# Obrigado!

- Canais de Comunicação;
- Horário de Atendimento.

