

Министерство цифрового развития, связи и массовых коммуникаций РФ  
Уральский технический институт связи и информатики (филиал) ФГБОУ ВО  
"Сибирский государственный университет телекоммуникаций и информатики" в  
г. Екатеринбурге  
(УрТИСИ СибГУТИ)

**Казанцев М.Ю.**

# **ОСНОВЫ ПРОЕКТИРОВАНИЯ БАЗ ДАННЫХ**

Методические указания по выполнению практических работ  
студентов очной и заочной формы обучения  
«Информатика и вычислительная техника»  
Методические указания для выполнения практической работы №4

**Екатеринбург 2024**

## **Практическая работа №4**

### **«Создание и управление базами данных в pgAdmin 4»**

#### **Цель практической работы:**

Цель этой практической работы - научиться создавать и управлять базами данных, таблицами и схемами с использованием графического интерфейса pgAdmin 4.

#### **Теоретический материал:**

Базы данных представляют собой структурированные наборы данных, предназначенные для хранения, управления и обработки информации. Они играют критически важную роль в современных приложениях, обеспечивая надежное и эффективное хранение данных. В основе большинства приложений лежат базы данных, которые позволяют организовать и быстро извлекать необходимые данные.

PostgreSQL — это мощная объектно-реляционная система управления базами данных с открытым исходным кодом, которая поддерживает расширяемость и соответствие стандартам SQL. Благодаря этим качествам PostgreSQL стал популярным выбором среди разработчиков и администраторов баз данных. Эта система позволяет создавать сложные запросы и управлять большими объемами данных с высокой производительностью и надежностью.

pgAdmin 4 выступает в качестве графического интерфейса для управления базами данных PostgreSQL. Он предоставляет удобные инструменты для администрирования и разработки баз данных, включая создание и изменение объектов, выполнение SQL-запросов и анализ данных. С pgAdmin 4 пользователи могут легко выполнять задачи, связанные с управлением базами данных, в интуитивно понятной среде.

Основные объекты базы данных в PostgreSQL включают базы данных, схемы, таблицы, столбцы и строки. База данных является контейнером для хранения данных, в то время как схема организует объекты внутри базы данных. Таблица, состоящая из строк и столбцов, служит основным объектом для хранения данных. Столбцы определяют типы данных и ограничения для каждого поля в таблице, а строки представляют собой записи данных.

В PostgreSQL реализованы различные операции для работы с данными. Создание базы данных осуществляется с помощью команды `CREATE DATABASE`, создающей новый контейнер для данных. Команда `CREATE SCHEMA` позволяет создать новую схему для организации объектов внутри базы данных, а `CREATE TABLE` используется для создания таблиц с заданными столбцами и типами

данных. Для добавления новых строк в таблицу применяется команда INSERT INTO, а для извлечения данных на основе условий используется команда SELECT.

Типы данных в PostgreSQL разнообразны и включают такие типы, как serial для автоматически увеличивающихся значений, часто используемых в качестве первичных ключей, varchar(n) для строковых значений с ограниченной длиной, integer для целых чисел и date для хранения дат. Понимание этих типов данных и их применение в PostgreSQL важно для эффективного управления данными. Практическая работа с использованием pgAdmin 4 позволит закрепить теоретические знания и научиться создавать и модифицировать базы данных, что является важным шагом на пути к эффективному управлению данными.

### **Задания:**

Ваше задание начинается с подключения к серверу PostgreSQL через pgAdmin 4, используя ваши учетные данные. Это позволит вам ознакомиться с процессом установки соединения с сервером и подготовит к дальнейшим действиям. Первым шагом является успешное подключение к локальному серверу, что необходимо для создания собственной базы данных.

Для начала работы вам необходимо подключиться к серверу PostgreSQL через pgAdmin 4. Откройте приложение pgAdmin 4 на вашем компьютере. После загрузки интерфейса вы увидите основное окно с панелью управления проектами, где будет отображаться список серверов.

Чтобы подключиться к серверу, найдите раздел "Servers" в левой части экрана. Нажмите правой кнопкой мыши на "Servers" и выберите "Create" и затем "Server". Появится окно, в котором нужно указать имя соединения, такое как "MyLocalServer", чтобы вы могли легко его идентифицировать. Перейдите на вкладку "Connection" для ввода параметров подключения.

Введите localhost в поле "Host name/address", если подключаетесь к серверу на вашем компьютере. Если же нет, уточните данные для подключения у старосты группы либо у преподавателя. Оставьте значение порта 5432, если оно не изменялось при установке PostgreSQL. Введите postgres в поле "Maintenance database" для использования стандартной базы данных обслуживания. Укажите ваш PostgreSQL логин в поле "Username", который обычно равен postgres, и введите пароль, заданный при установке PostgreSQL. Логин и пароль если вы работаете в аудитории также уточните у преподавателя.

После ввода всех необходимых данных нажмите "Save", чтобы сохранить настройки соединения. Ваш сервер появится в списке "Servers", и вы можете дважды кликнуть по нему, чтобы подключиться. Убедитесь, что все параметры введены правильно, чтобы избежать проблем с подключением. После успешного

подключения вы сможете увидеть список баз данных на сервере и приступить к выполнению задания.

После успешного подключения к серверу PostgreSQL с помощью pgAdmin 4 вам необходимо создать новую базу данных. Это важный шаг, поскольку каждая база данных служит контейнером для хранения и управления данными. В интерфейсе pgAdmin 4 найдите в левой части экрана список серверов и разверните его, чтобы увидеть подраздел "Databases". Это место, где будут отображаться все существующие базы данных на вашем сервере.

Чтобы создать новую базу данных, кликните правой кнопкой мыши на "Databases" и выберите "Create" и затем "Database". В появившемся окне вам будет предложено ввести имя для новой базы данных. Рекомендуется использовать уникальное имя, которое будет включать ваши инициалы и номер учебной группы, например, "383\_Kazantsev". Это поможет вам легко идентифицировать вашу базу данных среди других.

В этом же окне выберите владельца базы данных из выпадающего списка. По умолчанию это будет пользователь "postgres", если вы не создавали других пользователей. Вы также можете настроить дополнительные параметры, такие как кодировка и шаблон, но для базовой настройки эти параметры можно оставить по умолчанию.

После внесения всех необходимых данных нажмите "Save", чтобы завершить создание базы данных. Новая база данных появится в списке под "Databases". Вы можете развернуть ее, чтобы увидеть структуру и начать добавлять схемы и таблицы.

После создания базы данных следующим шагом будет создание таблицы, которая будет использоваться для хранения данных. В pgAdmin 4 разверните вашу базу данных в списке "Databases", затем найдите и разверните раздел "Schemas", после чего выберите "public" или другую схему, которую вы создали ранее. Внутри схемы найдите раздел "Tables".

Чтобы создать новую таблицу, щелкните правой кнопкой мыши на разделе "Tables" и выберите "Create" и затем "Table". В открывшемся окне введите имя таблицы, например, students. Это имя должно отражать содержимое таблицы и быть понятным для других пользователей базы данных.

Перейдите на вкладку "Columns", где вы сможете добавить столбцы для вашей таблицы. Начните с добавления столбца id, который будет использоваться как первичный ключ. Установите для него тип данных serial, чтобы значения автоматически увеличивались с каждой новой записью, обеспечивая уникальность каждого идентификатора.

Добавьте другие столбцы, такие как `name`, `age`, и `enrollment_date`. Для `name` выберите тип данных `varchar` и укажите максимальную длину, например, 50 символов, чтобы обеспечить достаточно места для имени студента. `age` должен быть целочисленным полем, поэтому выберите тип `integer`. Для `enrollment_date` используйте тип `date`, чтобы правильно хранить информацию о дате зачисления.

После добавления всех необходимых столбцов установите необходимые ограничения, такие как `NOT NULL`, если данные в определенных столбцах должны быть обязательными. Убедитесь, что все настройки проверены и корректны, затем нажмите "Save", чтобы сохранить таблицу.

После создания таблицы следующим шагом является вставка данных, что позволяет начать наполнять таблицу информацией. В pgAdmin 4 вы можете использовать SQL Query Tool для выполнения SQL-запросов на вставку данных в таблицу. Этот инструмент доступен через меню "Tools" в верхней части интерфейса, выберите "Query Tool", чтобы открыть окно для ввода SQL-кода.

Для начала вставки данных в таблицу `students` убедитесь, что выбрана нужная база данных. Введите SQL-запрос, используя команду `INSERT INTO`, чтобы добавить новые строки данных. Структура команды будет выглядеть следующим образом:

## SQL

```
INSERT INTO students (name, age, enrollment_date) VALUES
('Алиса', 20, '2023-09-01'),
('Боб', 22, '2023-09-01'),
('Чарли', 19, '2023-09-01');
```

В этом запросе `students` — это имя таблицы, а в скобках перечислены столбцы, в которые вы будете вставлять данные. После ключевого слова `VALUES` следуют скобки с данными, которые вы хотите добавить. Убедитесь, что порядок данных соответствует порядку столбцов, указанному в запросе.

Когда ваш SQL-запрос готов, нажмите на кнопку "Execute" (обычно это значок молнии) в верхней части окна Query Tool. Это выполнит ваш запрос и вставит данные в таблицу. Если все выполнено корректно, вы увидите сообщение об успешном завершении операции в нижней части окна.

После выполнения запроса вы можете проверить вставленные данные, выбрав таблицу `students` в объектном браузере, щелкнув правой кнопкой мыши и выбрав "View/Edit Data" > "All Rows". Это действие откроет таблицу в режиме редактирования, где вы сможете увидеть и подтвердить, что данные были корректно добавлены.

В завершение работы подготовьте отчет, включающий скриншоты базы данных, схемы и таблицы, а также выполненных SQL-запросов. В отчете опишите все выполненные действия и продемонстрируйте результат.

## **Контрольные вопросы**

### **1. Какие шаги необходимы для подключения к серверу PostgreSQL с использованием pgAdmin 4?**

Открыть pgAdmin, в левом меню ПКМ на Servers - Register - Server. Укажите имя, хост (localhost), порт (5432), логин/пароль (обычно postgres). Нажмите Save.2.

### **2. Как правильно создать новую базу данных в pgAdmin 4 и какие параметры необходимо учитывать при ее создании?**

Имя базы данных, кодировка, доступ.

### **3. В чем заключается назначение схемы в базе данных и как ее создать в pgAdmin 4?**

Организация объектов в логические группы для ясности и простоты работы. Схемы помогают сегментировать объекты базы данных в отдельные пространства имён

### **4. Какие типы данных следует использовать для столбцов id, name, age, и enrollment\_date при создании таблицы students?**

Serial, varchar, integer, data.

### **5. Какова структура команды SQL для вставки данных в таблицу и как убедиться, что данные были успешно добавлены?**

INSERT INTO имя\_таблицы (столбец1, столбец2, ...)

VALUES (значение1, значение2, ...);

Проверить добавление данных таблицу можно при помощи команды Select.

### **6. Какие действия необходимо предпринять для просмотра и проверки данных, вставленных в таблицу students?**

ПКМ по таблице и выбрать View edit data – all rows.