

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

**Кафедра прикладной информатики**

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1  
ДИСЦИПЛИНЫ «Введение в NoSQL СУБД»

**Выполнил:**  
Студент группы ПИН-б-о-22-1  
Солдаткин Александр Алексеевич  
Проверил: Щеголев А. А.

Ставрополь, 2025

## Цель работы

Ознакомиться с функциями PostgreSQL.

**Задание:** Получить представление о метаданных, написать запрос, который найдет название страны, где проходит мероприятие 'Fight club', изменить таблицу venues, добавив в нее колонку типа Boolean, создать правило, которое перехватывает команды delete, заменить выборку из сводной таблицы функцией generate\_series(a, b), построить сводную таблицу, которая отображает каждый день в течение месяца, каждый день должен содержать количество событий.

## Выполнение работы

Основы работы с PostgreSQL

Подключение к серверу осуществляется через порт 5432. Чтобы подключиться, используйте команду:

Вывод команд в PostgreSQL включает имя базы данных и символы, указывающие на уровень доступа (хэш для администратора и доллар для обычного пользователя).

CRUD операции

CRUD — это основные операции с данными:

- Создать (Create)
- Прочитать (Read)
- Обновить (Update)
- Удалить (Delete)

Эти операции позволяют управлять записями в базе данных.

Например, команда для создания таблицы стран:

```
CREATE TABLE countries (  
    country_code CHAR(2) PRIMARY KEY,  
    country_name VARCHAR(100) UNIQUE  
);
```

The screenshot shows a PostgreSQL query editor interface. The top toolbar includes icons for file operations, a filter icon, a dropdown menu with 'No limit', and execution controls. The main editor area has tabs for 'Query', 'Query History', and 'Scratch Pad'. The 'Query' tab is active, displaying a SQL statement to create a table named 'countries'. The statement is as follows:

```
1 CREATE TABLE countries (  
2     country_code CHAR(2) PRIMARY KEY,  
3     country_name VARCHAR(100) UNIQUE  
4 );  
5
```

### Примеры операций

1. Добавление данных:

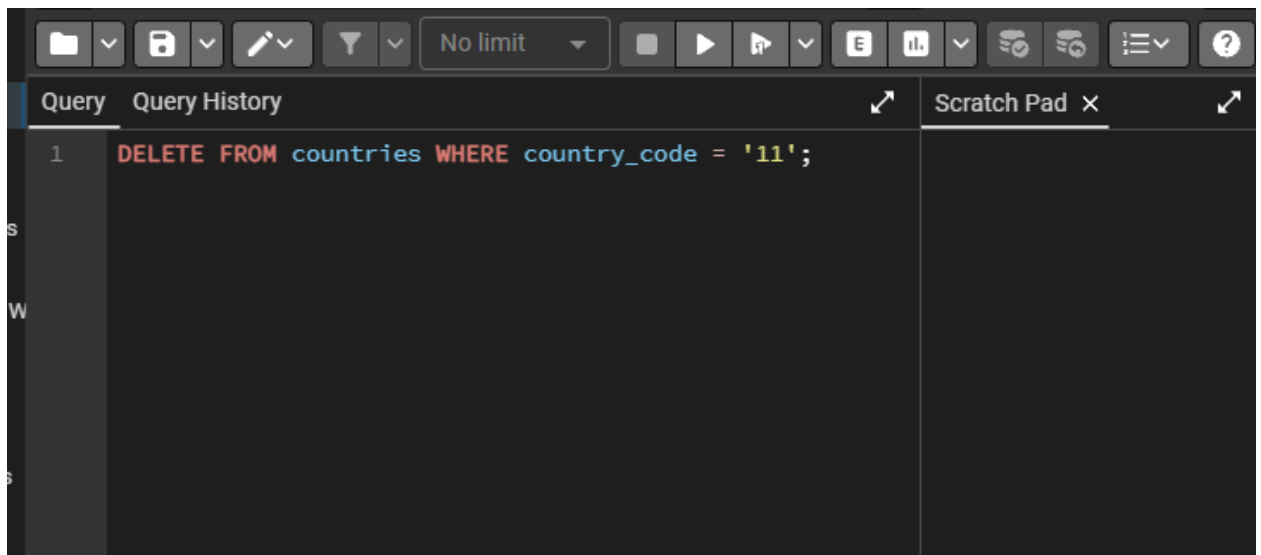
```
INSERT INTO countries (country_code, country_name)  
VALUES ('US', 'United States'), ('MX', 'Mexico');
```

The screenshot shows the same PostgreSQL query editor interface. The 'Query' tab is active, displaying an SQL statement to insert data into the 'countries' table. The statement is as follows:

```
1 INSERT INTO countries (country_code, country_name)  
2 VALUES ('US', 'United States'), ('MX', 'Mexico');  
3
```

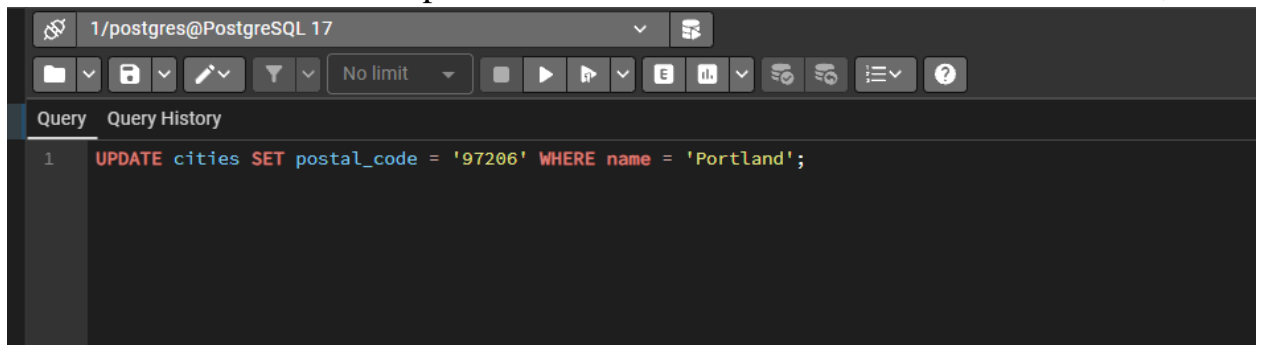
2. Удаление данных:

```
DELETE FROM countries WHERE country_code = '11';
```



3. Обновление данных:

`UPDATE cities SET postal_code = '97206' WHERE name = 'Portland';`



4. Чтение данных:

`SELECT * FROM countries;`

The screenshot shows a database query editor with a dark theme. At the top, there are tabs for 'Query' and 'Query History', and a 'Scratch Pad' button. The main query area contains the following SQL statement:

```
1 SELECT * FROM countries;
```

Below the query area, there are tabs for 'Data Output', 'Messages', and 'Notifications'. The 'Data Output' tab is active, showing a table of results. The table has two columns: 'country\_code' (PK) character (2) and 'country\_name' character varying (100). The results are as follows:

	country_code [PK] character (2)	country_name character varying (100)
1	US	United States
2	MX	Mexico

Соединение таблиц

```
SELECT cities.*, countries.country_name
```

```
FROM cities
```

```
INNER JOIN countries ON cities.country_code = countries.country_code;
```

The screenshot shows a database query editor with a dark theme. The main query area contains the following SQL statement:

```
1 SELECT cities.*, countries.country_name
2 FROM cities
3 INNER JOIN countries ON cities.country_code = countries.country_code;
4
```

Вывод: в этой главе мы рассмотрели основные концепции реляционных баз данных и основные операции CRUD.

