МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Лабораторная работа №11

**Основы SQL**

**по дисциплине**

**«Введение в информационные технологии»**

Выполнил: студент гр. БВТ2403

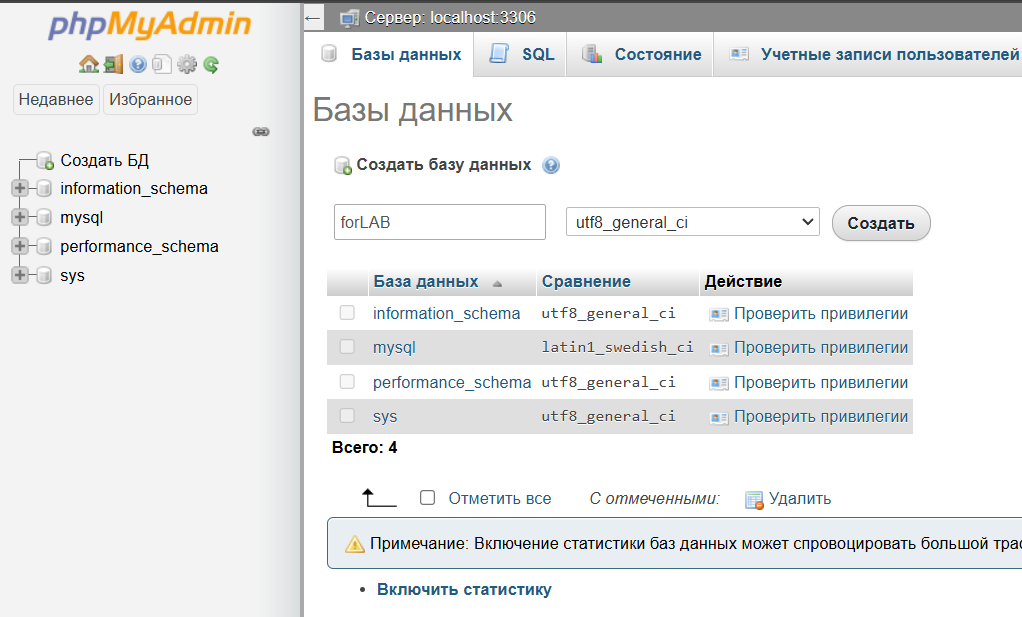
Лашков Ю.Е.

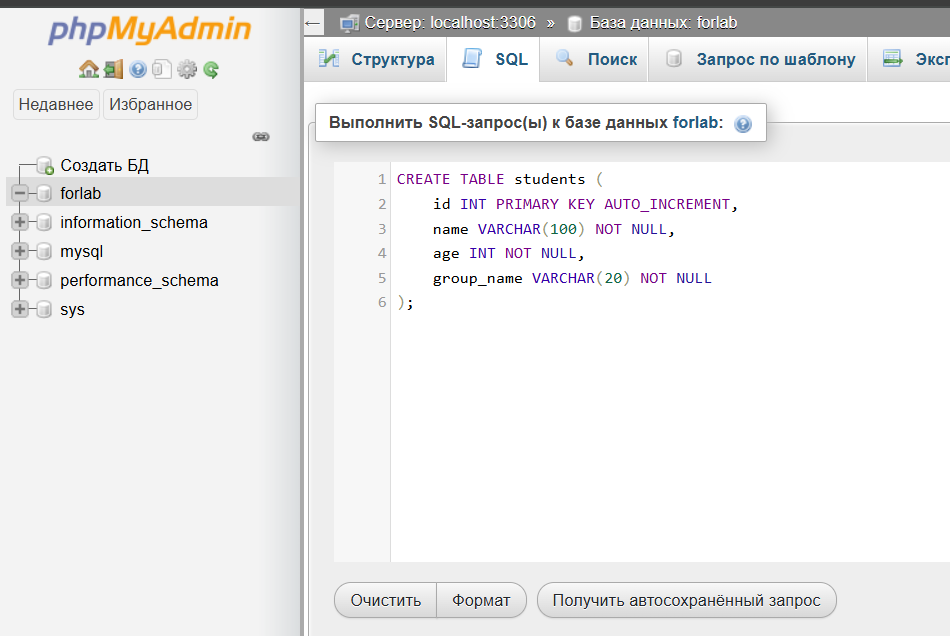
Проверил: Павликов А.Е.

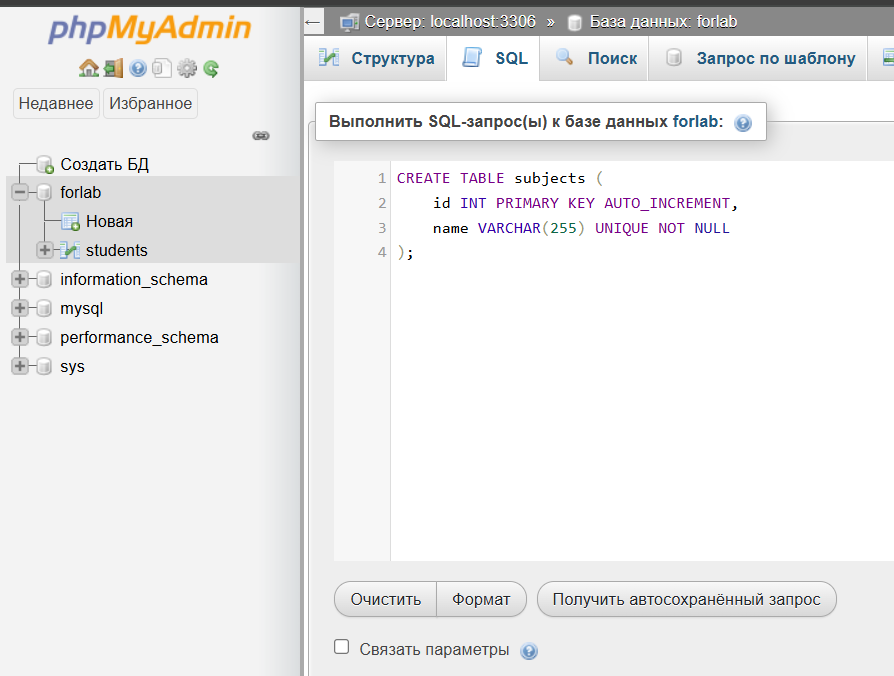
Москва, 2025 г.

**Цель работы**: Познакомить студентов с основными операциями работы с базами данных SQL, включая создание таблиц, добавление данных, выполнение запросов и работу с отношениями между таблицами.  
**Оборудование:** Компьютер с ОС Windows, установленный сервер баз данных (например, MySQL, PostgreSQL) и клиентское приложение (например, phpMyAdmin).  
**Введение:** SQL — это язык для управления базами данных. Он позволяет создавать, изменять и извлекать данные из реляционных баз данных. SQL помогает пользователям организовывать данные, обеспечивать их целостность, а также эффективно находить и анализировать необходимую информацию.

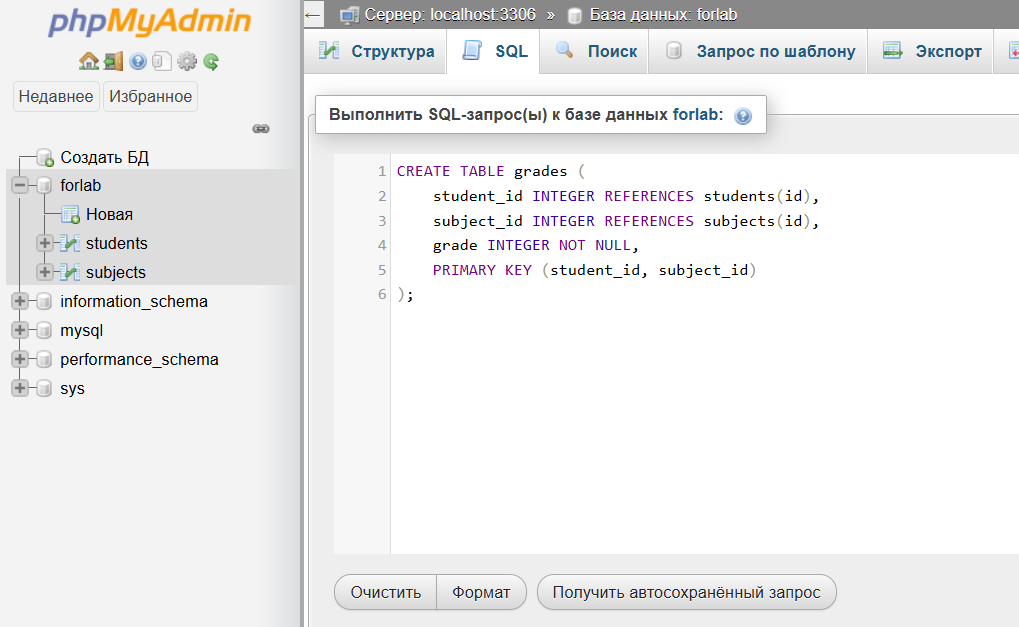
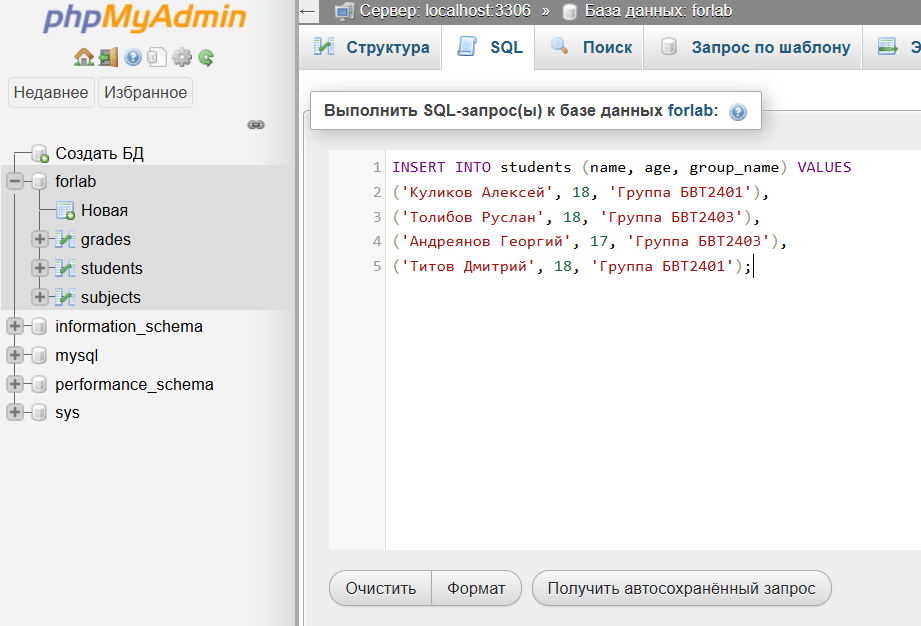
**Практическое задание**

  
Чтобы начать работу с базами данных в phpMyAdmin, нам нужно создать новую базу данных под названием "forLAB"

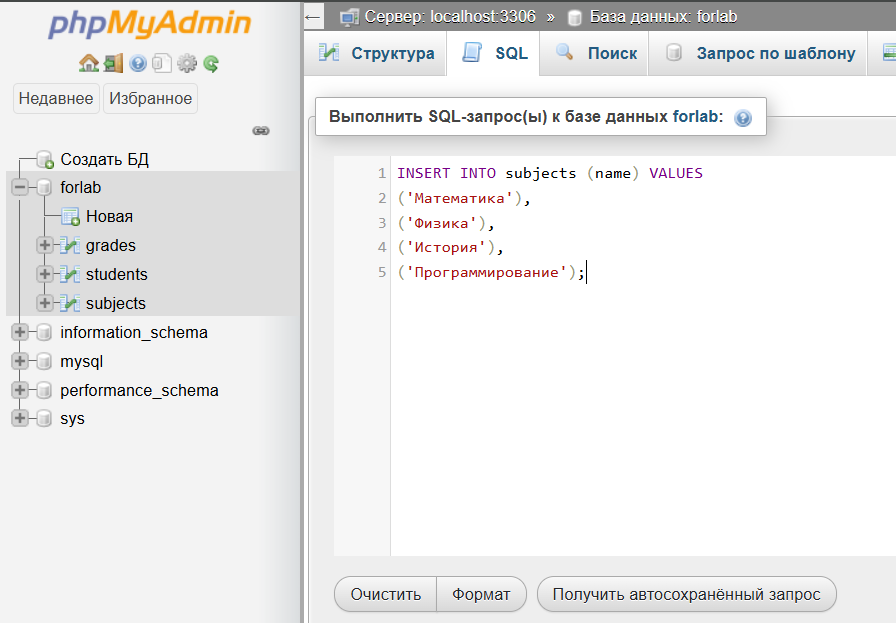
  
Чтобы создать таблицу "students" в базе данных "forLAB", я использую SQL-запрос CREATE TABLE. В этом запросе я определяю структуру таблицы, указывая для каждого столбца (id, name, age, group\_name) его тип данных (INT или VARCHAR) и ограничения (PRIMARY KEY, AUTO\_INCREMENT, NOT NULL). id будет автоматически генерируемым уникальным идентификатором студента, name - именем, age - возрастом, а group\_name - названием группы. При этом, столбцы name, age и group\_name обязательны для заполнения.

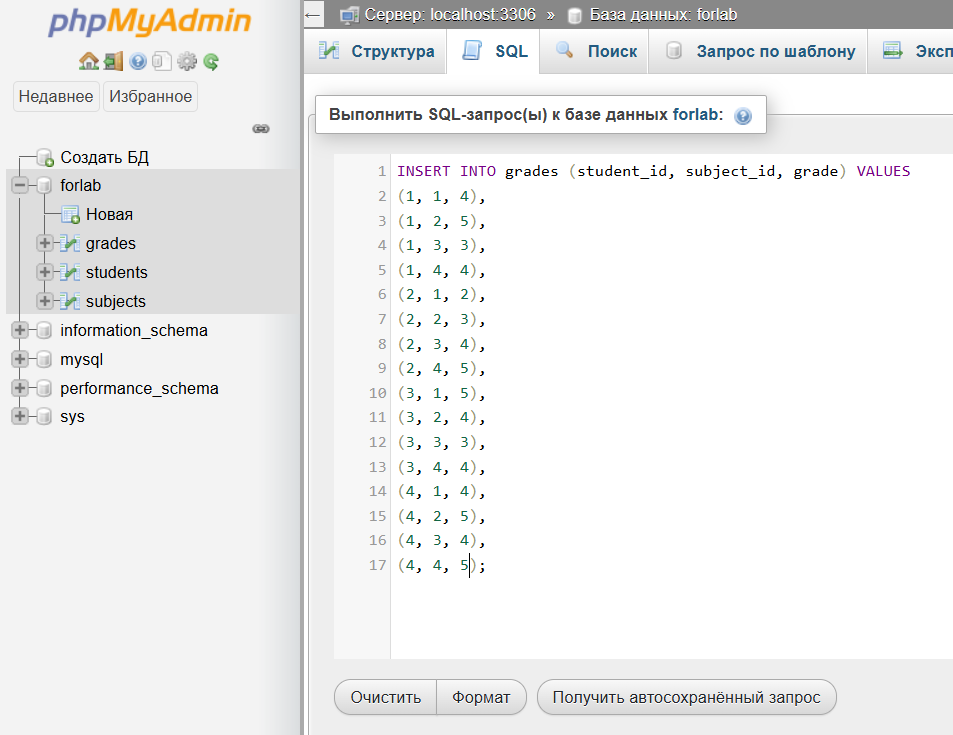


Для хранения списка предметов создаю таблицу subjects с помощью SQL запроса CREATE TABLE. Она содержит столбцы id (уникальный, автогенерируемый идентификатор) и name (название), которое обязательно и уникально.

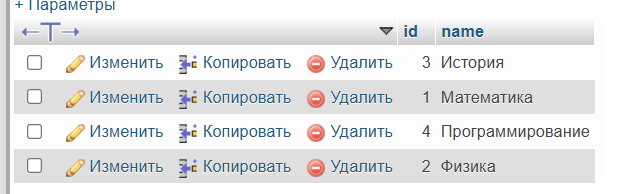
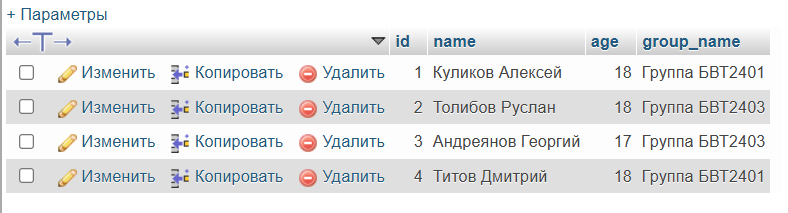
  
Чтобы хранить информацию об оценках, я использую запрос CREATE TABLE grades. Он создает таблицу, в которой будут столбцы: student\_id и subject\_id (указывают, кому и по какому предмету поставлена оценка), а также сама grade. Очень важно, что student\_id и subject\_id не просто цифры, а ссылки на таблицы students и subjects (это делается с помощью REFERENCES), то есть нельзя поставить оценку несуществующему студенту или по несуществующему предмету. Ну и, конечно, сама оценка (grade) должна быть обязательно указана.  


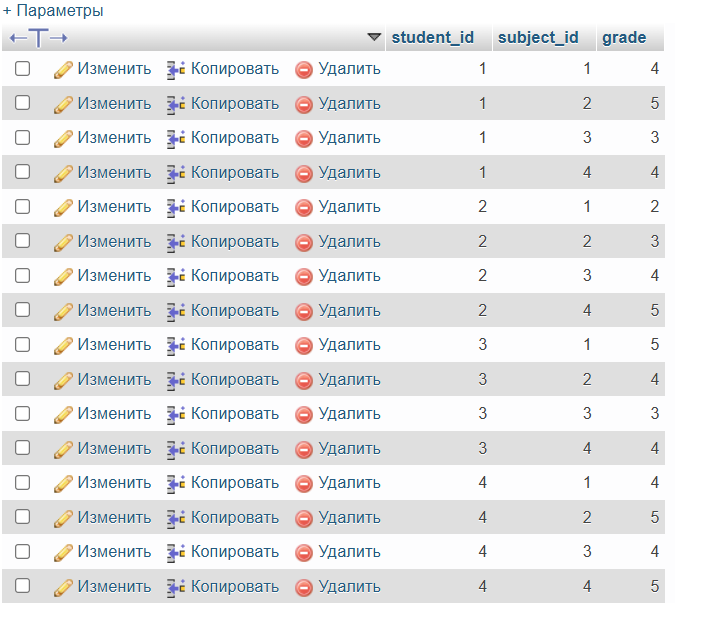
Чтобы добавить информацию о новых студентах в таблицу students, я использую запрос INSERT INTO. В этом запросе я указываю имена, возраст и группы для каждого студента, и эти данные будут добавлены в таблицу.

  
Чтобы заполнить таблицу subjects названиями предметов, я использую команду INSERT INTO subjects (name) VALUES. Для каждого предмета (математика, физика, история, программирование) добавляется отдельная строка с его названием.

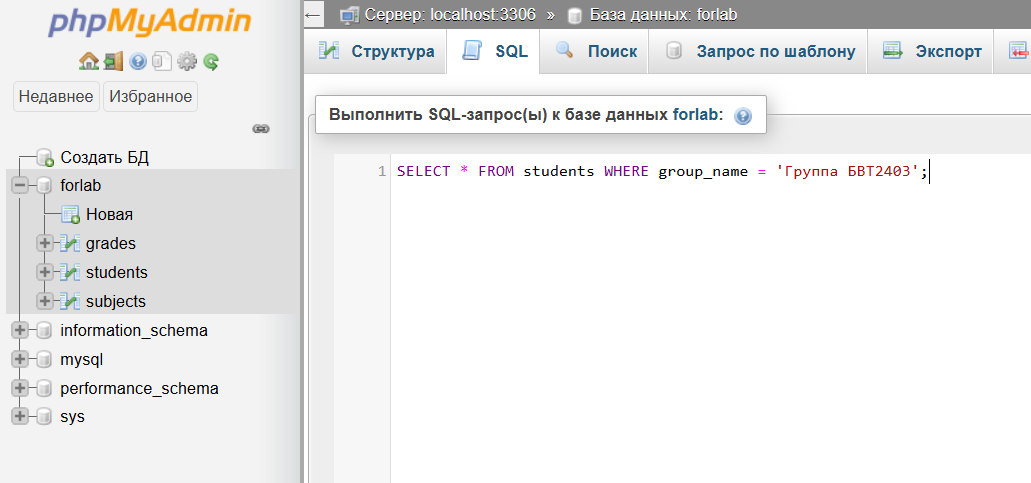


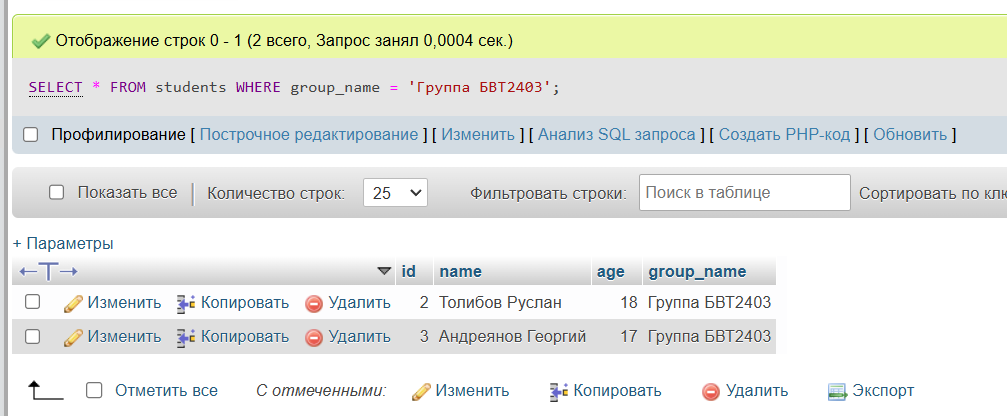
Чтобы внести оценки студентов по разным предметам, я использую команду INSERT INTO grades. Для каждой оценки я указываю student\_id, subject\_id и саму grade, тем самым связывая студента, предмет и его оценку.

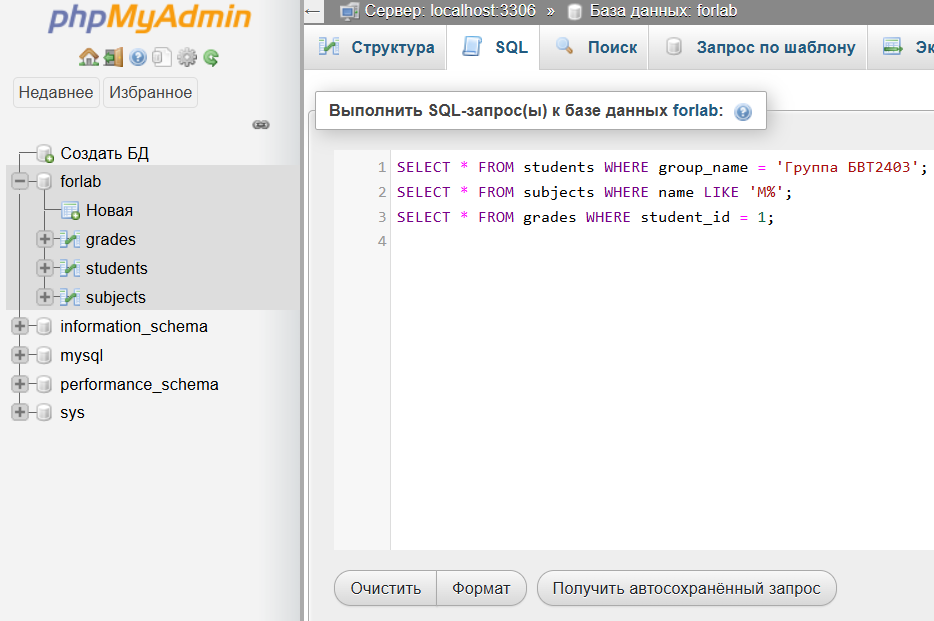


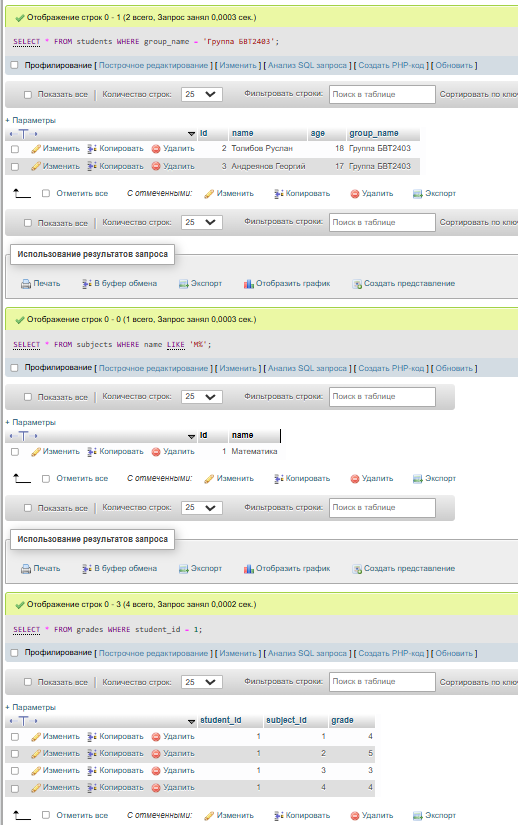


Тут наглядно видно, что таблицы students, subjects и grades успешно заполнены данными, отражая информацию о студентах, предметах и их оценках.



  
С помощью SQL запроса SELECT \* FROM students WHERE group\_name = 'Группа БВТ2403', я наглядно показываю, как можно отфильтровать данные и получить информацию только о студентах из определенной группы.



  
Результат работы запросов SELECT: мы видим студентов только из группы "БВТ2403", предметы, название которых начинается с "М" (в данном случае "Математика"), и все оценки студента с student\_id = 1.

**Заключение:** В ходе лабораторной работы мы научились создавать базы данных и таблицы, определять типы данных и ограничения, а также добавлять и извлекать информацию с помощью языка SQL. Мы освоили основные команды CREATE TABLE, INSERT INTO и SELECT, что позволяет эффективно управлять и анализировать данные в реляционных базах данных.