

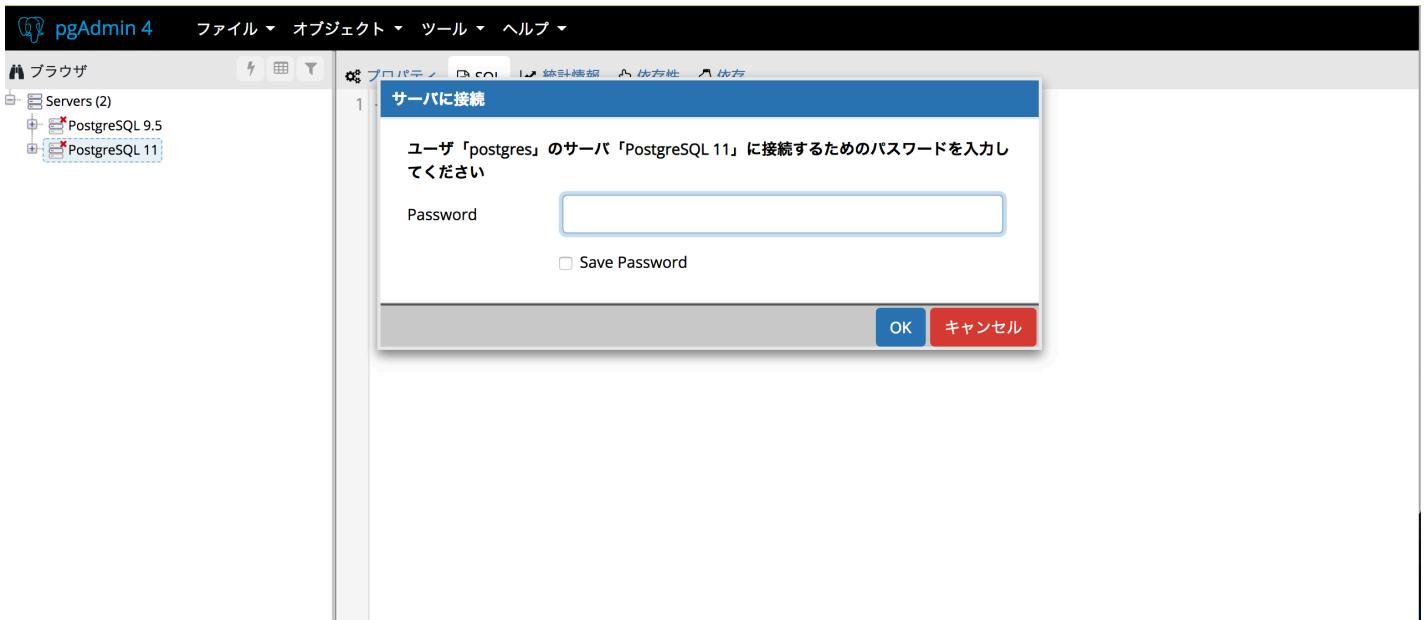
4-4-2.データベースを扱うための準備

Step1 : テーブルの作成

1. まずはpgAdmin4を起動させ、サーバに接続し、すでに作成してあるデータベース【lesson_db】を選択します。

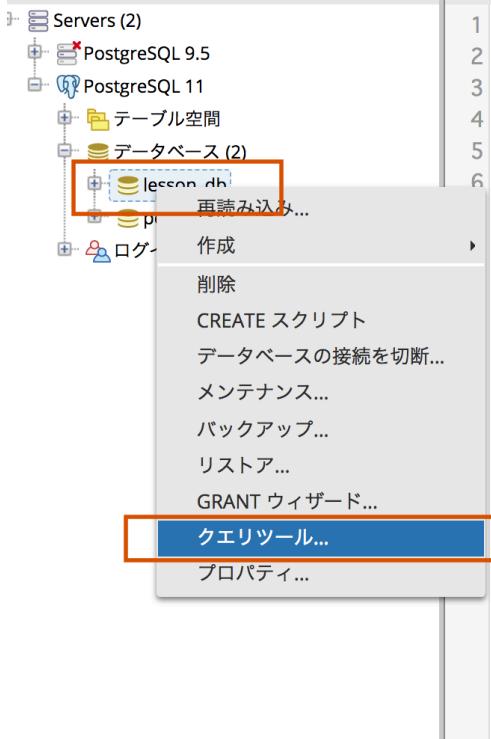
※まだデータベースを作成していない方や、サーバーの接続方法を忘れてしまった方は環境構築のページを確認して下さい。

※もしもパスワードを問われたら **postgres** と入力しましょう！



2. 「lesson_db」を選択して右クリック。

メニューバーにあるクエリツールをクリックして、SQL実行用のウィンドウを開きます。



```
-- Database: lesson_db
-- DROP DATABASE lesson_db;

CREATE DATABASE lesson_db
    WITH
    OWNER = postgres
    ENCODING = 'UTF8'
    LC_COLLATE = 'ja_JP.UTF-8'
    LC_CTYPE = 'ja_JP.UTF-8'
    TABLESPACE = pg_default
    CONNECTION LIMIT = -1;
```

3. クエリツールの画面に、下記コードを丸ごとコピーして貼り付けて下さい。

【雷マーク】をクリックして、クエリを実行します。

The screenshot shows the pgAdmin 4 interface with the SQL tab selected. The code for creating the 'lesson_db' database is pasted into the query editor. A red box highlights the execute button (a lightning bolt icon) in the toolbar above the editor.

```
CREATE TABLE Staff(Id CHAR(4) NOT NULL,Name  VARCHAR(50) NOT NULL,Entrance_year  INTEGER,Section  VARCHAR(30),CONSTRAINT pk_staff PRIMARY KEY(Id));
```

```
CREATE TABLE Area(AreaCode  CHAR(2) NOT NULL,AreaName  VARCHAR(20) NOT NULL,CONSTRAINT pk_area PRIMARY KEY(AreaCode));
```

```
CREATE TABLE Shop(ShopCode CHAR(3) NOT NULL,ShopName VARCHAR(50) NOT NULL,ShopNameAbc VARCHAR(50) NOT NULL,AreaCode CHAR(2) NOT NULL,UpdateDate DATE NOT NULL,CONSTRAINT pk_shop PRIMARY KEY(ShopCode),CONSTRAINT fk_areacode FOREIGN KEY(AreaCode) REFERENCES Area(AreaCode));
```

```
CREATE TABLE Goods(GoodsCode CHAR(4) NOT NULL,GoodsName VARCHAR(50) NOT NULL,UnitPrice INTEGER default 0,UpdateDate DATE NOT NULL,CONSTRAINT pk_goods PRIMARY KEY(GoodsCode));
```

```
CREATE TABLE Stocks(GoodsCode CHAR(4) NOT NULL,ShopCode CHAR(3) NOT NULL,Quantity INTEGER default 0,UpdateDate DATE NOT NULL,CONSTRAINT pk_stocks PRIMARY KEY(GoodsCode, ShopCode),CONSTRAINT fk_goodscode FOREIGN KEY(GoodsCode) REFERENCES Goods(GoodsCode),CONSTRAINT fk_shopcode FOREIGN KEY(ShopCode) REFERENCES Shop(ShopCode));
```

```
CREATE TABLE Purchaser(PurchaserCode CHAR(4) NOT NULL,PurchaserName VARCHAR(50) NOT NULL,Address VARCHAR(100) NOT NULL,Telephone VARCHAR(15),CONSTRAINT pk_purchaser PRIMARY KEY(PurchaserCode));
```

```

CREATE TABLE Order_Header(OrderNo INTEGER NOT NULL,PurchaserCode CHAR(4) NOT NULL,ShopCode CHAR(3) NOT NULL,OrderTimestamp Date NOT
NULL,Total INTEGER,CONSTRAINT pk_order_header PRIMARY KEY(OrderNo),CONSTRAINT fk_purchasercode FOREIGN KEY(PurchaserCode) REFERENCES
Purchaser(PurchaserCode));

CREATE TABLE Order_Detail(OrderNo INTEGER NOT NULL,DetailNo INTEGER NOT NULL,GoodsCode CHAR(4) NOT NULL,Quantity INTEGER NOT
NULL,SubTotal INTEGER,CONSTRAINT pk_order_detail PRIMARY KEY(OrderNo, DetailNo),CONSTRAINT fk_order_header FOREIGN KEY(OrderNo)
REFERENCES Order_header(OrderNo),CONSTRAINT fk_goodscodeorder FOREIGN KEY(GoodsCode) REFERENCES Goods(GoodsCode));

commit;

```

Step2 : テーブルの確認

【lesson_db】の階層を開き 【テーブル】という項目を開きます。

8項目のテーブルが作成されていることを確認して下さい。

The screenshot shows the PostgreSQL 11 interface in pgAdmin. On the left, a tree view displays the database structure. A red box highlights the 'Tables' section under the 'lesson_db' schema. Inside this box, the 'employee' table is underlined with a red line. The right side of the interface shows a query editor window titled 'lesson_db on postgres@PostgreSQL'. The window has tabs for 'EXPLAIN', 'メッセージ' (Messages), and '通' (Transactions). The main area of the window is currently empty.

Step3 : データの登録

1. テーブルの作成時と同様にデータベース【lesson_db】に対してクエリツールを起動します。
2. 下記のコードをコピーし、テーブル作成時と同様に上部のSQL実行用のウィンドウに貼り付けたら【雷マーク】をクリックしてクエリを実行します。

The screenshot shows the pgAdmin 4 interface. On the left is a tree view of servers and databases. The main window shows a query editor titled "lesson_db on postgres@PostgreSQL 11". The toolbar has several icons, with the last one (a magnifying glass) highlighted by a red box. The code area contains multiple INSERT statements for the Staff, Area, Shop, Goods, and Stocks tables. Below the code are tabs for "データ出力", "EXPLAIN", "メッセージ", "通知", and "クエリの履歴".

```

INSERT INTO Staff VALUES('0001','鈴木一郎',1995,'人事部');

INSERT INTO Staff VALUES('0002','鈴木次郎',1995,'経理部');

INSERT INTO Staff VALUES('0003','田中健一',1995,'営業部');

INSERT INTO Staff VALUES('0004','佐藤健二',1999,'総務部');

INSERT INTO Staff VALUES('0005','雪村花子',2000,'経理部');

INSERT INTO Staff VALUES('0006','草野由紀',2003,'営業部');

INSERT INTO Staff VALUES('9999','テスト',NULL,NULL);

INSERT INTO Area VALUES('01','北海道');

INSERT INTO Area VALUES('02','東北');

INSERT INTO Area VALUES('03','信越');

INSERT INTO Area VALUES('04','関東');

INSERT INTO Area VALUES('05','北陸');

INSERT INTO Area VALUES('06','東海');

INSERT INTO Area VALUES('07','近畿');

INSERT INTO Area VALUES('08','中国');

INSERT INTO Area VALUES('09','四国');

INSERT INTO Area VALUES('10','九州');

INSERT INTO Area VALUES('11','沖縄');

INSERT INTO Area VALUES('12','海外');

INSERT INTO Shop VALUES('001','新宿','SHINJUKU','04', '2000-01-01');

INSERT INTO Shop VALUES('002','横浜','YOKOHAMA','04', '2003-04-01');

INSERT INTO Shop VALUES('003','福岡','FUKUOKA','10', '2005-08-10');

INSERT INTO Shop VALUES('004','松山','MATSUYAMA','09', '2005-08-10');

INSERT INTO Shop VALUES('005','ロンドン','LONDON','12', '2005-12-31');

INSERT INTO Goods VALUES('1001','ベスト','7800', '2004-09-15');

INSERT INTO Goods VALUES('1002','セーター','12800', '2003-03-30');

INSERT INTO Goods VALUES('1003','ボレロ','5500', '2004-09-15');

INSERT INTO Goods VALUES('2001','パイロット帽','4000', '2000-01-01');

INSERT INTO Goods VALUES('2002','どんがり帽','3000', '2000-01-01');

INSERT INTO Goods VALUES('3001','手袋','2000', '2000-10-30');

INSERT INTO Goods VALUES('3002','手袋(指なし)','2000', '2000-10-30');

INSERT INTO Stocks VALUES('1001','001',20, current_date);

INSERT INTO Stocks VALUES('1002','001',25, current_date);

```

```
INSERT INTO Stocks VALUES('1003','001',10, current_date);
INSERT INTO Stocks VALUES('2001','002',5, current_date);
INSERT INTO Stocks VALUES('2002','002',12, current_date);
INSERT INTO Stocks VALUES('1002','002',14, current_date);
INSERT INTO Stocks VALUES('2001','003',2, current_date);
INSERT INTO Stocks VALUES('2002','003',1, current_date);
INSERT INTO Stocks VALUES('3001','003',18, current_date);
INSERT INTO Stocks VALUES('3002','003',9, current_date);
INSERT INTO Stocks VALUES('1001','005',5, current_date);
INSERT INTO Stocks VALUES('1002','005',0, current_date);
INSERT INTO Stocks VALUES('1003','005',30, current_date);
INSERT INTO Stocks VALUES('3001','005',22, current_date);
INSERT INTO Stocks VALUES('3002','005',10, current_date);
INSERT INTO Purchaser VALUES('0501','辛順勲','東京都新宿区','0300000000');
INSERT INTO Purchaser VALUES('0502','太宰満天','大分県別府市','0890000000');
INSERT INTO Purchaser VALUES('0503','豊臣秀子','神奈川県小田原市','0480000000');
INSERT INTO Purchaser VALUES('0504','John Devon','Buckingham Palace Road London','02073212233');
INSERT INTO Purchaser VALUES('0505','瀬田茅','東京都世田谷区','0330000000');
INSERT INTO Order_Header VALUES(420,'0504','005',TO_DATE('2000-05-23','YYYY-MM-DD'),4000);
INSERT INTO Order_Header VALUES(421,'0505','001',TO_DATE('2005-01-31','YYYY-MM-DD'),12800);
INSERT INTO Order_Header VALUES(422,'0501','002',TO_DATE('2005-02-06','YYYY-MM-DD'),8000);
INSERT INTO Order_Header VALUES(423,'0505','002',TO_DATE('2005-04-01','YYYY-MM-DD'),9500);
INSERT INTO Order_detail VALUES(420,1,'3001',1,2000);
INSERT INTO Order_detail VALUES(420,2,'3002',1,2000);
INSERT INTO Order_detail VALUES(421,1,'1002',1,12800);
INSERT INTO Order_detail VALUES(422,1,'2001',2,8000);
INSERT INTO Order_detail VALUES(423,1,'1003',2,5500);
INSERT INTO Order_detail VALUES(423,2,'2001',2,4000);
commit;
```

Step4 : データの確認

1.Step3で実行した文章をすべて削除して `select * from 【テーブル名】` で全てのテーブルを確認して行きます。

試しに、`select * from staff` と入力して【雷マーク】をクリックしてクエリを実行してみましょう。
上記の通りにデータが表示されればしっかりとデータが登録できています。

The screenshot shows a PostgreSQL client interface. In the top bar, there are tabs for 'プロパティ' (Properties), 'SQL', '統計情報' (Statistics), '依存性' (Dependencies), '依存' (Dependency), and 'クエリ - lesson_db on postgres@PostgreSQL 11 *'. Below the tabs, a toolbar has icons for file operations like open, save, search, and refresh. A dropdown menu shows '無制限' (Unlimited). The main area displays a query window with the command 'select * from staff'. The results are shown in a table titled 'データ出力' (Data Output) with columns: id, name, entrance_year, and section. The table contains 7 rows of data. The entire results table is highlighted with a red box.

	id character (4)	name character varying (50)	entrance_year integer	section character varying (30)
1	0001	鈴木一郎	1995	人事部
2	0002	鈴木次郎	1995	経理部
3	0003	田中健一	1995	営業部
4	0004	佐藤健二	1999	総務部
5	0005	雪村花子	2000	経理部
6	0006	草野由紀	2003	営業部
7	9999	テスト	[null]	[null]

同じ要領で、合計 8個 のテーブルを確認して行きましょう。

area , goods , order_detail , order_header , purchaser , shop , staff , stocks

全てのテーブルで、データが表示されれば、データの確認は以上となります。

これでデータベースの準備はOKです。課題を進めて行きましょう！

最終更新日時: 2022年 09月 10日(土曜日) 07:14