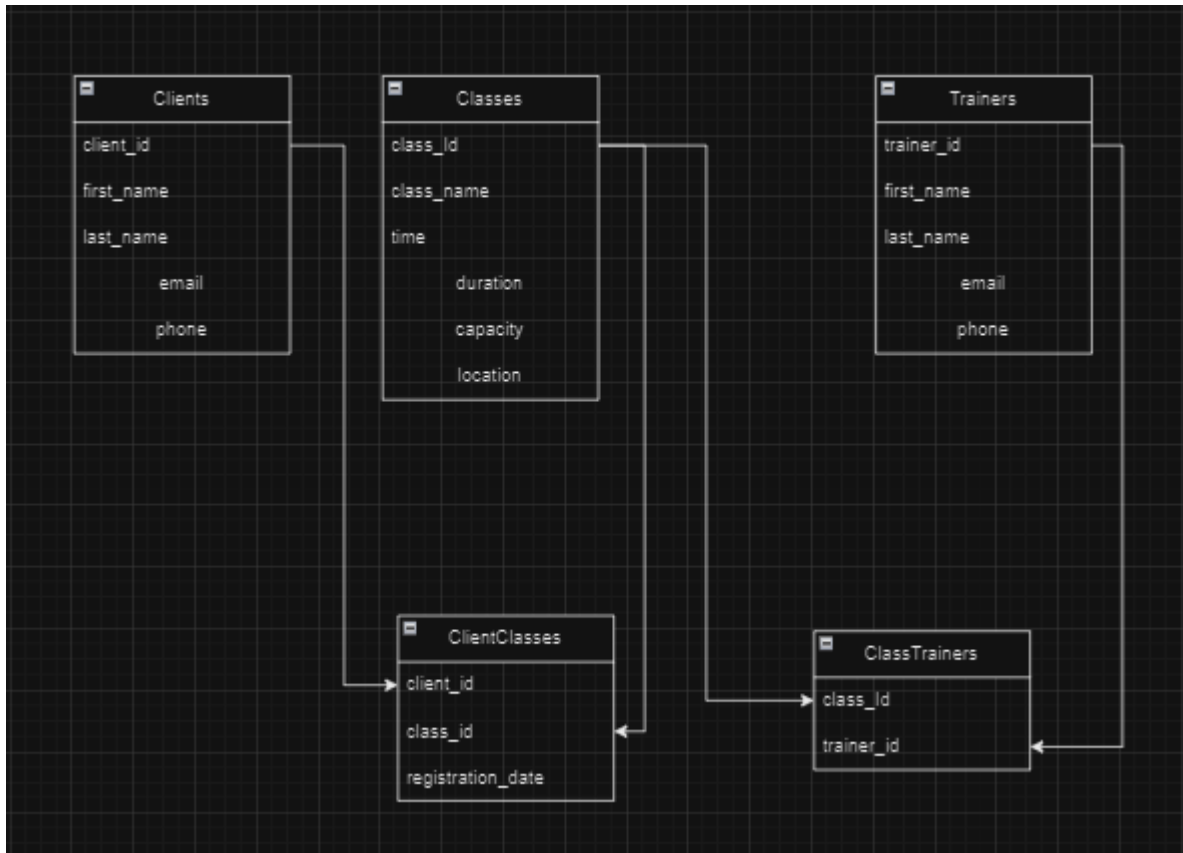
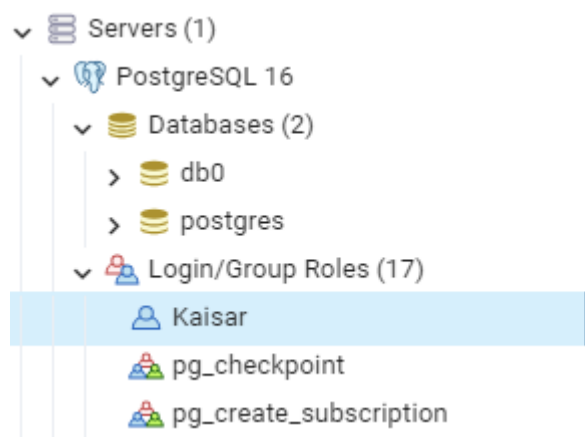


Лабораторная работа №6



Создание новой БД и пользователя через pgAdmin4



SQL-запрос на создание таблиц в соответствие со схемой

```
Query Query History
1 CREATE TABLE if not exists clients(id SERIAL
2 PRIMARY KEY, first_name VARCHAR(100) NOT
3 NULL,last_name VARCHAR(100) NOT
4 NULL, email VARCHAR(100), phone INT);
5
6 CREATE TABLE if not exists classes(id SERIAL
7 PRIMARY KEY, class_name VARCHAR(100) NOT
8 NULL,time INT, duration TIMESTAMP, capacity TIMESTAMP, location VARCHAR(100));
9
10 CREATE TABLE if not exists trainers(id SERIAL
11 PRIMARY KEY, first_name VARCHAR(100) NOT NULL, last_name VARCHAR(100),
12                                     email VARCHAR(100),phone INT);
13 CREATE TABLE if not exists clientclasses (client_id SERIAL PRIMARY KEY REFERENCES clients(id),
14                                             class_id SERIAL,
15                                             registration_date DATE);
16 CREATE TABLE if not exists classtrainers (class_id SERIAL PRIMARY KEY REFERENCES classes(id),
17                                             trainer_id SERIAL);

Data Output Messages Notifications
CREATE TABLE
Query returned successfully in 104 msec.
```

```
CREATE TABLE if not exists clients(id SERIAL
PRIMARY KEY, first_name VARCHAR(100) NOT
NULL,last_name VARCHAR(100) NOT
NULL, email VARCHAR(100), phone INT);
```

```
CREATE TABLE if not exists classes(id SERIAL
PRIMARY KEY, class_name VARCHAR(100) NOT
NULL,time INT, duration TIMESTAMP, capacity TIMESTAMP, location VARCHAR(100));
```

```
CREATE TABLE if not exists trainers(id SERIAL
PRIMARY KEY, first_name VARCHAR(100) NOT NULL, last_name VARCHAR(100),
                                     email VARCHAR(100),phone INT);
```

```
CREATE TABLE if not exists clientclasses (client_id SERIAL PRIMARY KEY REFERENCES clients(id),
                                             class_id SERIAL,
                                             registration_date
DATE);
```

```
CREATE TABLE if not exists classtrainers (class_id SERIAL PRIMARY KEY REFERENCES classes(id),
```

trainer_id SERIAL);

Добавление данных в таблицу clients

Foreign Tables

Functions

Materialized Views

Operators

Procedures

Sequences

Tables (5)

Trigger Functions

Types

Views

Subscriptions

postgres

pgin/Group Roles (17)

kaisar

pg_checkpoint

pg_create_subscription

1

2

SELECT * FROM public.clients

ORDER BY id ASC

Data Output

Messages

Notifications

	id	first_name	last_name	email	phone
	[PK] integer	character varying (100)	character varying (100)	character varying (100)	integer
1	1	Кайсар	Ерденбек	erdenbekkaisar@mail.ru	77777777

INSERT INTO clients(first_name, last_name, email, phone) VALUES ('Кайсар','Ерденбек','erdenbekkaisar@mail.ru','+77777777');

Добавление данных в таблицу Trainers

Foreign Tables

Functions

Materialized Views

Operators

Procedures

Sequences

Tables (5)

Trigger Functions

Types

Views

Subscriptions

postgres

pgin/Group Roles (17)

Kaisar

pg_checkpoint

pg_create_subscription

1

2

SELECT * FROM public.trainers

ORDER BY id ASC

Data Output

Messages

Notifications

	id	first_name	last_name	email	phone
	[PK] integer	character varying (100)	character varying (100)	character varying (100)	integer
1	1	Кайсар	Ерденбек	erdenbekkaisar@mail.ru	77777777

```
INSERT INTO trainers(first_name, last_name, email, phone) VALUES  
('Кайсар','Ерденбек','erdenbekkaisar@mail.ru','+77777777');
```