

u585 ● 第13章mrpデータベース構築

PostgreSQLのmrpデータベースにEMPLOYEESテーブルを作成します。

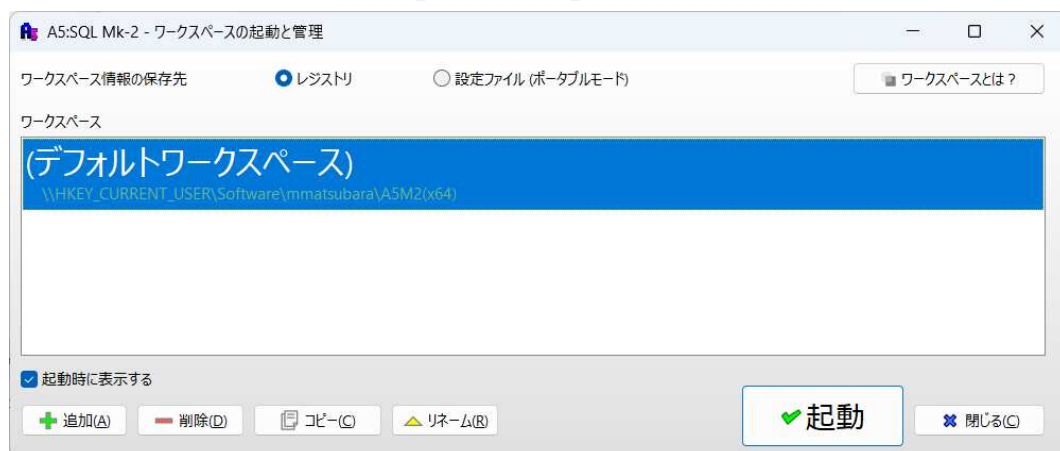
1. (準備)新しいドキュメントルートjdbcを作成

```
root@aso:webapps# cd /var/lib/tomcat9/webapps
root@aso:webapps# mkdir jdbc
root@aso:webapps# mkdir jdbc/WEB-INF
root@aso:webapps# mkdir jdbc/WEB-INF/classes
root@aso:webapps# mkdir jdbc/WEB-INF/jsp
root@aso:webapps# cd jdbc/WEB-INF/classes
root@aso:classes# cp /usr/share/tomcat9/lib/servlet-api.jar .
root@aso:classes# jar -xvf servlet-api.jar
```

2. データベースサーバ(PostgreSQL)に接続

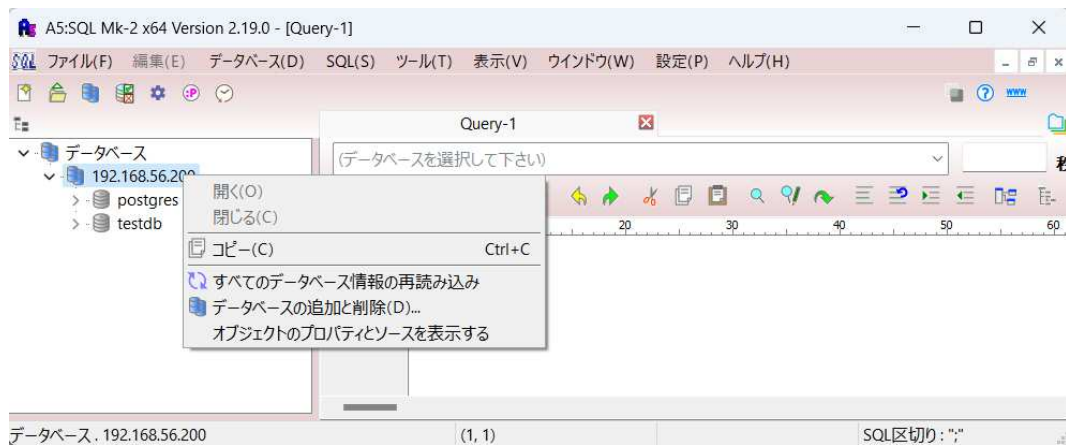
・ WindowsクライアントのA5M2で接続確認

WindowsクライアントのA5M2(**配布フォルダ内のa5m2_2.19.0_x64フォルダにあるA5M2.exeをダブルクリック**)を「デフォルトワークスペース」で「起動」します。

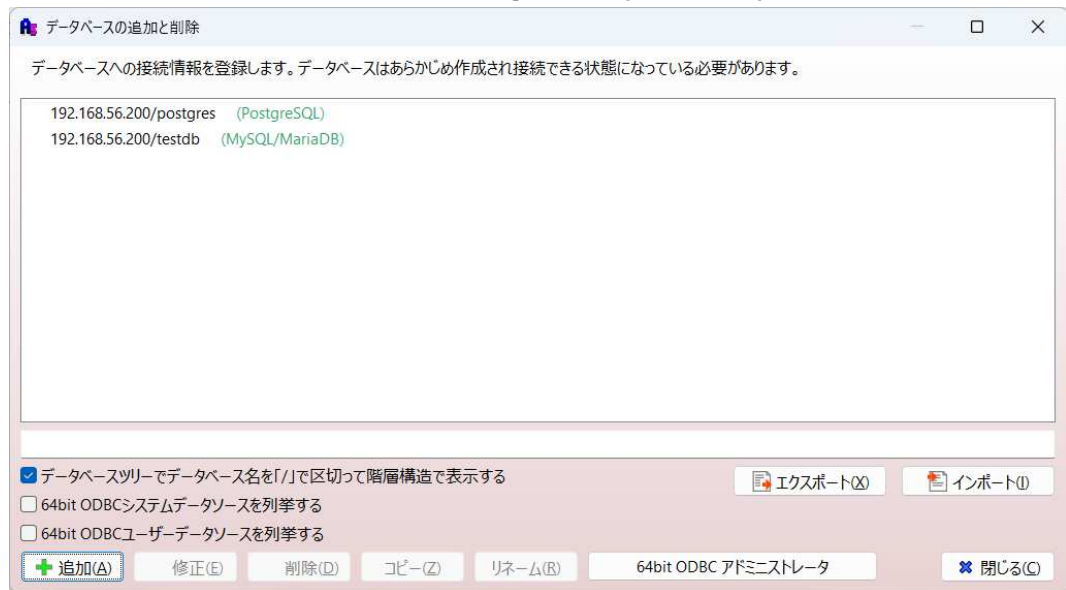


↓

「データベース」→「UbuntuのIPアドレスを右クリック」→「データベースの追加と削除」を行います。



「追加ボタンをクリック」→「PostgreSQL(直接接続)」でDBに接続します

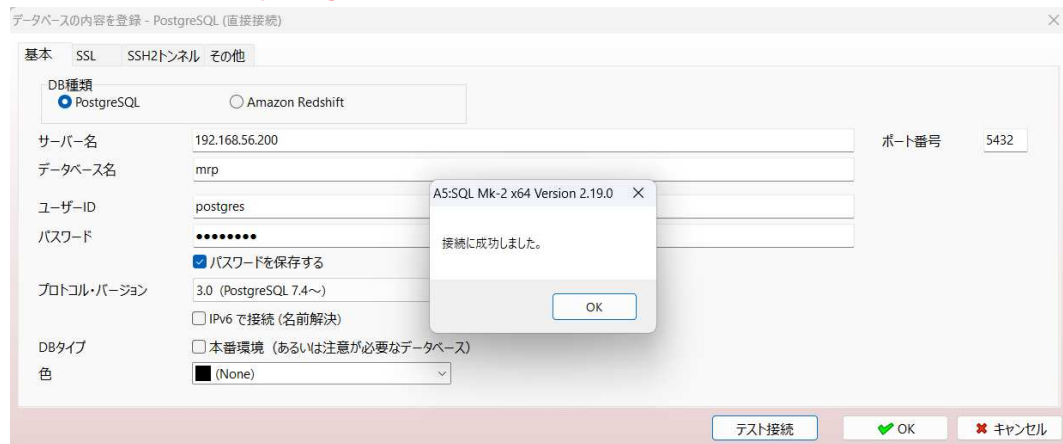


以下のように設定して「テスト接続」をクリックします→**成功することを確認します**

※「サーバ名」のIPアドレスは各自の値で設定します

※「データベース名」は「**mrp**」です

※パスワードは「**postgres**」です



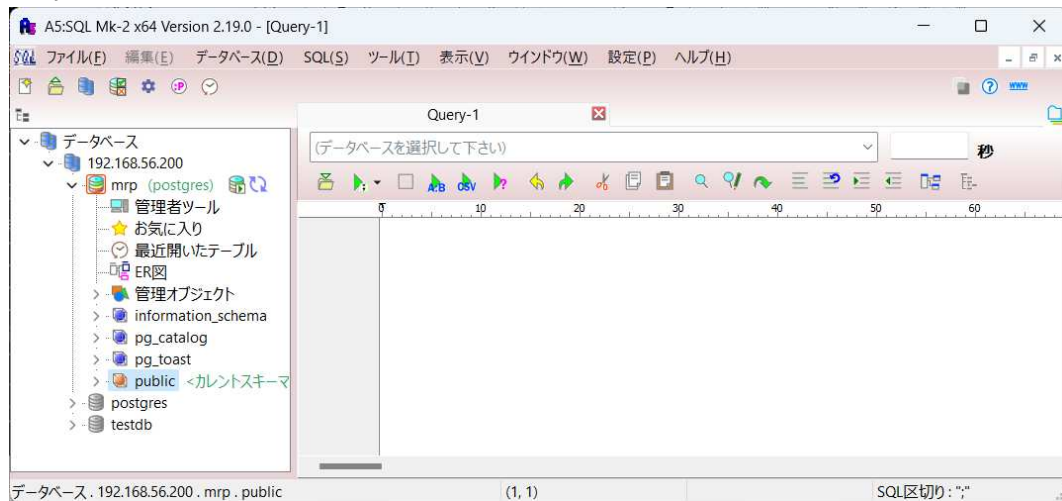
接続成功を確認後OKをクリックして「データベースの追加と削除」を閉じます



3. テーブルの作成とデータ入力

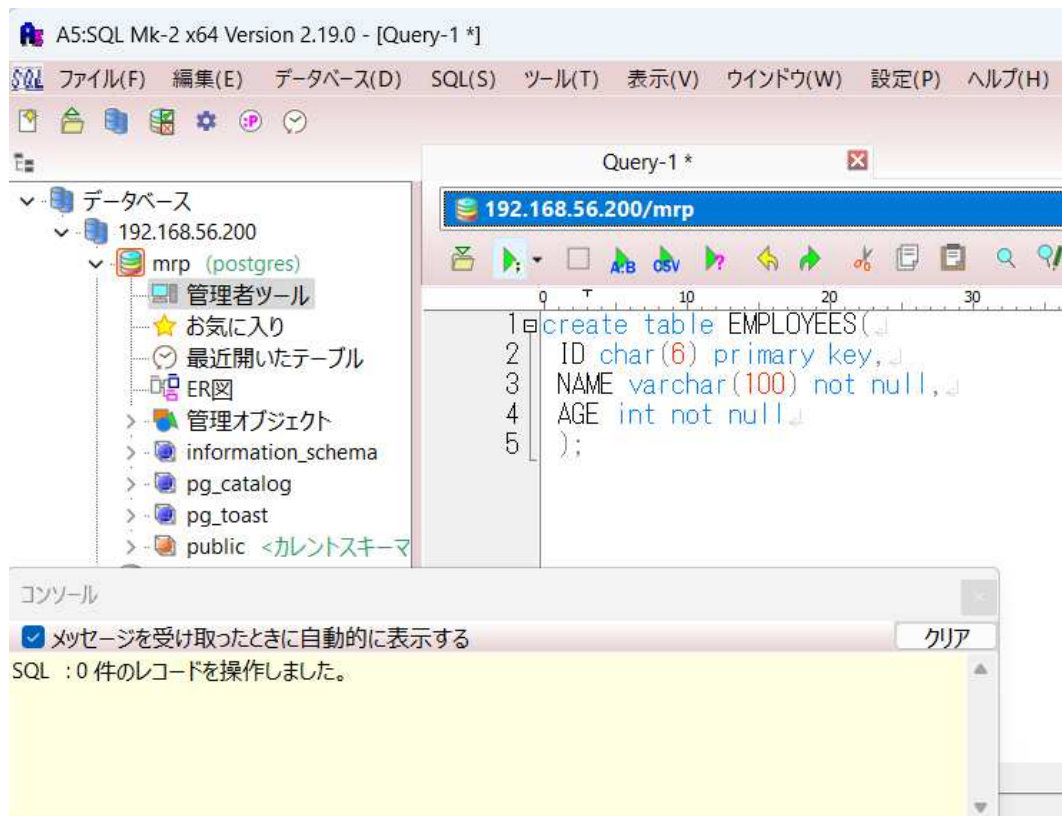
- WindowsクライアントのA5M2からEMPLOYEESテーブルを作成

mrpデータベースを展開（接続）します

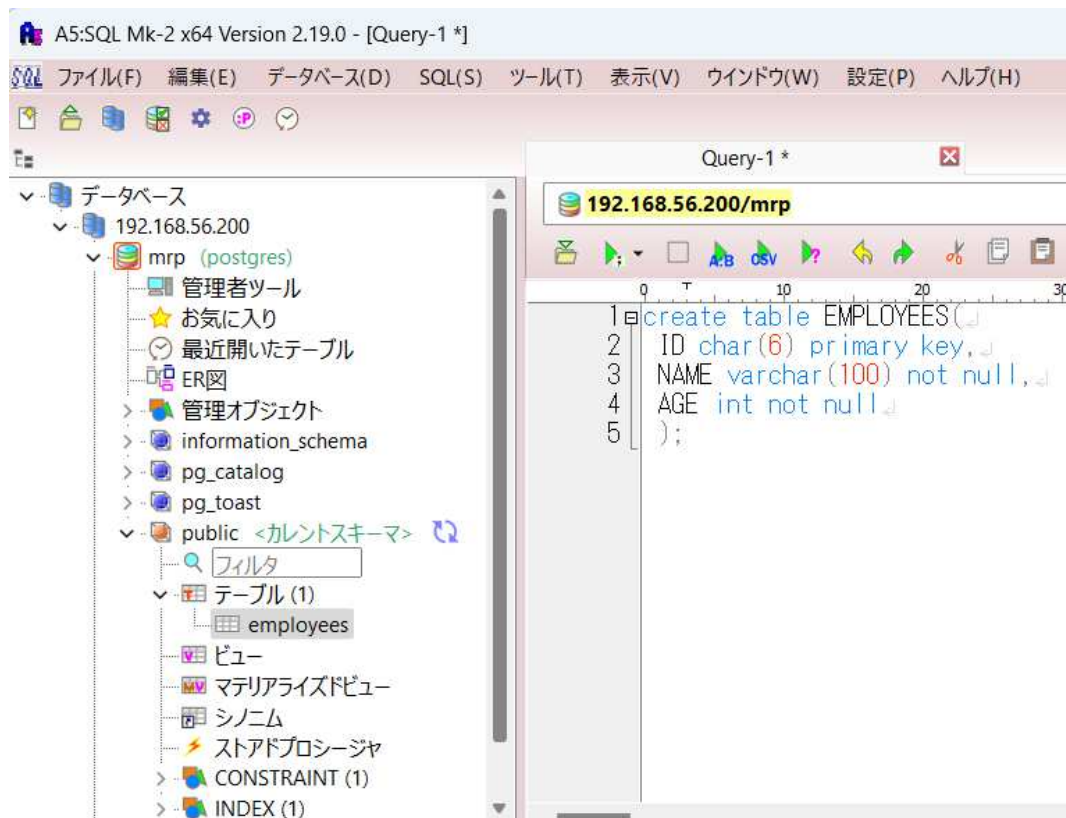


データベースを選択し以下のSQL-DDLを入力して ▶コマンドで実行します。

※キャレット位置からの実行ですのでご注意ください
(先頭から全ての実行に変えてもよいです)

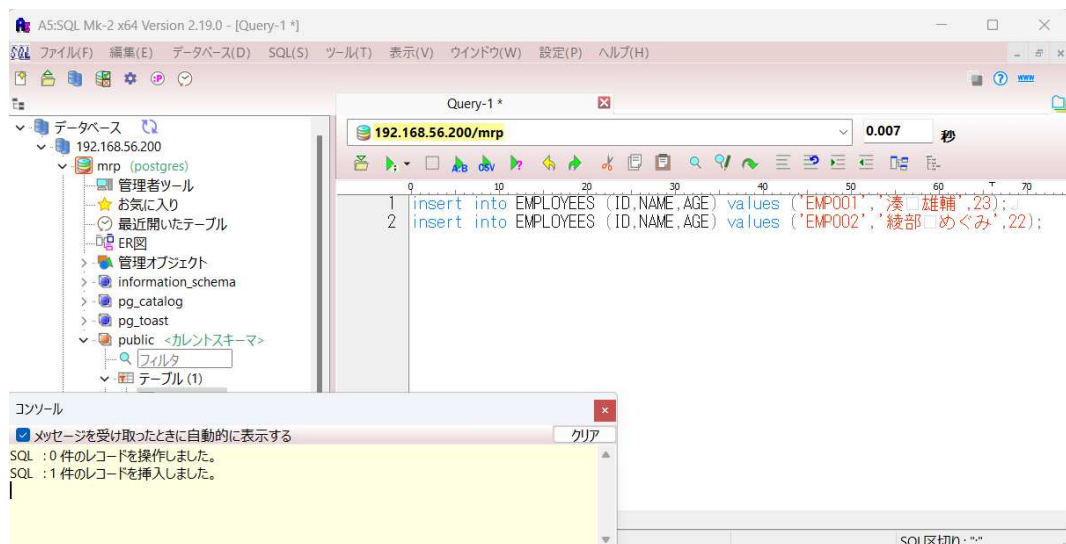


「public<カレントスキーマ>」→「テーブル」を順次展開して
「employees」テーブルの存在を確認します



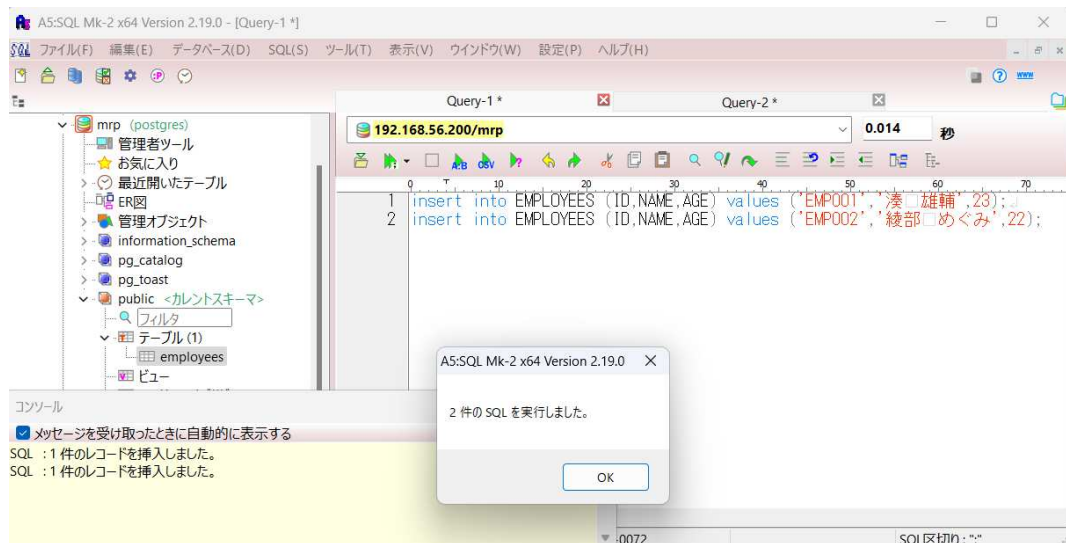
・ WindowsクライアントのA5M2からEMPLOYEESテーブルにデータを入力

以下のSQL-DMLを入力して ► コマンドで実行します。※実行位置を**キャレット位置以降**に変更してキャレットを実行したい行に置いて実行します



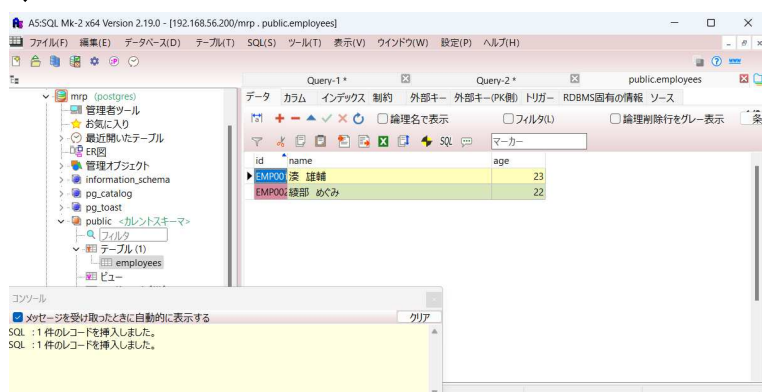
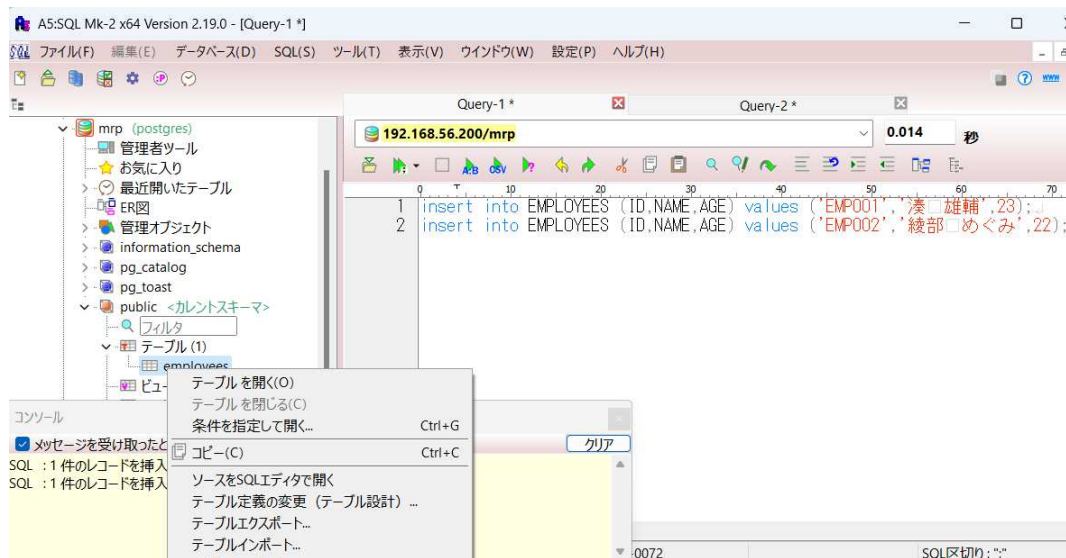
2 件のSQLが実行（2件のデータが入力） されないといけません
→失敗したら一旦データを削除してやり直した方が安全です

※データの削除の仕方「delete from employees」



・ Windowsクライアント のA5M2からEMPLOYEESテーブルのデータを 確認

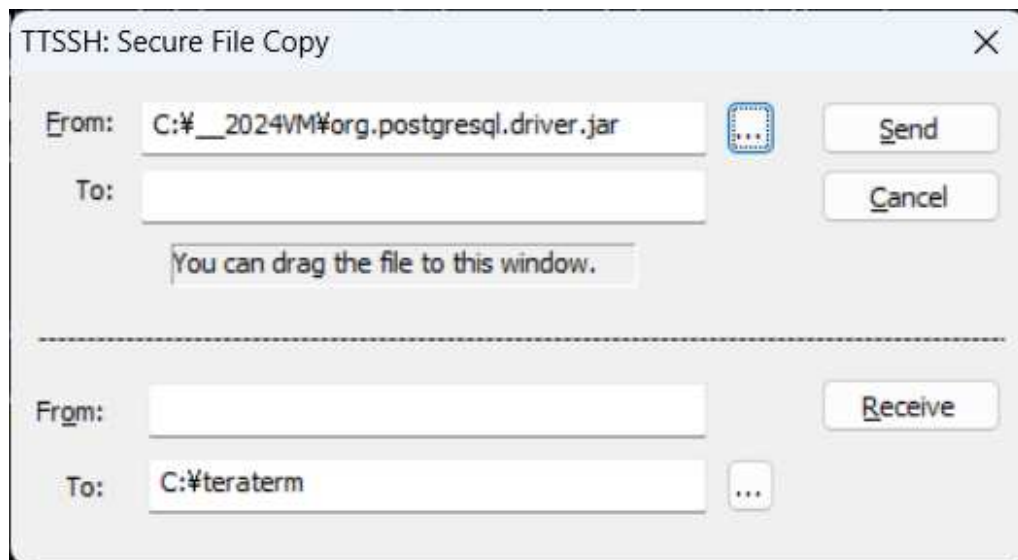
テーブルのEMPLOYEESを右クリックで選択して「テーブルを開く」でデータが入力されていることを確認します



4. PostgreSQL用Pure-Java-Driverを配置

・TeraTermの「ssh SCP」機能でドライバをUbuntuに送りコンパイル位置に展開する

Windowsの配布フォルダにある**org-postgresql-driver.jar**ファイルをUbuntu側へ転送します。



ssh SCP の転送先はホームディレクトリとなります。

つまり、TeraTermをguestでログインしているときは、転送先は**/home/guest**となります。

よって、Servletのコンパイル位置である**classes**に**org-postgresql-driver.jar**コピーして展開します。

最後にorgディレクトリが作成されていることを確認します。

```
[root@aso classes]# pwd
/var/lib/tomcat9/webapps/jdbc/WEB-INF/classes
[root@aso classes]# cp /home/guest/org.postgresql.driver.jar .
[root@aso classes]# jar xvf org.postgresql.driver.jar
[root@aso classes]# ls
HelloServlet.class  javax      module-
info.class          org.postgresql.driver.jar
HelloServlet.java   META-INF  org          servlet-api.jar
```

5. DBアクセスプログラム（JDBC）を作成（コード13-1）

```
[root@aso classes]# vi SelectEmployees.java
```

コード13-1を実装します。↓11行目と16行目の部分を変更してください。

```
192.168.56.200 - root@aso: /var/lib/tomcat9/webapps/jdbc/WEB-INF/c VT
ファイル(F) 編集(E) 設定(S) コントロール(C) ウィンドウ(W) 漢字コード(K) ヘルプ(H)
1 import java.sql.Connection;
2 import java.sql.DriverManager;
3 import java.sql.PreparedStatement;
4 import java.sql.ResultSet;
5 import java.sql.SQLException;
6
7 public class SelectEmployees {
8     public static void main(String[] args) {
9         // JDBCドライバを読み込む
10        try {
11            Class.forName("org.postgresql.Driver"); //koko
12        } catch (ClassNotFoundException e) {
13            throw new IllegalStateException("JDBCドライバを読み込めませんでした");
14        }
15        // データベースに接続
16        try (Connection conn = DriverManager.getConnection("jdbc:postgresql://192.168
.56.200:5432/mrp", "postgres", "postgres")) { //koko
17
18            // SELECT文を準備
19            String sql = "SELECT ID,NAME,AGE FROM EMPLOYEES";
20            PreparedStatement pstmt = conn.prepareStatement(sql);
21
22            // SELECTを実行し、結果表 (ResultSet) を取得
23            ResultSet rs = pstmt.executeQuery();
24
25            // 結果表に格納されたレコードの内容を表示
26            while (rs.next()) {
27                String id = rs.getString("ID");
28                String name = rs.getString("NAME");
29                int age = rs.getInt("AGE");
30
31                // 取得したデータを出力
32                System.out.println("ID:" + id);
33                System.out.println("名前:" + name);
34                System.out.println("年齢:" + age + "¥n");
35            }
36        } catch (SQLException e) {
37            e.printStackTrace();
38        }
39    }
40 }
```

【参考】

●PostgreSQLに接続するコード（重要なのはココ）

```
Class.forName("org.postgresql.Driver");
Connection conn =
DriverManager.getConnection("jdbc:postgresql://192.168.56.200:5432/mrp",
"postgres", "postgres")
```

6. DBアクセスプログラムを実行して確認

SelectEmployees.javaをコンパイルして実行します。コンパイルにはjava.sqlのモジュール追加オプションを指定しパッケージ対応で行います。もし、**module-info.class**があれば**rm**コマンドで削除してください（そうすれば**--add-modules java.sql**オプションはいりません）。


```
root@aso:classes# javac --add-modules java.sql -d .  
SelectEmployees.java  
root@aso:classes# java SelectEmployees  
ID:EMP001  
名前:湊 雄輔  
年齢:23  
  
ID:EMP002  
名前:綾部 めぐみ  
年齢:22  
root@aso:/var/lib/tomcat9/webapps/jdbc/WEB-INF/classes#
```

7. コード 1 3 – 2、3、4 を実装

コード 1 3 – 2 : Employee.java
そのまま実装後コンパイル

```
root@aso:classes# vi Employee.java  
root@aso:classes# javac -d . Employee.java
```

コード 1 3 – 3 : EmployeesDAO.java
※データベース接続部分のコード変更が必要です！！
コード変更

```
root@aso:classes# vi EmployeesDAO.java
```

※14～16行目、22行目を変更します

```
12 public class EmployeesDAO {  
13     // データベース接続に使用する情報  
14     private final String JDBC_URL = "jdbc:postgresql://192.168.56.10:5432/mrp";  
15     private final String DB_USER = "postgres";  
16     private final String DB_PASS = "postgres";  
17  
18     public List<Employee> findAll() {  
19         List<Employee> empList = new ArrayList<Employee>();  
20         // JDBCドライバを読み込む  
21         try {  
22             Class.forName("org.postgresql.Driver");  
23         } catch (ClassNotFoundException e) {  
24             throw new IllegalStateException("JDBCドライバを読み込めませんでした");  
25         }  
26     }  
27 }
```

コンパイル

```
root@aso:classes# javac --add-modules java.sql -d .  
EmployeesDAO.java
```

コード 13 - 4 : SelectEmployees.java → **SelectEmployeesDAO.java**に
名前を変更

コードの内容は13-4SelectEmployees.javaのクラス宣言だけ変更

コード変更

※ 5 行目のみ変更

```
5 public class SelectEmployeesDAO {  
6     public static void main(String[] args) {  
7         // EMPLOYEESテーブルの全レコードを取得  
8         EmployeesDAO empDAO = new EmployeesDAO();  
9         List<Employee> empList = empDAO.findAll();  
    }
```

コンパイル

```
root@aso:classes# javac -d . SelectEmployeesDAO.java
```

8. 実行確認後結果を報告

起動するクラスは？ mainメソッドを持つクラスですよ！

```
root@aso:classes# java xxxxxxxxxxxxxxxx
```

結果は?????

実行した結果のスクショを送ってください！！

【重要】

●PostgreSQLに接続するコード（重要なのはココ）

```
Class.forName("org.postgresql.Driver");
```

```
Connection conn =
```

```
DriverManager.getConnection("jdbc:postgresql://192.168.56.200:5432/mrp",  
"postgres", "postgres")
```