

08144 – PARADIGMAS DE PROGRAMAÇÃO B

AULA 04 – LABORATÓRIO 2

Prof. Ademar Takeo Akabane

2020

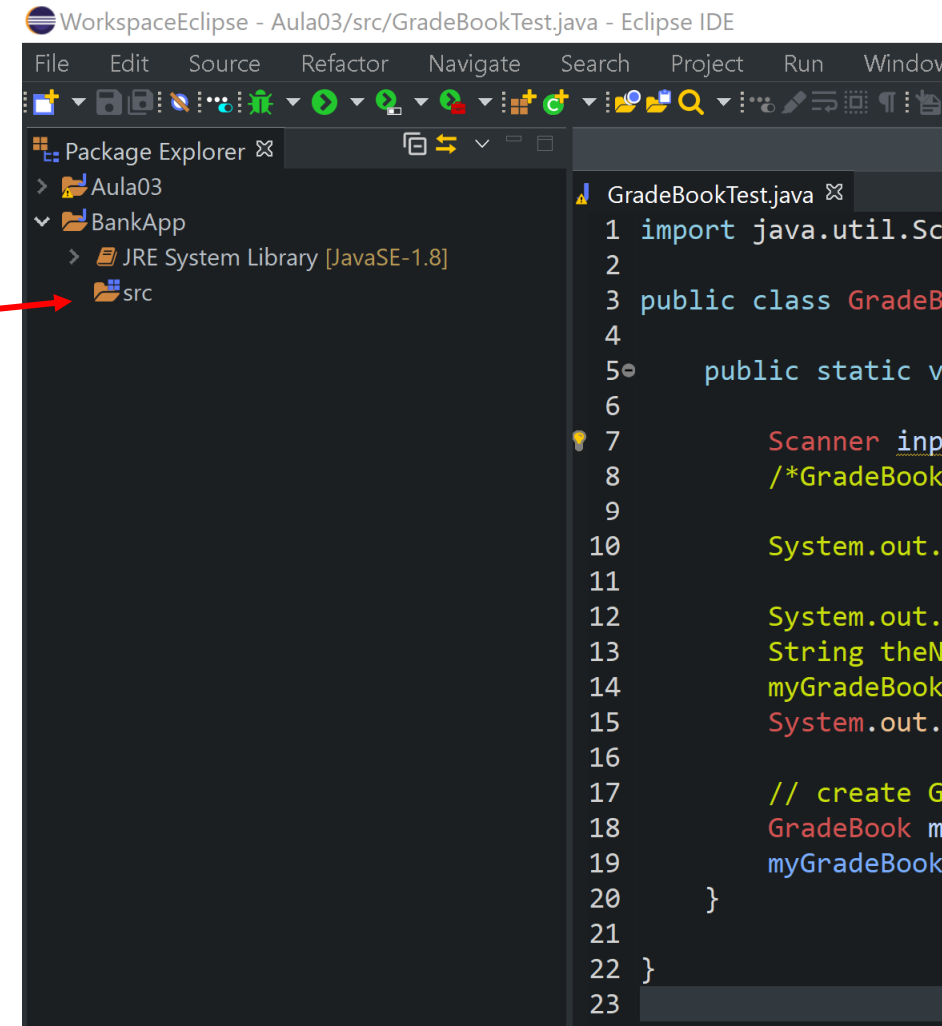
CLASSES E OBJETOS

CRIAÇÃO DE UM PROJETO

- Criar um projeto chamado: **BankApp**
 - File → New → Java Project
 - No campo Project Name digite: BankApp
 - ✓ Clique em: Finish
- Criar duas Classes
 - BankAccount
 - **Variáveis de instância** e **Métodos** da classe
 - BankAccountTest
 - Onde estará a **main()**

CRIAÇÃO DE CLASSES

- Criação das classes
 - Clicar com o botão direito do mouse em cima da pasta **src**
 - New → Class

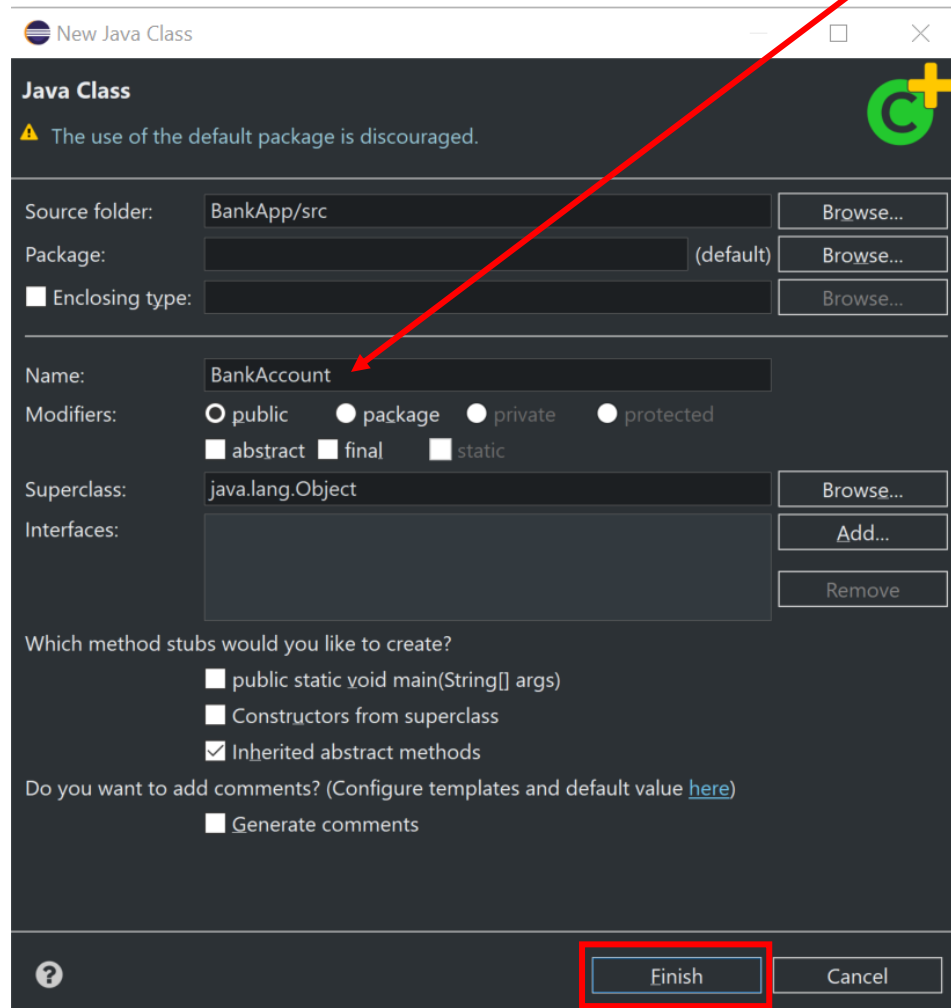


CRIAÇÃO DE CLASSES

- Criação das classes

BankAccount

BankAccountTest



The screenshot shows the 'New Java Class' dialog box. The 'Name' field is filled with 'BankAccount'. A red arrow points from the 'BankAccount' text above to this field. The 'Finish' button at the bottom is highlighted with a red rectangle.

Java Class

⚠ The use of the default package is discouraged.

Source folder: BankApp/src Browse...

Package: (default) Browse...

☐ Enclosing type: Browse...

Name: BankAccount

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)

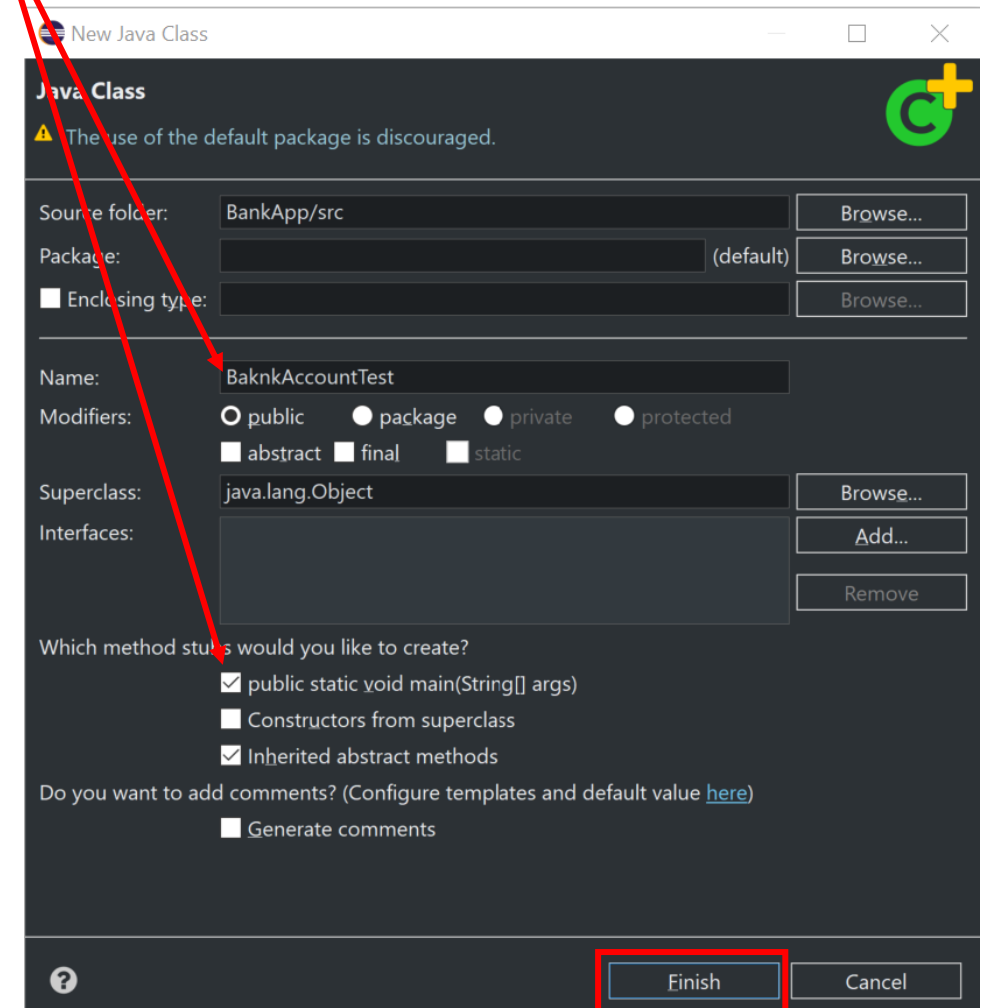
☐ Constructors from superclass

☒ Inherited abstract methods

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

☐ Generate comments

? Finish Cancel



The screenshot shows the 'New Java Class' dialog box. The 'Name' field is filled with 'BankAccountTest'. A red arrow points from the 'BankAccountTest' text above to this field. The 'Finish' button at the bottom is highlighted with a red rectangle.

Java Class

⚠ The use of the default package is discouraged.

Source folder: BankApp/src Browse...

Package: (default) Browse...

☐ Enclosing type: Browse...

Name: BankAccountTest

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

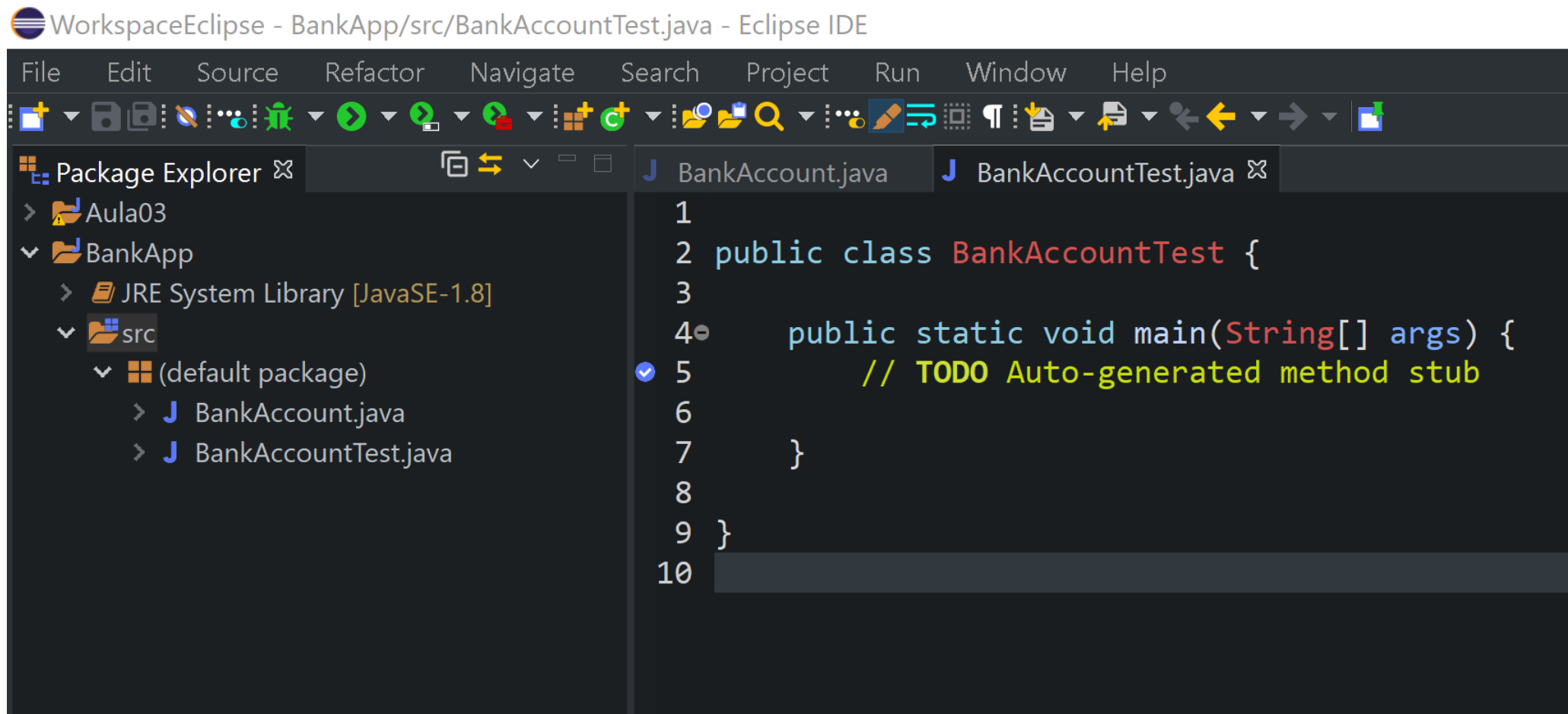
☒ Inherited abstract methods

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

☐ Generate comments

? Finish Cancel

VISÃO GERAL DO ECLIPSE



CLASSE BankAccount

- Na classe BankAccount criar as seguintes variáveis de instância **privadas**
 - Número da conta
 - Nome do correntista
 - Saldo
 - Limite
- Criar os **getters** e **setters** para as variáveis de instância
 - Métodos acessadores (get) e modificadores (set)
- Na classe Conta, criar um método que realiza a **impressão** de todas as variáveis de instância da classe

CLASSE BankAccountTest

- No método **main**
 - Instanciar 3 objetos BankAccount
 - Atribuir valores às variáveis de cada objeto
 - Imprimir as informações de cada objeto

CONTINUAÇÃO

CLASSE BankAccount

- Considerando a Classe BankAccount, desenvolvida na aula anterior, implemente
 - Inclua a variável de classe **senha** (String, private)
 - Esta variável **não** terá métodos set e get
 - **Construtores** para a classe
 - Forneça 2 construtores para inicialização dos objetos da classe, um sem parâmetros
 - ✓ Outro com parâmetros (nome do correntista, número da conta e limite)
 - O saldo deve ser **setado** obrigatoriamente como 0.0
 - A senha terá valor inicial 0000
 - O método setSaldo não deverá mais existir

CLASSE BankAccount

- Considerando a Classe BankAccount, desenvolvida na aula anterior, implemente
 - Desenvolva os métodos `withdraw(double value)` e `deposit(double value)`
 - O usuário só poderá sacar algum valor se houver saldo, ou seu limite cobrir o valor sacado, ficando sua conta negativa
 - Desenvolva um método `changePassword(String oldPass, String newPass)`
 - Permite que o usuário altere a senha, desde que conheça a senha antiga
 - Caso a senha antiga esteja incorreta, avisar ao usuário e não realizar a alteração