

# DB接続アプリ実習（在庫検索）

## DB接続アプリ実習（在庫検索）

以下の仕様を実現するJavaアプリケーションを作成します。

### 【仕様】

在庫データベースに接続しコマンドラインから入力した商品番号の在庫状況を取得するJavaアプリケーションを作成する。

※エラー処理は、とりあえず抜きにしましょう

• データベース(PostgreSQL)の制作

(1) まずは在庫テーブルを作成します

• WindowsクライアントからCSEで接続確認

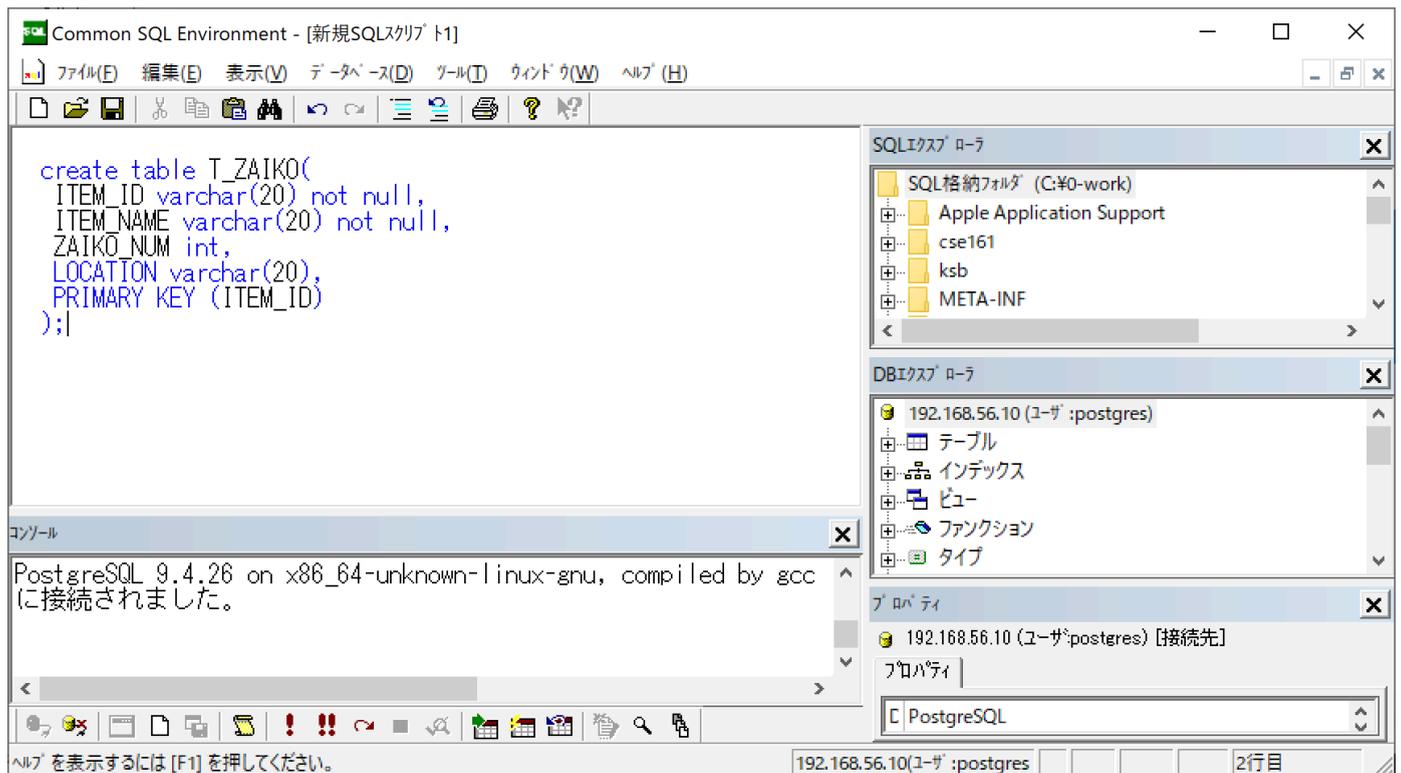
まずは、WindowsのCSEからWindowsのPostgreSQLのlibraryデータベースにアクセスして問題なく接続されることを確認する。

※libraryデータベースが存在しない場合は「**create database library**」コマンドでlibraryデータベースを作成してください。

その後、接続を切断してDB名：をlibraryに変更して新たに接続し直します。

• WindowsクライアントからCSEで在庫テーブルを作成

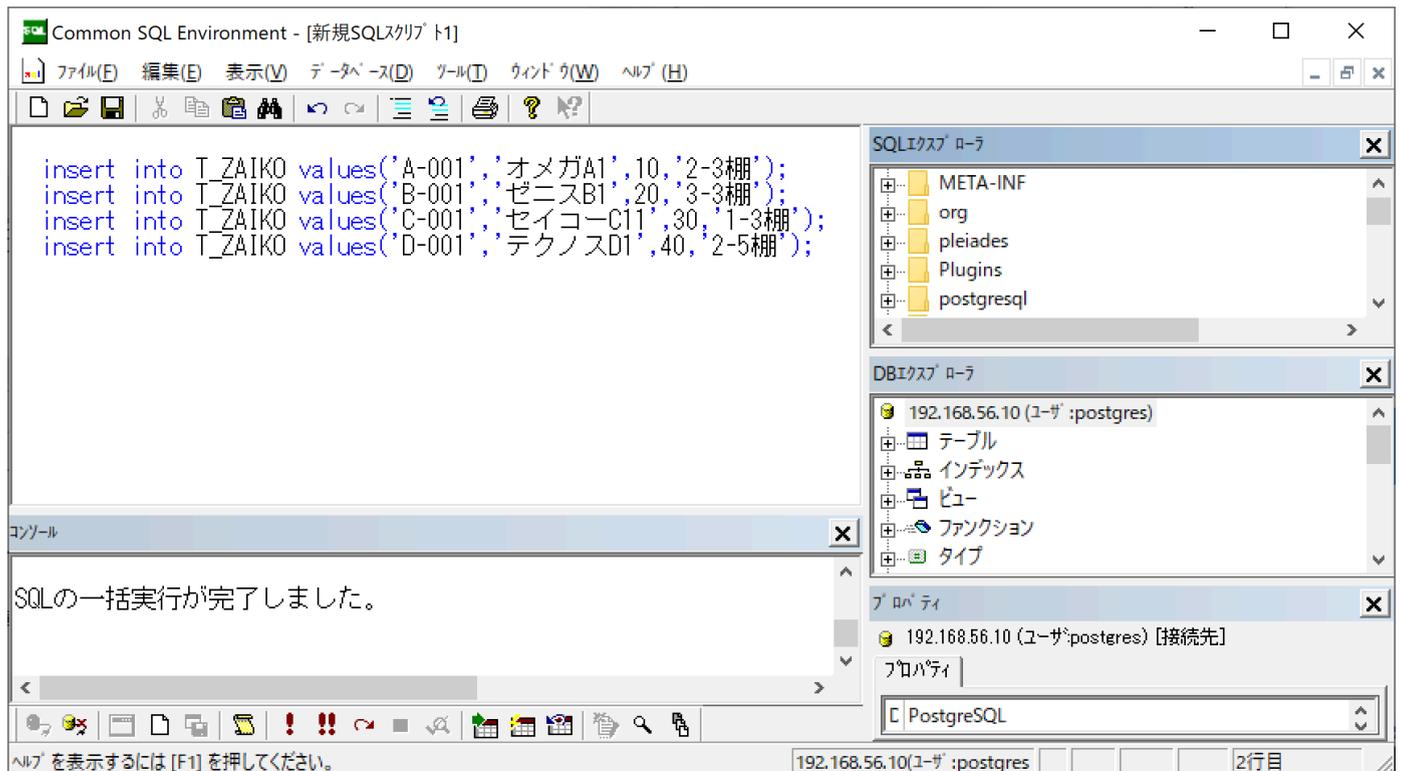
以下のSQL-DDLを入力して！コマンドで実行します。



(2)次に在庫テーブルにデータを入力します

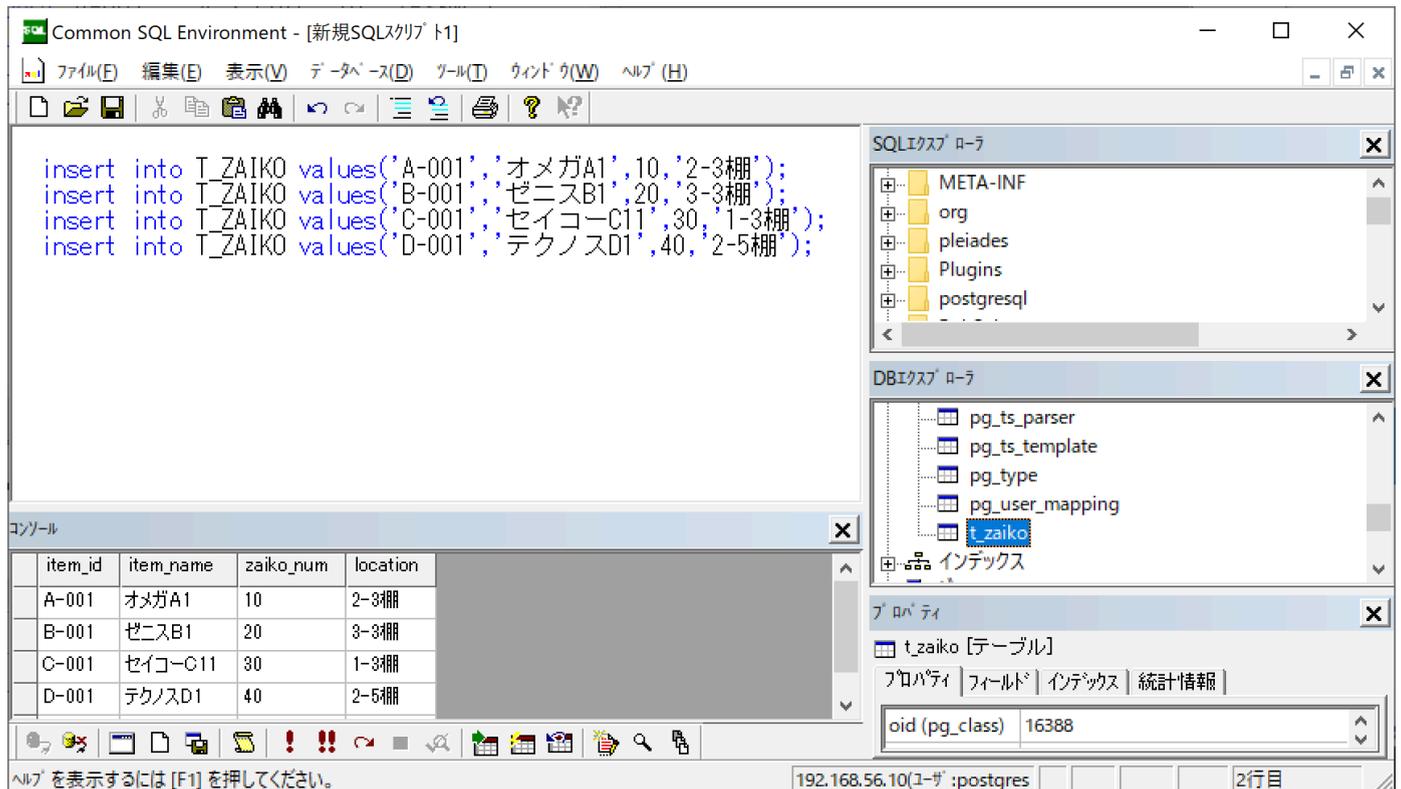
- ・ WindowsクライアントからCSEで在庫テーブルにinsert文を発行

以下のSQL-DMLを入力して！！コマンドで実行します。



- ・ WindowsクライアントからCSEで在庫テーブルのデータを確認

右側の「DBエクスプローラ」からt\_zaikoを右クリックで選択して「全データを開く」でテーブルにデータが入力されていることを確認する



### (3) PostgreSQL用Pure-Java-Driverを配置します

配布されたorg-postgresql-driver.jarファイルを適切に配置します。  
適切なコンパイルルートに org-postgresql-driver.jar コピーして展開してください。

```
[root@ksb classes]# jar xvf org.postgresql.driver.jar
```

```
META-INFが作成されました
META-INF/MANIFEST.MFが展開されました
orgが作成されました
org/postgresqlが作成されました
org/postgresql/coreが作成されました
org/postgresql/core/BaseConnection.classが展開されました
.....
.....
.....
```

```
[[root@ksb classes]#
```

#### ●モデル(Java)の制作

(1)Javaプログラムを作成します(最初からPackage化しておきます)

```
[root@ksb classes]# vi Zaiko.java
```

ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) 漢字コード(K) ヘルプ(H)

```
1 package jp.ict.ksb;
2 import java.sql.*;
3 import java.util.*;
4 public class Zaiko {
5     public static void main(String args[]){
6         String item_id="";
7         String item_name="";
8         int zaiko_num=0;
9         String location="";
10    try{
11        /**
12        * ●入力処理セクション
13        */
14        item_id = args[0];
15        /**
16        * ●業務処理セクション
17        */
18        /*
19        SQL-DDL
20        create table T_ZAIKO(
21            ITEM_ID varchar(20) not null,
22            ITEM_NAME varchar(20) not null,
23            ZAIKO_NUM int,
24            LOCATION varchar(20),
25            PRIMARY KEY (ITEM_ID)
26        );
27        SQL-DML
28        insert into T_ZAIKO values('A-001','オメガA1',10,'2-3棚');
29        insert into T_ZAIKO values('B-001','ゼニスB1',20,'3-3棚');
30        insert into T_ZAIKO values('C-001','セイコーC11',30,'1-3棚');
31        insert into T_ZAIKO values('D-001','テクノスD1',40,'2-5棚');
32        */
33        //[1]JDBCドライバのロード
34        Class.forName("org.postgresql.Driver");
35        //[2]JRDBへの接続
-- INSERT --
```

```
192.168.56.10 - root@aso:/usr/local/src/apache-tomcat-9.0.0.M17/w VT
ファイル(F) 編集(E) 設定(S) コントロール(O) ウィンドウ(W) 漢字コード(K) ヘルプ(H)
35 // [2] JDBCへの接続
36 Connection connection = DriverManager.getConnection
37     ("jdbc:postgresql://192.168.56.10:5432/library","postgres","
    postgres");
38 // [3] SQL文のコンテナ作成
39 Statement stmt = connection.createStatement();
40 // [4] SQL文を実行する
41 ResultSet rs = stmt.executeQuery
42     ("select * from T_ZAIKO where ITEM_ID = '" + item_id + "'");
43 // [5] JDBCの検索結果を取り出す
44 while( rs.next() ){
45     item_name = rs.getString("ITEM_NAME");
46     zaiko_num = rs.getInt( "ZAIKO_NUM" );
47     location = rs.getString( "LOCATION" );
48 }
49 // [6] データベースから切断する
50 stmt.close();
51 connection.close();
52
53 /**
54 * ● 出力処理セクション
55 System.out.println("在庫状況を確認します!");
56 System.out.println("");
57 System.out.println("商品名: " + item_name);
58 System.out.println("在庫数: " + zaiko_num);
59 System.out.println("保管棚: " + location);
60 System.out.println("");
61 System.out.println("処理が終了しました。");
62 */
63
64 } catch ( Exception e ) {
65     e.printStackTrace();
66 }
67 }
68 }
-- INSERT --
```

```
[root@ksb classes]# javac -d . Zaiko.java
```

```
[root@ksb classes]# java jp.ict.ksb.Zaiko ←←← 引数が必要ですよ！！
```

```
java.lang.ArrayIndexOutOfBoundsException: 0
```

```
    at jp.ict.ksb.Zaiko.main(Zaiko.java:14)
```

```
[root@ksb classes]# java jp.ict.ksb.Zaiko A-001
```

```
在庫状況を確認します!
```

```
商品名 : オメガA1
```

```
在庫数 : 10
```

```
保管棚 : 2-3棚
```

処理が終了しました。

[root@ksb classes]#

### 【演習 1】

以下のような形式で、すべてのアイテムを出力するようにプログラムを変更してください。

出力結果を提出すること！

#### (実行例)

在庫状況を確認します!

\* 商品ID : A-001

商品名 : オメガA1

在庫数 : 10

保管棚 : 2-3棚

\* 商品ID : B-001

商品名 : ゼニスB1

在庫数 : 20

保管棚 : 3-3棚

\* 商品ID : C-001

商品名 : セイコーC11

在庫数 : 30

保管棚 : 1-3棚

\* 商品ID : D-001

商品名 : テクノスD1

在庫数 : 40

保管棚 : 2-5棚

処理が終了しました。

### 【演習 2】

前回作成済の libraryデータベース に接続して v\_利用者貸出し履歴 から利用者電話番号 045-123-4567

のデータ（氏名、所属、書名、貸出日、返却日）を取得するJavaアプリケーションを作成してください。

プログラム名はZaikoLibrary.javaとします。

ソースコードを提出すること！

#### (実行例)

貸出し状況を確認します!

電 話 : 045-123-4567  
氏 名 : 鈴木紀子  
所 属 : 技術  
書 名 : WindowsNT4.0システム管理入門 (基礎編)  
貸出日 : 1999-12-20  
返却日 : null

電 話 : 045-123-4567  
氏 名 : 鈴木紀子  
所 属 : 技術  
書 名 : 総合化MRPシステム - 設計と導入 -  
貸出日 : 1998-09-07  
返却日 : 1998-10-07

処理が終了しました。