

Министерство образования и науки Российской Федерации  
федеральное государственное автономное образовательное учреждение  
высшего образования  
«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ ИТМО»  
Факультет среднего профессионального образования

ОТЧЁТ  
ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 4  
по теме: Создание таблиц БД POSTGRESQL. Заполнение таблиц  
рабочими данными  
по дисциплине: Основы проектирования баз данных

Специальность 09.02.07 Информационные системы и программирование

Проверил:  
\_\_\_\_\_ Говоров А. И.  
Дата: «\_\_\_\_\_» \_\_\_\_\_ 2020г.  
Оценка \_\_\_\_\_

Выполнил:  
студент группы Y2336  
\_\_\_\_\_ Наумов М.А.

Санкт-Петербург 2020

## ЦЕЛЬ РАБОТЫ

Цель практической работы №4: овладеть практическими навыками создания таблиц базы данных PostgreSQL 10 (11), заполнения их рабочими данными, резервного копирования и восстановления БД.

## ПРАКТИЧЕСКОЕ ЗАДАНИЕ

1. Создать базу данных с использованием pgadmin 4 (согласно индивидуальному заданию).
2. Создать схему в составе базы данных.
3. Создать таблицы базы данных.
4. Заполнить таблицы БД рабочими данными.
5. Создать резервную копию БД.
6. Восстановить БД на другом ПК..

## DUMP, СОДЕРЖАЩИЙ СКРИПТЫ РАБОТЫ С БД

### СОЗДАНИЕ СТРУКТУРЫ ТАБЛИЦ

```
CREATE DATABASE "NAUMOV" WITH TEMPLATE = TEMPLATE0 ENCODING = 'UTF8' LC_COLLATE = 'RUSSIAN_RUSSIA.1251' LC_CTYPE = 'RUSSIAN_RUSSIA.1251';
```

---

```
CREATE TABLE PUBLIC.CICERONE (  
    ID_CICERONE INTEGER NOT NULL,  
    NAME_CICERONE "CHAR"[],  
    WORK_EXPERIENCE INTERVAL,  
    CONTACT_CICERONE "CHAR"[],  
    PORTFOLIO TEXT  
);
```

---

```
CREATE TABLE PUBLIC.CLIMBER (  
    ID_CLIMBER INTEGER NOT NULL,  
    NAME_CLIMBER "CHAR"[] NOT NULL,  
    ID_CLUB INTEGER NOT NULL,  
    CONTACT_CLIMBER "CHAR"[] NOT NULL,  
    ADDRESS_CLIMED "CHAR"[] NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC.CLIMBING (  
    ID_CLIMBING INTEGER NOT NULL,  
    ID_GROUP INTEGER NOT NULL,  
    START_PLAN_CLIMBING DATE NOT NULL,  
    START_FACT_CLIMBING DATE NOT NULL,  
    END_PLAN_CLIMBING DATE NOT NULL,
```

```
END_FACT_CLIMBING DATE NOT NULL,  
DURATION_CLIMBING INTERVAL NOT NULL,  
BRIEF_INFORMATION_CLIMBING TEXT NOT NULL,  
ID_ROUTE INTEGER  
);
```

---

```
CREATE TABLE PUBLIC.CLUB (  
  "ID_CLUB" INTEGER NOT NULL,  
  CITY_CLUB "CHAR"[] NOT NULL,  
  CONTACT_CLUB_NAME "CHAR"[] NOT NULL,  
  COUNTRY_CLUB "CHAR"[] NOT NULL,  
  EMAIL_CLUB "CHAR"[] NOT NULL,  
  NAME_CLUB "CHAR"[] NOT NULL,  
  PHONE_NUMBER_CLUB INTEGER NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC.CONTRACT_CICERONE (  
  ID_AGENCY INTEGER NOT NULL,  
  ID_CICERONE INTEGER NOT NULL,  
  TERMS_CONTRACT_CICERONE TEXT NOT NULL,  
  ID_CONTRACT_CICERONE INTEGER NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC.CONTRACT_CLIMBER (  
  ID_CLIMBER INTEGER NOT NULL,  
  ID_CICERONE INTEGER NOT NULL,  
  ID_GROUP INTEGER NOT NULL,  
  ID_AGENCY INTEGER NOT NULL,
```

```
TERMS_CONTRACT_CLIMBER TEXT NOT NULL,  
ID_CONTRACT_CLIMBER INTEGER NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC.EMERGANCY_SITUATION (  
  "ID_SITUATION" INTEGER NOT NULL,  
  TYPE_SITUATION "CHAR"[] NOT NULL,  
  ID_CLIMBER INTEGER NOT NULL,  
  ID_CLIMBING INTEGER NOT NULL,  
  TIME_SITUATION TIME WITH TIME ZONE NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC."GROUP" (  
  ID_GROUP INTEGER NOT NULL,  
  QUANTITY_MEMBER INTEGER  
);
```

---

```
CREATE TABLE PUBLIC.INSURANCE_CASE (  
  ID_INSURANCE_CASE INTEGER NOT NULL,  
  ID_SITUATION INTEGER NOT NULL,  
  SUM_REIMBURSEMENT INTEGER NOT NULL,  
  DATE_REIMBURSEMENT DATE NOT NULL,  
  ID_INSURANCE_CONTRACT INTEGER NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC.INSURANCE_CONTRACT (  
  ID_AGENCY INTEGER NOT NULL,  
  ID_CLIMBER INTEGER NOT NULL,
```

```
INSURANCE_CONTRACT_TERMS TEXT NOT NULL,  
ID_INSURANCE_CONTRACT INTEGER NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC.MOUNTAIN (  
  COUNTRY_MOUNTAIN "CHAR"[] NOT NULL,  
  AREA_MOUNTAIN "CHAR"[] NOT NULL,  
  ID_MOUNTAIN INTEGER NOT NULL,  
  NAME_MOUNTAIN "CHAR" NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC.PEAK (  
  NAME_PEAK "CHAR"[] NOT NULL,  
  COUNTRY_PEAK "CHAR"[] NOT NULL,  
  AREA_PEAK "CHAR"[] NOT NULL,  
  HIGHT_PEAK INTEGER NOT NULL,  
  ID_MOUNTAIN INTEGER NOT NULL,  
  ID_PEAK INTEGER NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC.TRAVEL_AGENCY (  
  ID_AGENCY INTEGER NOT NULL,  
  NUMBER_OF_CICERONE INTEGER NOT NULL,  
  NAME_AGENCY "CHAR"[] NOT NULL  
);
```

---

```
CREATE TABLE PUBLIC.ROUTE (  
  ID_ROUTE INTEGER NOT NULL,  
  DESCRIPTION_ROUTE TEXT NOT NULL,  
  NAME_ROUTE "CHAR"[] NOT NULL,  
  "ID_PEAK" INTEGER NOT NULL  
);
```



## СОЗДАНИЕ ПЕРВИЧНЫХ КЛЮЧЕЙ

```
ALTER TABLE ONLY PUBLIC.CLUB  
ADD CONSTRAINT "CLUB_PKEY" PRIMARY KEY ("ID_CLUB");
```

---

```
ALTER TABLE ONLY PUBLIC.CICERONE  
ADD CONSTRAINT CICERONE_PKEY PRIMARY KEY (ID_CICERONE);
```

---

```
ALTER TABLE ONLY PUBLIC."ALLOWANCE"  
ADD CONSTRAINT "ALLOWANCE_PKEY" PRIMARY KEY ("ID_ALLOWANCE");  
ALTER TABLE ONLY PUBLIC."ALLOWANCE" DROP CONSTRAINT "ALLOWANCE_PKEY";
```

---

```
ALTER TABLE ONLY PUBLIC.CLIMBER  
ADD CONSTRAINT CLIMBER_PKEY PRIMARY KEY (ID_CLIMBER);
```

---

```
ALTER TABLE ONLY PUBLIC.CLIMBING  
ADD CONSTRAINT CLIMBING_PKEY PRIMARY KEY (ID_CLIMBING);
```

---

```
AALTER TABLE ONLY PUBLIC.CONTRACT_CICERONE  
ADD CONSTRAINT CONTRACT_CICERONE_PKEY PRIMARY KEY (ID_CONTRACT_CICERONE);
```

---

```
ALTER TABLE ONLY PUBLIC.CONTRACT_CLIMBER  
ADD CONSTRAINT CONTRACT_CLIMBER_PKEY PRIMARY KEY (ID_CONTRACT_CLIMBER);
```

---

```
ALTER TABLE ONLY PUBLIC.EMERGANCY_SITUATION  
ADD CONSTRAINT EMERGANCY_SITUATION_PKEY PRIMARY KEY ("ID_SITUATION");
```

---

```
ALTER TABLE ONLY PUBLIC."GROUP"  
ADD CONSTRAINT GROUP_PKEY PRIMARY KEY (ID_GROUP);
```

---

```
ALTER TABLE ONLY PUBLIC.INSURANCE_CASE  
ADD CONSTRAINT INSURANCE_CASE_PKEY PRIMARY KEY (ID_INSURANCE_CASE);
```

---

```
ALTER TABLE ONLY PUBLIC.INSURANCE_CONTRACT  
ADD CONSTRAINT INSURANCE_CONTRACT_PKEY PRIMARY KEY (ID_INSURANCE_CONTRACT);
```

---

```
ALTER TABLE ONLY PUBLIC.MOUNTAIN  
ADD CONSTRAINT MOUNTAIN_PKEY PRIMARY KEY (ID_MOUNTAIN);
```

---

```
ALTER TABLE ONLY PUBLIC.PEAK  
ADD CONSTRAINT PEAK_PKEY PRIMARY KEY (ID_PEAK);
```

---

```
ALTER TABLE ONLY PUBLIC.ROUTE  
ADD CONSTRAINT ROUTE_NAME_ROUTE_KEY UNIQUE (NAME_ROUTE);
```

---

```
ALTER TABLE ONLY PUBLIC.ROUTE  
ADD CONSTRAINT ROUTE_PKEY PRIMARY KEY (ID_ROUTE);
```

---

```
ALTER TABLE ONLY PUBLIC.TRAVEL_AGENCY  
ADD CONSTRAINT TRAVEL_AGENCY_PKEY PRIMARY KEY (ID_AGENCY)
```

---

## СОЗДАНИЕ ВНЕШНИХ КЛЮЧЕЙ

```
ALTER TABLE ONLY PUBLIC.CLIMBING  
ADD CONSTRAINT CLIMBING_ID_GROUP_FKEY FOREIGN KEY (ID_GROUP) REFERENCES PUBLIC."GROUP"(ID_GROUP);
```

---

```
ALTER TABLE ONLY PUBLIC.CLIMBING  
ADD CONSTRAINT CLIMBING_ID_ROUTE_FKEY FOREIGN KEY (ID_ROUTE) REFERENCES PUBLIC.ROUTE(ID_ROUTE) ON UPDATE CASCADE ON DELETE CASCADE;
```

---

```
ALTER TABLE ONLY PUBLIC.EMERGANCY_SITUATION  
ADD CONSTRAINT EMERGANCY_SITUATION_ID_CLIMBER_FKEY FOREIGN KEY (ID_CLIMBER) REFERENCES PUBLIC.CLIMBER(ID_CLIMBER);
```

---

```
ALTER TABLE ONLY PUBLIC.EMERGANCY_SITUATION  
ADD CONSTRAINT EMERGANCY_SITUATION_ID_CLIMBING_FKEY FOREIGN KEY (ID_CLIMBING) REFERENCES PUBLIC.CLIMBING(ID_CLIMBING);
```

---

```
ALTER TABLE ONLY PUBLIC.INSURANCE_CONTRACT  
ADD CONSTRAINT ID_AGENCY FOREIGN KEY (ID_AGENCY) REFERENCES PUBLIC.TRAVEL_AGENCY(ID_AGENCY);
```

---

```
ALTER TABLE ONLY PUBLIC.CONTRACT_CICERONE  
ADD CONSTRAINT ID_AGENCY FOREIGN KEY (ID_AGENCY) REFERENCES PUBLIC.TRAVEL_AGENCY(ID_AGENCY);
```

---

```
ALTER TABLE ONLY PUBLIC.CONTRACT_CLIMBER
  ADD CONSTRAINT ID_AGENCY FOREIGN KEY (ID_AGENCY) REFERENCES PUB-
LIC.TRAVEL_AGENCY(ID_AGENCY);
```

---

```
ALTER TABLE ONLY PUBLIC.CONTRACT_CICERONE
  ADD CONSTRAINT ID_CICERONE FOREIGN KEY (ID_CICERONE) REFERENCES PUBLIC.CICERONE(ID_CIC-
ERONE);
```

---

```
ALTER TABLE ONLY PUBLIC.CONTRACT_CLIMBER
  ADD CONSTRAINT ID_CICERONE FOREIGN KEY (ID_CICERONE) REFERENCES PUBLIC.CICERONE(ID_CIC-
ERONE);
```

---

```
ALTER TABLE ONLY PUBLIC.INSURANCE_CONTRACT
  ADD CONSTRAINT ID_CLIMBER FOREIGN KEY (ID_CLIMBER) REFERENCES PUB-
LIC.CLIMBER(ID_CLIMBER);
```

---

```
ALTER TABLE ONLY PUBLIC.CONTRACT_CLIMBER
  ADD CONSTRAINT ID_CLIMBER FOREIGN KEY (ID_CLIMBER) REFERENCES PUB-
LIC.CLIMBER(ID_CLIMBER);
```

---

```
ALTER TABLE ONLY PUBLIC.CLIMBER
  ADD CONSTRAINT ID_CLUB FOREIGN KEY (ID_CLUB) REFERENCES PUBLIC.CLUB("ID_CLUB");
```

---

```
ALTER TABLE ONLY PUBLIC.CONTRACT_CLIMBER
  ADD CONSTRAINT ID_GROUP FOREIGN KEY (ID_GROUP) REFERENCES PUBLIC."GROUP"(ID_GROUP);
```

---

```
ALTER TABLE ONLY PUBLIC.INSURANCE_CASE
```

```
ADD CONSTRAINT ID_INSURANCE_CONTRACT FOREIGN KEY (ID_INSURANCE_CONTRACT) REFER-  
ENCES PUBLIC.INSURANCE_CONTRACT(ID_INSURANCE_CONTRACT);
```

---

```
ALTER TABLE ONLY PUBLIC.INSURANCE_CASE  
ADD CONSTRAINT ID_SITUATION FOREIGN KEY (ID_SITUATION) REFERENCES PUBLIC.EMERGANCY_SIT-  
UATION("ID_SITUATION");
```

---

```
ALTER TABLE ONLY PUBLIC.PEAK  
ADD CONSTRAINT PEAK_ID_MOUNTAIN_FKEY FOREIGN KEY (ID_MOUNTAIN) REFERENCES PUB-  
LIC.MOUNTAIN(ID_MOUNTAIN);
```

---

```
ALTER TABLE ONLY PUBLIC.ROUTE  
ADD CONSTRAINT "ROUTE_ID_PEAK_FKEY" FOREIGN KEY ("ID_PEAK") REFERENCES PUB-  
LIC.PEAK(ID_PEAK);
```

---

## ЗАПОЛНЕНИЕ ТАБЛИЦ

COPY PUBLIC.CICERONE (ID\_CICERONE, NAME\_CICERONE, WORK\_EXPERIENCE, CONTACT\_CICERONE, PORTFOLIO) FROM STDIN;

1	{A}	00:00:05	{A}	КАКОЙ-ТО ТЕКСТ
2	{B}	00:00:06	{B}	КАКОЙ-ТО ТЕКСТ
3	{C}	00:00:04	{C}	КАКОЙ-ТО ТЕКСТ
4	{D}	00:00:06	{D}	КАКОЙ-ТО ТЕКСТ
5	{E}	00:00:07	{E}	КАКОЙ-ТО ТЕКСТ

\.

COPY PUBLIC.CLIMBER (ID\_CLIMBER, NAME\_CLIMBER, ID\_CLUB, CONTACT\_CLIMBER, ADDRESS\_CLIMED) FROM STDIN;

1	{M}	1	{M}	{P}
2	{M}	2	{M}	{A}
3	{A}	3	{A}	{S}
4	{A}	4	{A}	{O}
5	{J}	5	{A}	{B}

\.

COPY PUBLIC.CLIMBING (ID\_CLIMBING, ID\_GROUP, START\_PLAN\_CLIMBING, START\_FACT\_CLIMBING, END\_PLAN\_CLIMBING, END\_FACT\_CLIMBING, DURATION\_CLIMBING, BRIEF\_INFORMATION\_CLIMBING, ID\_ROUTE) FROM STDIN;

1	1	2020-02-02	2020-02-02	2020-02-05	2020-02-06	00:00:03	КАКОЙ-ТО ТЕКСТ	1
---	---	------------	------------	------------	------------	----------	----------------	---

2	2	2020-03-04	2020-03-04	2020-03-11	2020-03-11	00:00:07	КАКОЙ-ТО ТЕКСТ	2
3	3	2020-01-03	2020-01-05	2020-01-15	2020-01-14	00:00:12	КАКОЙ-ТО ТЕКСТ	3
4	4	2020-01-05	2020-01-03	2020-03-21	2020-03-21	00:01:16	КАКОЙ-ТО ТЕКСТ	4
5	5	2019-03-08	2019-03-08	2019-04-05	2019-04-10	00:00:28	КАКОЙ-ТО ТЕКСТ	5

\.

COPY PUBLIC.CLUB ("ID\_CLUB", CITY\_CLUB, CONTACT\_CLUB\_NAME, COUNTRY\_CLUB, EMAIL\_CLUB, NAME\_CLUB, PHONE\_NUMBER\_CLUB) FROM STDIN;

1	{S}	{I}	{R}	{T}	{T}	5353535
2	{S}	{M}	{R}	{M}	{M}	5362182
3	{M}	{J}	{R}	{G}	{G}	4525314
4	{M}	{M}	{R}	{P}	{T}	5462134
5	{I}	{N}	{R}	{I}	{H}	4589561

\.

COPY PUBLIC.CONTRACT\_CICERONE (ID\_AGENCY, ID\_CICERONE, TERMS\_CONTRACT\_CICERONE, ID\_CONTRACT\_CICERONE) FROM STDIN;

1	1	УСЛОВИЯ 1
2	2	УСЛОВИЯ 2
3	3	УСЛОВИЯ 3
4	4	УСЛОВИЯ 4
5	5	УСЛОВИЯ 5

\.

COPY PUBLIC.CONTRACT\_CLIMBER (ID\_CLIMBER, ID\_CICERONE, ID\_GROUP, ID\_AGENCY, TERMS\_CONTRACT\_CLIMBER, ID\_CONTRACT\_CLIMBER) FROM STDIN;

1	1	1	1	УСЛОВИЯ 1
2	2	2	2	УСЛОВИЯ 2
3	3	3	3	УСЛОВИЯ 3
4	4	4	4	УСЛОВИЯ 4
5	5	5	5	УСЛОВИЯ 5

\.

COPY PUBLIC.EMERGANCY\_SITUATION ("ID\_SITUATION", TYPE\_SITUATION, ID\_CLIMBER, ID\_CLIMBING, TIME\_SITUATION) FROM STDIN;

1	{A}	1	1	15:15:00+14:59
2	{B}	2	2	15:00:00+03
3	{C}	3	3	10:31:00+03
4	{D}	4	4	08:03:00+04
5	{E}	5	5	03:17:00+02

\.

COPY PUBLIC."GROUP" (ID\_GROUP, QUANTITY\_MEMBER) FROM STDIN;

1	15
2	12
3	13
4	15
5	16

\.



```
COPY PUBLIC.INSURANCE_CASE (ID_INSURANCE_CASE, ID_SITUATION, SUM_REIMBURSEMENT,
DATE_REIMBURSEMENT, ID_INSURANCE_CONTRACT) FROM STDIN;
```

```
1 1 55000 2020-05-30 1
2 2 31500 2020-06-15 2
3 3 90000 2020-09-01 3
4 4 38000 2020-06-01 4
5 5 70000 2019-05-01 5
\.
```

```
COPY PUBLIC.INSURANCE_CONTRACT (ID_AGENCY, ID_CLIMBER, INSURANCE_CONTRACT_TERMS, ID_IN-
SURANCE_CONTRACT) FROM STDIN;
```

```
1 1 УСЛОВИЯ 1
2 2 УСЛОВИЯ\N 2
3 3 УСЛОВИЯ 3
4 4 УСЛОВИЯ 4
5 5 УСЛОВИЯ 5
\.
```

```
COPY PUBLIC.MOUNTAIN (COUNTRY_MOUNTAIN, AREA_MOUNTAIN, ID_MOUNTAIN, NAME_MOUNTAIN)
FROM STDIN;
```

```
{R} {C} 2 {
{F} {S} 3 {
{U} {W} 4 {
{E} {N} 5 {
{R} {S} 1 {
```

\\.

COPY PUBLIC.PEAK (NAME\_PEAK, COUNTRY\_PEAK, AREA\_PEAK, HIGHT\_PEAK, ID\_MOUNTAIN, ID\_PEAK)  
FROM STDIN;

{K}	{R}	{1}	5134 1	1
{R}	{R}	{1}	1234 2	2
{R}	{F}	{1}	1567 3	3
{R}	{U}	{1}	1894 4	4
{R}	{E}	{1}	5423 5	5

\\.

COPY PUBLIC.ROUTE (ID\_ROUTE, DESCRIPTION\_ROUTE, NAME\_ROUTE, "ID\_PEAK") FROM STDIN;

1	КАКОЙ-ТО ТЕКСТ	{A}	1
2	КАКОЙ-ТО ТЕКСТ	{B}	2
3	КАКОЙ-ТО ТЕКСТ	{C}	3
4	КАКОЙ-ТО ТЕКСТ	{D}	4
5	КАКОЙ-ТО ТЕКСТ	{E}	5

\\.

COPY PUBLIC.TRAVEL\_AGENCY (ID\_AGENCY, NUMBER\_OF\_CICERONE, NAME\_AGENCY) FROM STDIN;

1	1	{A}
2	1	{B}
3	1	{C}
4	1	{D}
5	1	{E}

\.

## ВЫВОД

В практической работе №4 были получены практические навыки создания таблиц базы данных PostgreSQL 10 (11), заполнения их рабочими данными, резервного копирования и восстановления БД.