

SQLBird

(para MySQL e PostgreSQL)

Luis Camilo Jussiani Moreira : 206316

João Victor Laskoski : 1906470

Curitiba, 2020

Introdução :

O programa foi feito através da biblioteca Java Swing (utilizando linguagem Java), no qual tem como intuito explorar os dados e consultas de um banco de dados. Esse programa possui integração tanto com o MySQL quanto com o PostgreSQL.

Execução do SQLBird :

.Consideração inicial para o funcionamento :

Para que o programa grave as conexões inseridas pelo usuário, foi criado um banco de dados para guardar a url, usuário e senha da conexão. Sendo assim, através do MySQL Workbench, foi executado o seguinte código :

```
CREATE DATABASE users_connection;  
CREATE TABLE connection (  
    url varchar(100) ,  
    usuario varchar(100) ,  
    senha varchar(50),  
    primary key (url, usuario, senha)  
);
```

Tal código também pode ser criado usando como sistema o Postgre. Logo, antes da execução do programa para que possua o funcionamento adequado, deve-se executar o comando para a criação do banco de dados e a tabela de conexão e além disso, modificar 2 linhas de código encontradas no seguintes locais.

.Arquivo NewDialog1.java - linha 197

```
196 |  
197 | boolean testeConexao; //COLOCAR OS ERROS quando a conexao ja existe DE FATO TALVEZ EM UMA JANELA  
198 | try {  
199 |     con = DriverManager.getConnection("jdbc:mysql://localhost:3306/users_connection", "root", "031"  
200 |     testeConexao = testaConexao(url, usuario, senha);  
201 |  
202 |     if(testeConexao){ //FAZER DEPOIS UMA JANELA QUE APAREÇA A MENSAGEM DE ERRO  
203 |         addConexaoBd(url, usuario, senha);  
204 |         dispose();  
205 |         conexaoAberta = true;  
206 |         con.close();  
207 |     }
```

Dentro da função “testeConexao”, deve-se mudar a url, usuário e senha para a função *getConnection*, de acordo com os mesmos utilizados para o banco de dados “users_connection” criado previamente.

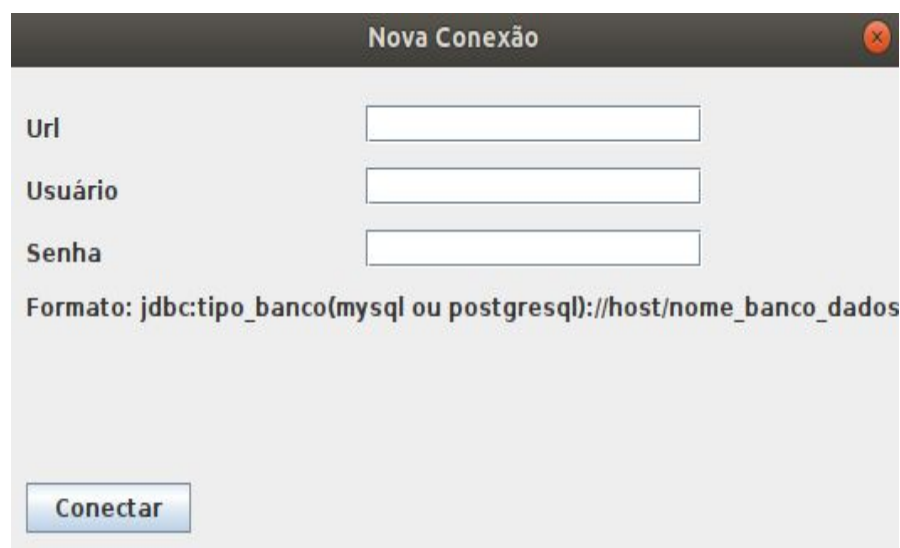
.Arquivo Conexoes.java - linha 32

```
23 |  
24 | public void conectar() throws SQLException{  
25 |     con = DriverManager.getConnection("jdbc:mysql://localhost:3306/users_connection", "root", "03178964");  
26 | }
```

Na função conectar, deve-se mudar de mesmo modo a url, usuário e senha, de acordo com os mesmos usados para o banco de dados para a conexão de usuários. Após isso, a execução do programa já pode ser feita. Caso não seja criado o banco de dados “users_connection”, será apresentado no momento de fazer a conexão, um erro apontando a não existência do banco de conexões.

Conexões :

Ao executar o programa, aparecerá a janela de conexões, em que pode-se escolher uma conexão já utilizada ou inserir uma nova conexão.

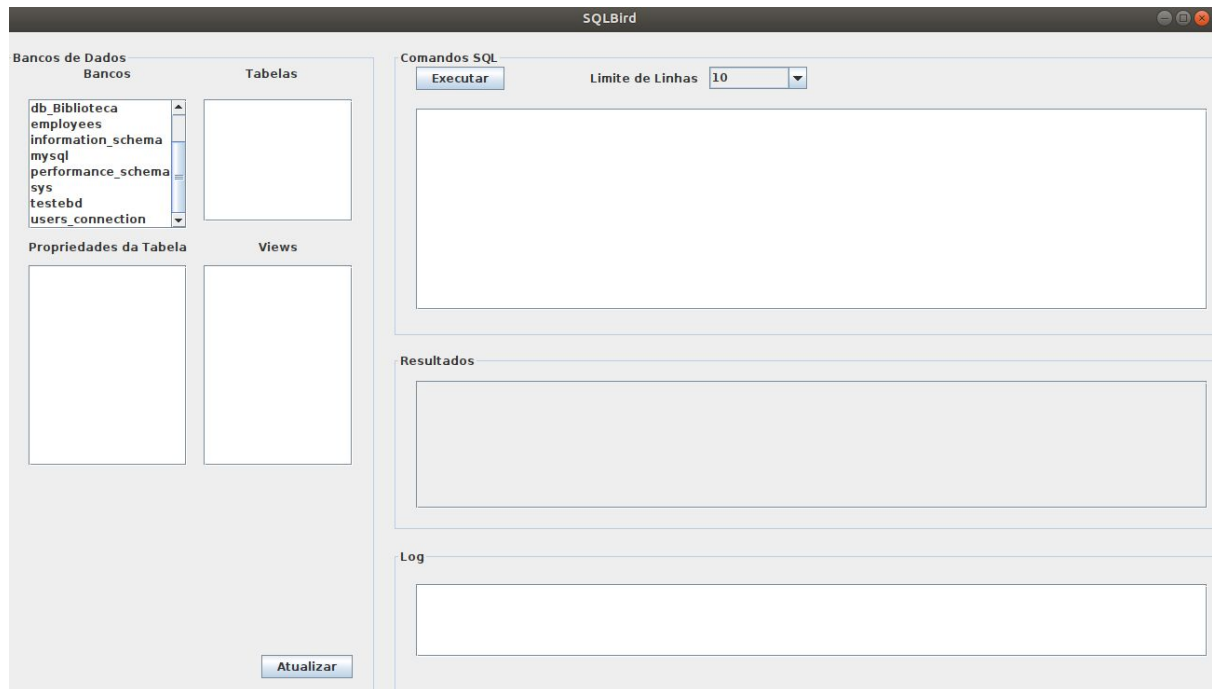
A imagem mostra uma janela de diálogo intitulada "Nova Conexão". Ela possui um cabeçalho escuro com o título e um botão de fechar (X) no canto superior direito. O corpo da janela é cinza claro e contém três campos de entrada de texto rotulados "Url", "Usuário" e "Senha". Abaixo desses campos, há um texto de exemplo: "Formato: jdbc:tipo_banco(mysql ou postgresql)//host/nome_banco_dados". No canto inferior esquerdo, há um botão azul com o texto "Conectar".

Janela Nova Conexão

Nessa janela, há um formato específico da url para ser inserida que está no texto Formato, em que é preciso definir o tipo do banco a trabalhar, o host e o nome do banco de dados que irá executar inicialmente, caso inserir algum campo que seja inválido, a conexão será testada e aparecerá uma janela de erro. Inserindo uma conexão válida mas que a url não siga o padrão colocado, o programa apresentará problemas na funcionalidade durante a execução, como por exemplo, não demonstrar certamente os bancos, tabelas e views do banco utilizado no momento.

Consequinte, colocando os campos corretos, será salvo os dados no banco de dados “users_connection” e o usuário será direcionado para Janela Principal, contudo a nova conexão só será vista no campo de conexões anteriores (na Janela Conexão) na próxima execução do SQLBird.

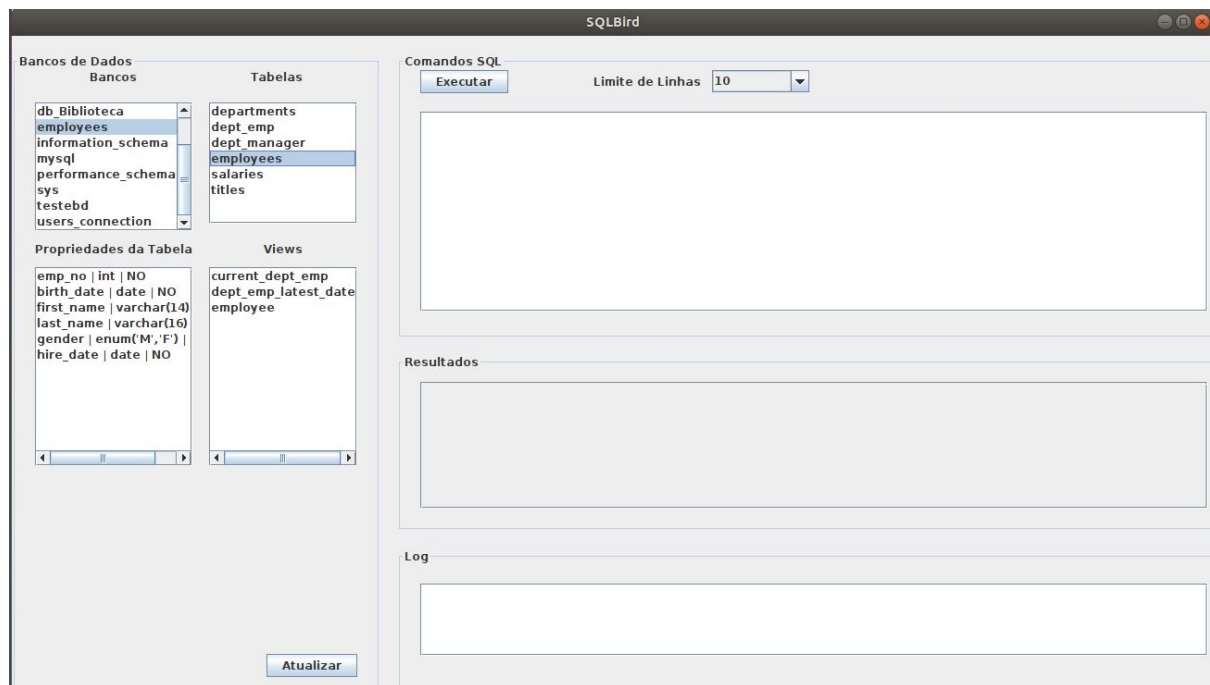
Janela Principal (Workbench) :



Janela Principal

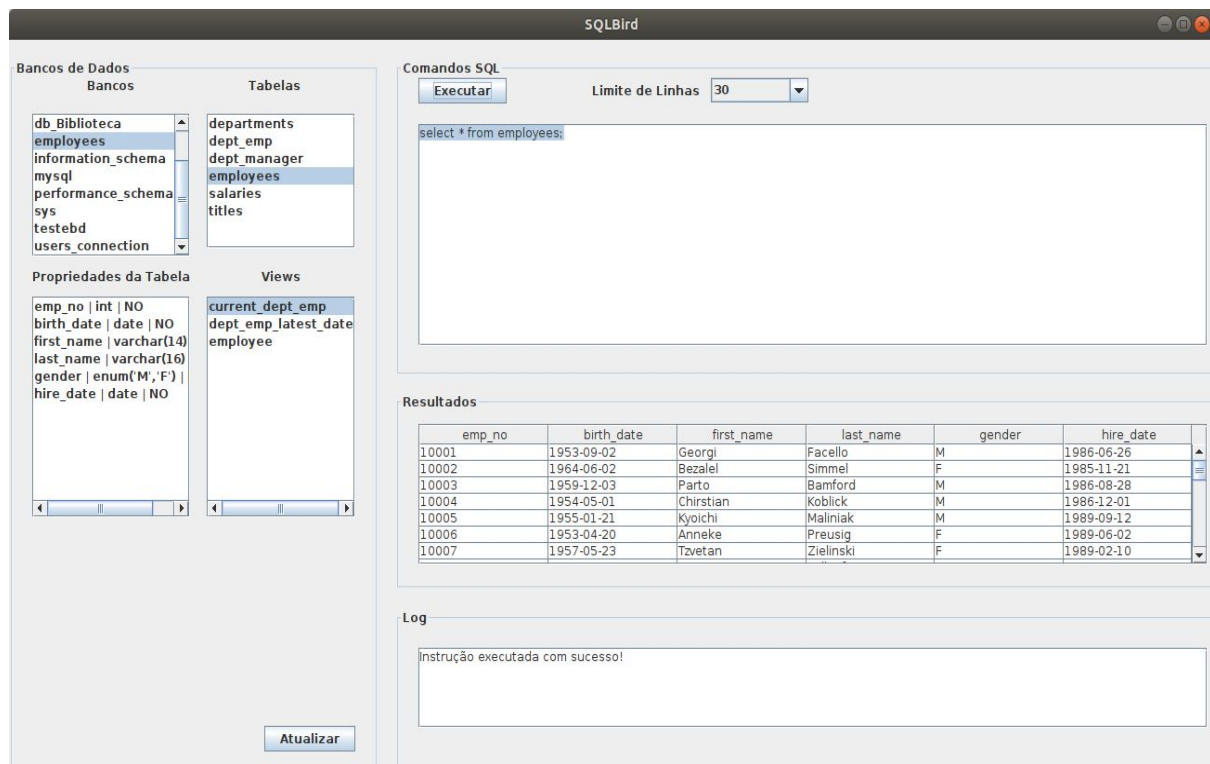
Nessa janela, é onde poderemos trabalhar com os dados e consultas de fato. Primeiramente no campo da lista de Bancos, caso a conexão utilizada seja de um banco do MySQL, irá mostrar todos os bancos criados no mesmo sistema. Por outro lado, caso a conexão inicial seja para um banco PostgreSQL, será exibido os bancos criados no Postgre.

Ao selecionar com o clique do mouse um banco, o mesmo será selecionado para o uso, além de mostrar as tabelas e views correspondentes. Ainda mais, escolhendo as tabelas de mesmo modo, serão expostas as propriedades da Tabela, com os campos (chave primária, nome do campo, tipo do campo, é nulo). Ainda mais, é possível selecionar os bancos e as tabelas com as setas (para cima e para baixo).



Janela Principal exibindo tabelas, views e propriedades da tabela

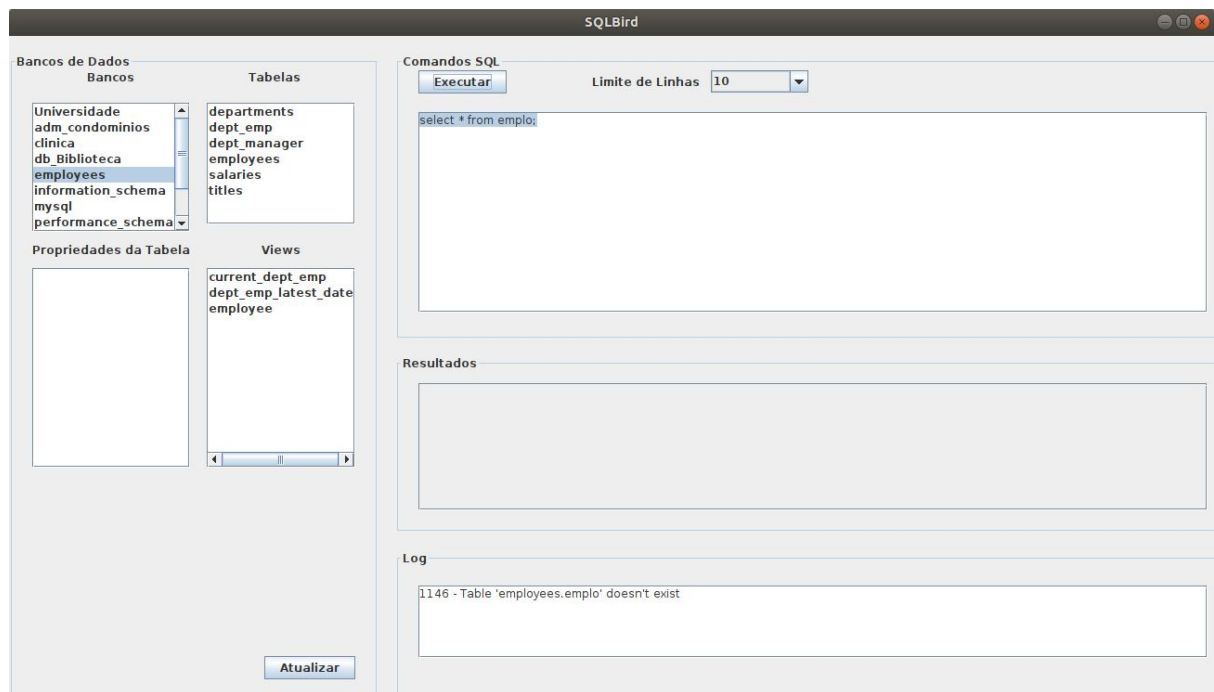
Tratando sobre os comandos SQL de fato, em que podem ser inseridos no campo de texto dentro da partição “Comandos SQL”, de primeiro modo, pode-se optar pelos limites de linhas demonstradas nos resultados. Após escolher o limite e digitar o comando, deve-se selecionar todo o texto pretendido para executar com o mouse e clicar no botão Executar (são executados apenas os comandos selecionados com o mouse, os demais são ignorados). Contudo, na situação de nenhum comando específico ser selecionado, será executado todo o texto contido no campo.



Execução de uma query e demonstração do resultado

Todos os comandos SQL, englobando inserção, criação, alteração, entre outros podem ser executados. Caso alterado um banco de dados, tabela, entre outros, não aparecerá de imediato nas listas que as compõem. Sendo assim, é necessário apenas clicar com o mouse no botão Atualizar, que será mostrado os novos resultados.

Para queries não executadas com êxito, no campo Log irá aparecer o número do erro e a mensagem do mesmo.



Demonstração do resultado de uma query sem êxito de execução