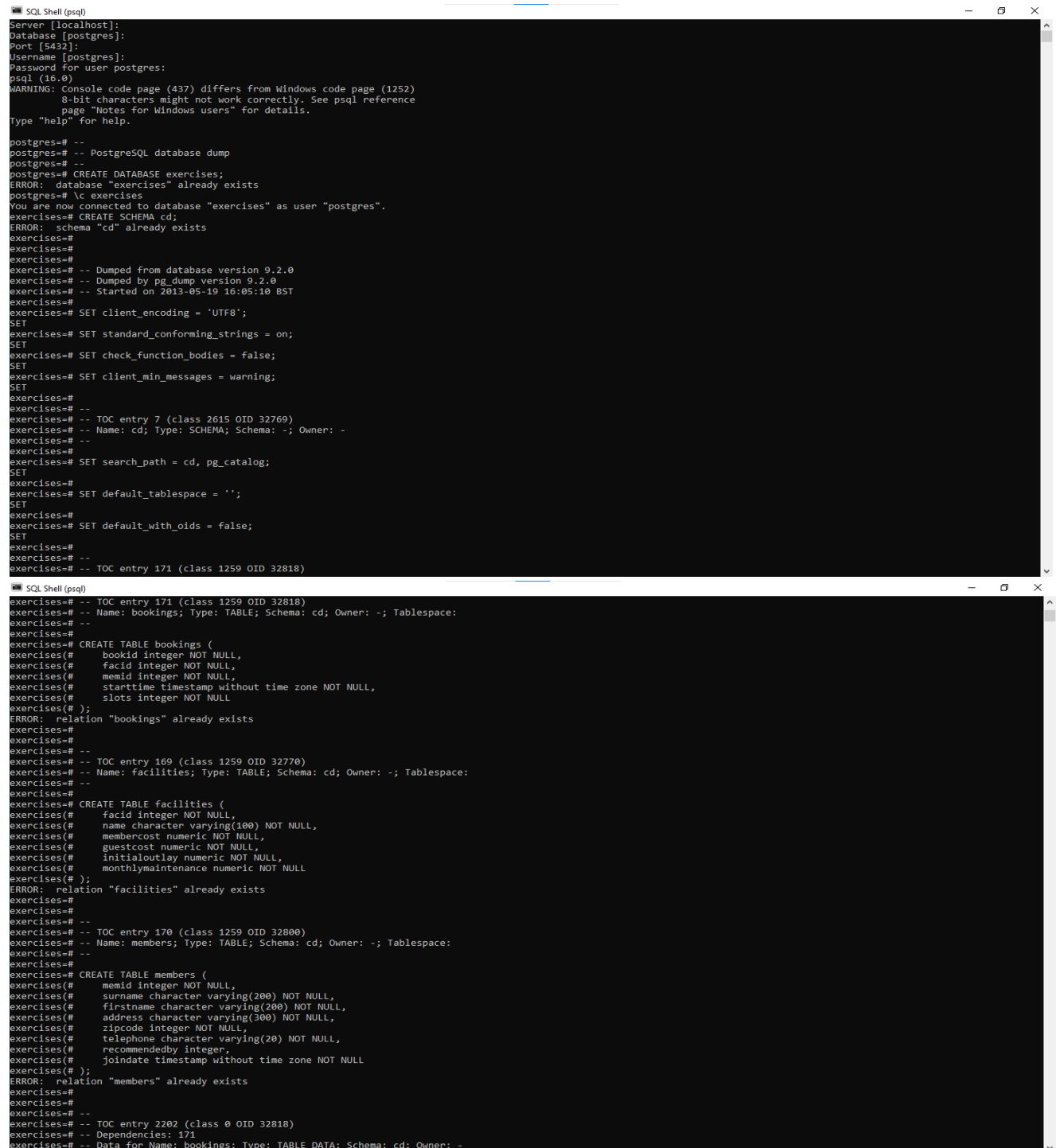


# PostgreExercisesAssignment

First I insert clubdata file:

ScreenShots:



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (16.0)
WARNING: Console code page (437) differs from Windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

postgres=# --
postgres=# -- PostgreSQL database dump
postgres=# --
postgres=# CREATE DATABASE exercises;
ERROR:  database "exercises" already exists
postgres=# \c exercises
You are now connected to database "exercises" as user "postgres".
exercises=# CREATE SCHEMA cd;
ERROR:  schema "cd" already exists
exercises=#
exercises=#
exercises=# -- Dumped from database version 9.2.0
exercises=# -- Dumped by pg_dump version 9.2.0
exercises=# -- Started on 2013-05-19 16:05:10 BST
exercises=#
exercises=# SET client_encoding = 'UTF8';
SET
exercises=# SET standard_conforming_strings = on;
SET
exercises=# SET check_function_bodies = false;
SET
exercises=# SET client_min_messages = warning;
SET
exercises=#
exercises=# --
exercises=# -- TOC entry 7 (class 2615 OID 32769)
exercises=# -- Name: cd; Type: SCHEMA; Schema: -; Owner: -
exercises=# --
exercises=#
exercises=# SET search_path = cd, pg_catalog;
SET
exercises=#
exercises=# SET default_tablespace = '';
SET
exercises=#
exercises=# SET default_with_oids = false;
SET
exercises=#
exercises=# --
exercises=# -- TOC entry 171 (class 1259 OID 32818)
exercises=# -- Name: bookings; Type: TABLE; Schema: cd; Owner: -; Tablespace:
exercises=# --
exercises=#
exercises=# CREATE TABLE bookings (
exercises(#   bookid integer NOT NULL,
exercises(#   facid integer NOT NULL,
exercises(#   memid integer NOT NULL,
exercises(#   starttime timestamp without time zone NOT NULL,
exercises(#   slots integer NOT NULL
exercises(# );
ERROR:  relation "bookings" already exists
exercises=#
exercises=# --
exercises=# -- TOC entry 169 (class 1259 OID 32770)
exercises=# -- Name: facilities; Type: TABLE; Schema: cd; Owner: -; Tablespace:
exercises=# --
exercises=#
exercises=# CREATE TABLE facilities (
exercises(#   facid integer NOT NULL,
exercises(#   name character varying(100) NOT NULL,
exercises(#   membercost numeric NOT NULL,
exercises(#   guestcost numeric NOT NULL,
exercises(#   initialoutlay numeric NOT NULL,
exercises(#   monthlymaintenance numeric NOT NULL
exercises(# );
ERROR:  relation "facilities" already exists
exercises=#
exercises=# --
exercises=# -- TOC entry 170 (class 1259 OID 32800)
exercises=# -- Name: members; Type: TABLE; Schema: cd; Owner: -; Tablespace:
exercises=# --
exercises=#
exercises=# CREATE TABLE members (
exercises(#   memid integer NOT NULL,
exercises(#   surname character varying(200) NOT NULL,
exercises(#   firstname character varying(200) NOT NULL,
exercises(#   address character varying(300) NOT NULL,
exercises(#   zipcode integer NOT NULL,
exercises(#   telephone character varying(20) NOT NULL,
exercises(#   recommendedby integer,
exercises(#   joindate timestamp without time zone NOT NULL
exercises(# );
ERROR:  relation "members" already exists
exercises=#
exercises=# --
exercises=# -- TOC entry 2202 (class 0 OID 32818)
exercises=# -- Dependencies: 171
exercises=# -- Data for Name: bookings; Type: TABLE DATA; Schema: cd; Owner: -
```

```
SQL Shell (psql)
exercices=# -- Data for Name: bookings; Type: TABLE DATA; Schema: cd; Owner: -
exercices=# --
exercices=#
exercices=# INSERT INTO bookings (bookid, facid, memid, starttime, slots) VALUES
exercices=# (0, 3, 1, '2012-07-03 11:00:00', 2),
exercices=# (1, 4, 1, '2012-07-03 08:00:00', 2),
exercices=# (2, 6, 0, '2012-07-03 18:00:00', 2),
exercices=# (3, 7, 1, '2012-07-03 19:00:00', 2),
exercices=# (4, 8, 1, '2012-07-03 10:00:00', 1),
exercices=# (5, 8, 1, '2012-07-03 15:00:00', 1),
exercices=# (6, 0, 2, '2012-07-04 09:00:00', 3),
exercices=# (7, 0, 2, '2012-07-04 15:00:00', 3),
exercices=# (8, 4, 3, '2012-07-04 13:00:00', 2),
exercices=# (9, 4, 0, '2012-07-04 15:00:00', 2),
exercices=# (10, 4, 0, '2012-07-04 17:30:00', 2),
exercices=# (11, 6, 0, '2012-07-04 12:30:00', 2),
exercices=# (12, 6, 0, '2012-07-04 14:00:00', 2),
exercices=# (13, 6, 1, '2012-07-04 15:30:00', 2),
exercices=# (14, 7, 2, '2012-07-04 14:00:00', 2),
exercices=# (15, 8, 2, '2012-07-04 12:00:00', 1),
exercices=# (16, 8, 3, '2012-07-04 18:00:00', 1),
exercices=# (17, 1, 0, '2012-07-05 17:30:00', 3),
exercices=# (18, 2, 1, '2012-07-05 09:30:00', 3),
exercices=# (19, 3, 3, '2012-07-05 09:00:00', 2),
exercices=# (20, 3, 1, '2012-07-05 19:00:00', 2),
exercices=# (21, 4, 3, '2012-07-05 18:30:00', 2),
exercices=# (22, 6, 0, '2012-07-05 13:00:00', 2),
exercices=# (23, 6, 1, '2012-07-05 14:30:00', 2),
exercices=# (24, 7, 2, '2012-07-05 18:30:00', 2),
exercices=# (25, 8, 3, '2012-07-05 12:30:00', 1),
exercices=# (26, 0, 0, '2012-07-06 08:00:00', 3),
exercices=# (27, 0, 0, '2012-07-06 14:00:00', 3),
exercices=# (28, 0, 2, '2012-07-06 15:30:00', 3),
exercices=# (29, 2, 1, '2012-07-06 17:00:00', 3),
exercices=# (30, 3, 1, '2012-07-06 11:00:00', 2),
exercices=# (31, 4, 3, '2012-07-06 12:00:00', 2),
exercices=# (32, 6, 1, '2012-07-06 14:00:00', 2),
exercices=# (33, 7, 2, '2012-07-06 08:30:00', 2),
exercices=# (34, 7, 2, '2012-07-06 13:30:00', 2),
exercices=# (35, 8, 3, '2012-07-06 15:30:00', 1),
exercices=# (36, 0, 2, '2012-07-07 08:30:00', 3),
exercices=# (37, 0, 0, '2012-07-07 12:30:00', 3),
exercices=# (38, 0, 2, '2012-07-07 14:30:00', 3),
exercices=# (39, 1, 3, '2012-07-07 08:30:00', 3),
exercices=# (40, 2, 1, '2012-07-07 09:00:00', 3),
exercices=# (41, 2, 1, '2012-07-07 11:30:00', 3),
exercices=# (42, 2, 1, '2012-07-07 16:00:00', 3),
exercices=# (43, 3, 2, '2012-07-07 12:30:00', 2),
exercices=# (44, 4, 3, '2012-07-07 11:30:00', 2),
exercices=# (45, 4, 3, '2012-07-07 14:00:00', 2),
exercices=# (46, 4, 0, '2012-07-07 17:30:00', 2),
exercices=# (47, 6, 0, '2012-07-07 08:30:00', 2),
```

```
SQL Shell (psql)
exercices=# (47, 6, 0, '2012-07-07 08:30:00', 2),
exercices=# (48, 6, 1, '2012-07-07 10:30:00', 2),
exercices=# (49, 6, 1, '2012-07-07 14:30:00', 2),
exercices=# (50, 6, 0, '2012-07-07 16:00:00', 2),
exercices=# (51, 7, 2, '2012-07-07 11:30:00', 2),
exercices=# (52, 8, 3, '2012-07-07 16:00:00', 1),
exercices=# (53, 8, 3, '2012-07-07 17:30:00', 2),
exercices=# (54, 0, 3, '2012-07-08 13:00:00', 3),
exercices=# (55, 0, 2, '2012-07-08 17:30:00', 3),
exercices=# (56, 1, 1, '2012-07-08 15:00:00', 3),
exercices=# (57, 1, 1, '2012-07-08 17:30:00', 3),
exercices=# (58, 3, 1, '2012-07-08 11:30:00', 2),
exercices=# (59, 3, 3, '2012-07-08 18:30:00', 2),
exercices=# (60, 3, 1, '2012-07-08 19:30:00', 2),
exercices=# (61, 4, 0, '2012-07-08 11:00:00', 2),
exercices=# (62, 4, 2, '2012-07-08 16:30:00', 2),
exercices=# (63, 4, 0, '2012-07-08 18:00:00', 2),
exercices=# (64, 4, 0, '2012-07-08 19:30:00', 2),
exercices=# (65, 6, 0, '2012-07-08 14:00:00', 2),
exercices=# (66, 6, 0, '2012-07-08 18:30:00', 2),
exercices=# (67, 7, 2, '2012-07-08 11:00:00', 2),
exercices=# (68, 7, 1, '2012-07-08 16:30:00', 2),
exercices=# (69, 8, 3, '2012-07-08 10:00:00', 1),
exercices=# (70, 8, 3, '2012-07-08 16:30:00', 1),
exercices=# (71, 0, 2, '2012-07-09 12:30:00', 3),
exercices=# (72, 0, 2, '2012-07-09 15:30:00', 3),
exercices=# (73, 0, 2, '2012-07-09 19:00:00', 3),
exercices=# (74, 1, 0, '2012-07-09 13:00:00', 3),
exercices=# (75, 1, 1, '2012-07-09 19:00:00', 3),
exercices=# (76, 2, 1, '2012-07-09 09:00:00', 6),
exercices=# (77, 2, 0, '2012-07-09 19:00:00', 3),
exercices=# (78, 3, 3, '2012-07-09 17:00:00', 2),
exercices=# (79, 3, 3, '2012-07-09 18:30:00', 2),
exercices=# (80, 4, 2, '2012-07-09 11:00:00', 2),
exercices=# (81, 4, 3, '2012-07-09 14:30:00', 2),
exercices=# (82, 6, 0, '2012-07-09 14:30:00', 2),
exercices=# (83, 7, 1, '2012-07-09 15:30:00', 2),
exercices=# (84, 7, 0, '2012-07-09 18:30:00', 4),
exercices=# (85, 8, 3, '2012-07-09 09:30:00', 1),
exercices=# (86, 8, 3, '2012-07-09 16:30:00', 1),
exercices=# (87, 8, 3, '2012-07-09 20:00:00', 1),
exercices=# (88, 0, 0, '2012-07-10 11:30:00', 3),
exercices=# (89, 0, 0, '2012-07-10 16:00:00', 3),
exercices=# (90, 3, 2, '2012-07-10 08:00:00', 2),
exercices=# (91, 3, 1, '2012-07-10 11:00:00', 2),
exercices=# (92, 3, 3, '2012-07-10 15:30:00', 2),
exercices=# (93, 3, 2, '2012-07-10 16:30:00', 2),
exercices=# (94, 3, 1, '2012-07-10 18:00:00', 2),
exercices=# (95, 4, 0, '2012-07-10 10:00:00', 2),
exercices=# (96, 4, 4, '2012-07-10 11:30:00', 2),
exercices=# (97, 4, 0, '2012-07-10 15:00:00', 2),
exercices=# (98, 4, 3, '2012-07-10 17:00:00', 4),
```

```
SQL Shell (psql)
exercises=# (98, 4, 3, '2012-07-10 17:00:00', 4),
exercises=# (99, 5, 0, '2012-07-10 08:30:00', 2),
exercises=# (100, 6, 0, '2012-07-10 14:30:00', 2),
exercises=# (101, 6, 0, '2012-07-10 19:00:00', 2),
exercises=# (102, 7, 4, '2012-07-10 08:30:00', 2),
exercises=# (103, 7, 2, '2012-07-10 17:30:00', 2),
exercises=# (104, 8, 0, '2012-07-10 11:30:00', 1),
exercises=# (105, 8, 3, '2012-07-10 12:00:00', 1),
exercises=# (106, 8, 3, '2012-07-10 19:30:00', 1),
exercises=# (107, 0, 4, '2012-07-11 08:00:00', 3),
exercises=# (108, 0, 2, '2012-07-11 18:00:00', 3),
exercises=# (109, 0, 0, '2012-07-11 12:00:00', 3),
exercises=# (110, 0, 0, '2012-07-11 14:00:00', 3),
exercises=# (111, 0, 2, '2012-07-11 15:30:00', 3),
exercises=# (112, 0, 2, '2012-07-11 18:30:00', 3),
exercises=# (113, 1, 0, '2012-07-11 12:30:00', 3),
exercises=# (114, 1, 0, '2012-07-11 16:00:00', 2),
exercises=# (115, 4, 1, '2012-07-11 08:00:00', 2),
exercises=# (116, 4, 0, '2012-07-11 09:00:00', 2),
exercises=# (117, 4, 3, '2012-07-11 11:00:00', 2),
exercises=# (118, 4, 0, '2012-07-11 15:00:00', 2),
exercises=# (119, 5, 4, '2012-07-11 17:00:00', 2),
exercises=# (120, 6, 0, '2012-07-11 14:00:00', 2),
exercises=# (121, 6, 0, '2012-07-11 19:30:00', 2),
exercises=# (122, 7, 0, '2012-07-11 08:00:00', 2),
exercises=# (123, 7, 0, '2012-07-11 14:00:00', 2),
exercises=# (124, 7, 0, '2012-07-11 16:30:00', 2),
exercises=# (125, 8, 4, '2012-07-11 11:00:00', 1),
exercises=# (126, 8, 3, '2012-07-11 13:00:00', 1),
exercises=# (127, 0, 0, '2012-07-12 13:30:00', 3),
exercises=# (128, 0, 2, '2012-07-12 16:30:00', 3),
exercises=# (129, 1, 1, '2012-07-12 11:30:00', 3),
exercises=# (130, 2, 1, '2012-07-12 09:00:00', 3),
exercises=# (131, 2, 1, '2012-07-12 18:30:00', 3),
exercises=# (132, 3, 3, '2012-07-12 18:00:00', 2),
exercises=# (133, 4, 1, '2012-07-12 16:00:00', 2),
exercises=# (134, 6, 0, '2012-07-12 12:00:00', 4),
exercises=# (135, 7, 2, '2012-07-12 08:00:00', 2),
exercises=# (136, 7, 4, '2012-07-12 13:30:00', 2),
exercises=# (137, 7, 4, '2012-07-12 16:00:00', 2),
exercises=# (138, 8, 3, '2012-07-12 16:30:00', 1),
exercises=# (139, 0, 2, '2012-07-13 18:30:00', 3),
exercises=# (140, 0, 4, '2012-07-13 14:00:00', 3),
exercises=# (141, 0, 3, '2012-07-13 17:00:00', 3),
exercises=# (142, 1, 1, '2012-07-13 15:00:00', 3),
exercises=# (143, 2, 1, '2012-07-13 09:00:00', 3),
exercises=# (144, 2, 0, '2012-07-13 15:00:00', 3),
exercises=# (145, 2, 1, '2012-07-13 16:30:00', 3),
exercises=# (146, 4, 0, '2012-07-13 11:00:00', 2),
exercises=# (147, 4, 0, '2012-07-13 13:30:00', 2),
exercises=# (148, 4, 0, '2012-07-13 15:00:00', 2),
exercises=# (149, 4, 3, '2012-07-13 16:00:00', 2),

SQL Shell (psql)
exercises=# (152, 7, 0, '2012-07-13 08:00:00', 2),
exercises=# (153, 7, 1, '2012-07-13 11:00:00', 2),
exercises=# (154, 7, 4, '2012-07-13 12:30:00', 2),
exercises=# (155, 8, 0, '2012-07-13 15:30:00', 1),
exercises=# (156, 8, 2, '2012-07-13 18:30:00', 1),
exercises=# (157, 0, 2, '2012-07-14 08:30:00', 3),
exercises=# (158, 0, 4, '2012-07-14 11:30:00', 3),
exercises=# (159, 0, 3, '2012-07-14 15:00:00', 3),
exercises=# (160, 1, 3, '2012-07-14 18:30:00', 3),
exercises=# (161, 1, 3, '2012-07-14 12:30:00', 3),
exercises=# (162, 1, 0, '2012-07-14 14:30:00', 3),
exercises=# (163, 2, 1, '2012-07-14 08:30:00', 3),
exercises=# (164, 3, 2, '2012-07-14 16:00:00', 2),
exercises=# (165, 4, 3, '2012-07-14 08:00:00', 2),
exercises=# (166, 4, 1, '2012-07-14 14:30:00', 2),
exercises=# (167, 6, 0, '2012-07-14 09:30:00', 2),
exercises=# (168, 6, 1, '2012-07-14 12:30:00', 2),
exercises=# (169, 6, 0, '2012-07-14 15:00:00', 2),
exercises=# (170, 7, 2, '2012-07-14 12:30:00', 2),
exercises=# (171, 7, 2, '2012-07-14 15:00:00', 2),
exercises=# (172, 7, 4, '2012-07-14 16:30:00', 2),
exercises=# (173, 7, 1, '2012-07-14 19:00:00', 2),
exercises=# (174, 8, 3, '2012-07-14 09:00:00', 1),
exercises=# (175, 8, 1, '2012-07-14 17:00:00', 1),
exercises=# (176, 0, 2, '2012-07-15 08:00:00', 3),
exercises=# (177, 0, 0, '2012-07-15 16:00:00', 3),
exercises=# (178, 0, 2, '2012-07-15 19:00:00', 3),
exercises=# (179, 1, 0, '2012-07-15 10:00:00', 3),
exercises=# (180, 1, 0, '2012-07-15 12:00:00', 3),
exercises=# (181, 1, 3, '2012-07-15 15:30:00', 3),
exercises=# (182, 2, 1, '2012-07-15 13:00:00', 3),
exercises=# (183, 3, 1, '2012-07-15 17:30:00', 2),
exercises=# (184, 4, 3, '2012-07-15 11:30:00', 2),
exercises=# (185, 4, 0, '2012-07-15 15:00:00', 2),
exercises=# (186, 4, 3, '2012-07-15 17:30:00', 2),
exercises=# (187, 7, 4, '2012-07-15 14:30:00', 2),
exercises=# (188, 7, 4, '2012-07-15 17:00:00', 2),
exercises=# (189, 8, 4, '2012-07-15 10:00:00', 1),
exercises=# (190, 8, 2, '2012-07-15 12:00:00', 1),
exercises=# (191, 8, 3, '2012-07-15 12:30:00', 1),
exercises=# (192, 8, 3, '2012-07-15 13:30:00', 1),
exercises=# (193, 8, 5, '2012-07-16 11:00:00', 3),
exercises=# (194, 0, 5, '2012-07-16 19:00:00', 3),
exercises=# (195, 1, 1, '2012-07-16 08:00:00', 3),
exercises=# (196, 1, 0, '2012-07-16 12:30:00', 3),
exercises=# (197, 2, 1, '2012-07-16 16:30:00', 3),
exercises=# (198, 4, 3, '2012-07-16 09:00:00', 2),
exercises=# (199, 4, 1, '2012-07-16 11:00:00', 2),
exercises=# (200, 4, 3, '2012-07-16 12:00:00', 2),
exercises=# (201, 4, 3, '2012-07-16 17:30:00', 2),
exercises=# (202, 6, 0, '2012-07-16 18:30:00', 2),
exercises=# (203, 7, 4, '2012-07-16 08:00:00', 2),
```

## To Choose Database Exercise:

```
postgres=# \c exercises
You are now connected to database "exercises" as user "postgres".
exercises=#
```

# Simple SQL Queries

## 1.Retrieve everything from a table

Query: Select \* from facilities

SQLOUTPUT:

```
exercises=# select * from facilities;
```

| facid | name            | membercost | guestcost | initialoutlay | monthlymaintenance |
|-------|-----------------|------------|-----------|---------------|--------------------|
| 0     | Tennis Court 1  | 5          | 25        | 10000         | 200                |
| 1     | Tennis Court 2  | 5          | 25        | 8000          | 200                |
| 2     | Badminton Court | 0          | 15.5      | 4000          | 50                 |
| 3     | Table Tennis    | 0          | 5         | 320           | 10                 |
| 4     | Massage Room 1  | 35         | 80        | 4000          | 3000               |
| 5     | Massage Room 2  | 35         | 80        | 4000          | 3000               |
| 6     | Squash Court    | 3.5        | 17.5      | 5000          | 80                 |
| 7     | Snooker Table   | 0          | 5         | 450           | 15                 |
| 8     | Pool Table      | 0          | 5         | 400           | 15                 |

(9 rows)

## 2.Retrieve specific columns from a table

Query: select name, membercost from facilities;

SQLOUTPUT:

```
exercises=# select name, membercost from facilities;
```

| name            | membercost |
|-----------------|------------|
| Tennis Court 1  | 5          |
| Tennis Court 2  | 5          |
| Badminton Court | 0          |
| Table Tennis    | 0          |
| Massage Room 1  | 35         |
| Massage Room 2  | 35         |
| Squash Court    | 3.5        |
| Snooker Table   | 0          |
| Pool Table      | 0          |

(9 rows)

## 3.Control which rows are retrieved

Query: select \* from facilities where membercost > 0;

SQLOUTPUT:

```

exercises=# select * from facilities where membercost > 0;
 facid |      name      | membercost | guestcost | initialoutlay | monthlymaintenance
-----+-----+-----+-----+-----+-----
      0 | Tennis Court 1 |          5 |         25 |         10000 |             200
      1 | Tennis Court 2 |          5 |         25 |          8000 |             200
      4 | Massage Room 1 |         35 |         80 |          4000 |            3000
      5 | Massage Room 2 |         35 |         80 |          4000 |            3000
      6 | Squash Court   |         3.5 |        17.5 |          5000 |             80
(5 rows)

exercises=#

```

## 4.Control which rows are retrieved - part 2

**Query:** select facid, name, membercost, monthlymaintenance from facilities where membercost > 0 and (membercost < monthlymaintenance/50.0);

**SQLOUTPUT:**

```

exercises=# select facid, name, membercost, monthlymaintenance from facilities where membercost > 0 and (membercost < monthlymaintenance/50.0);
 facid |      name      | membercost | monthlymaintenance
-----+-----+-----+-----
      4 | Massage Room 1 |         35 |            3000
      5 | Massage Room 2 |         35 |            3000
(2 rows)

exercises=#

```

## 5.Basic string searches

**Query:** select \* from facilities where name like '%Tennis%';

**SQLOUTPUT:**

```

exercises=# select * from facilities where name like '%Tennis%';
 facid |      name      | membercost | guestcost | initialoutlay | monthlymaintenance
-----+-----+-----+-----+-----+-----
      0 | Tennis Court 1 |          5 |         25 |         10000 |             200
      1 | Tennis Court 2 |          5 |         25 |          8000 |             200
      3 | Table Tennis   |          0 |          5 |           320 |              10
(3 rows)

```

## 6.Matching against multiple possible values

**Query:** select \* from facilities where facid in (1,5);

## SQLOUTPUT:

```
exercises=# select * from facilities where facid in (1,5);
 facid |      name      | membercost | guestcost | initialoutlay | monthlymaintenance
-----+-----+-----+-----+-----+-----
      1 | Tennis Court 2 |          5 |         25 |          8000 |             200
      5 | Massage Room 2 |         35 |         80 |          4000 |             3000
(2 rows)
```

## 7. Classify results into buckets

**Query:** select name, case when (monthlymaintenance > 100) then 'expensive' else 'cheap' end as cost from facilities;

## SQLOUTPUT:

```
exercises=# select name, case when (monthlymaintenance > 100) then 'expensive' else 'cheap' end as cost from facilities;
 name      | cost
-----+-----
Tennis Court 1 | expensive
Tennis Court 2 | expensive
Badminton Court | cheap
Table Tennis | cheap
Massage Room 1 | expensive
Massage Room 2 | expensive
Squash Court | cheap
Snooker Table | cheap
Pool Table | cheap
(9 rows)
```

## 8. Working with dates

**Query:** select memid, surname, firstname, joindate from members where joindate >= '2012-09-01';

## SQLOUTPUT:

```
exercises=# select memid, surname, firstname, joindate from members where joindate >= '2012-09-01';
 memid | surname      | firstname | joindate
-----+-----+-----+-----
     24 | Sarwin      | Ramnaresh | 2012-09-01 08:44:42
     26 | Jones      | Douglas   | 2012-09-02 18:43:05
     27 | Rumney     | Henrietta | 2012-09-05 08:42:35
     28 | Farrell    | David     | 2012-09-15 08:22:05
     29 | Worthington-Smyth | Henry   | 2012-09-17 12:27:15
     30 | Purview    | Millicent | 2012-09-18 19:04:01
     33 | Tupperware | Hyacinth  | 2012-09-18 19:32:05
     35 | Hunt       | John      | 2012-09-19 11:32:45
     36 | Crumpet    | Erica     | 2012-09-22 08:36:38
     37 | Smith      | Darren    | 2012-09-26 18:08:45
(10 rows)
```

## 9. Removing duplicates, and ordering results

**Query:** select distinct surname from members order by surname limit 10;

## SQLOUTPUT:

```
exercises=# select distinct surname from members order by surname limit 10;
 surname
-----
 Bader
 Baker
 Boothe
 Butters
 Coplin
 Crumpet
 Dare
 Farrell
 Genting
 GUEST
(10 rows)
```

## 10. Combining results from multiple queries

**Query:** select surname from members union select name from facilities;

## SQLOUTPUT:

```
exercises=# select surname from members union select name from facilities;
 surname
-----
 Hunt
 Farrell
 Tennis Court 2
 Table Tennis
 Dare
 Rownam
 GUEST
 Badminton Court
 Smith
 Tupperware
 Owen
 Worthington-Smyth
 Butters
 Rumney
 Tracy
 Crumpet
 Purview
 Massage Room 2
 Sarwin
 Baker
 Pool Table
 Snooker Table
 Jones
 Coplin
 Mackenzie
 Boothe
 Joplette
 Stibbons
 Squash Court
 Tennis Court 1
 Pinker
 Genting
 Bader
 Massage Room 1
(34 rows)
```

## 11.Simple aggregation

**Query:** select max(joindate) as latest from members;

### SQLOUTPUT:

```
exercises=# select max(joindate) as latest from members;
         latest
-----
2012-09-26 18:08:45
(1 row)
```

## 12.More aggregation

**Query:** select firstname, surname, joindate from members order by joindate desc limit 1;

### SQLOUTPUT:

```
exercises=# select firstname, surname, joindate from members order by joindate desc limit 1;
firstname | surname |      joindate
-----+-----+-----
Darren    | Smith   | 2012-09-26 18:08:45
(1 row)
```