

## SI400 – Programação Orientada a Objetos II

Prof. André F. de Angelis  
Oferecimento 2S2022

### Projeto II: Acesso a Banco de Dados

Objetivo: desenvolver um programa em **Java** com interface Swing para realizar consulta a um banco de dados relacional MariaDB rodando remotamente em Linux.

No banco de dados, há uma série de textos da literatura brasileira. Cada texto foi atribuído a um grupo<sup>1</sup> e separado em linhas individuais, cada uma correspondendo a uma linha na tabela. As linhas foram embaralhadas na base de dados.

O programa deve recuperar o texto do grupo<sup>2</sup>, ordenando as linhas via comando SQL e reconstruindo a obra original, que deverá ser gravada em um arquivo local (.txt).

Os dados de conexão são os seguintes:

- Endereço IP: <a ser publicado>
- Porta do serviço: 3306
- RDBMS: mariadb
- Tipo de driver: jdbc
- Base de Dados: SI400
- Tabela: Fragmentos

Os dados de autenticação são os seguintes:

- Usuário: si400\_2022
- Senha: si400\_2022

O programa pode congelar no código (*hardcoded*) as informações de conexão ou solicitá-las numa tela de diálogo.

Porém, os dados de autenticação devem, obrigatoriamente, ser solicitados via tela de diálogo própria, feita pelo grupo, a cada nova execução.

Os campos de dados de interesse na tabela são:

- groupId: int identificação do grupo
- file: char nome do arquivo original de texto
- line: int número sequencial da linha dentro do texto
- text: char linha de texto

O programa deve distribuir as responsabilidades entre classes de forma adequada, considerando inicialização, conexão ao banco de dados, controle, gravação de arquivos, interface gráfica, etc. Se achar conveniente, use pacotes (*packages*) para a organização do código.

Deve ser usado o padrão DAO - *Data Access Object*, construindo uma hierarquia de classes com, ao menos, uma classe (ou interface) DAO abstrata; uma classe concreta para acesso real ao banco de dados; e uma classe concreta para acesso a um banco simulado em memória (fase de desenvolvimento e testes).

Use um *factory method* (ver *Design Patterns*<sup>3</sup>) para a construção polimórfica da instância apropriada da classe concreta, conforme a situação (desenvolvimento ou produção).

---

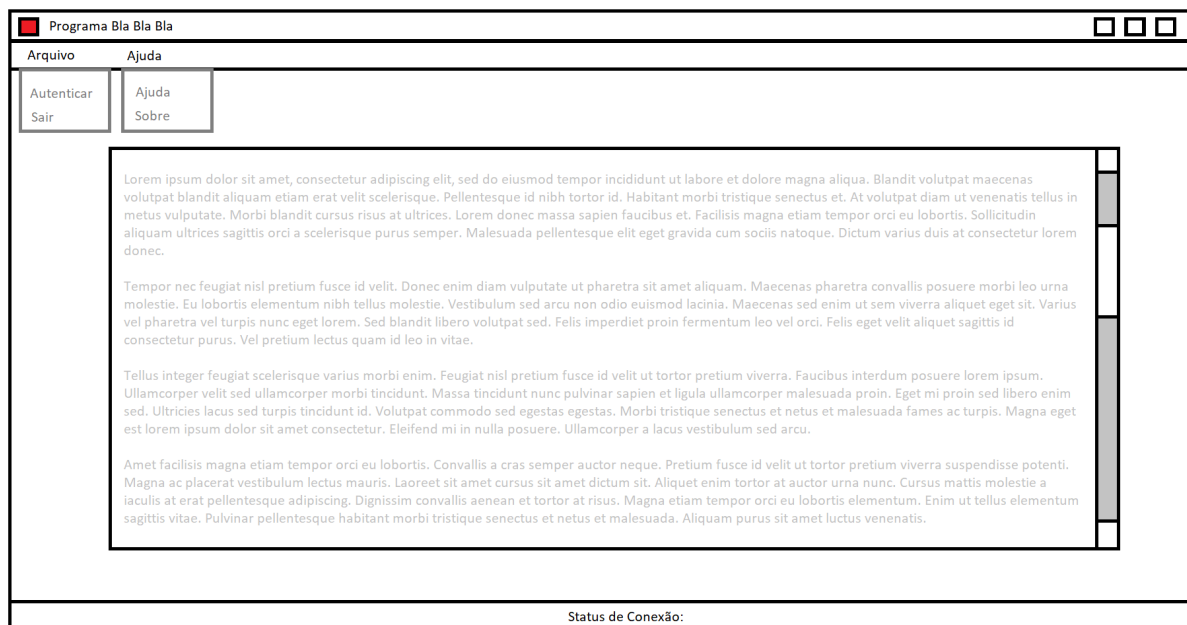
1 Para grupos cujo nome é uma letra, considerar a ordem numérica: A → 1, B → 2, ...

2 O acesso de leitura é livre para todos os grupos e, portanto, todos os textos podem ser recuperados e lidos como cultura geral e lazer literário.

3 Informações adicionais no Café Forte Academic Channel – vídeos 45 a 47.

O programa deve ter uma interface gráfica de usuário (GUI) baseada em componentes Swing, tendo como janela principal uma derivação personalizada de *JFrame*, com menus<sup>4</sup>.

O texto recuperado, devidamente ordenado, deve ser mostrado na área de fundo da janela principal por um componente *TextArea* ou *Table*, qualquer um deles com barras de rolagem, a critério do grupo. Veja protótipo da tela principal a seguir:



São necessárias, **ao menos**, as seguintes opções:

- |                 |                         |                                   |
|-----------------|-------------------------|-----------------------------------|
| • Menu Arquivo: | Autenticação do usuário | Abre diálogo de autenticação      |
| •               | Saída do programa       | Fecha conexões e termina programa |
| • Menu Ajuda    | Ajuda                   | Texto explicativo do programa     |
| •               | Sobre                   | Versão e créditos de autoria      |

Na tela de créditos, devem ser apresentados os nomes e as fotos de cada integrante do grupo (é parte do exercício trabalhar com imagens).

Caso o grupo opte por recuperar também outros textos, devem ser usados diálogos para selecionar a obra consultada (campo groupId) e definir o arquivo de saída.

Implemente e depure o programa, gere a documentação com a ferramenta *javadoc*.

A entrega do projeto será feita como segue, no Moodle:

- 1 arquivo compactado contendo todo o código fonte ( zip);
- 1 arquivo jar executável (.jar);
- 1 arquivo do relatório de contribuição (.pdf);
- 1 arquivo de figura com o diagrama de classes UML do projeto (.jpg);
- 1 arquivo de texto (.txt) com o texto do grupo recuperado do banco de dados.
- 1 link para o vídeo demonstrativo de funcionamento (todas as telas e funções) do programa

## **Critérios de Avaliação**

- Só serão considerados os programas que compilem sem erro e executem corretamente a(s) funcionalidade(s) requerida(s).

#	Item	Max. pontos
01	Entrega completa de todos os itens solicitados	1,0
02	Qualidade da interface gráfica	1,5
03	Arquitetura de classes (estruturação, divisão de responsabilidades, diagrama UML)	2,0
04	Qualidade do código (estilo, indentação, organização, limpeza)	1,5
05	Demonstração do programa em execução (vídeo)	1,5
06	Recursos extras significativos	1,0
07	Aspecto geral do trabalho	1,5
	Total	10,0