



EducaCiência FastCode

Fala Galera,

- Artigo: 19/2020 Data: Abril/2020
- Público Alvo: Desenvolvedores – Iniciantes
- Tecnologia: Java
- Tema: Usando clausula “Try/Catch” na conexão com MySql
- Link: <https://github.com/perucello/DevFP>

Neste artigo vamos desenvolver um Script de conexão com o Banco de Dados MySql usando de uma cláusula try/Catch porém, iremos trazer algumas maneiras de manipular esta String.

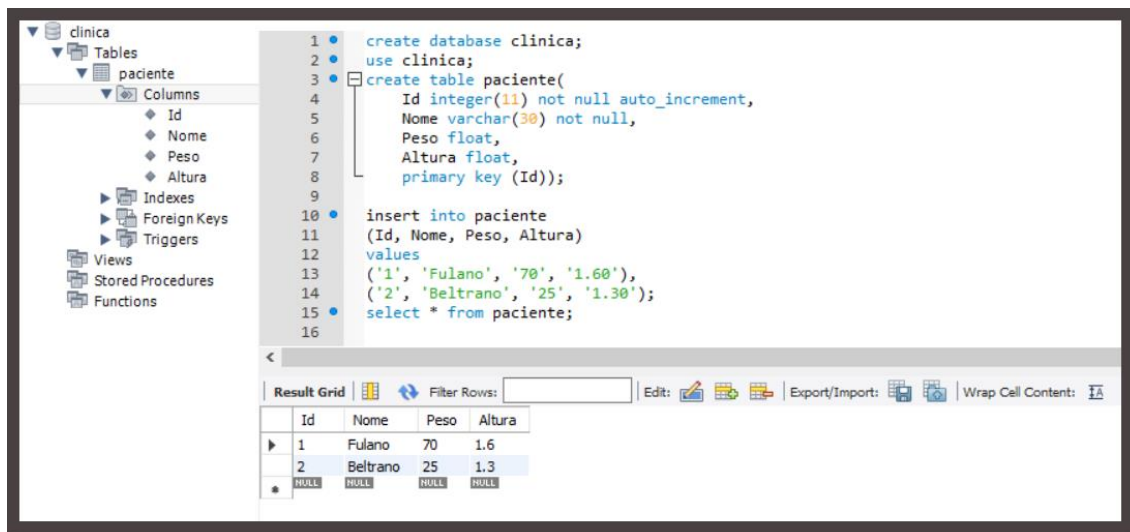
Vamos utilizar de “sout” e também da biblioteca swing “JOptionPane”.

Lembrando que o artigo de número 18 – abordei uma maneira pouco performática utilizando “if”, se tiver curioso, basta baixar o artigo 18 e os Scripts e bons estudos.

Para este ambiente , já temos nosso Banco de Dados criado e chamado “clinica”. Elaboraremos nossa String de Conexão a partir de agora, porém , utilizaremos de uma condição que ao meu ponto de ver , é muito mais utilizada , onde eu mesmo utilizei deste método de conexão , mas, visto que estou ministrando curso de Java , resolvi desenvolver algumas maneiras de conectar ao Banco de Dados para os alunos e vendo os comentários, optei em trazer em caráter de Artigo para toda a comunidade.

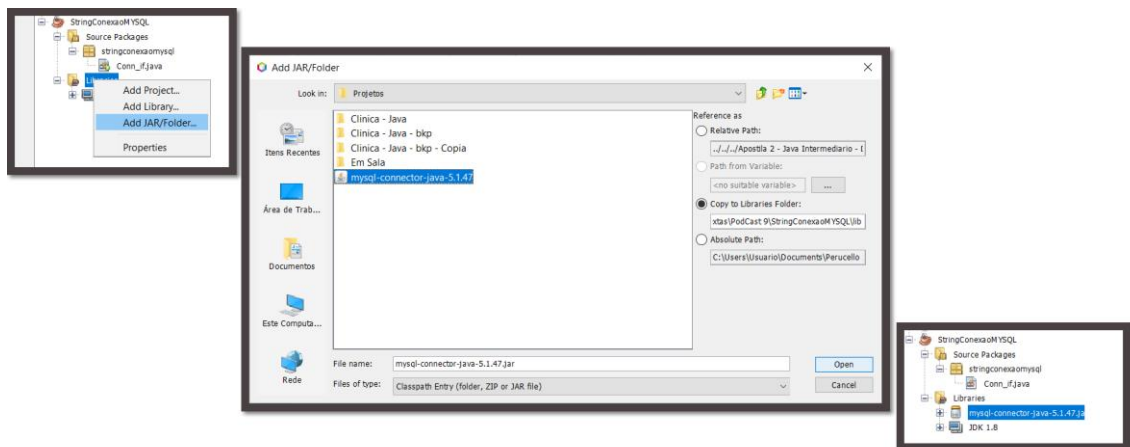


Nosso Ambiente – Banco de Dados MySQL utilizando Workbench:



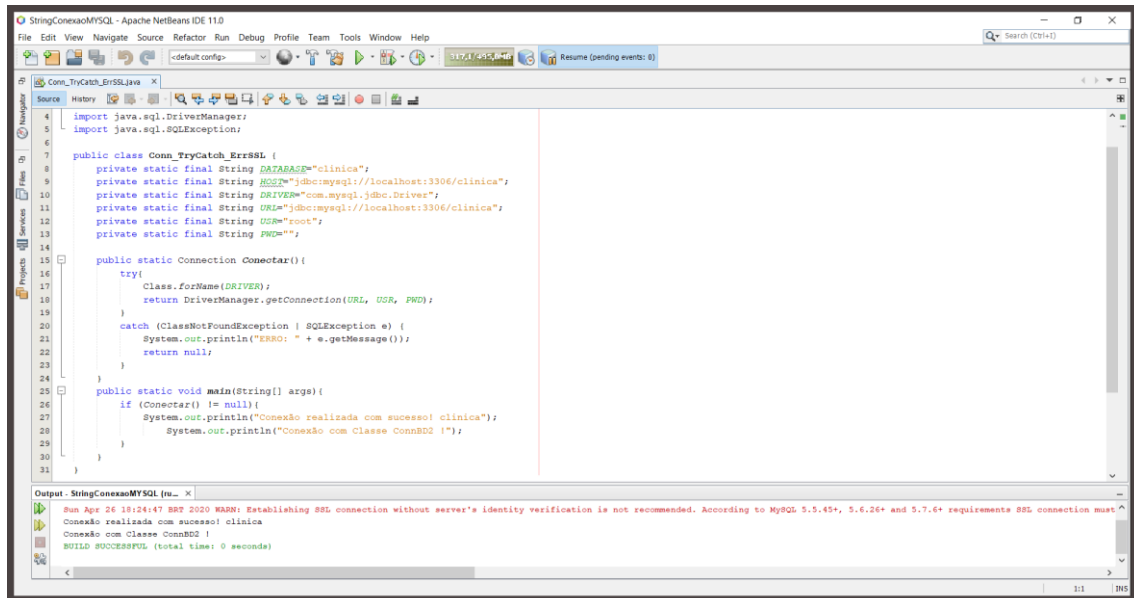
Agora, vamos ao que interessa, mais uma vez , ressalvo que em meu ponto de vista, esta forma de desenvolver Script de Conexão não é a usual, porém , esta foi uma maneira que escrevemos apenas para mostrarmos que também é possível.

Antes , vale lembrar que temos que adicionar nossa biblioteca “jar” de Conexão. Vamos lá.



Neste ambiente, já temos nosso projeto criado “StringConexaoMYSQL” no Netbeans, e neste momento, criamos diversas “classes” Classe Java para manipularmos nossa String de Conexão. Para maior clareza, vamos “comentar as linhas”.

Conn_TryCatch_ErrSSL.java



```

1 import java.sql.DriverManager;
2 import java.sql.SQLException;
3
4 public class Conn_TryCatch_ErrSSL {
5     private static final String DATABASE="clinica";
6     private static final String HOST="jdbc:mysql://localhost:3306/clinica";
7     private static final String DRIVER="com.mysql.jdbc.Driver";
8     private static final String URL="jdbc:mysql://localhost:3306/clinica";
9     private static final String USER="root";
10    private static final String PWD="";
11
12    public static Connection Conectar() {
13        try {
14            Class.forName(DRIVER);
15            return DriverManager.getConnection(URL, USER, PWD);
16        } catch (ClassNotFoundException | SQLException e) {
17            System.out.println("ERRO: " + e.getMessage());
18            return null;
19        }
20    }
21
22    public static void main(String[] args) {
23        if (Conectar() != null) {
24            System.out.println("Conexão realizada com sucesso! clinica");
25            System.out.println("Conexão com Classe ConnBD2 !");
26        }
27    }
28 }

```

Output - StringConexaoMYSQL [n... X]

```

Sun Apr 26 15:24:47 SBT 2020 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must
Conexão realizada com sucesso! clinica
Conexão com Classe ConnBD2 !
BUILD SUCCESSFUL (total time: 0 seconds)

```

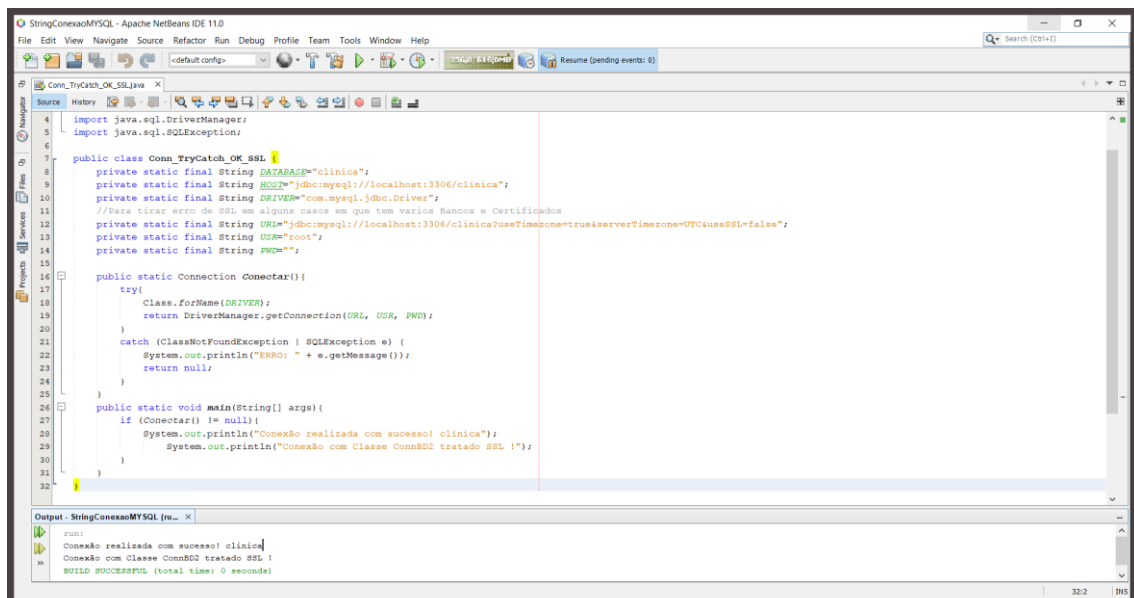
Notamos que apareceu um erro de certificado SSL, vamos manipular para que não ocorra

```

//Para tirar erro de SSL em alguns casos em que tem varios Bancos e Certificados
private static final String URL="jdbc:mysql://localhost:3306/clinica?useTimezone=true&serverTimezone=UTC&useSSL=false";

```

Conn_TryCatch_OK_SSL.java



```

1 import java.sql.DriverManager;
2 import java.sql.SQLException;
3
4 public class Conn_TryCatch_OK_SSL {
5     private static final String DATABASE="clinica";
6     private static final String HOST="jdbc:mysql://localhost:3306/clinica";
7     private static final String DRIVER="com.mysql.jdbc.Driver";
8     private static final String URL="jdbc:mysql://localhost:3306/clinica?useTimezone=true&serverTimezone=UTC&useSSL=false";
9     private static final String USER="root";
10    private static final String PWD="";
11
12    public static Connection Conectar() {
13        try {
14            Class.forName(DRIVER);
15            return DriverManager.getConnection(URL, USER, PWD);
16        } catch (ClassNotFoundException | SQLException e) {
17            System.out.println("ERRO: " + e.getMessage());
18            return null;
19        }
20    }
21
22    public static void main(String[] args) {
23        if (Conectar() != null) {
24            System.out.println("Conexão realizada com sucesso! clinica");
25            System.out.println("Conexão com Classe ConnBD2 tratado SSL !");
26        }
27    }
28 }

```

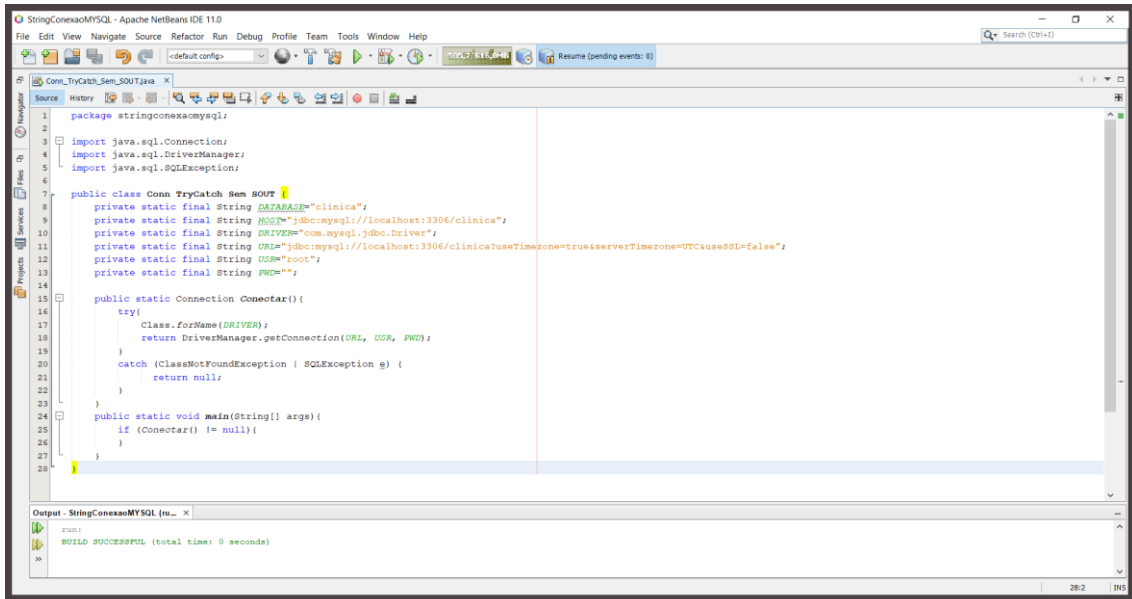
Output - StringConexaoMYSQL [n... X]

```

Sun Apr 26 15:24:47 SBT 2020 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must
Conexão realizada com sucesso! clinica
Conexão com Classe ConnBD2 tratado SSL !
BUILD SUCCESSFUL (total time: 0 seconds)

```

Conn_TryCatch_Sem_SOUT.java



```

1 package stringconexaomysql;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5 import java.sql.SQLException;
6
7 public class Conn_TryCatch_Sem_SOUT {
8     private static final String DATABASE="clinica";
9     private static final String HOST="jdbc:mysql://localhost:3306/clinica";
10    private static final String DRIVER="com.mysql.jdbc.Driver";
11    private static final String URL="jdbc:mysql://localhost:3306/clinica?useTimezone=true&serverTimezone=UTC&useSSL=false";
12    private static final String USER="root";
13    private static final String PWD="";
14
15    public static Connection Conectar() {
16        try {
17            Class.forName(DRIVER);
18            return DriverManager.getConnection(URL, USER, PWD);
19        } catch (ClassNotFoundException | SQLException e) {
20            return null;
21        }
22    }
23
24    public static void main(String[] args) {
25        if (Conectar() != null) {
26            //
27        }
28    }
29 }

```

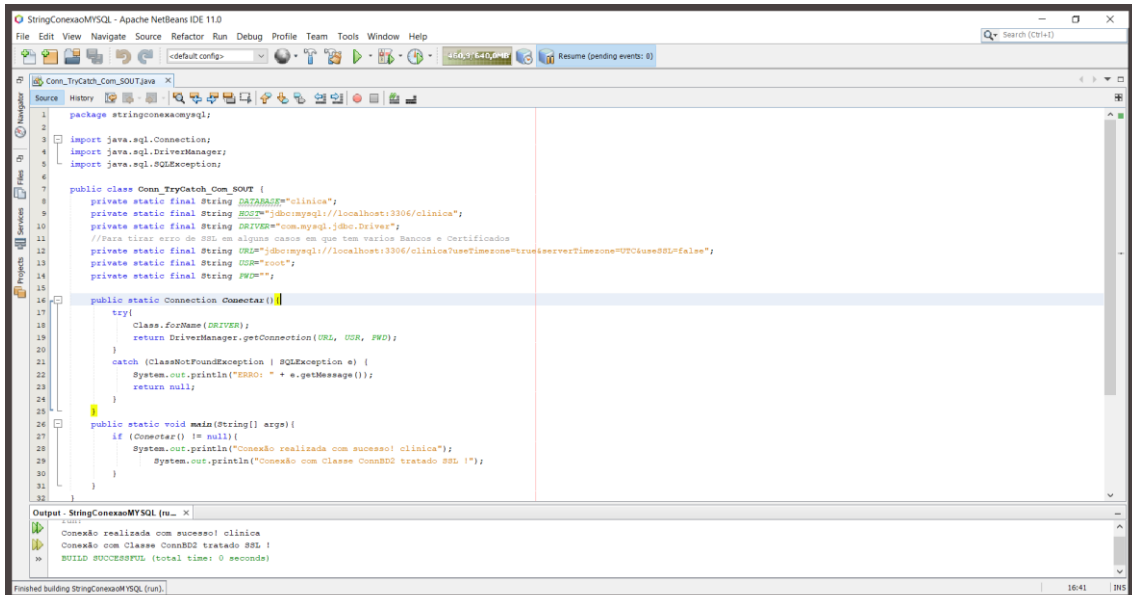
Output - StringConexaoMySQL (run) X

```

run
BUILD SUCCESSFUL (total time: 0 seconds)

```

Conn_TryCatch_Com_SOUT.java



```

1 package stringconexaomysql;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5 import java.sql.SQLException;
6
7 public class Conn_TryCatch_Com_SOUT {
8     private static final String DATABASE="clinica";
9     private static final String HOST="jdbc:mysql://localhost:3306/clinica";
10    private static final String DRIVER="com.mysql.jdbc.Driver";
11    //Para tirar erro de SSL em alguns casos em que tem varios Bancos e Certificados
12    private static final String URL="jdbc:mysql://localhost:3306/clinica?useTimezone=true&serverTimezone=UTC&useSSL=false";
13    private static final String USER="root";
14    private static final String PWD="";
15
16    public static Connection Conectar() {
17        try {
18            Class.forName(DRIVER);
19            return DriverManager.getConnection(URL, USER, PWD);
20        } catch (ClassNotFoundException | SQLException e) {
21            System.out.println("ERRO: " + e.getMessage());
22            return null;
23        }
24    }
25
26    public static void main(String[] args) {
27        if (Conectar() != null) {
28            System.out.println("Conexão realizada com sucesso! clinica");
29            System.out.println("Conexão com Classe ConnBD2 tratado SSL !");
30        }
31    }
32 }

```

Output - StringConexaoMySQL (run) X

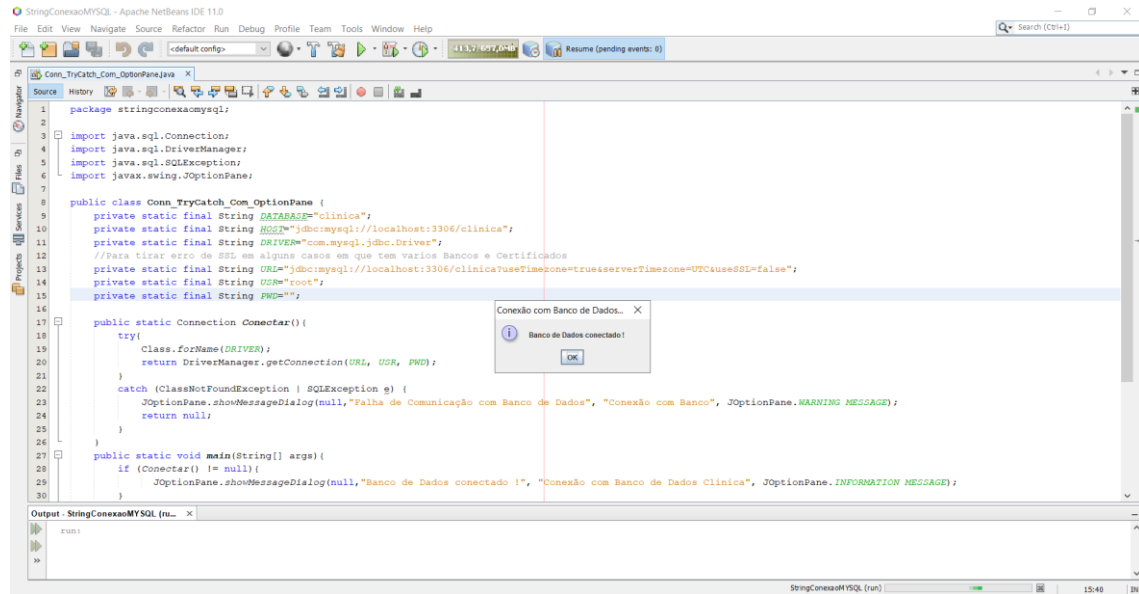
```

run
Conexão realizada com sucesso! clinica
Conexão com Classe ConnBD2 tratado SSL !
BUILD SUCCESSFUL (total time: 0 seconds)

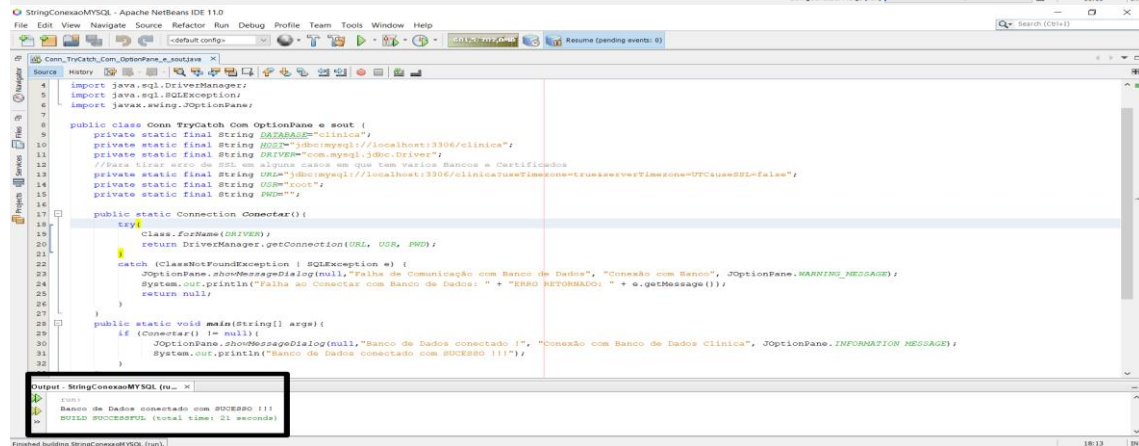
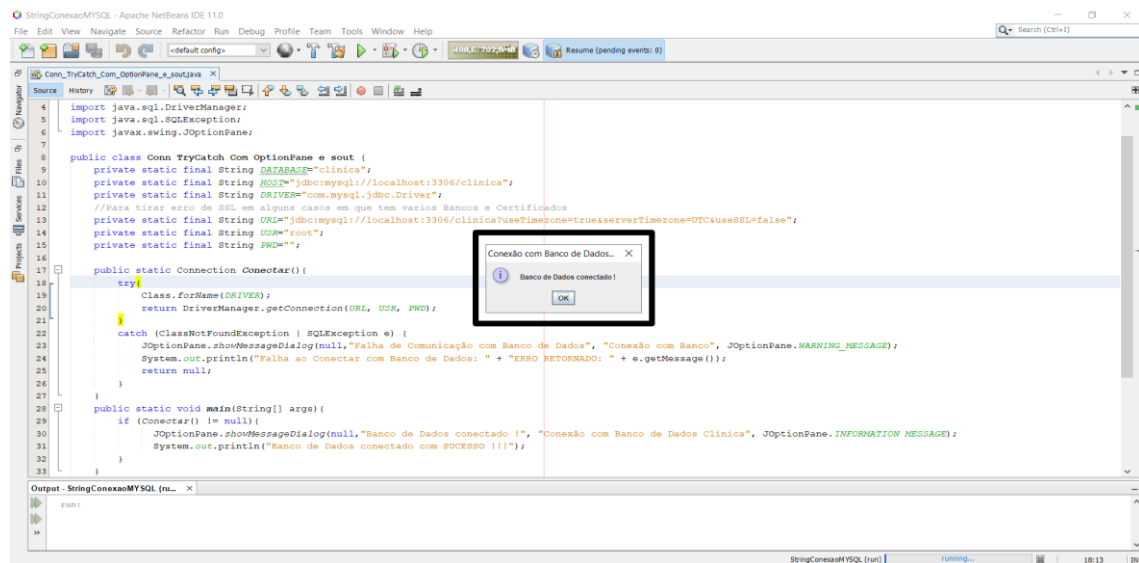
```

Finished building StringConexaoMySQL (run).

Conn_TryCatch_Com_OptionPane.java

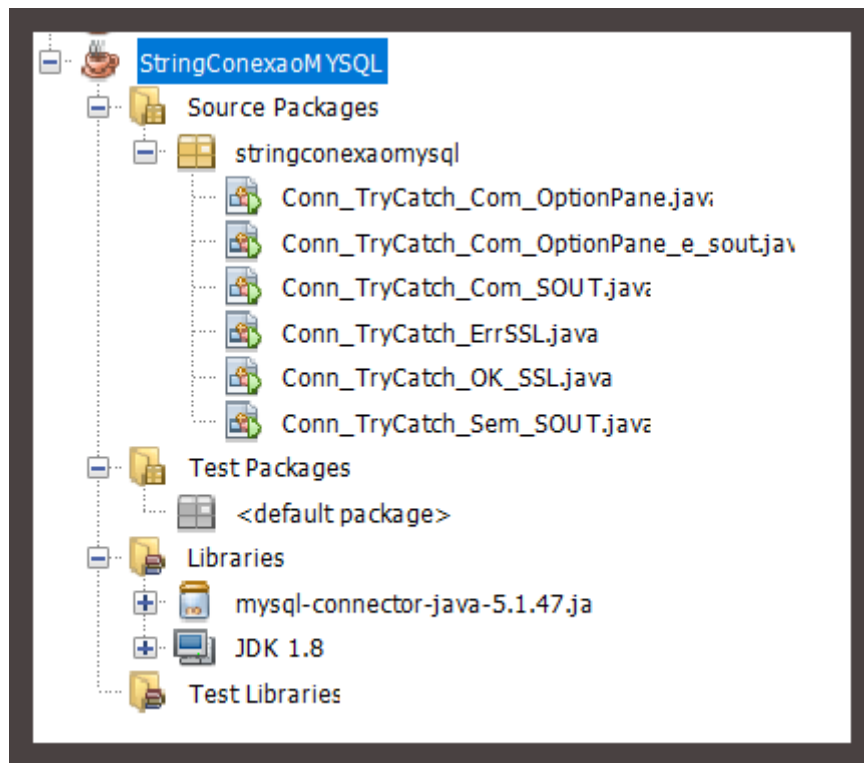


Conn_TryCatch_Com_OptionPane_e_sout.java



Para este tema, desenvolvemos algumas maneiras de realizar a conexão com o Banco de Dados, agora basta você escolher a que melhor que encaixa !

Criamos para este artigo, todas as classes Java e as disponibilizaremos no link do artigo.



Os códigos estarão disponíveis no Git.

Até mais !
Espero ter ajudado !