Отчет по лабораторной работе 1, вариант 7.

Выполнил: Золотухин Андрей Александрович КС-36
Преподаватель: Семёнов Геннадий Николаевич

ЗАДАНИЕ

7. 1. Разработать схему базы данных на основе следующих данных:
Общежитие:

Дата предоставления	Группа	Студент	Сосед по комнате	Удалённость собственного жилья от института, км.	№ комнаты	Стипендия, руб.
3.08.05	КС-10	Васильев	Гришин	200	14	2000
15.08.05	КС-14	Петрова	Сидорова	435	37	2570
2.08.05	КС-14	Сидоров	Иванов	112	25	2000
2.08.05	КС-10	Иванов	Сидоров	240	25	2240
14.08.05	КС-10	Сидорова	Петрова	1200	37	4500
4.08.05	КС-14	Гришин	Васильев	780	14	4000

2. В СУБД PostrgeSQL создайте базу данных, содержащую не менее 3-х таблиц в 3-й нормальной форме.
3. Заполните таблицы данными.

Создание базы данных

Пароль к серверу и базе данных: 123

```
CREATE DATABASE “Var7”
    WITH
    OWNER = postgres
    ENCODING = ‘UTF8’
    TABLESPACE = pg_default
    CONNECTION LIMIT = -1
    IS_TEMPLATE = False;
```

Создание таблиц

1) **Группы**

```
    CREATE TABLE IF NOT EXISTS groups
    (
        group_id integer NOT NULL GENERATED ALWAYS AS
IDENTITY (INCREMENT 1 START 1 MINVALUE 1 MAXVALUE
2147483647),
        group_name character varying(5) NOT NULL,
```

```
        CONSTRAINT groups_pkey PRIMARY KEY (group_id)
    );
```

2) Комнаты

```
    CREATE TABLE IF NOT EXISTS rooms
    (
        room_id integer NOT NULL GENERATED ALWAYS AS
IDENTITY (INCREMENT 1 START 1 MINVALUE 1 MAXVALUE
2147483647),
        room_number integer NOT NULL,
        CONSTRAINT rooms_pkey PRIMARY KEY (room_id)
    );
```

3) Студенты

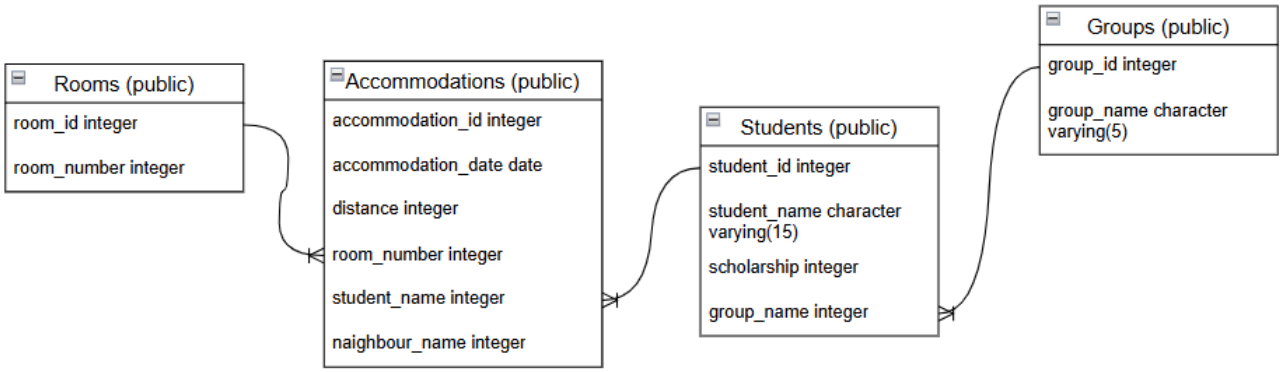
```
    CREATE TABLE IF NOT EXISTS students
    (
        student_id integer NOT NULL GENERATED ALWAYS
AS IDENTITY (INCREMENT 1 START 1 MINVALUE 1 MAXVALUE
2147483647),
        student_name character varying(15) NOT NULL,
        scholarship integer NOT NULL,
        group_name integer REFERENCES
groups(group_id) NOT NULL,
        CONSTRAINT students_pkey PRIMARY KEY
(student_id)
    );
```

4) Размещения

```
    CREATE TABLE IF NOT EXISTS accommodations
    (
```

```
        accommodation_id integer NOT NULL GENERATED
ALWAYS AS IDENTITY (INCREMENT 1 START 1 MINVALUE 1
MAXVALUE 2147483547)
        accommodation_date date NOT NULL,
        distance integer NOT NULL,
        room_number integer REFERENCES rooms(room_id)
NOT NULL,
        student_name integer REFERENCES
students(student_id) NOT NULL,
        neighbour_name integer REFERENCES
students(student_id) NOT NULL,
    );
```

Схема базы данных



Полный код создания таблиц

ГРУППЫ

```
    CREATE TABLE IF NOT EXISTS public.groups
    (
        group_id SERIAL,
        group_name character varying(5) NOT NULL,
        CONSTRAINT groups_pkey PRIMARY KEY (group_id)
```

```
    );
```

КОМНАТЫ

```
    CREATE TABLE IF NOT EXISTS public.rooms
    (
        room_id SERIAL,
        room_number integer NOT NULL,
        CONSTRAINT room_number_pkey PRIMARY KEY
(room_id)
    );
```

СТУДЕНТЫ

```
    CREATE TABLE IF NOT EXISTS public.students
    (
        student_id SERIAL,
        student_name character varying(15) NOT NULL,
        scholarship integer NOT NULL,
        group_name integer NOT NULL,
        CONSTRAINT students_pkey PRIMARY KEY
(student_id),
        CONSTRAINT group_group_name_fkey FOREIGN KEY
(group_name)
            REFERENCES public.groups(group_id) MATCH
        SIMPLE
            ON UPDATE NO ACTION
            ON DELETE NO ACTION
    );
```

РАЗМЕЩЕНИЯ

```
CREATE TABLE IF NOT EXISTS public.accommodations
(
    accommodation_id SERIAL,
    accommodation_date date NOT NULL,
    distance integer NOT NULL,
    room_number integer NOT NULL,
    student_name integer NOT NULL,
    neighbour_name integer NOT NULL,
    CONSTRAINT room_room_number_fkey FOREIGN KEY
(room_number)
        REFERENCES public.rooms(room_id) MATCH
    SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION,
    CONSTRAINT student_student_name_fkey FOREIGN
KEY (student_name)
        REFERENCES public.students(student_id)
    MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION,
    CONSTRAINT neighbour_neighbour_name_fkey
FOREIGN KEY (neighbour_name)
        REFERENCES public.students(student_id)
    MATCH SIMPLE
        ON UPDATE NO ACTION
        ON DELETE NO ACTION
);
```

Заполнение таблиц

1) **Группы**

```
INSERT INTO groups (group_name) VALUES
('CS-10'),
```

```
('CS-14');
```

2) Комнаты

```
INSERT INTO rooms (room_number) VALUES
(14),
(37),
(25);
```

3) Студенты

```
INSERT INTO students (student_name, scholarship, group_name) VALUES
('Vasilyev', 2000, 1),
('Petrova', 2570, 2),
('Sidorov', 2000, 2),
('Ivanov', 2240, 1),
('Sidorova', 4500, 1),
('Grishin', 4000, 2);
```

4) Размещения

```
INSERT INTO accommodations (accommodation_date, distance, room_number, student_name, neighbour_name) VALUES
('2005-08-03', 200, 1, 1, 6),
('2005-08-15', 435, 2, 2, 5),
('2005-08-02', 112, 3, 3, 4),
('2005-08-02', 240, 3, 4, 3),
('2005-08-14', 1200, 2, 5, 2),
('2005-08-04', 780, 1, 6, 1);
```

Результат заполнения таблиц

```
Var7=# SELECT * FROM groups;
 group_id | group_name
----------+------------
        1 | CS-10
        2 | CS-14
(2 rows)
```

```
Var7=# SELECT * FROM rooms;
 room_id | room_number
---------+-------------
       1 |          14
       2 |          37
       3 |          25
(3 rows)
```

```
Var7=# SELECT * FROM students;
 student_id | student_name | scholarship | group_name
------------+--------------+-------------+------------
          1 | Vasilyev     |        2000 |          1
          2 | Petrova      |        2570 |          2
          3 | Sidorov      |        2000 |          2
          4 | Ivanov       |        2240 |          1
          5 | Sidorova     |        4500 |          1
          6 | Grishin      |        4000 |          2
(6 rows)
```

```
Var7=# SELECT * FROM accommodations;
 accommodation_id | accommodation_date | distance | room_number | student_name | neighbour_name
------------------+--------------------+----------+-------------+--------------+---------------
                1 | 2005-08-03         |      200 |           1 |            1 |             6
                2 | 2005-08-15         |      435 |           2 |            2 |             5
                3 | 2005-08-02         |      112 |           3 |            3 |             4
                4 | 2005-08-02         |      240 |           3 |            4 |             3
                5 | 2005-08-14         |     1200 |           2 |            5 |             2
                6 | 2005-08-04         |      780 |           1 |            6 |             1
(6 rows)
```