### **Documentação do Código Java**

O Java possui uma ferramenta padrão para geração de documentação chamada **Javadoc**. Você já começou um bom trabalho de documentação diretamente no seu código com os comentários /\*\* ... \*/. O Javadoc utiliza essas *doc comments* para gerar páginas HTML bem formatadas que descrevem suas classes, métodos, atributos, etc.

**Passos para gerar a documentação Javadoc:**

1. **Certifique-se de que todos os seus arquivos .java tenham comentários Javadoc adequados.** Isso inclui:
   1. Comentários de nível de classe (/\*\* Classe principal ... \*/).
   2. Comentários de nível de método (/\*\* Método de inicialização ... @param stage ... \*/).
   3. Comentários para atributos (se necessário).
   4. Use as tags @param, @return, @throws, @author, @version, etc., conforme apropriado.
2. **Utilize a ferramenta Javadoc.** A ferramenta javadoc é geralmente incluída no seu JDK (Java Development Kit). Você pode executá-la a partir da linha de comando.

Navegue até o diretório raiz do seu projeto (onde a pasta org reside) e execute o seguinte comando:

Bash

javadoc -d doc -sourcepath src org.example.gestaodehorario

* 1. -d doc: Especifica o diretório de saída para os arquivos de documentação (neste caso, uma pasta chamada doc será criada).
  2. -sourcepath src: Indica onde os arquivos de código fonte .java estão localizados.
  3. org.example.gestaodehorario: Especifica os pacotes para os quais você deseja gerar a documentação. Você pode especificar pacotes individuais ou usar . para documentar todos os pacotes e subpacotes.

1. **Abra os arquivos HTML gerados.** Após a execução do comando, uma pasta doc será criada no seu diretório raiz. Dentro dela, você encontrará um arquivo chamado index.html. Abra este arquivo em seu navegador para visualizar a documentação gerada.

**Exemplo da documentação Javadoc para a classe Main:**

A partir dos seus comentários, o Javadoc geraria uma página HTML para a classe Main que incluiria:

* Uma descrição geral da classe: "Classe principal que inicia a aplicação JavaFX e configura o ambiente inicial. Responsável por: Inicializar o palco (stage) principal, Configurar o tratamento de exceções não capturadas, Gerenciar a navegação entre telas."
* Detalhes sobre o método start(Stage stage), incluindo a descrição "@param stage Palco principal fornecido pelo JavaFX".
* Detalhes sobre o método main(String[] args), incluindo a descrição "@param args Argumentos de linha de comando".

**Exemplo da documentação Javadoc para a classe ScreenManager:**

Similarmente, para a classe ScreenManager, a documentação incluiria:

* Uma descrição geral da classe: "Classe utilitária para gerenciamento de telas e navegação na aplicação JavaFX. Responsável por: Configurar o palco (stage) principal, Carregar telas FXML, Aplicar estilos CSS, Manter configurações consistentes de exibição."
* Detalhes sobre o método setPrimaryStage(Stage stage), incluindo a descrição "@param stage Palco JavaFX a ser configurado".
* Detalhes sobre o método changeScreen(String fxmlPath, String cssPath), incluindo as descrições "@param fxmlPath Caminho relativo do arquivo FXML (a partir de resources/)", "@param cssPath Caminho opcional para arquivo CSS personalizado (a partir de resources/)", e "@throws RuntimeException Se ocorrer erro no carregamento do FXML/CSS".

### **Documentação da Base de Dados MySQL**

A documentação da sua base de dados MySQL é crucial para entender a estrutura, os relacionamentos e o propósito dos dados. Aqui estão algumas maneiras de documentar seu banco de dados:

1. **Diagrama Entidade-Relacionamento (DER):** Um DER é uma representação visual da estrutura do seu banco de dados, mostrando tabelas, colunas, tipos de dados, chaves primárias e estrangeiras, e os relacionamentos entre as tabelas. Existem várias ferramentas que podem gerar DERs a partir de um banco de dados existente (por exemplo, MySQL Workbench). Manter o DER atualizado é fundamental.
2. **Dicionário de Dados:** Um dicionário de dados é um documento textual que descreve cada tabela e suas colunas em detalhes. Para cada tabela, você deve incluir:
   1. Nome da tabela
   2. Descrição da finalidade da tabela
   3. Para cada coluna:
      1. Nome da coluna
      2. Tipo de dados
      3. Restrições (NOT NULL, UNIQUE, PRIMARY KEY, FOREIGN KEY)
      4. Descrição do propósito da coluna
      5. Valores padrão (se houver)
      6. Exemplos de dados (opcional)
3. **Comentários no SQL:** Você pode adicionar comentários diretamente no seu código SQL ao criar as tabelas e colunas. Embora isso não seja uma documentação externa formal, é útil para quem consulta o esquema diretamente.

SQL

CREATE TABLE usuarios (  
 id INT AUTO\_INCREMENT PRIMARY KEY COMMENT 'Identificador único do usuário',  
 nome VARCHAR(255) NOT NULL COMMENT 'Nome completo do usuário',  
 email VARCHAR(255) UNIQUE NOT NULL COMMENT 'Endereço de email único do usuário',  
 data\_cadastro TIMESTAMP DEFAULT CURRENT\_TIMESTAMP COMMENT 'Data e hora em que o usuário foi cadastrado'  
);

1. **Ferramentas de Documentação de Banco de Dados:** Existem ferramentas específicas para documentar bancos de dados, como SchemaSpy, que podem gerar documentação HTML a partir do seu esquema MySQL, incluindo diagramas, detalhes de tabelas e relacionamentos.

**Recomendações para a Documentação do Banco de Dados:**

* **Mantenha a documentação atualizada:** Qualquer alteração no esquema do banco de dados deve ser refletida na documentação.
* **Escolha o formato adequado:** Um DER é ótimo para uma visão geral, enquanto um dicionário de dados fornece detalhes completos. Uma combinação de ambos é ideal.
* **Centralize a documentação:** Armazene a documentação em um local acessível a todos os membros da equipe (por exemplo, em um sistema de gerenciamento de documentos ou em um wiki interno).