

## **Отчетный лист по предмету "Базы данных"**

Лабораторная работа № 3

СОЗДАНИЕ ТАБЛИЦ БАЗЫ ДАННЫХ POSTGRESQL. ЗАПОЛНЕНИЕ  
ТАБЛИЦ РАБОЧИМИ ДАННЫМИ

Группа: K3242

Студент: Здоров Филипп Кириллович

# Оглавление

[Цель работы](#)

[Задание](#)

[Скрипт работы с БД](#)

[Вывод](#)

# Цель работы

Овладеть практическими навыками создания таблиц базы данных PostgreSQL 10 (11), заполнения их рабочими данными, резервного копирования и восстановления БД.

## Задание

### Скрипт работы с БД

```
-- Создаем базу данных "Lab"
CREATE DATABASE "Lab"
WITH
  OWNER = postgres
  ENCODING = 'UTF8'
  LC_COLLATE = 'Russian_Russia.1251'
  LC_CTYPE = 'Russian_Russia.1251'
  TABLESPACE = pg_default
  CONNECTION LIMIT = -1;

-----

-- Создаем схему "Public"
CREATE SCHEMA public
  AUTHORIZATION postgres;

COMMENT ON SCHEMA public
  IS 'standard public schema';

GRANT ALL ON SCHEMA public TO PUBLIC;

GRANT ALL ON SCHEMA public TO postgres;

-----

-- Создаем таблицы!

-- Создаем таблицу "Номер"
CREATE TABLE public."Номер"
(
  "физический_номер" integer NOT NULL,
```

```

    "телефонный_номер" integer,
    "этаж" integer NOT NULL,
    "id_типа_номера" integer NOT NULL,
    PRIMARY KEY ("физический_номер")
);

ALTER TABLE public."Номер"
    OWNER to postgres;

COMMENT ON TABLE public."Номер"
    IS 'Таблица, содержащая информацию о
    номерах гостиницы';

-- Создаем таблицу "Тип_номера"
CREATE TABLE public."Тип_номера"
(
    "id_типа_номера" integer NOT NULL,
    "количество_мест" integer NOT NULL,
    "стоимость_проживания" integer NOT NULL,
    PRIMARY KEY ("id_типа_номера")
);

ALTER TABLE public."Тип_номера"
    OWNER to postgres;

COMMENT ON TABLE public."Тип_номера"
    IS 'Таблица, содержащая информацию о типах
    номеров гостиницы';

-- Создаем таблицу "Служащий"
CREATE TABLE public."Служащий"
(
    "табельный_номер" integer NOT NULL,
    "фамилия" character(30) NOT NULL,
    "имя" character(30) NOT NULL,
    "отчество" character(30),
    PRIMARY KEY ("табельный_номер")
);

ALTER TABLE public."Служащий"
    OWNER to postgres;

COMMENT ON TABLE public."Служащий"
    IS 'Таблица, содержащая информацию о
    служащих гостиницы';

```

```
-- Создаем таблицу "Смена"
CREATE TABLE public."Смена"
(
    "id_смены" integer NOT NULL,
    "табельный_номер" integer NOT NULL,
    "физический_номер" integer NOT NULL,
    "дата" date NOT NULL,
    PRIMARY KEY ("id_смены")
);

ALTER TABLE public."Смена"
    OWNER to postgres;

COMMENT ON TABLE public."Смена"
    IS 'Таблица, содержащая информацию о сменах
    служащих';

-- Создаем таблицу "Клиент"
CREATE TABLE public."Клиент"
(
    "номер_паспорта" integer NOT NULL,
    "фамилия" character(30) NOT NULL,
    "имя" character(20) NOT NULL,
    "отчество" character(30),
    PRIMARY KEY ("номер_паспорта")
);

ALTER TABLE public."Клиент"
    OWNER to postgres;

COMMENT ON TABLE public."Клиент"
    IS 'Таблица, содержащая информацию о
    клиентах гостиницы';

-- Создаем таблицу "Клиент"
CREATE TABLE public."Проживание"
(
    "id_проживания" integer NOT NULL,
    "номер_паспорта" integer NOT NULL,
    "физический_номер" integer NOT NULL,
```

```

"д а т а_з а с е л е н и я" date NOT NULL,
"д а т а_в ы с е л е н и я" date NOT NULL,
"г о р о д_п р и б ы т и я" character(30),
PRIMARY KEY ("id_п р о ж и в а н и я")
);

ALTER TABLE public."П р о ж и в а н и е"
    OWNER to postgres;

COMMENT ON TABLE public."П р о ж и в а н и е"
    IS 'Т а б л и ц а, с о д е р ж а щ а я и н ф о р м а ц и ю о
    р е з е р в и р о в а н и и м е с т в г о с т и н и ц е';

-----

-----

-- Создаем ограничения!

-- Создаем внешний ключ для таблицы "Номер"
ALTER TABLE public."Н о м е р"
    ADD CONSTRAINT "id_т и п а_н о м е р а" FOREIGN KEY
("id_т и п а_н о м е р а")
    REFERENCES public."Т и п_н о м е р а" ("id_т и п а_н о м е р а") MATCH
SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;

-- Создаем внешний ключ для таблицы "Смена"
ALTER TABLE public."С м е н а"
    ADD CONSTRAINT "ф и з и ч е с к и й_н о м е р" FOREIGN KEY
("ф и з и ч е с к и й_н о м е р")
    REFERENCES public."Н о м е р" ("ф и з и ч е с к и й_н о м е р") MATCH
SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;

ALTER TABLE public."С м е н а"
    ADD CONSTRAINT "т а б е л ь н ы й_н о м е р" FOREIGN KEY
("т а б е л ь н ы й_н о м е р")
    REFERENCES public."С л у ж а щ и й" ("т а б е л ь н ы й_н о м е р")
MATCH SIMPLE
    ON UPDATE NO ACTION

```

```
ON DELETE NO ACTION;
```

```
-- Создаем внешний ключ для таблицы
"Проживание"
ALTER TABLE public."Проживание"
    ADD CONSTRAINT "номер_паспорта" FOREIGN KEY
("номер_паспорта")
    REFERENCES public."Клиент" ("номер_паспорта") MATCH
SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;

ALTER TABLE public."Проживание"
    ADD CONSTRAINT "физический_номер" FOREIGN KEY
("физический_номер")
    REFERENCES public."Номер" ("физический_номер") MATCH
SIMPLE
    ON UPDATE NO ACTION
    ON DELETE NO ACTION;
```

```
-----
-----

-- Заполнение данных
```

```
-- Клиент
INSERT INTO public."Клиент" (
    "номер_паспорта", "фамилия", "имя",
    "отчество")
    VALUES (654992, 'Здоров', 'Филипп', 'Кириллович');

INSERT INTO public."Клиент" (
    "номер_паспорта", "фамилия", "имя")
    VALUES (836503, 'Сульфитжад', 'Ибрагим');

INSERT INTO public."Клиент" (
    "номер_паспорта", "фамилия", "имя",
    "отчество")
    VALUES (835603, 'Говоров', 'Антон', 'Игоревич');

INSERT INTO public."Клиент" (
    "номер_паспорта", "фамилия", "имя",
    "отчество")
    VALUES (111111, 'Анонимов', 'Аноним',
    'Аноимович');
```

```
INSERT INTO public."Клиент"(  
    "номер_паспорта", "фамилия", "имя",  
    "отчество")  
    VALUES (232323, 'Лапенко', 'Антон',  
    'Вячеславович');  
  
-- Тип_номера  
INSERT INTO public."Тип_номера"(  
    "id_типа_номера", "количество_мест",  
    "стоимость_проживания")  
    VALUES (1, 1, 1200);  
  
INSERT INTO public."Тип_номера"(  
    "id_типа_номера", "количество_мест",  
    "стоимость_проживания")  
    VALUES (2, 2, 2200);  
  
INSERT INTO public."Тип_номера"(  
    "id_типа_номера", "количество_мест",  
    "стоимость_проживания")  
    VALUES (3, 3, 4000);  
  
-- Служащий  
INSERT INTO public."Служащий"(  
    "табельный_номер", "фамилия", "имя",  
    "отчество")  
    VALUES (1, 'Самощенко', 'Алексей',  
    'Алексеевич');  
  
INSERT INTO public."Служащий"(  
    "табельный_номер", "фамилия", "имя",  
    "отчество")  
    VALUES (2, 'Горшков', 'Артем', 'Сергеевич');  
  
INSERT INTO public."Служащий"(  
    "табельный_номер", "фамилия", "имя")  
    VALUES (3, 'Мэнсон', 'Мерилин');  
  
INSERT INTO public."Служащий"(  
    "табельный_номер", "фамилия", "имя")  
    VALUES (4, 'Паркер', 'Питер');  
  
INSERT INTO public."Служащий"(  
    "табельный_номер", "фамилия", "имя",  
    "отчество")  
    VALUES (5, 'Поставьте', 'Пожалуйста',  
    'Пятерку');
```



```
-- Но м е р
INSERT INTO public."Но м е р"("физический_номер", "телефонный_номер",
"этаж", "id_типа_номера")
VALUES (1, 79523538666, 1, 1);

INSERT INTO public."Но м е р"("физический_номер", "телефонный_номер",
"этаж", "id_типа_номера")
VALUES (2, 79523538665, 1, 3);

INSERT INTO public."Но м е р"("физический_номер", "телефонный_номер",
"этаж", "id_типа_номера")
VALUES (3, 79523538664, 2, 2);

INSERT INTO public."Но м е р"("физический_номер", "телефонный_номер",
"этаж", "id_типа_номера")
VALUES (4, 79523538663, 3, 2);

INSERT INTO public."Но м е р"("физический_номер", "телефонный_номер",
"этаж", "id_типа_номера")
VALUES (5, 79523538662, 4, 1);

-- С м е н а
INSERT INTO public."С м е н а"("id_смены", "табельный_номер",
"физический_номер", "дата")
VALUES (1, 5, 5, '22.06.2000');

INSERT INTO public."С м е н а"("id_смены", "табельный_номер",
"физический_номер", "дата")
VALUES (2, 4, 4, '22.06.2000');

INSERT INTO public."С м е н а"("id_смены", "табельный_номер",
"физический_номер", "дата")
VALUES (3, 3, 1, '22.06.2000');

INSERT INTO public."С м е н а"("id_смены", "табельный_номер",
"физический_номер", "дата")
```

```
VALUES (4, 2, 2, '22.06.2000');

INSERT INTO public."Смена"(
    "id_смены", "табельный_номер",
    "физический_номер", "дата")
VALUES (5, 1, 3, '22.06.2000');

-- Проживание
INSERT INTO public."Проживание"(
    "id_проживания", "номер_паспорта",
    "физический_номер", "дата_заселения",
    "дата_выселения", "город_прибытия")
VALUES (1, 654992, 1, '22.06.2000', '15.10.2020', 'Допса');

INSERT INTO public."Проживание"(
    "id_проживания", "номер_паспорта",
    "физический_номер", "дата_заселения",
    "дата_выселения", "город_прибытия")
VALUES (2, 836503, 2, '22.11.2020', '22.12.2020', 'Магадан');

INSERT INTO public."Проживание"(
    "id_проживания", "номер_паспорта",
    "физический_номер", "дата_заселения",
    "дата_выселения", "город_прибытия")
VALUES (3, 835603, 1, '16.10.2020', '16.10.3030',
'Надо-Было-Раньше-Думать');

INSERT INTO public."Проживание"(
    "id_проживания", "номер_паспорта",
    "физический_номер", "дата_заселения",
    "дата_выселения", "город_прибытия")
VALUES (4, 111111, 3, '02.08.2020', '02.12.2020',
'Йошкар-Ола');

INSERT INTO public."Проживание"(
    "id_проживания", "номер_паспорта",
    "физический_номер", "дата_заселения",
    "дата_выселения")
VALUES (5, 232323, 3, '02.08.2020', '02.12.2020');
```

## Вывод

В ходе лабораторной работы была создана база данных PostgreSQL для управления гостиницей. Также, были созданы таблицы и занесены некоторые данные.