Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Базы данных

ОТЧЁТ

по лабораторной работе №4

«Реализация SQL-запросов на простую выборку данных»

на тему

«Сеть клубов карате»

Студент Р.С. Кочеров

Преподаватель Д.В. Куприянова

­­­­

Минск 2025СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_Toc192854753)

[1 SQL-ЗАПРОСЫ 4](#_Toc192854754)

[1.1 WHERE 4](#_Toc192854755)

[1.2 WHERE и ORDER BY 5](#_Toc192854756)

[1.3 LIKE 6](#_Toc192854757)

[1.4 LIKE с дополнительным условием 7](#_Toc192854758)

[1.5 INNER JOIN 8](#_Toc192854759)

[1.6 INNER JOIN (2) 10](#_Toc192854760)

[1.7 LEFT OUTER JOIN 11](#_Toc192854761)

[1.8 LEFT OUTER JOIN (2) 13](#_Toc192854762)

[1.9 ORDER BY 14](#_Toc192854763)

[1.10 ORDER BY (2) 16](#_Toc192854764)

[1.11 RIGHT OUTER JOIN 17](#_Toc192854765)

[1.12 FULL OUTER JOIN 18](#_Toc192854766)

[1.13 FULL OUTER JOIN (2) 20](#_Toc192854767)

[1.14 CROSS JOIN 22](#_Toc192854768)

[ЗАКЛЮЧЕНИЕ 24](#_Toc192854769)

# **ВВЕДЕНИЕ**

Для работы с данными используются системы управления базами данных (СУБД), которые позволяют хранить, обрабатывать и извлекать информацию эффективно и структурированно. Одним из основных инструментов взаимодействия с базами данных является язык структурированных запросов. SQL предоставляет широкие возможности для выборки, фильтрации, сортировки и агрегации данных, что делает его незаменимым инструментом для аналитиков, разработчиков и администраторов баз данных.

Целью данной лабораторной работы является освоение базовых принципов составления SQL-запросов на выборку данных с использованием таких операторов, как SELECT FROM, WHERE, ORDER BY, а также различных типов соединений таблиц (JOIN, INNER JOIN, OUTER LEFT JOIN, OUTER RIGHT JOIN, FULL OUTER JOIN, CROSS JOIN). В процессе выполнения работы будут рассмотрены примеры запросов, которые позволяют извлекать данные из одной или нескольких таблиц, фильтровать их по условиям, сортировать и объединять на основе связей между таблицами.

В рамках лабораторной работы используются заранее подготовленные таблицы, моделирующие работу сети клубов по карате. Эти таблицы содержат информацию о клубах, учениках, тренерах, мероприятиях, местах проведения мероприятий, тренировках, оплате тренировок. На основе этих данных будут составлены SQL-запросы, демонстрирующие различные способы выборки данных.

Выполнение данной работы позволит получить практические навыки работы с SQL, понять принципы построения запросов и научиться извлекать необходимую информацию из базы данных. Это важный этап в освоении технологий работы с данными, который будет полезен для дальнейшего изучения более сложных аспектов SQL и проектирования баз данных.

# **1 SQL-ЗАПРОСЫ**

## 1.1 WHERE

Выбрать всех учеников, которым больше 21 года (рисунки 1.1-1.2).

SQL-Команда: SELECT \* FROM student WHERE age > 21;

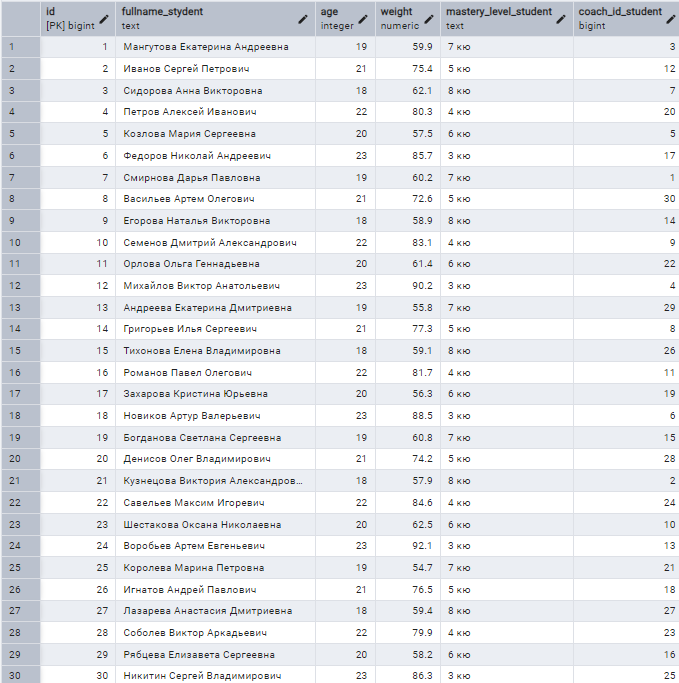


Рисунок 1.1 – Исходная таблица «student»



Рисунок 1.2 – Выполнение SQL запроса

## 1.2 WHERE и ORDER BY

Выбрать все клубы стиль которых кёкусинкай и отсортировать по алфавиту названия клуба (рисунки 1.3-1.4).

SQL-команда: SELECT \* FROM club WHERE karate\_style = 'кёкусинкай' ORDER BY name\_club ASC;

Изображение выглядит как текст, меню, снимок экрана, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.3 – Исходная таблица «club»

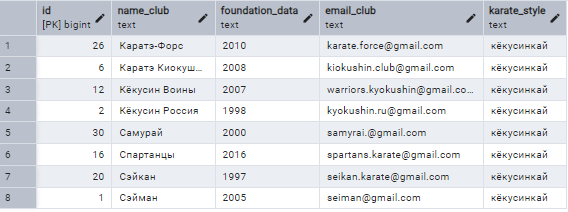


Рисунок 1.4 – Выполнение SQL запроса

## 1.3 LIKE

Найти тренеров, у которых фамилия начинается на «К» (рисунки 1.5-1.6).

SQL-команда: SELECT \* FROM coach WHERE name\_coach LIKE 'К%';



Рисунок 1.5 – Исходная таблица «coach»

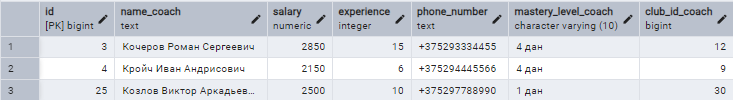


Рисунок 1.6 – Выполнение SQL запроса

## 1.4 LIKE с дополнительным условием

Найти тренеров, у которых имя начинается на «Л» c опытом работы больше 10 лет (рисунки 1.7-1.8).

SQL-команда: SELECT \* FROM coach WHERE name\_coach LIKE 'Л%' AND experience > 10;



Рисунок 1.7 – Исходная таблица «coach»

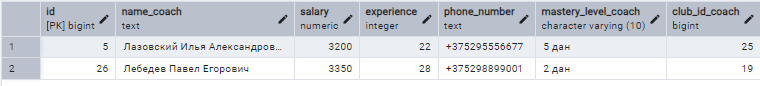


Рисунок 1.8 – Выполнение SQL запроса

## 1.5 INNER JOIN

Выбрать все клубы с их тренерами (рисунки 1.9-1.11).

SQL-команда: SELECT cl.name\_club, co.name\_coach, co.mastery\_level\_coach FROM club cl INNER JOIN coach co ON co.club\_id\_coach = cl.id;

Изображение выглядит как текст, меню, снимок экрана, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.9 – Исходная таблица «club»

Изображение выглядит как текст, меню, снимок экрана, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.10 – Исходная таблица «coach»



Рисунок 1.11 – Выполнение SQL запроса

## 1.6 INNER JOIN (2)

Выбрать все клубы и их мероприятия типа «кумите» (рисунки 1.12-1.14).

SQL-команда: SELECT cl.name\_club, ev.name\_event, ev. type\_event FROM club cl INNER JOIN event ev ON ev.club\_id\_event = cl.id WHERE type\_event = 'кумите';

Изображение выглядит как текст, меню, снимок экрана, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.12 – Исходная таблица «club»

Изображение выглядит как текст, снимок экрана, меню, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.13 – Исходная таблица «event»

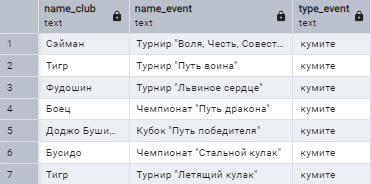


Рисунок 1.14 – Выполнение SQL запроса

## 1.7 LEFT OUTER JOIN

Получить список мест проведения мероприятия, где вместимость больше 300 человек и уровень мероприятия международный (рисунки 1.15-1.17).

SQL-команда: SELECT e.name\_event, l.address, l.capacity, e.level FROM event\_location l LEFT JOIN event e ON e.evenlocation\_id\_event = l.id WHERE capacity > 300 AND level = 'международный';

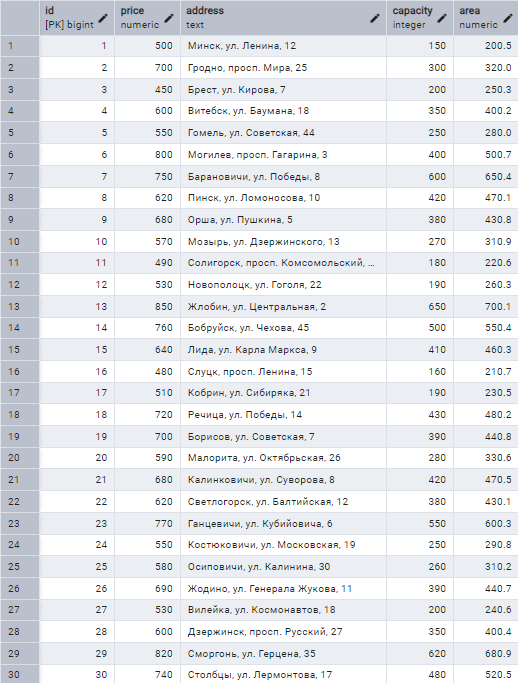


Рисунок 1.15 – Исходная таблица «event\_location»

Изображение выглядит как текст, снимок экрана, меню, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.16 – Исходная таблица «event»

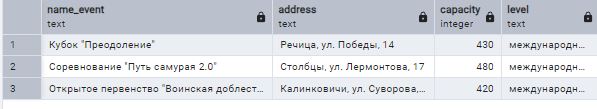


Рисунок 1.17 – Выполнение SQL запроса

## 1.8 LEFT OUTER JOIN (2)

Вывести список студентов у которых ежемесячный платёж больще 50 и у которых 8 кю(рисунки 1.18-1.20).

SQL-команда:SELECT s.fullname\_stydent, s.mastery\_level\_ student, p.amount FROM student s LEFT JOIN payment p ON p.student\_id\_payment = s.id WHERE amount > 50 AND mastery\_level\_student = '8 кю';

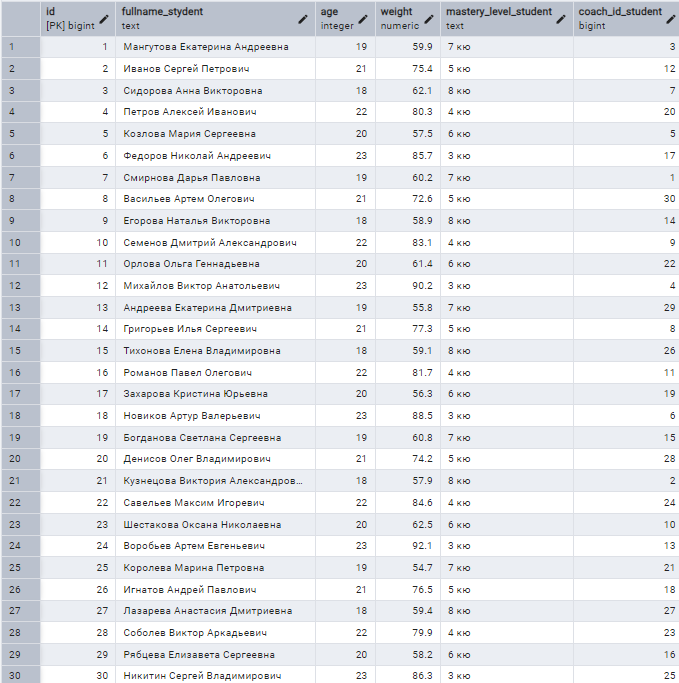


Рисунок 1.18 – Исходная таблица «student»

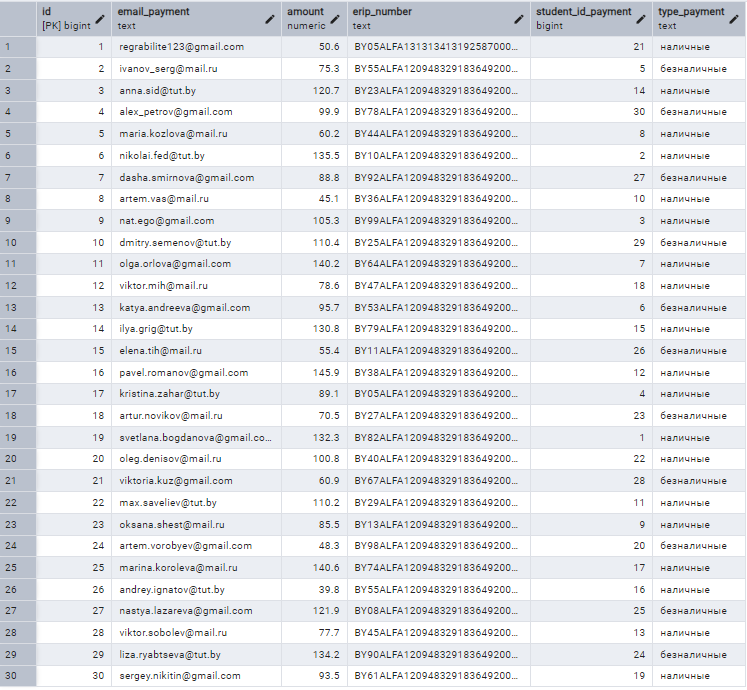


Рисунок 1.19 – Исходная таблица «payment»

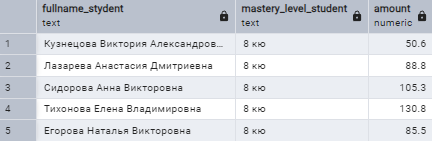


Рисунок 1.20 – Выполнение SQL запроса

## 1.9 ORDER BY

Вывести места проведения мероприятий по вместимости (возрастание) (рисунки 1.21-1.22).

SQL-команда: SELECT \* FROM event\_location ORDER BY capacity ASC;

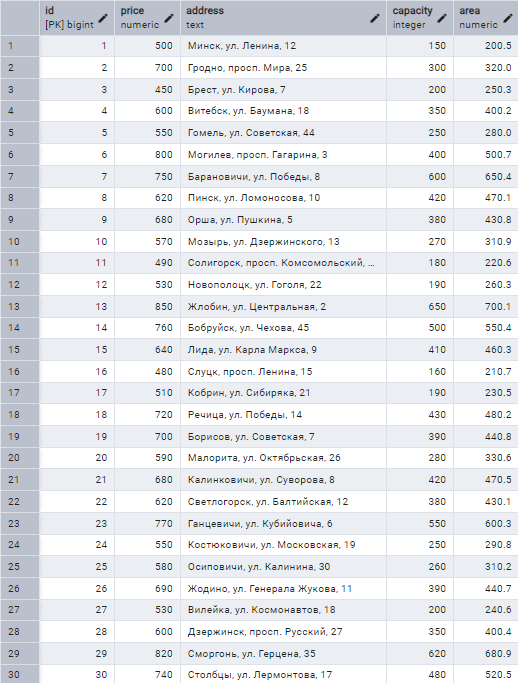


Рисунок 1.21 – Исходная таблица «event\_location»

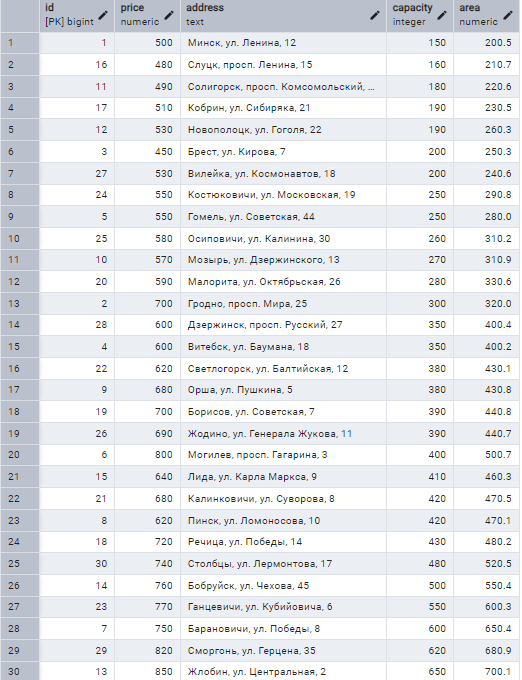


Рисунок 1.22 – Выполнение SQL запроса

## 1.10 ORDER BY (2)

Вывести список тренеров, отсортированный по поясам (возрастание) и фамилии (убывание) (рисунки 1.23-1.24).

SQL-команда: SELECT \* FROM coach ORDER BY mastery\_level\_coach ASC, name\_coach DESC;

Изображение выглядит как текст, меню, снимок экрана, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.23 – Исходная таблица «coach»



Рисунок 1.24 – Выполнение SQL запроса

## 1.11 RIGHT OUTER JOIN

Вывести список тренеров, которые проводят боевые тренировки (рисунки 1.25-1.27).

SQL-команда: SELECT coach.name\_coach, coach.phone\_number, workout.type, workout.group\_name FROM coach RIGHT JOIN workout ON workout.coach\_id\_workout = coach.id WHERE workout.type = 'боевая';



Рисунок 1.25 – Исходная таблица «workout»

Изображение выглядит как текст, меню, снимок экрана, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.26 – Исходная таблица «coach»

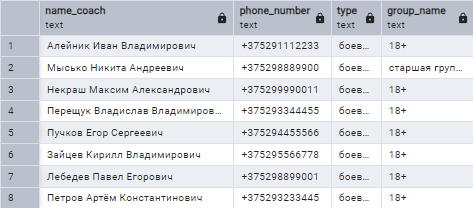


Рисунок 1.27 – Выполнение SQL запроса

## 1.12 FULL OUTER JOIN

Вывести все клубы и все мероприятия, включая клубы без мероприятий и мероприятия, которые не принадлежат никаким клубам, у которых тип кате (рисунки 1.28-1.30).

SQL-команда: SELECT c.name\_club, c.karate\_style, e.name\_event, e.type\_event FROM club c FULL JOIN event e ON c.id = e.club\_id\_event WHERE e.type\_event = 'кате';

Изображение выглядит как текст, снимок экрана, меню, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.28 – Исходная таблица «event»

Изображение выглядит как текст, меню, снимок экрана, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.29 – Исходная таблица «club»

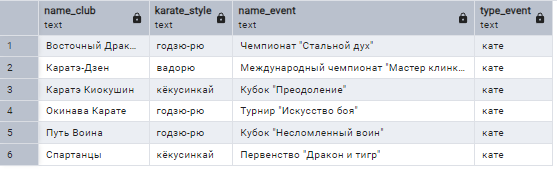


Рисунок 1.30 – Выполнение SQL запроса

## 1.13 FULL OUTER JOIN (2)

Вывести всех тренеров и всех учеников у которых вес меньше 80 килограмм (рисунки 1.31-1.33).

SQL-команда: SELECT

c.name\_coach AS coach\_name,

s.fullname\_stydent AS student\_name,

s.weight

FROM coach c

FULL JOIN student s ON c.id = s.coach\_id\_student

WHERE s.weight < 80 OR s.weight IS NULL;

Изображение выглядит как текст, меню, снимок экрана, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.31 – Исходная таблица «coach»

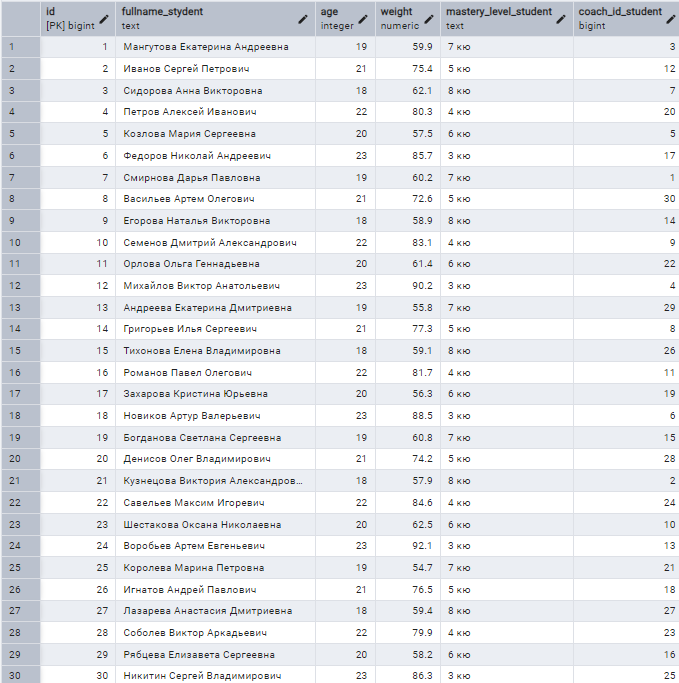


Рисунок 1.32 – Исходная таблица «student»

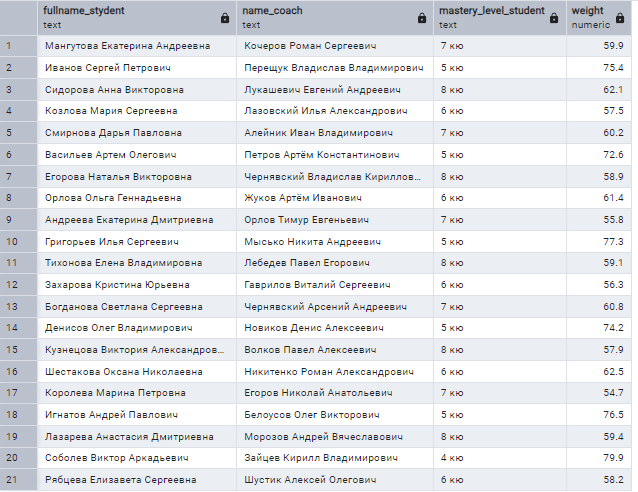


Рисунок 1.33 – Выполнение SQL запроса

## 1.14 CROSS JOIN

Получить список возможных учеников, которые примут участие в мероприятиях типа «кумите» в возрасте 18 лет (рисунки 1.34-1.36).

SQL-команда: SELECT s.fullname\_stydent, e.name\_event, e.type\_event FROM student s CROSS JOIN event e WHERE e.type\_event = 'кумите' AND s.age =18;

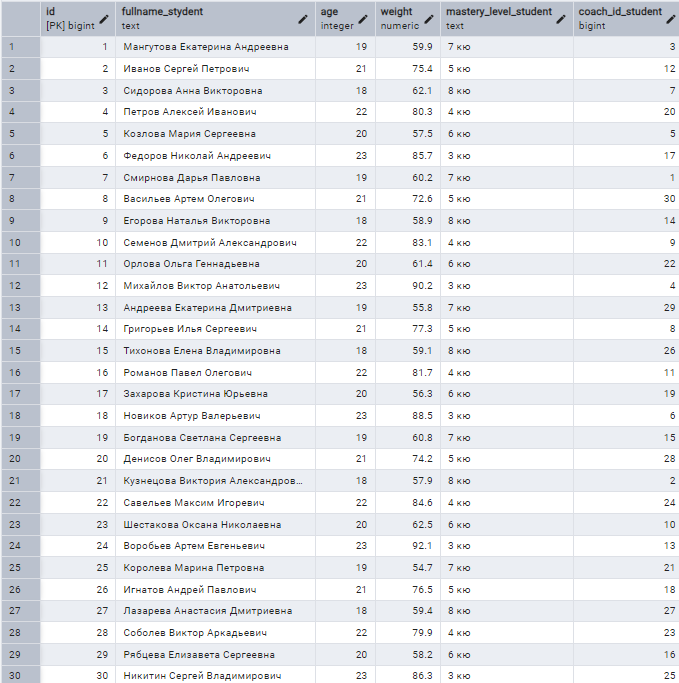


Рисунок 1.34 – Исходная таблица «student»

Изображение выглядит как текст, снимок экрана, меню, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1.35 – Исходная таблица «event»

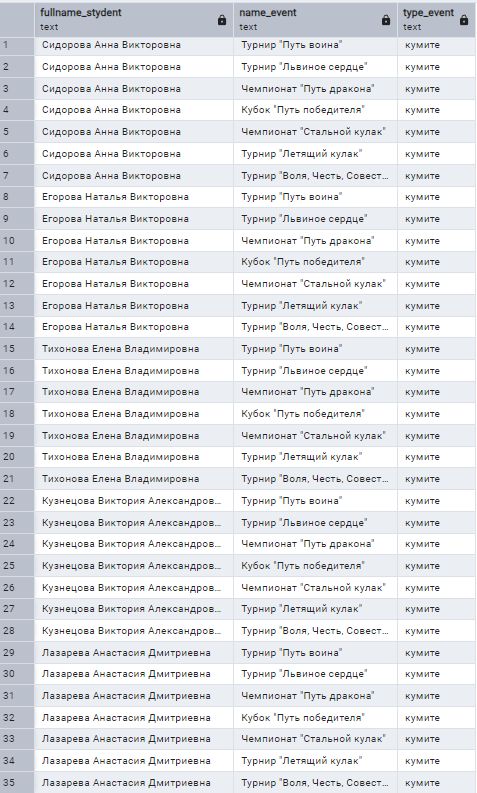


Рисунок 1.36 – Выполнение SQL запроса

**ЗАКЛЮЧЕНИЕ**

В ходе выполнения лабораторной работы были изучены и применены на практике основные принципы составления SQL-запросов на выборку данных. Эти инструменты позволяют эффективно извлекать, фильтровать, сортировать и объединять данные из одной или нескольких таблиц, что является важным навыком для работы с базами данных.

На примере модели базы данных сети клубов карате были составлены и выполнены запросы, которые демонстрируют различные сценарии работы с данными: от простой выборки информации из одной таблицы до сложных запросов с объединением нескольких таблиц и использованием условий фильтрации. Это позволило на практике понять, как SQL может быть использован для решения реальных задач, таких как поиск тренировок, анализ мероприятий, работа с тренерами.

Выполнение данной работы способствовало развитию навыков работы с SQL, пониманию логики построения запросов и умению анализировать структуру базы данных для извлечения необходимой информации. Полученные знания и опыт будут полезны для дальнейшего изучения более сложных аспектов SQL, таких как агрегация данных, оптимизация запросов и проектирование баз данных.

Таким образом, лабораторная работа успешно достигла своей цели, предоставив практические навыки работы с SQL и углубив понимание принципов взаимодействия с реляционными базами данных. Эти навыки являются важной основой для дальнейшего профессионального роста в области анализа данных, разработки программного обеспечения и управления информационными системами.