

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 (УрТИСИСибГУТИ)



Уральский технический
институт связи
и информатики

Кафедра информационных
систем и
технологий



ОТЧЕТ

По дисциплине «Основы проектирования баз данных»
Практическая работа № 5

Выполнил:

Студент

Кононов С.Д.

Проверил:

Преподаватель

Казанцев М.Ю.

Екатеринбург

2025 г.

Тема:

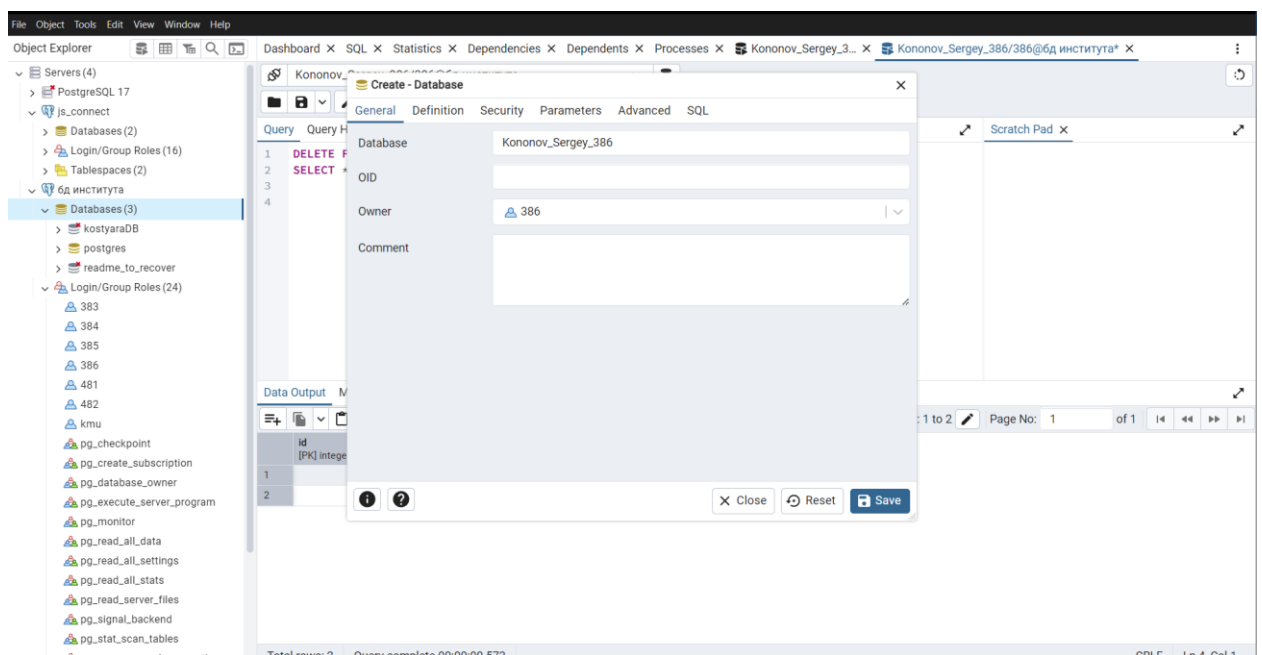
Запросы SQL: Изучение и выполнение основных SQL-запросов

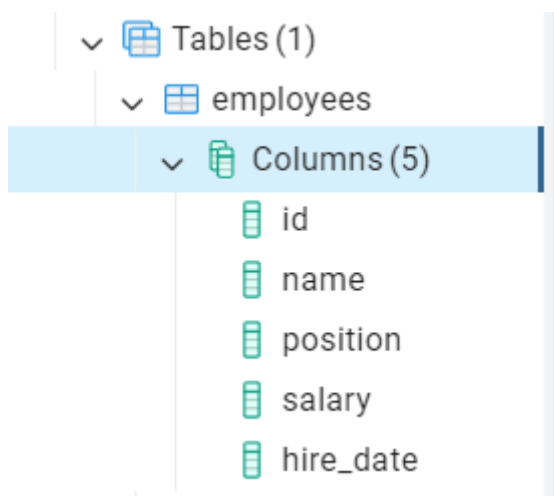
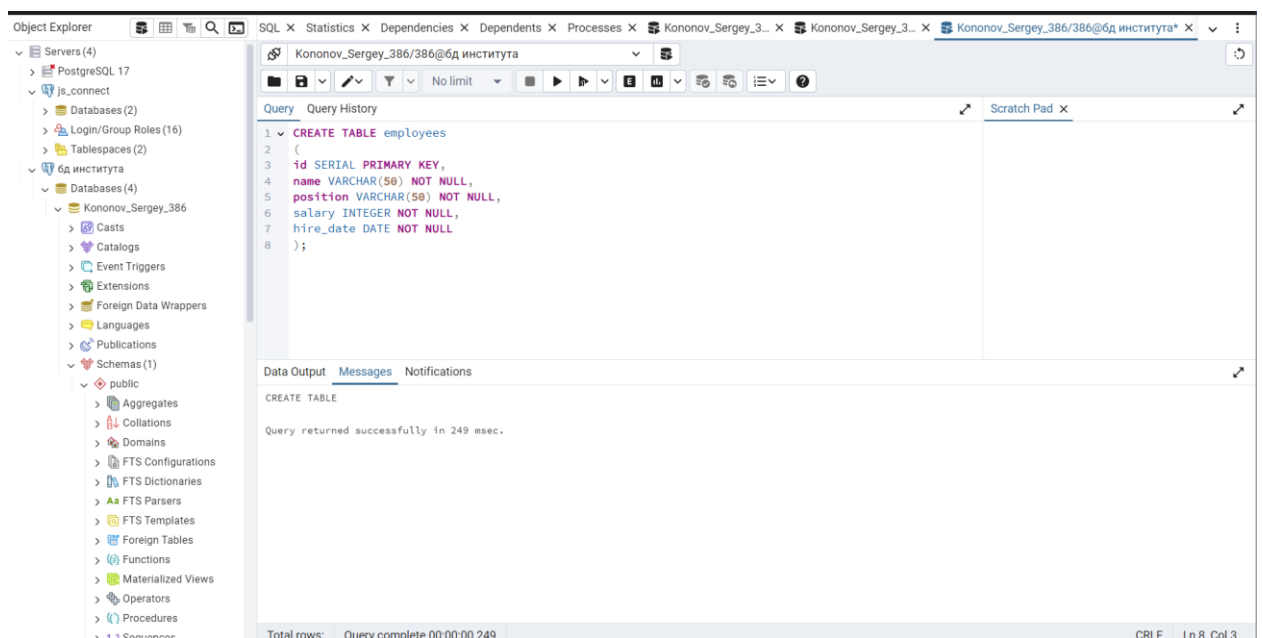
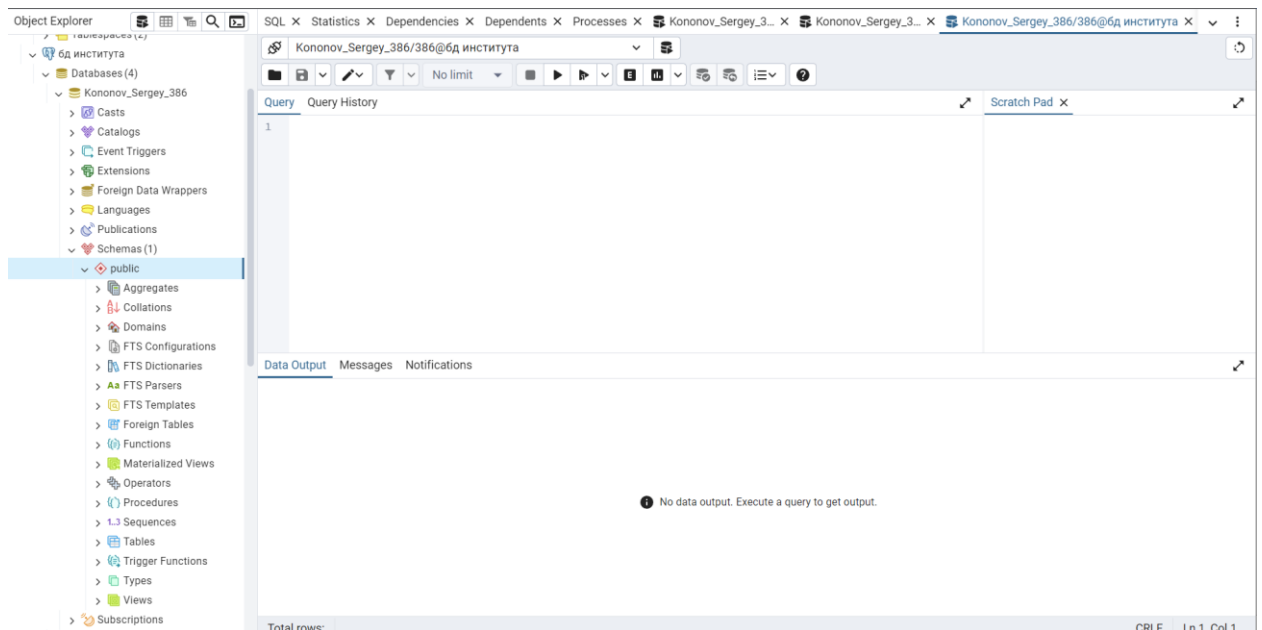
Цель практической работы:

Цель этой практической работы - научиться использовать основные SQL-запросы (SELECT, INSERT, UPDATE, DELETE) для работы с данными в базе данных PostgreSQL с использованием pgAdmin 4.

Задания:

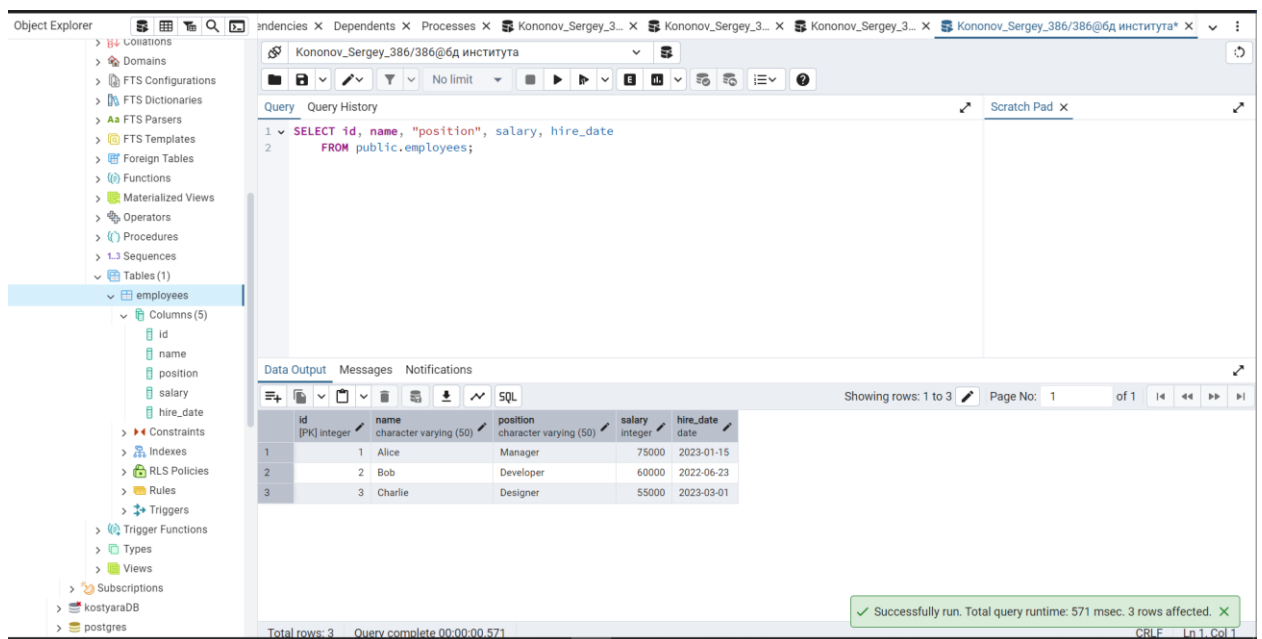
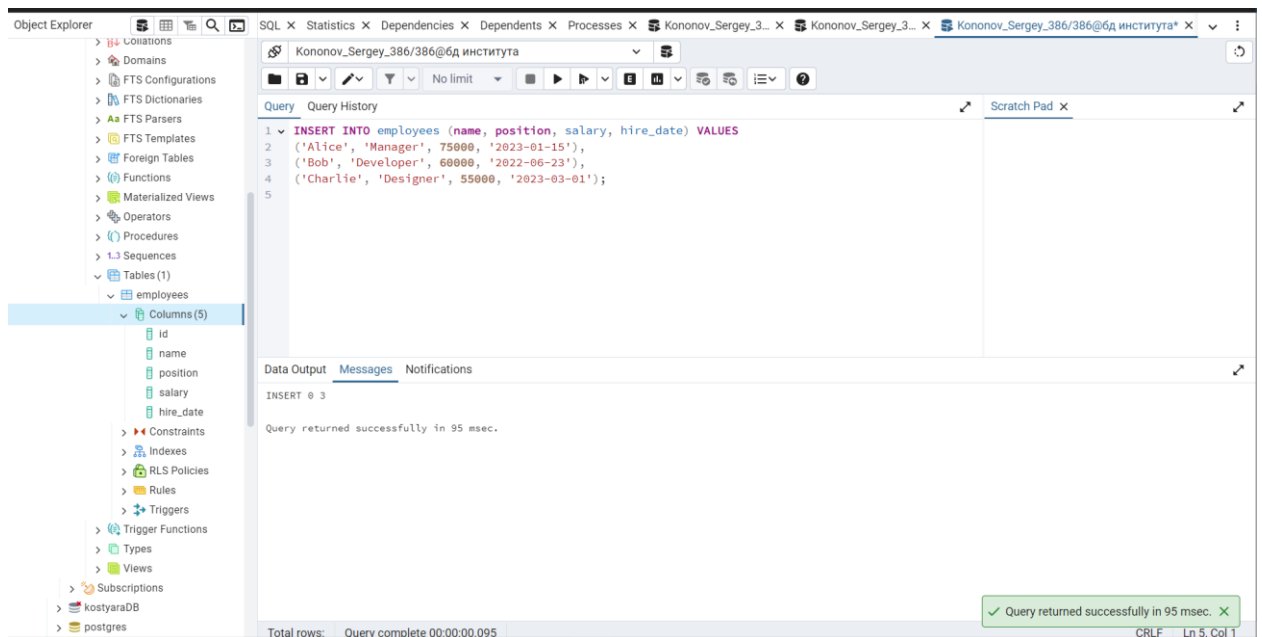
В этой практической работе вы начнете с создания новой базы данных, которая будет служить контейнером для ваших данных. Имя базы данных должно отражать вашу учебную группу и инициалы, чтобы обеспечить уникальность и облегчить идентификацию. После создания базы данных вам предстоит создать таблицу employees, которая будет содержать информацию о сотрудниках. Таблица должна включать столбцы id для уникального идентификатора, name для имени сотрудника, position для должности, salary для зарплаты и hire_date для даты найма. Это позволит вам структурировать данные и подготовить их для дальнейших операций.





Следующим шагом будет вставка данных в таблицу. С помощью SQL-запроса INSERT INTO вам нужно добавить несколько записей, например,

информацию о сотрудниках с именами Alice, Bob и Charlie, их должностями, зарплатами и датами найма. После выполнения запроса убедитесь, что данные были успешно добавлены в таблицу, это можно сделать, выполнив запрос на извлечение всех данных.



Далее вы будете извлекать данные из таблицы employees с помощью SQL-запросов SELECT. Начните с выполнения запроса, который извлечет все данные из таблицы. Затем выберите только те записи, где зарплата сотрудников превышает определенное значение. Также вам нужно будет посчитать общее количество сотрудников в таблице. Эти операции помогут вам освоить базовые техники выборки данных и агрегатных функций.

Object Explorer

- > Databases
- > Domains
- > FTS Configurations
- > FTS Dictionaries
- > FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > Sequences
- > Tables (1)
 - employees
 - Columns (5)
 - id
 - name
 - position
 - salary
 - hire_date
 - Constraints
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
 - Trigger Functions
 - Types
 - Views
 - Subscriptions
 - kostyaraDB
 - postgres

Query

```
1 SELECT * FROM employees;
```

Query History

Scratch Pad

Data Output Messages Notifications

Showing rows: 1 to 3 Page No: 1 of 1

id	name	position	salary	hire_date
1	Alice	Manager	75000	2023-01-15
2	Bob	Developer	60000	2022-06-23
3	Charlie	Designer	55000	2023-03-01

Total rows: 3 Query complete 00:00:00.596

Successfully run. Total query runtime: 596 msec. 3 rows affected.

Object Explorer

- > Databases
- > Domains
- > FTS Configurations
- > FTS Dictionaries
- > FTS Parsers
- > FTS Templates
- > Foreign Tables
- > Functions
- > Materialized Views
- > Operators
- > Procedures
- > Sequences
- > Tables (1)
 - employees
 - Columns (5)
 - id
 - name
 - position
 - salary
 - hire_date
 - Constraints
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
 - Trigger Functions
 - Types
 - Views
 - Subscriptions
 - kostyaraDB
 - postgres

Query

```
1 SELECT name, position FROM employees WHERE salary > 60000;
```

Query History

Scratch Pad

Data Output Messages Notifications

Showing rows: 1 to 1 Page No: 1 of 1

name	position
Alice	Manager

Total rows: 1 Query complete 00:00:00.483

Successfully run. Total query runtime: 483 msec. 1 rows affected.

Object Explorer

- Domains
- FTS Configurations
- FTS Dictionaries
- FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- Operators
- Procedures
- Sequences
- Tables (1)
 - employees
 - Columns (5)
 - id
 - name
 - position
 - salary
 - hire_date
 - Constraints
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
 - Trigger Functions
 - Types
 - Views
 - Subscriptions
 - kostyaraDB
 - postgres

Query Query History

```
1 SELECT COUNT(*) FROM employees;
```

Data Output Messages Notifications

count	bigint
1	3

Showing rows: 1 to 1 Page No: 1 of 1

Successfully run. Total query runtime: 202 msec. 1 rows affected.

Total rows: 1 Query complete 00:00:00.202

После этого вы перейдете к обновлению данных. Используйте SQL-запрос UPDATE, чтобы изменить зарплату сотрудника с именем Bob. Измените зарплату на новое значение и убедитесь, что изменения применены, выполнив запрос на извлечение данных. Это упражнение продемонстрирует вам, как можно изменять существующие записи в таблице.

Object Explorer

- Domains
- FTS Configurations
- FTS Dictionaries
- FTS Parsers
- FTS Templates
- Foreign Tables
- Functions
- Materialized Views
- Operators
- Procedures
- Sequences
- Tables (1)
 - employees
 - Columns (5)
 - id
 - name
 - position
 - salary
 - hire_date
 - Constraints
 - Indexes
 - RLS Policies
 - Rules
 - Triggers
 - Trigger Functions
 - Types
 - Views
 - Subscriptions
 - kostyaraDB
 - postgres

Query Query History

```
1 UPDATE employees SET salary = 65000 WHERE name = 'Bob';
```

Data Output Messages Notifications

UPDATE 1

Query returned successfully in 125 msec.

Total rows: 1 Query complete 00:00:00.125

The screenshot shows the PostgreSQL IDE interface. On the left, the 'Object Explorer' pane displays the database structure, with the 'employees' table selected under 'Tables (1)'. The 'Query' pane contains the following SQL statement:

```
1 SELECT id, name, "position", salary, hire_date
2 FROM public.employees;
```

The 'Data Output' pane shows the results of the query, displaying 3 rows. The table has the following structure:

id	name	position	salary	hire_date
1	Alice	Manager	75000	2023-01-15
2	Bob	Developer	65000	2022-06-23
3	Charlie	Designer	55000	2023-03-01

The status bar at the bottom indicates 'Total rows: 3' and 'Query complete 00:00:00.617'.

Наконец, вам предстоит удалить данные из таблицы. Выполните SQL-запрос DELETE, чтобы удалить запись о сотруднике с именем Charlie. После удаления убедитесь, что запись была удалена, выполнив запрос на извлечение всех данных. Это действие покажет вам, как можно удалять ненужные или устаревшие данные из базы.

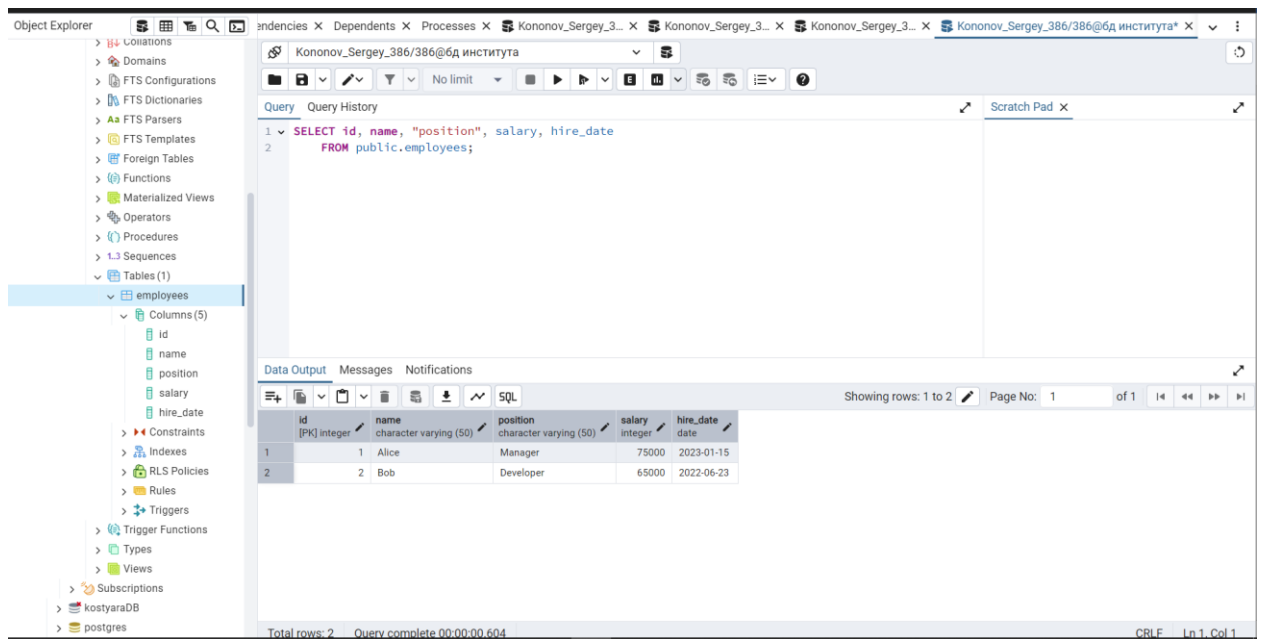
The screenshot shows the PostgreSQL IDE interface after executing a DELETE query. The 'Query' pane contains the following SQL statement:

```
1 DELETE FROM employees WHERE name = 'Charlie';
```

The 'Messages' pane shows the execution result:

```
DELETE 1
Query returned successfully in 143 msec.
```

A green status bar at the bottom right confirms: 'Query returned successfully in 143 msec.' The status bar at the bottom indicates 'Total rows: 0' and 'Query complete 00:00:00.143'.



Контрольные вопросы:

Какой синтаксис используется для извлечения данных из таблицы с помощью команды SELECT?

SELECT столбцы FROM таблица [WHERE условие] [ORDER BY сортировка] [LIMIT количество];

В чем заключается процесс вставки новых данных в таблицу с помощью команды INSERT INTO?

Процесс вставки новых данных в таблицу с помощью команды INSERT INTO заключается в добавлении одной или нескольких строк в указанную таблицу.

Как с помощью команды UPDATE изменить данные в существующей записи таблицы?

UPDATE table_name

SET column1 = value1, column2 = value2, ...

[WHERE условие]

[ORDER BY ...]

[LIMIT количество];

Какие условия необходимо задать при выполнении команды DELETE для удаления определенных строк из таблицы?

При выполнении команды DELETE для удаления определённых строк из таблицы необходимо задать условие WHERE, иначе удалятся все данные в таблице!